Remigiusz
Żulicki

Data science: najseksowniejszy zawód XXI wieku w Polsce

Big data, sztuczna inteligencja i PowerPoint

Data science: najseksowniejszy zawód XXI wieku w Polsce

WYDAWNICTWO
UNIWERSYTETU
ŁÓDZKIEGO

Remigiusz
Żulicki

Data science: najseksowniejszy zawód XXI wieku w Polsce

Big data, sztuczna inteligencja i PowerPoint

WYDAWNICTWO
UNIWERSYTETU
ŁÓDZKIEGO
Łódź 2022

Remigiusz Żulicki (ORCID: 0000-0003-2624-2422) – Uniwersytet Łódzki
Wydział Ekonomiczno-Socjologiczny, Katedra Metod i Technik Badań Społecznych
90-214 Łódź, ul. Rewolucji 1905 r. nr 41/43

RECENZENCI
Dariusz Jemielniak, *Kazimierz Krzysztofek*

REDAKTOR INICJUJĄCY
Katarzyna Włodarczyk-Gil

OPRACOWANIE REDAKCYJNE
Monika Poradecka

SKŁAD I ŁAMANIE
AGENT PR

KOREKTA TECHNICZNA
Wojciech Grzegorczyk

PROJEKT OKŁADKI
Tomasz Kochelski

https://doi.org/10.18778/8331-110-4

Wydane przez Wydawnictwo Uniwersytetu Łódzkiego
Wydanie I. W.10865.22.0.M

Ark. wyd. 23,5; ark. druk. 21,75

ISBN 978-83-8331-110-4
e-ISBN 978-83-8331-111-1

Wydawnictwo Uniwersytetu Łódzkiego
90-237 Łódź, ul. Matejki 34A
www.wydawnictwo.uni.lodz.pl
e-mail: ksiegarnia@uni.lodz.pl
tel. 42 635 55 77

Spis treści

Podziękowania

Choć jestem jedynym autorem tej książki, to powstała ona dzięki pomocy wielu osób, którym chciałbym w tym miejscu podziękować.

Dziękuję za wsparcie i konsultacje: Tadeuszowi Antczakowi (relacyjne bazy danych), Julicie Czerneckiej (pytania drażliwe w wywiadach swobodnych), Aleksandrze Ełbakian (dostęp do literatury), Katarzynie Grzeszkiewicz-Radulskiej (wyjaśnianie statystyki i zwrot w koncepcji pracy), Dariuszowi Jemielniakowi (jasna krytyka i dodanie odwagi), Annie Kacperczyk (przewodnictwo po społecznych światach), Dariuszowi Koperczakowi (wyrozumiałość szefa), Kazimierzowi Krzysztofkowi (zombie i wskazanie pominięć), Piotrowi Migdałowi (laptop z naklejkami i kontakt mailowy), Arturowi Suchwałko (dyscyplina czasowa i rozmowy telefoniczne), Izabeli Staszczyszyn (serwery i bazy nierelacyjne). Pozwalam sobie pominąć Państwa tytuły naukowe i stanowiska, ponieważ pomagaliście mi po ludzku, także poza obowiązkami zawodowymi. Dziękuję Wam.

Bardzo dziękuję Kazimierzowi Kowalewiczowi, promotorowi mojej pracy doktorskiej, na podstawie której powstała ta książka. Bez zaufania, cierpliwości, porad i spokoju Pana Profesora nigdy bym jej nie napisał. Bez Pana Profesora wcale zresztą nie ukończyłbym socjologii. Dziękuję Panu za krytykę, za czytanie setek stron, dziękuję za kasety magnetofonowe, dziękuję za wszystko!

Najmocniej dziękuję osobom, które brały udział w badaniach, rozmawiały ze mną, objaśniały i pokazywały data science. Starałem się patrzeć na Wasz świat Waszymi oczami i sukcesem dla mnie będzie, jeżeli znajdziecie choć odrobinę tej perspektywy w niniejszej książce.

Remigiusz Żulicki

Rozdział 1
Uwagi wstępne

Czy sztuczna inteligencja pozbawia nas pracy? Czy algorytmy przejmują władzę nad światem? Czy big data sprawia, że jesteśmy bezustannie inwigilowani? Czy ogromna ilość danych zastępuje ekspertów i naukowców? Cokolwiek sądzi się na te tematy, istnieje heterogeniczne środowisko ludzi zajmujących się tzw. sztuczną inteligencją czy big data od strony technicznej i metodologicznej. To środowisko posługuje się narzędziami matematycznymi i informatycznymi, wykonując operacje na danych cyfrowych za pomocą skryptowych języków programowania. Nie należy mylić ich z programistami (*software developers*). Pole ich działania nazywane jest data science (dalej DS), a oni data scientists. Niniejsza publikacja poświęcona jest właśnie im – polskiemu środowisku DS, które traktuję jako świat społeczny, wzorując się głównie na podejściu Adele E. Clarke. Celem głównym opracowania jest opis etnograficzny polskiego społecznego świata DS.

Data science to dynamiczne, zmienne, nowe, bardzo słabo rozpoznane i niejasne zjawisko społeczne o poważnych konsekwencjach dla innych sfer życia. Niejasność ta spowodowana jest:

1) wysokim poziomem zaawansowania matematyczno-informatycznego w substancjalnej sferze działalności data scientistów;
2) omawianymi poniżej sporami definicyjnymi;
3) ogromem szumu – nie tylko medialnego – i entuzjazmu wobec tej branży;
4) charakterystyczną dla wysokobudżetowej branży nowoczesnych technologii kulturą tajemnicy.

> Tak jak czarnymi skrzynkami są często technologie AI, tak samo są nimi przemysłowe kultury, w których te technologie powstają. Rzadko możliwy jest wgląd w szczegóły systemów i ich otoczenia, właśnie z powodu korporacyjnych przepisów tajności (Crawford i in., 2018: 11).

Oczywiste jest, że DS to forma komputerowej analizy danych cyfrowych. Skoro zatem pewni ludzie określają swoją działalność terminem „data science", a siebie samych „data scientists", skoro inni ludzie chcą korzystać z usług tych pierwszych albo dołączyć do nich i stać się data scientistami, skoro wiadomo (choć niezbyt precyzyjnie), co robią data scientiści, to mowa o DS jako o całości społecznej, wyróżnianej z uwagi na działanie jej uczestników.

Rezultatem pracy data scientistów są elementy systemów technologicznych, nazywane potocznie algorytmami lub sztuczną inteligencją, a technicznie – modelami uczenia maszynowego (*machine learning* – ML), z którymi na co dzień, często nieświadomie, w trakcie korzystania z takich narzędzi jak wyszukiwarka internetowa Google lub aparat fotograficzny w smartfonie, stykają się miliardy ludzi na całym świecie:

> Algorytmy w ostatnich latach stały się hasłem przewodnim (*catchword*): ogniskiem publicznej fascynacji; rzadkim artefaktem, który wymaga nadzwyczaj wysokich wynagrodzeń dla tych, którzy je tworzą; piorunochronem dla tajemnicy biznesowej i „magiczną czarną skrzynką" dla tych, którzy z nich korzystają. Przy tym wszystkim, **bierzemy je za pewnik** [podkr. R.Ż.]. Kształtują dla nas kanały informacyjne (*newsfeeds*), personalizują nasze listy życzeń, włączają nam grzejniki i zoptymalizują nam ćwiczenia fizyczne. [Algorytmy – przyp. R.Ż.] To rzeczy codziennego użytku (Thomas, Nafus, Sherman, 2018: 1).

Dojrzewanie i rozpowszechnianie się tzw. sztucznej inteligencji sprawia, że algorytmy (modele ML) wtapiają się w ludzkie doświadczenie i otoczenie jako mediatorzy. Wpływają silnie, ale w sposób słabo zauważalny, bo wygodny, na interakcje pomiędzy ludźmi i między ludźmi a otoczeniem (Taddeo, Floridi, 2018). Systemy te znajdują zastosowanie w rozmaitych dziedzinach życia i stanowią kolejne technologie ogólnego zastosowania, tak jak rozpoczęta kilkadziesiąt lat temu cyfryzacja jako stosowanie komputerów i internetu. Rezultaty pracy data scientistów wdrażane są nie tylko w sferze dóbr i usług konsumpcyjnych, ale także we wrażliwych obszarach, np. zatrudnienia, edukacji, ochrony zdrowia, opieki społecznej, transportu, polityki, administracji publicznej, wymiaru sprawiedliwości.

Z uwagi na narosły wokół DS entuzjazm, dynamiczny rozwój tej dziedziny, a przy tym poważne konsekwencje społeczne badanie etnograficzne, nastawione głównie na ogląd świata DS „od wewnątrz", uznaję za istotny problem dla socjologicznej pracy badawczej.

1.1. Historia terminów związanych z data science

Terminy „big data", „sztuczna inteligencja", „uczenie maszynowe" i „data science" występują zarówno w mediach głównego nurtu, jak i publikacjach akademickich. Są stosowane przez uczestników społecznego świata DS. W ostatnich dwunastu latach każdy z nich zyskał na popularności, biorąc pod uwagę względną liczbę dotyczących ich zapytań wpisywanych w wyszukiwarkę internetową Google (Google Trends, 2020a; 2020b). Różni autorzy nadają tym czterem terminom odmienne znaczenia i na wiele sposobów określają relacje zachodzące między nimi. Jeszcze większy chaos panuje w dyskursie publicznym, w mediach i ogólnie w gronie nietechnicznym, także w naukach społecznych.

W tej książce przyjrzę się ewolucji analizowanych terminów i niejasnościom, które ich dotyczą. Ich znajomość jest drobnym, niezbędnym elementem wiedzy uczestnika świata społecznego DS. Bez poznania kilku znaczeń czterech wyróżnionych terminów nie można mówić o świecie społecznym DS z perspektywy jego uczestników, a taką perspektywę przyjmuję. Pokażę między innymi, co oznacza, że uczestnicy społecznego świata DS kojarzą termin „big data" z narzędziem Spark, a termin „sztuczna inteligencja" z prezentacjami w PowerPoincie.

1.1.1. Big data

Jak podaje jeden z jego „ojców chrzestnych" (Diebold, 2012), termin „big data" prawdopodobnie ukuł John R. Mashey podczas przerwy na lunch w firmie Silicon Graphics Inc. w połowie lat dziewięćdziesiątych ubiegłego wieku. Pierwsze referencje akademickie w dziedzinie informatyki to praca Sholoma M. Weissa i Nitina Indurkhya (*Predictive Data Mining. A practical guide* z 1998 roku), w statystyce/ekonometrii zaś Francisa X. Diebolda (*Big Data Dynamic Factor Models for Macroeconomic Measurement and Forecasting* z 2000 roku). Diebold użył go przypadkiem, nie słysząc terminu wcześniej, a samo określenie uznał za „trafne, dźwięczne i intrygująco orwellowskie, szczególnie gdy zapisze się je z wielkich liter" (Diebold, 2012: 2). Pojęcie „big data" zostało zdefiniowane w 2001 roku przez Douglasa Laneya z firmy doradczej Gartner jako dane, które charakteryzuje: wielkość (*volume*), różnorodność (*variety*), szybkość (*velocity*) (Laney, 2001). Definicja była powtarzana jako „3 Vs of big data" (Chen, Chiang, Storey, 2012; Soubra, 2012; TechAmerica, 2012; Berman, 2013; Ohlhorst, 2013; Cukier, Mayer-Schönberger, 2014; Kwon, Lee, Shin, 2014). Cechy te opisywano w następujący sposób:

1) wielkość (*volume*) – rozmiar danych; bywa rozumiana jako zbiory większe niż 1 TB oraz takie, których nie da się przetwarzać za pomocą tradycyjnych narzędzi (relacyjnych baz danych i arkuszy kalkulacyjnych) (Gandomi, Haider, 2015);
2) różnorodność (*variety*) – zróżnicowanie; różne struktury, formaty i charakter danych (np. teksty i fotografie pochodzące z blogów, stron WWW, mediów społecznościowych);
3) szybkość (*velocity*) – szybkość pojawiania się nowych danych; ciągły ich napływ, powodujący, że potrzebne są metody umożliwiające wydobywanie informacji z danych w czasie rzeczywistym (Cukier, Mayer-Schönberger, 2014; Gandomi, Haider, 2015).

Powyższe wymiary pozostają w zależności – zmiana jednego powoduje zmianę drugiego. Douglas Laney (2001) obrazuje to w postaci trzech różnych osie w trójwymiarowym układzie współrzędnych. Nie ma jednak kryteriów dotyczących tego, gdzie „zaczynają się" big data (Gandomi, Haider, 2015), choć można znaleźć żartobliwe stwierdzenia: „big data są wtedy, gdy dane nie mieszczą się w Excelu" czy poważniejsze: „dane są wielkie, gdy wielkość danych zaczyna być problemem"

(Dutcher, 2014). Definicje te mówią nie o wielkości bezwzględnej, a o relacji w stosunku do dysponowanej mocy obliczeniowej, co prowadzi do rozwoju sprzętu i oprogramowania, które mają tę wielkość uczynić możliwą do „skonsumowania" (Floridi, 2012: 435–436). Pojawiają się także propozycje kolejnych cech big data. Są to (Gandomi, Haider, 2015):

1) wiarygodność (*veracity*) – błędy w danych i ich prawdziwości;
2) zróżnicowanie (*variability and complexity*) – duże zróżnicowanie wartości zmiennych i złożoność danych, które spowodowane są m.in. tym, że analizie poddawana jest populacja, a nie próba, tzw. N = all (Cukier, Mayer-Schönberger, 2014);
3) wartość (*value*) – potencjał biznesowy, możliwości generowania zysków dzięki informacjom wydobytym z danych.

Big data to niekoniecznie tylko dane, jak wskazuje definicja 3Vs:

> […] powinno się zatem definiować Big Data jako układ składający się z: danych opisanych własnościami 3Vs (5Vs), metod składowania i przetwarzania danych, technik zaawansowanej analizy danych oraz wreszcie całego środowiska sprzętu informatycznego. Jest to zatem połączenie nowoczesnej technologii i teorii analitycznych, które pomagają optymalizować masowe procesy związane z dużą liczbą klientów czy użytkowników (Przanowski, 2014: 13).

Celem tego rodzaju analiz jest zazwyczaj stworzenie modelu matematycznego, wykonującego jakieś zadanie na podstawie danych.

Z punktu widzenia socjologa ciekawe jest to, że zdaniem entuzjastów big data doprowadziło do przełomu cywilizacyjnego porównywalnego z wynalezieniem druku, maszyny parowej czy internetu (Minelli, Dhiraj, Chambers, 2013; Cukier, Mayer-Schönberger, 2014). Big data to „rewolucja, która zmieni nasze myślenie, życie i pracę" (Cukier, Mayer-Schönberger, 2014); „big data może poznać nas lepiej, niż sami siebie znamy" (Gardner, 2012: 14). Wypowiedzi Drew Conwaya (z firmy Project Florida): „big data to ruch kulturowy, za pomocą którego kontynuujemy odkrywanie tego, jak ludzkość i świat oddziałują na siebie nawzajem", czy też Daniela Gillicka (z firmy Google): „reprezentuje zmianę kulturową […], coraz więcej decyzji podejmowanych jest za pomocą algorytmów działających na podstawie udokumentowanych, niezmiennych dowodów" (Dutcher, 2014) również wskazują na zmianę. Termin „big data" ogromną popularność zdobył w szeroko pojętym biznesie:

> Big data ma szanse zostać słowem roku. Chyba trzeba spędzić ostatnie miesiące na Księżycu, żeby nie myśleć o wdrożeniu big data w swojej branży. […] Big data zmieniło historię: chodzi mi o wybór Donalda Trumpa i brexit [mowa o istotnej roli analizy danych big data w wywieraniu wpływu na głosujących – przyp. R.Ż.]. Big data jest buzzwordem, nie ma konferencji biznesowej bez tego tematu[1] {notatka badacza, obserwacja 10}.

1 Otwierająca spotkanie wypowiedź Kamili Łepkowskiej, prowadzącej panel „Cyfrowe ślady, big data i polityka" w ramach wydarzenia „Igrzyska Wolności 2017 RE:wolucja / Cyfrowa Rzeczywistość" w Łodzi, zanotowana przeze mnie podczas obserwacji uczestniczącej

Mówi się o końcu zapotrzebowania na ekspertów dziedzinowych – big data ma zapewniać lepsze decyzje, bez odwoływania się do jakoby ograniczających teorii czy intuicji; wystarczy „pozwolić przemówić danym" (Cukier, Mayer-Schönberger, 2014: 19, 35, 185).

Coraz częściej pojawiają się teksty mówiące o światopoglądzie, filozofii czy religijnej wierze w dane – dataizmie (*dataism*), który zakładać ma, że każdy układ czy zjawisko o dowolnej naturze sprowadza się do przetwarzania danych. „Dataiści są sceptyczni wobec ludzkiej wiedzy czy mądrości, skłaniając się do zaufania w Big Data i algorytmy komputerowe" (Harari, 2017: 213–214). Istnieje strona internetowa dataists.com, na której opublikowano m.in. bardzo popularny diagram, prezentujący, czym jest DS (Conway, 2010). Entuzjazm wokół big data był przedmiotem żartów już w 2013 roku – „big data jest jak seks nastolatków: wszyscy o tym rozmawiają; nikt naprawdę nie wie, jak się to robi; każdy sądzi, że robią to inni, więc wszyscy twierdzą, że również to robią" (Ariely, 2013).

Powstały prace akcentujące zarówno pozytywne (Gardner, 2012; Eagle, Greene, 2014), jak i negatywne skutki big data (O'Neil, 2013; 2017; Surma, 2017). W pierwszych mówi się np. o zdrowszym i łatwiejszym życiu jednostek (Eagle, Greene, 2014: 31–50), optymalizacji ruchu ulicznego czy wykrywaniu aktywności przestępczych w przestrzeni miejskiej (Eagle, Greene, 2014: 85–98) – ogólnie o zmienianiu świata na lepsze dzięki rozwiązywaniu wcześniej nierozwiązywalnych problemów (Gardner, 2012: 14–15). Drugie poruszają np. kwestię niepewności i niedokładności wyników analiz dużych ilości danych (O'Neil, 2013; Surma, 2017), przedkładania skuteczności i szybkości działania algorytmów nad sprawiedliwość, co skutkuje m.in. kryminalizacją biednych dzielnic (O'Neil, 2017: 134–139), czy zamieniania nieświadomych tego klientów w produkty poprzez sprzedawanie ich uwagi reklamodawcom (Surma, 2017: 66–69).

Temat „big data" był od lutego 2012 do marca 2017 roku popularniejszy w wyszukiwarce Google niż „data science" i „uczenie maszynowe" (rys. 1.1). Później jego popularność spadła nieznacznie, ale silnie wzrosła dla dwóch wymienionych tematów. Niezmiennie najpopularniejszym z analizowanych był jednak temat „sztuczna inteligencja". W maju 2020 roku, gdy jego popularność[2] osiągnęła maksymalną wartość 100, dla tematu „big data" popularność była równa 10, dla „uczenia maszynowego" 21, a dla „data science" 16.

---

21 października 2017 roku około godziny 18:15. Cytowania materiałów pozyskanych w ramach badań własnych oznaczono w tekście jako {technika badawcza numer}, np. {obserwacja 8}, i w odróżnieniu od cytatów z literatury wyróżniono graficznie pionową szarą linią z lewej strony tekstu. Informacje o dacie i innych cechach każdego aktu badawczego podane zostały w aneksie.

2 Miara nazywana przez Google'a popularnością (oś y na rys. 1) reprezentuje względną częstość wyszukiwania. Przyjmuje wartości od 0 do 100 z dokładnością do 1; pojawia się też wartość „< 1". Miara uwzględnia skalę korzystania z wyszukiwarki Google ogółem, przedstawia wartość w liczbach względnych. Liczba 100 reprezentuje więc maksymalną popularność zanotowaną dla danego okresu i zakresu geograficznego. Wartość 50 oznacza, że dany temat był o połowę mniej popularny. Zero oznacza, że nie było wystarczającej ilości danych dla tego tematu (Google Trends, 2020a).

1.1.2. Sztuczna inteligencja

Sztuczna inteligencja (*artificial intelligence* – AI) to termin powiązany z big data. Za twórcę określenia uznawany jest John McCarthy, który użył go w opublikowanym 31 sierpnia 1955 roku zaproszeniu do udziału w warsztatach badawczych (McCarthy i in., 1955). Współcześnie AI definiowana jest zazwyczaj na cztery sposoby. Mówi się o systemach komputerowych, które myślą/działają w ludzki sposób lub racjonalnie (Russel, Norvig, 2009: 1–2). Sztuczna inteligencja służy zastosowaniom praktycznym: „automatyzacji różnego rodzaju rutynowej pracy, rozpoznawaniu obrazów, rozpoznawaniu mowy, wsparciu diagnoz medycznych czy badań naukowych" (Goodfellow, Bengio, Courville, 2016: 1–2). Początkowo AI rozwiązywała problemy „intelektualnie trudne dla ludzi, ale stosunkowo proste dla komputerów", czyli takie, które można opisać w sposób formalny (jak gra w szachy). Obecnie „wyzwaniem dla AI stały się problemy łatwe do wykonania dla ludzi, ale trudne do opisania formalnie" – takie, które ludzie rozwiązują intuicyjnie (rozpoznawanie mowy czy twarzy na obrazach) (Goodfellow, Bengio, Courville, 2016: 1–2). Popularność tematu „sztuczna inteligencja" w wyszukiwarce Google była wyższa niż pozostałych trzech analizowanych. Temat był co najmniej czterokrotnie bardziej popularny niż najpopularniejszy z pozostałych trzech tematów – w styczniu 2020 roku popularność „sztucznej inteligencji" była równa 91, podczas gdy dla „uczenia maszynowego" było to 21 – to najniższa różnica popularności (rys. 1.1A). W analizowanym okresie średnia popularność tematu „sztuczna inteligencja" wynosiła 76, dla „uczenia maszynowego" było to 8, dla „big data" 7 i dla „data science" 5 (Google Trends, 2020a).

Apokalipsą ludzkości wywołaną przez AI grozili m.in. przedsiębiorca technologiczny Elon Musk[3] i astrofizyk Stephen Hawkins[4]. To z ich udziałem powstały „23 reguły AI z Asilomar":

> AI dostarczyła już korzystnych narzędzi, które są używane codziennie przez ludzi na całym świecie. Jej dalszy rozwój, kierowany następującymi zasadami, będzie oferował niesamowite możliwości pomocy i wsparcia ludzi w nadchodzących dziesięcioleciach i wiekach (Future of Life Institute, 2017).

Podobnie jak w przypadku entuzjastów big data, tam również mówi się o rewolucji czy przełomie. Andrew Ng, uznany badacz (Uniwersytet Stanforda) i praktyk (wcześniej Google, później chiński konkurent Google'a, firma Baidu) sztucznej inteligencji twierdzi, że AI zmieni oblicze biznesu, tak jak przed stu laty zrobiła to elektryfikacja (Ng, 2017). Temat AI podejmowany jest też w medialnych dyskusjach o tym, czy „roboty zabiorą nam pracę". Od sierpnia 2017 roku pisały o tym

3 „AI jest naszym największym zagrożeniem egzystencjalnym", „za pomocą AI przywołujemy demona" (UNECE, 2014).

4 „Myślę, że rozwój pełnej sztucznej inteligencji może oznaczać koniec rasy ludzkiej" (Clark, 2014).

m.in. magazyny: „Wired" (Surowiecki, 2017), „Forbes" (Ozimek, 2017), „Independent" (Barnet, 2017), „The New York Times" (Williams, 2017) czy „The Guardian" (Elliott, 2018). Istnieje strona internetowa willrobotstakemyjob.com, na której można sprawdzić ryzyko automatyzacji dla konkretnego zawodu. Strona powstała na podstawie prognozy naukowców z Uniwersytetu Oksfordzkiego, którzy podjęli ten temat z uwagi na rozwój ML, będącego subdyscypliną AI, oraz rozwój big data (Frey, Osborne, 2017). Inny zespół przeprowadził badanie ankietowe na próbie 352 naukowców z dziedziny ML. Respondenci sądzili, że istnieje 50% szans na AI przewyższającą ludzi we wszystkich zadaniach w ciągu 45 lat od roku 2016, a na automatyzację wszystkich ludzkich stanowisk pracy w ciągu 120 lat (Grace i in., 2018: 1–2).

Hiperentuzjastą sztucznej inteligencji jest Ray Kurzweil, którego cechuje religijna wiara w rozwój technologiczny. Jego pojęcie Osobliwości (*Singularity*) oznacza chwilę w rozwoju ludzkości, w której człowiek przekroczy granice biologiczne – ograniczeń swojego umysłu i ciała, z osiągnięciem nieśmiertelności włącznie (Kurzweil, 2016: 23–24).

Pojęcie AI obecne jest w popkulturze. Jak wskazuje Rozalia Knapik, teksty, w których nadejście tzw. Osobliwości jest traktowane jako istotne zagrożenie dla ludzkości, pojawiają się w kulturze popularnej o wiele częściej niż te pokazujące inteligentne maszyny z podziwem, pokorą czy pobłażaniem dla ich służebnej roli. Popularność negatywnych wizji świadczy o trudnościach ze skonstruowaniem przekonującej wizji afirmatywnej. Wskazując m.in. na dzieła Stanisława Lema, Philipa K. Dicka i rodzeństwa Wachowskich, Knapik zaznacza, że obawa przed Osobliwością jest niejednoznaczna i obok lęku zawiera też element fascynacji tym wytworem człowieka (Knapik, 2018: 11). Szczególnie sci-fi bazuje na obawie wobec nieprzygotowania ludzkości do korzystania ze zdobyczy technologii i zbyt ufnej symbiozie człowieka z technologią – przykładami mogą tu być *Nowy Wspaniały Świat* Aldousa Huxleya i *Rok 1984* Georga Orwella. Za współczesną grę z tymi dziełami autorka uznaje brytyjski serial *Czarne lustro* Charliego Brookera (Knapik, 2018: 29). Rolę Boga „odgrywają na przemian konstruktorzy myślących maszyn i same sztuczne inteligencje" (Knapik, 2018: 13). Bezcielesność sztucznej inteligencji ma stawiać ją w „drabinie bytów" wyżej, niż gdyby posiadała ciało. Przy tym, z uwagi na trudność w zrozumieniu AI, jej działanie stanowi dla laika pokaz magiczny; laik wierzy AI „na słowo". W tekstach kultury – filmach *Odyseja kosmiczna*, *Terminator*, *Ja, Robot*, *Colossus* – pojawia się wątek skrajnie racjonalnego zbawienia ludzkości w postaci unicestwienia ludzi (Knapik, 2018: 113–129). Prowadzi to do konkluzji, że takie przedstawienia stanowią „nową realizację prastarego zjawiska, jakim jest tworzenie sobie Boga" (Knapik, 2018: 130), a „ludzie obawiają się Boga, ale uparcie wypatrują jego nadejścia" (Knapik, 2018: 188).

1.1.3. Uczenie maszynowe

Uczenie maszynowe jest jedną z najsilniej powiązanych z big data subdyscyplin AI (Russel, Norvig, 2009: 2; Goodfellow, Bengio, Courville, 2016: 9).

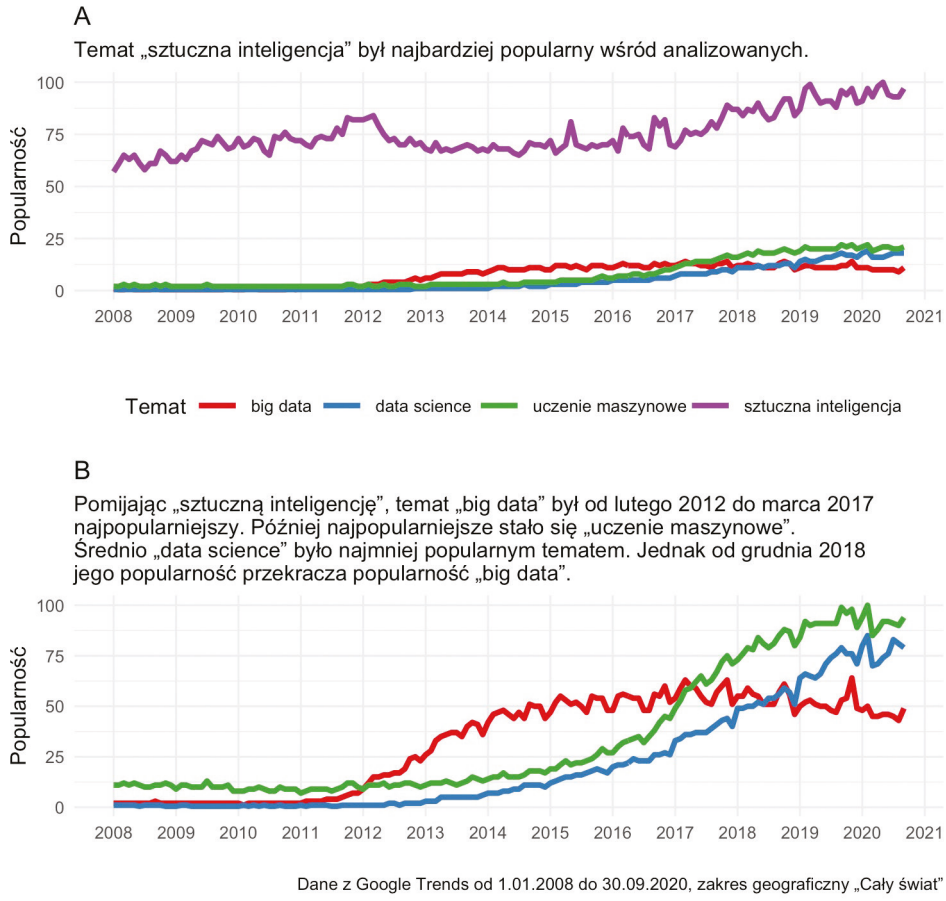
> W drugiej połowie XX w. uczenie maszynowe wyewoluowało z badań nad sztuczną inteligencją, w których projektowano samouczące się algorytmy, zdolne do pozyskiwania wiedzy z informacji oraz tworzenia na ich podstawie prognoz. Dzięki uczeniu maszynowemu nie trzeba zatrudniać ludzi do ręcznego określania reguł oraz tworzenia modeli poprzez analizowanie olbrzymich pokładów danych; omawiana dziedzina wiedzy oferuje efektywniejsze rozwiązanie polegające na stopniowym poprawianiu skuteczności modeli predykcyjnych oraz podejmowaniu decyzji na podstawie analizowanych danych (Raschka, 2018: 26).

Techniki ML to sposoby na budowanie tzw. modeli – od znanych również w naukach społeczno-ekonomicznych modeli statystycznych, np. regresji liniowej, regresji logistycznej, po złożone od strony matematycznej i informatycznej modele, np. sztucznych sieci neuronowych (*artificial neural networks*) czy maszyny wektorów nośnych (*support vector machine* – SVM) (Larose, 2005; 2012; Raschka, 2018). Pojawiają się opinie, że big data – w sensie ilości danych, ich dostępności i możliwości przetwarzania, ale również innego nastawienia[5] do analizy danych – umożliwia rozwój ML (Canton, 2016; Marr, 2016; Hansen, 2017). Przykładem tego jest Tłumacz Google. Dostęp do ogromnej ilości danych oraz przesunięcie akcentu z dokładności na szybkość obliczeń pozwoliły już na etapie prototypowej sieci neuronowej uzyskać tłumaczenie o 7 punktów BLEU lepsze[6] niż wcześniejsza wersja tłumacza, kiedy reguły programowano (Lewis-Kraus, 2017). Zainteresowanie tematem „uczenie maszynowe" w wyszukiwarce Google przekroczyło zainteresowanie „big data" w maju 2017 roku i utrzymywało się na poziomie wyższym niż dla tematów „big data" i „data science" do września 2020 roku (rys. 1.1). Wyłączając popularność „sztucznej inteligencji", w analizowanym okresie średnia popularność tematu „uczenie maszynowe" wynosiła 35, dla „big data" było to 33, a dla „data science" 22 (Google Trends, 2020b). W momentach maksymalnej popularności – we wrześniu 2019 i w lutym 2020 roku – temat „uczenie maszynowe" był dwa razy bardziej popularny niż „big data".

---

5 Mówi się o trzech zmianach: analizowaniu całej populacji zamiast próby, zmniejszeniu dokładności obliczeń na rzecz ich szybkości i zmianie pytania będącego podstawą budowania modeli z „dlaczego?" na „co?" (por. Cukier, Mayer-Schönberger, 2014).

6 Test dla tłumaczenia z języka angielskiego na francuski; „poprawa o 1 punkt była postrzegana jako spory sukces, poprawę o 2 punkty postrzegano jako wybitną" (Lewis-Kraus, 2017: 153).



Uwaga 1: Poprawiony system gromadzenia danych Google od 1 stycznia 2016 roku.

Uwaga 2: Nazwę tematu „data science" zmieniono z tłumaczenia Google'a „danologia", aby zachować przyjętą konwencję.

Uwaga 3: Google udostępnia dane z dokładnością do 1, więc wartość zmiennej Popularność równą < 1 zmieniono na 0,5, aby zachować ilościowy charakter zmiennej.

Rysunek 1.1. Popularność tematów „big data", „uczenie maszynowe", „data science", „sztuczna inteligencja" w wyszukiwarce Google: A – popularność dla czterech tematów; B – popularność z wyłączeniem tematu „sztuczna inteligencja"

Źródło: opracowanie własne za pomocą GNU R na podstawie: A – Google Trends, 2020a, B – Google Trends, 2020b.

Przedstawione na rysunku 1.1 linie obrazują popularność jako względną częstość wyszukiwania tematów (nie jest ważna wielkość liter) w wyszukiwarce Google na całym świecie. Na osi y przedstawiono czas. Wykorzystano dane w interwale miesiąca od początku stycznia 2008 do końca września 2020 roku.

1.1.4. Data science

Data science to dziedzina określana jako integrująca i wdrażająca w praktyce wszystkie omawiane powyżej: big data, AI i ML. Nie znaczy to jednak, że każdy człowiek zajmujący się DS wykorzystuje np. big data, czy każdy pracujący z big data lub ML zajmuje się DS. W ogóle z tak ujętą integracją czy wdrażaniem AI, big data i ML przez DS niekoniecznie zgadzają się uczestnicy świata społecznego DS w Polsce.

Osoby zajmujące się DS nazywane są data scientists. Termin ten zazwyczaj nie jest tłumaczony na język polski, choć bywa stosowany jako inżynier danych, analityk, mistrz danych (Przanowski, 2014; Wikipedia, b.d.). W Polsce wśród uczestników świata społecznego DS termin bywa odmieniany zgodnie z regułami języka polskiego, np. „kilku data scientistów", „data scientistka", „data scientiści", „[kogo?] data scientisty", „[jestem] data scientistą/scientistką", co wiem z badań własnych. Pojęcia DS po raz pierwszy użyto w 1974 roku:

> Data science to nauka zajmująca się danymi po ich pozyskaniu i zapisaniu, podczas gdy problematyka związku danych z tym, co mają one reprezentować, jest pozostawiona innym polom i dziedzinom nauki (Naur, 1974: 30).

Pierwsze użycie pojęcia we współczesnym sensie datowane jest na rok 2001 (Donoho, 2015: 13; Andrus, Cook, Sood, 2017: 1). Chodzi o pracę *Data Science: An Action Plan for Expanding the Technical Areas of the field of Statistics* (Cleveland, 2001). Autor – William S. Cleveland – wieloletni współpracownik Johna Tuckeya w laboratorium badawczym firmy Bell i profesor statystyki, postuluje rozszerzenie technicznych obszarów statystyki uniwersyteckiej tak, by lepiej wspierać analityków danych (*data analyst*) pracujących dla rządowych lub komercyjnych podmiotów (Donoho, 2015: 13). Proponuje sześć obszarów aktywności dla uniwersytetów, sugerując rozkład pracy: badania multidyscyplinarne (25%), metody i modele analizy danych (20%), problemy obliczeniowe dla danych (15%), nauczanie (15%), ewaluacja narzędzi analitycznych (5%), teoria statystyczna (20%). David Donoho (2015: 13) wskazuje, że czternaście lat po publikacji artykułu Clevelanda zdecydowana większość statystyki akademickiej w USA wciąż poświęca 100% swojej pracy na teorię. Donoho powołuje się na pracę Leo Breimana (2001), również praktyka analiz w biznesie i profesora statystyki, wskazującego na istnienie dwóch kultur modelowania statystycznego. Jedna z nich, kultura modelowania generatywnego (*generative modeling*), ma być nastawiona na wnioskowanie (*inference*) o naturze zjawiska stojącego za związkiem statystycznym pomiędzy zmiennymi; podejście to ma przejawiać 98% statystyków akademickich. Z kolei kultura modelowania algorytmicznego ma na celu wyłącznie uzyskanie wyniku o jak najwyższym dopasowaniu do danych, przy całkowitym przemilczeniu rzeczywistych zjawisk przyrodniczych czy społecznych „stojących za" danymi. To podejście jest charakterystyczne dla 2% statystyków akademickich, ale także dla informatyków

(*computer scientists*) i praktyków statystyki w biznesie (*industrial statisticians*). Tu stosuje się ML, nazywane epicentrum kultury modelowania algorytmicznego (Donoho, 2015). Donoho dodaje, że ML od strony akademickiej zazwyczaj zajmują się wydziały informatyczne (*computer science departments*), a nie statystyczne (Donoho, 2015: 14–15). W pracy *Big Data: rewolucja...* autorzy powtarzają, że w erze big data obowiązuje pytanie „co?" zamiast pytania „dlaczego?" (por. Cukier, Mayer-Schönberger, 2014: 19, 35, 185). Pytanie „co?" jest zatem charakterystyczne dla kultury modelowania algorytmicznego (a więc raczej także dla DS), a „dlaczego?" dla modelowania generatywnego.

Do spopularyzowania terminu „data science" przyczyniły się dwa artykuły prasowe: *What is Data Science?* (Loukides, 2010) i *Data Scientist: The Sexiest Job of the 21st Century* (Davenport, Patil, 2012). Wskazują je prace poświęcone historii DS (Donoho, 2015; Andrus, Cook, Sood, 2017; Cao, 2017a). W artykułach tych przytoczono słowa Hala Variana, głównego ekonomisty Google'a, z wywiadu dla „The New York Times":

> W ciągu najbliższej dekady seksownym zajęciem będzie statystyka. Ludzie myślą, że żartuję, ale kto odgadłby, że seksownym zajęciem lat 90. będzie inżynieria informatyczna? (Lohr, 2009).

W obu artykułach mowa o możliwościach, jakie dają big data i DS, oraz o rosnącym w niesamowitym tempie zapotrzebowaniu na specjalistów z dziedziny DS. Choć Varian używa słowa „statystyk", to Mike Loukides oraz Thomas H. Davenport i D.J. Patij posługują się już terminem „data scientist". W drugim tekście przeinaczono oryginalną wypowiedź (poza nazwą profesji zmieniono najbliższą dekadę na całe stulecie), a dodatkowo wrażenie na czytelniku wywiera zawarcie wszystkich kluczowych słów w tytule *Data Scientist: The Sexiest Job of the 21st Century*.

W wyszukiwarce Google temat „data science" był w analizowanym okresie zawsze mniej popularny niż „sztuczna inteligencja" i „uczenie maszynowe" (rys. 1.1). W grudniu 2018 roku jego popularność przekroczyła popularność tematu „big data" i utrzymuje się na wyższym niż dla niego poziomie. Średnio „data science" to najmniej popularny temat spośród omawianych (Google Trends, 2020a; 2020b). Niemniej od czerwca 2012 roku jego popularność w wyszukiwarce Google ma trend rosnący. Jest silnie dodatnio skorelowana[7] z popularnością tematu „uczenie maszynowe".

Data science jest często definiowane jako nowa dyscyplina nauki i praktyki, rodząca się na styku bardziej dojrzałych pól. Calvin Andrus, Jon Cook i Suresh Sood używają metafory „dziecka" powstałego z „rodziców": ogólnej metodologii nauk, inżynierii danych, inżynierii oprogramowania, statystyki, wizualizacji danych

7 Dla danych o trzech opisywanych tematach (Google Trends, 2020b) współczynnik korelacji popularności tematów „data science" i „uczenie maszynowe" metodą rang Spearmana wynosi $r_s = 0{,}96$.

i hakerstwa (jak podkreślają „w pozytywnym tego słowa znaczeniu") (Andrus, Cook, Sood, 2017). Ponieważ książka jest wprowadzeniem do DS, autorzy zaznaczają złożoność tej nowej dyscypliny, mówiąc, że zapoznanie się z podręcznikiem nie uczyni ze studenta eksperta, bo „prawdziwe DS jest bardziej odpowiednio nauczane na poziomie magisterskim i doktoranckim". Data science powstaje „w świecie big data" i wymaga m.in. wiedzy i umiejętności z dziedziny rozproszonego przetwarzania petabajtowych zbiorów danych pochodzących z baz nierelacyjnych, znajomości ML i zaawansowanej statystyki (Andrus, Cook, Sood, 2017). Zdobycie umiejętności potrzebnych do pracy na stanowisku data scientisty wymagać może lat doświadczenia zawodowego, już z dyplomem magistra (Donoho, 2017: 8–9). Są jednak autorzy sugerujący, że pracować w DS mogą z powodzeniem zarówno absolwenci informatyki (*computer science major*) – tych nazwano hakerami, jak i statystycy (*statistics major*) – których nazwano piszącymi skrypty, oraz absolwenci kierunków managerskich (MBA) – użytkownicy oprogramowania statystycznego (Barlow, 2013: 8–9).

Niejednoznaczność połączona z coraz większym rozgłosem i entuzjazmem wokół terminów „big data" i „data science" prowadzić może m.in. do problemów w zatrudnianiu data scientistów (Harris, Murphy, Vaisman, 2013). Trudno dobrać właściwą osobę do projektu, kiedy pracodawca i kandydaci nadają terminom różne znaczenia. Te popularne terminy nazwano brutalnie „maszynką do mielenia modnych powiedzonek" (*buzzword meat grinder*). Podano przykład rozmowy kwalifikacyjnej, na której rekruter opisał, jakiego pracownika szuka:

> Chcę BOGA! Chcę gwiazdora rocka w programowaniu, który stworzył najbardziej wyszukane algorytmy uczenia maszynowego, zbudował platformę big data do rozproszonych obliczeń i założył swoją firmę! (Harris, Murphy, Vaisman, 2013: 3–6).

Jerzy Surma we wstępie do swojej krytycznej pracy ujął rzecz następująco:

> […] ta książka to rodzaj repulsji na to, co głoszą media głównego nurtu na temat Big Data. […] Metody te niejednokrotnie są znane i stosowane od lat, ale w narracji nieświadomych dziennikarzy są emanacją inteligencji porównywalnej z ludzką. **Ten bezkrytyczny opis metod sztucznej inteligencji, irytująca antropomorfizacja, przy jednoczesnym braku podstawowej wiedzy na temat funkcjonowania tego typu algorytmów, przekroczyły – moim zdaniem – granicę akceptowalnej kuriozalności** [podkr. R.Ż.] (Surma, 2017: 7–8).

Na antropomorfizowanie technologii, szczególnie tych jakoby inteligentnych i szczególnie przez nietechnicznych laików, data scientiści również zwracają uwagę (o czym piszę w części *Magia i majsterkowanie*). W literaturze wskazuje się także na technomorfizm antropologiczny. Jest to zjawisko niejako przeciwne antropomorfizowaniu technologii. Polega ono na upodabnianiu i dopasowywaniu się człowieka do technologii (Myoo, 2013: 172–174). Przejawem tego zjawiska mogą być np. stosowane potocznie metafory mózgu ludzkiego jako komputera, resetowania się, programowania się na coś (por. Postman, 2004). Technomorfizm nie jest

zastosowany w tej książce jako kategoria analityczna. W świecie DS mówi się co prawda o tym, że nauka programowania w Pythonie lub R zmienia sposób myślenia o pracy z danymi (w porównaniu do używania GUI). Niemniej posługując się ramą teoretyczną społecznych światów, w dalszej części książki opisałem używanie Pythona/R m.in. w kategoriach „odpowiednich" narzędzi wyznaczających granice świata społecznego DS oraz jako wyraz wartości świata DS.

Spory, niejednoznaczność i wysoki poziom emocji towarzyszą próbom zdefiniowania, czym są DS i data scientist. Szeroko powtarzana jest definicja (Conway, 2010), w której za pomocą diagramu Venna dyscyplinę ukazano na przecięciu wiedzy matematyczno-statystycznej, umiejętności hakerskich i doświadczenia dziedzinowego (rys. 1.2).

Rysunek 1.2. Definiowanie data science – popularny diagram: umiejętności hakerskie – *hacking skills*, wiedza matematyczno-statystyczna – *math & statistics knowledge*, doświadczenie dziedzinowe – *substantive expertise*, uczenie maszynowe – *machine learning*, tradycyjne badania – *traditional research*, strefa zagrożenia – *danger zone*

Źródło: opracowanie własne na podstawie Conway, 2010.

Prezentowany na rysunku 1.2 diagram powstał w 2010 roku. Sześć lat później na portalu Kdnuggets[8] opublikowano utrzymany w żartobliwym tonie artykuł *Battle of the Data Science Venn Diagrams* (Taylor, 2016), w którym omawiany jest ukazany powyżej, najwcześniejszy diagram Conwaya oraz dwanaście (sic!) innych,

8 Nagradzanym w branżowych konkursach „wiodącym portalu internetowym poświęconym analityce biznesowej, big data, data mining, data science i uczeniu maszynowemu" (Kdnuggets, b.d.).

powstałych w kolejnych latach. Autor we wstępie zaznacza, że postanowił uniknąć kontrowersji związanych z terminem „data scientist" i nazywa siebie „grotołazem danych (*data spelunker*)". W kolejnym zdaniu pisze: „data minerzy[9] i tak wyszli już z mody (*data 'miners' are out of vogue anyway*) (Taylor, 2016).

W mniej żartobliwym tonie utrzymane są definicje DS (14 różnych) zebrane przez portal bigdata-madesimple.com (Baitu, 2014). Mówi się m.in. o DS jako o „rzadkiej hybrydzie", a o data scientiście jako „Kolumbie i Colombo w jednym – wygłodniałym badaczu i sceptycznym detektywie". Większość zebranych tam wypowiedzi stosuje podobną metaforykę. Pojawiają się definicje uznające data scientistów za „ludzi renesansu". Często powtarzana jest wypowiedź Josha Willsa: „data scientist to osoba lepsza w statystyce niż każdy programista, i lepsza w programowaniu od każdego statystyka" (Delapenha, 2017). Popularny żart definicyjny mówi, że „data scientist to analityk danych, który mieszka w Kalifornii" (Baitu, 2014; Jarvis, 2014). W podobnym tonie utrzymanych jest wiele wypowiedzi, np.: „data scientist to statystyk pracujący na MacBooku" (Big Data Borat, 2013), „data science to domena ludzi, którzy decydują się wydrukować 'data scientist' na swoich wizytówkach i dostać podwyżkę" (Taylor, 2016). Dyskusje toczą się wokół tego, czy DS to tylko nowa nazwa (*rebranding*) statystyki, czy też, że statystyka jest tylko mało znaczącą częścią DS (Hyndman, 2014; Donoho, 2015: 5–6). Josh Bloom, profesor astronomii i pracownik Berkeley Institute for Data Science, zasłynął wypowiedzią: „pierwszą zasadą data science jest: nie pytaj o definicję data science" (Azam, 2014). W dzisiejszych czasach analiza danych jest tak popularna, że właściwie każdy badacz zajmuje się DS. Sam termin nie ma zatem żadnego konkretnego znaczenia i oznacza „różne rzeczy dla różnych ludzi" (Azam, 2014). Pytanie „czym jest data science?" jest „motywem przewodnim, centralnym zagadnieniem i mantrą" pracy matematyczki Cathy O'Neil i statystyczki Rachel Schutt (2015: 303). Choć ogrom szumu medialnego wokół DS spowodował ich sceptyczne podejście, to ostatecznie doszły one do następującego wniosku:

> Data science[10] jest zbiorem najlepszych praktyk stosowanych w firmach technologicznych operujących w szerokiej przestrzeni problemów, które można rozwiązywać za pomocą danych, i być może niekiedy zasługuje na miano nauki. Mimo to czasami nie jest niczym więcej niż szumnym określeniem, którego powinniśmy się wystrzegać i którego dodawania na siłę powinniśmy unikać (O'Neil, Schutt, 2015: 305).

To „rozwiązywanie problemów" jest dla mnie szczególnie ważną kategorią analizy jakościowej, zarówno w odniesieniu do działania podstawowego DS jako świata społecznego, jak i wartości tego świata.

Choć może nie istnieje coś takiego jak DS, to niewątpliwie istnieje praca w DS i wielki popyt na data scientistów (O'Neil, Schutt, 2015: 26). Praca/posada (*job*)

9 Określenie „data mining" jest powszechnie używane w podręcznikach analizy danych i ML (por. Larose, 2005; 2012).

10 Przetłumaczone jako „badanie danych" – zmieniono, by zachować konwencję.

data scientisty została uznana przez firmę Glassdoor za najlepszą spośród około 1700 badanych posad trzykrotnie – w latach 2016, 2017 i 2018 (Piatetsky, 2018a). O gwałtownie rosnącym popycie na pracowników zespołów DS pisze się wiele. Mowa np. o tym, że kariera w DS staje się bardziej atrakcyjna niż prawo lub medycyna (Burtch, 2014); o bardzo silnym w tej branży rynku pracownika – doradza się pracodawcom strategie rekrutacyjne (PwC, 2015); o prognozie popytu na data scientistów dla USA w 2018 roku rzędu 440–490 tys. stanowisk przy podaży około 300 tys. osób (tym samym popyt przewyższa podaż o około 50%, tzn. prawie jedna trzecia oferowanych stanowisk ma być w 2018 nieobsadzona) (Manyika i in., 2011); o najszybszym tempie wzrostu liczby stanowisk pracy na portalu LinkedIn dla posad „Machine Learning Engineer" (9,8 razy więcej osób z tym tytułem w 2017 niż w 2012 roku) i „Data Scientist" (6,5 razy) (Economic Graph Team, 2017).

Popyt na pracowników pociąga za sobą popyt na edukację w dziedzinie DS. Na pierwszej stronie wyników wyszukiwarki Google znajdują się np.: lista 368 kursów online wraz z ich recenzjami – stronę śledzi 127 100 osób (Class Central, 2018), lista najlepszych kursów w tej dziedzinie – 52 pozycje (LearnDataSci, 2018), lista najlepszych bootcampów[11] DS – 38 pozycji (switchup, 2018), lista 23 najlepszych studiów magisterskich, gdzie nie zabrakło znakomitych uczelni, jak University of California-Berkeley i Carnegie Mellon University (masterindatascience, 2018).

1.2. Cel pracy

Celem głównym pracy jest **opis etnograficzny polskiego społecznego świata data science**.

Pięć celów szczegółowych wyznaczono na podstawie wybranych pojęć i procesów zaczerpniętych z teorii społecznych światów, w tym społecznych światów/aren Adele E. Clarke:

1. Rekonstrukcja tworzonych przez uczestników społecznego świata DS definicji działania podstawowego.
2. Zrozumienie roli technologii umożliwiających wykonanie działania podstawowego w odniesieniu do zachodzących w społecznym świecie procesów – przede wszystkim:
   a) wyznaczania granic społecznego świata;
   b) legitymizacji i zaświadczania o autentyczności;
   c) segmentacji – profesjonalizacji.

11 Bardzo intensywny kurs, stacjonarny, zaoczny lub online; słowo przeszło do biznesu z wojska i oznacza dosłownie „obóz rekrutów".

3. Rozpoznanie świata wartości uczestników, które rozumiane są jako „punkty orientacyjne, pozwalające dokonać oceny działania, a jednocześnie wskazujące, jak ma ono wyglądać i w jakich granicach przebiegać" (Kacperczyk, 2016: 44).
4. Określenie i opis głównych aren sporów.
5. Uchwycenie pełnej sytuacji badania.

Powyższe cele staną się zrozumiałe dla osób niezaznajomionych z koncepcjami społecznych światów po lekturze trzeciego rozdziału książki.

Samo „środowisko" DS wykonuje różnego rodzaju akty samowiedzy – od wspomnień, polemik i wywiadów (Loukides, 2010; Barlow, 2013; Provost, Fawcett, 2013; Brooks, 2014; Gutierrez, 2014; Chang, 2015; O'Neil, Schutt, 2015; Cao, 2017a; 2017b; Drejewicz, 2017; O'Neil, 2017; Alekseichenko, 2018), po zestawienia, sondaże i analizy danych (Barlow, 2013; Fox, Leanage, 2016; *Is it a job offer for a Data Scientist?*, 2016; King, Magoulas, 2016; CrowdFlower, 2017; Delapenha, 2017; Kaggle, 2017b; Onalytica, 2017; 2018; Prokulski, 2017; Sopyła, 2017; Pascnau, Patraucean, Precup, 2018; Rexer Analytics, 2018; Zhang, Muller, Wang, 2020). Dotychczas nie powstała jednak praca socjologiczna, w której podjęto by zadanie szerokiego oglądu tzw. środowiska/społeczności DS oczami jego uczestników. Niniejsza praca zamierza wypełnić tę lukę.

Rozdział 2
Data science w literaturze naukowej

2.1. Nauki ścisłe i techniczne

W niniejszym rozdziale prezentuję przegląd literatury z zakresu nauk ścisłych i technicznych związanych z DS. Koncentruję się na autorach polskich. Niemniej z uwagi na cel rozprawy, tj. opis świata społecznego, przedstawiam znaczące dla niego czasopisma, książki i postaci z całego globu.

Jednym z najbardziej rozpoznawalnych w badanym świecie społecznym naukowców zajmujących się zagadnieniami związanymi z DS, jest Andrew Yan-Tak Ng, znany jako Andrew Ng. Słyszałem o nim podczas prowadzonych wywiadów i obserwacji, widziałem jego wizerunek lub cytaty z jego wypowiedzi podczas prezentacji w różnych materiałach powstających w ramach społecznego świata, a także na tapetach pulpitów komputerowych osób uczestniczących w warsztatach czy w postaci memów internetowych. Ng jest profesorem Stanford University, dyrektorem Stanford Artificial Intelligence Lab (Ng, 2018a; b.d.). Jego badania dotyczą m.in. rozpoznawania tematów w tekście metodą *Latent Dirichlet Allocation* – LDA (Ng, Zheng, Jordan, 2001; Blei, Ng, Jordan, 2003), rozpoznawania obrazu i dźwięku za pomocą głębokich sieci neuronowych (*deep learning* – DL) (Lee i in., 2009; Saxena, Sun, Ng, 2009) czy wsparcia diagnostyki medycznej z użyciem DL (Kallenberg i in., 2016; Rajpurkar i in., 2018). Jest też współzałożycielem Google Brain Deep Learning Project, gdzie pracował w latach 2011–2012. Google Brain stworzyło wówczas sieć neuronową rozpoznającą obiekty na filmach z należącego do Google'a portalu YouTube (Le i in., 2011). Rozpoznawanie odbywało się na zasadzie uczenia nienadzorowanego – pracowano na danych nieoznaczonych, wynikiem działania modelu było zatem nadanie etykiet obiektom podobnym. W prasie określano odkrycie w następujący sposób: „komputer sam nauczył się rozpoznawać koty" (Clarke, 2012). Rozpoznawalność Ng jest spowodowana przez prowadzony przez niego kurs ML dla początkujących na platformie Coursera. W roku 2012, po sukcesach otwartych kursów internetowych oferowanych przez Stanford w 2011 roku, Ng założył platformę Coursera i opublikował bezpłatny, ogólnodostępny kurs. Wzorował się m.in. na zasadzie działania najpopularniejszego forum

internetowego dla osób programujących Stack Overflow, którego sam często używał (Ng, Widom, b.d.).

Na Stanford University działał również Peter Norvig. Był pracownikiem naukowym University of Southern California i University of California, Berkeley, badaczem i szefem działów badawczych w Sun Microsystems Labs oraz NASA (Norvig, 2018). Od 2001 roku pracuje w Google, gdzie obecnie jest szefem działu badań (Spector, Norvig, Petrov, 2012; Google, 2018). Norvig zajmował się analizą tekstów (Norvig, 1987; Halevy, Norvig, Pereira, 2009; Michel i in., 2011). Już w 1994 roku opublikował książkę *Artificial Intelligence: A Modern Approach* (Russel, Norvig, 2009). Jest ona „wiodącym podręcznikiem w dziedzinie sztucznej inteligencji", używanym na ponad 1,3 tys. uczelni w 110 krajach i czwartą najczęściej cytowaną pracą XXI wieku (Russel, Norvig, 2009). Norvig wszedł też w polemikę z Noamem Chomskym na temat dwóch podejść do matematycznego modelowania zjawisk (Norvig, 2012), o czym jako pierwszy wypowiadał się Leo Breiman (2001). Norvig zaangażowany jest w założony przez Petera Diamandisa i Raya Kurzweila Uniwersytet Osobliwości, organizację think-tank, która zgodnie z wizją Kurzweila przygotowuje ludzi na przyszłość poza granicami ludzkiej biologii (Singularity University, 2018).

Yann LeCun to badacz znany z uwagi na wkład w rozwój sieci neuronowych oraz pracę dla Facebooka. Pracował w laboratorium Geoffreya Hintona, nazywanego ojcem chrzestnym DL. Obecnie pracuje na New York University (Singularity University, 2018), tam w roku 2013 założył NYU Center for Data Science (2013). Został pierwszym dyrektorem badań nad AI w firmie Facebook (LeCun, b.d.), później pełnił podobne funkcje (Facebook, 2018). LeCun jest autorem rozwiązania powszechnie wykorzystywanego do rozpoznawania obiektów na zdjęciach – konwolucyjnych sieci neuronowych (*convolutional neural networks* – CNN). Jego pierwszy artykuł dotyczył problemu rozpoznawania pisanych odręcznie kodów pocztowych (LeCun i in., 1989). Pracując w AT&T Research, LeCun stworzył CNN o nazwie LeNet-5, służącą do rozpoznawania pisma odręcznego i wydruków (LeCun i in., 1998). Przeprowadzony przez Andrew Ng wywiad z LeCunem dotyczący podstaw CNN zawarto w materiałach kursowych specjalizacji DL na portalu Coursera (Ng, 2018b).

Geoffrey Hinton karierę akademicką zaczynał w latach siedemdziesiątych ubiegłego wieku z dyplomem studiów pierwszego stopnia z psychologii eksperymentalnej Uniwersytetu Cambridge i doktoratem z AI obronionym na Uniwersytecie w Edynburgu. Pod koniec lat osiemdziesiątych pracował na Uniwersytecie w Toronto, a w jego zespole był m.in. Yann LeCun. Z tą uczelnią Hinton związany jest zawodowo nieprzerwanie od 2001 roku (CS Department Toronto University, b.d.). Miano ojca chrzestnego DL (Lee, 2016; Mannes, 2017; Somers, 2017; Sorensen, 2017) przyniosła mu praca poświęcona zastosowaniu w wielowarstwowych sieciach neuronowych metody propagacji wstecznej (*backpropagation*) (Rumelhart, Hinton, Williams, 1986). Metoda umożliwia korektę wag sieci neuronowej w sposób optymalizacyjny, co nazywane jest uczeniem sieci neuronowej.

Rozważmy przykład uczenia nadzorowanego, z danymi zaetykietowanymi (przyjmijmy, że są to fotografie psów i kotów wraz z etykietami „pies" albo „kot"). Wagi wszystkich sztucznych neuronów w sieci są losowane i sprawdzany jest błąd (koty zaklasyfikowane jako psy lub psy jako koty), następnie wagi są korygowane i ponownie sprawdzany jest błąd, procedura jest powtarzana wielokrotnie. Kierunek korekcji wag – ich zmniejszanie lub zwiększanie – w celu zminimalizowania błędu ustalany jest za pomocą reguły największego spadku (*gradient descent*) (Larose, 2005: 135–138).

Hinton i współpracownicy spopularyzowali użycie propagacji wstecznej w sieciach neuronowych. Prace tego rodzaju podejmowano już od początku lat sześćdziesiątych (Schmidhuber, 2015). Podstawa propagacji wstecznej, czyli reguła największego spadku, to mająca korzenie w siedemnastowiecznych pracach Gottfrieda Wilhelma Leibniza i osiemnastowiecznej koncepcji równań Eluera-Lagrange'a metoda redukowania błędów (Schmidhuber, 2015).

Hadley Wickham jest naukowcem znanym wśród data scientistów, ponieważ „buduje narzędzia (obliczeniowe i poznawcze), które czynią data science łatwiejszym, szybszym i bardziej zabawnym" (Wickham, 2018a). Pracuje na stanowisku profesora statystyki w University of Auckland, Stanford University i Rice University (Wickham, 2018a). Narzędzia Wickhama są przeznaczone do pracy w języku R, a on sam pełni funkcję Chief Scientist w RStudio (2018a). Dwa najpopularniejsze w DS języki programowania[1] to R i Python (DeZyre, 2015; Kromme, 2017; Piatetsky, 2017; Hale, 2018; Sharp Sight, 2018; Silaparasetty, 2018; Muenchen, 2019). Język R powstał właśnie na uniwersytecie w Auckland (Ihaka, Gentleman, 1996), a Wickham uczył się go od twórców podczas swoich studiów magisterskich na tej uczelni (Kopf, 2015). RStudio to firma oferująca oprogramowanie umożliwiające użytkownikom efektywniejszą pracę z językiem R (RStudio, 2018a). Najpopularniejszym produktem firmy jest IDE RStudio Desktop (RStudio Team, 2016). Narzędzie to jest edytorem programistycznym dostępnym bezpłatnie do większości zastosowań (Biecek, 2015a). Istnieją wersje płatne o rozszerzonej funkcjonalności (RStudio, 2018b). RStudio, w tym Wickham, tworzą w języku R narzędzia w formie pakietów (*packages*). Pakiet jest zbiorem funkcji wykonujących określone zadania: „funkcjami możemy analizować dane, rysować dane, odsłuchiwać dane, wykonywać obliczenia, wysyłać maile, robić najróżniejsze rzeczy" (Biecek, 2015a). Domyślnie zainstalowany w R pakiet „base" zawiera funkcję `seq()`, za pomocą której można stworzyć sekwencję liczb. Sekwencję stu jeden liczb od 0 do 10 z krokiem 0,1 utworzy i przypisze do zmiennej *sekwencja* następująca instrukcja (Biecek, 2015a):

```
sekwencja <- seq(0, 10, 0.1)[2]
```

1 O technologiach świata społecznego data science traktuje osobna część tej książki.

2 Fragmenty kodu wyeksportowano do edytora tekstu Libre Office Writer za pomocą R Markdown (Xie, Allaire, Grolemund, 2018; Allaire i in., 2019).

Wickham jest autorem rodziny pakietów „tidyverse" (Wickham, 2017). Jak sam wskazuje, pakiety te pozwalają na wykonanie 80% pracy w każdym projekcie DS, choć zazwyczaj nie będą wystarczające do realizacji całości zadania (Wickham, Grolemund, 2017). Pakiety R Wickhama mają łącznie największą liczbę pobrań – ponad 113 milionów – i największą łącznie liczbę gwiazdek (*star*), będących wyrazem uznania na portalu z repozytorium kodu GitHub (Mortimer, 2018). Przy tym jego pakiet do wizualizacji danych „ggplot2" znajduje się na drugim miejscu w rankingu pobrań pakietów, a na pierwszym w rankingu gwiazdek (Mortimer, 2018).

Wickham jest zarówno autorem podręczników do nauki R i DS (Wickham, 2014a; Wickham, Grolemund, 2017), jak i prac naukowych, w których prezentuje podstawy teoretyczne i praktyczne tworzonych przez siebie narzędzi (Wickham, 2010; 2014c). Publikuje także polemiki, traktujące o pracy data scientistów czy o branży DS (Wickham, 2014b; 2018d; Bryan, Wickham, 2017). Podręcznik napisany wraz z Garrettem Grolemundem został przetłumaczony m.in. na język polski (Wickham, Grolemund, 2018), ale jedynie wersja anglojęzyczna dostępna jest w całości bezpłatnie na http://r4ds.had.co.nz/ (dostęp: 3.06.2020). Wickham ma w środowisku użytkowników języka R (co nie jest tożsame ze środowiskiem DS) status autorytetu i celebryty. W mediach społecznościowych pojawił się np. żart w postaci przeróbki katolickiej modlitwy „Ojcze nasz" na modlitwę do Hadleya Wickhama, odmawianą przed uruchomieniem kodu w R (Votta, 2018), a także fikcyjna okładka magazynu lifestyle'owego, na której obok różnych żartów dotyczących wizualizacji danych widnieje m.in. zapowiedź wywiadu z Wickhamem, który „ujawnia, dlaczego *nigdy* nie zmieni domyślnego stylu grafiki w ggplot"[3].

W języku Python nie ma jednego twórcy narzędzi, którego dorobek byłby porównywalny z dorobkiem Wickhama. Poniżej przedstawiam twórców pięciu najpopularniejszych pakietów Pythona do zastosowań w DS: „NumPy", „SciPy", „pandas", „matplotlib" i „scikit-learn" (Bobriakov, 2017; Li, Paczuski, 2017; Satyaseel, 2018).

Pakiety „NumPy" i „SciPy" umożliwiają pracę z danymi tabelarycznymi i obliczenia naukowe lub inżynieryjne. Jednym z inicjatorów powstania obu pakietów jest Travis Oliphant. Zaczynał on pracę z językiem Python w 1998 roku jako doktorant w dziedzinie obrazowania biomedycznego. Na potrzeby swojej pracy zaczął przepisywać do Pythona metody dostępne w językach C lub Fortran (Oliphant, 2006; 2007). Publikował w czasopismach medycznych (Oliphant i in., 2001), obecnie nie jest aktywny naukowo. Zajmuje się rozwojem związanego z Pythonem otwartego oprogramowania, jest jednym z założycieli Anacondy, dyrektorem Python Software Foundation i dyrektorem NumFOCUS (Anaconda, 2018b). Anaconda jest bezpłatną i wolną platformą do programowania i zarządzania pakietami oraz środowiskiem pracy DS dla języka Python, choć możliwa jest praca w R (Anaconda, 2018a). NumFOCUS jest organizacją non profit, której

3 https://twitter.com/W_R_Chase/status/1155212225621221376 (dostęp: 12.08.2019).

motto brzmi „Otwarty Kod = Lepsza Nauka". Oliphant założył ją w 2011 roku. Jej misją jest wspieranie wolnego oprogramowania na potrzeby nauki. Anaconda jest sponsorem NumFOCUS od 2012 roku (Anaconda, b.d.; NumFOCUS, 2018). NumFOCUS prowadzi międzynarodowy program edukacyjny o nazwie PyData, dotyczący pracy z danymi. Pod nazwą „SciPy" kryje się też ekosystem pakietów do zastosowań naukowych, w skład którego wchodzą m.in. wymienione już „SciPy" i „NumPy", a także „pandas", „scikit-learn" i „Matplotlib" (SciPy.org, b.d.).

„Matplotlib" to pakiet służący do wizualizacji danych. Prace nad jego rozwojem są sponsorowane przez NumFOCUS (Matplotlib, 2018). Głównym autorem był nieżyjący już John Hunter (2007), doktor neurobiologii (Princeton Alumni Weekly, 2012). Hunter, podobnie jak Oliphant, zaczął pracę nad pakietem z uwagi na ograniczenia dostępnych narzędzi, takich jak MATLAB (Hunter, Droettboom, 2012).

Pakiet „pandas" dostarcza wydajnych i prostych w użyciu struktur danych oraz narzędzi do analizy danych w języku Python (The pandas project, 2018). Rozwój „pandas" także finansowany jest przez NumFOCUS (The pandas project, 2018). Autorem narzędzia jest Wes McKinney (2010; 2011). Jest on z wykształcenia matematykiem, podjął studia doktoranckie w dziedzinie statystyki na Duke University (McKinney, 2018). Opublikował podręcznik do analizy danych z użyciem pakietów „pandas" i „NumPy" (McKinney, 2012). Współpracował z omawianym wyżej Wickhamem z RStudio, tworząc dla DS narzędzia niezależne od języka programowania – można używać ich w Pythonie, R i kilku innych (Dar, 2018; McKinney, 2018).

Pakiet „scikit-learn" to narzędzie do modelowania ML:

> Scikit-learn to moduł Pythona integrujący szeroką gamę najnowocześniejszych algorytmów uczenia maszynowego dla problemów nadzorowanych i nienadzorowanych w średniej skali. Pakiet ten koncentruje się na wprowadzaniu uczenia maszynowego dla osób niebędących specjalistami, używających wysokopoziomowego języka programowania ogólnego zastosowania[4] (Pedregosa i in., 2011: 2826).

Fabian Pedregosa jest osobą, która z udziałem finansowym INRIA – francuskiego instytutu badawczego nauk cyfrowych – doprowadziła do publikacji pakietu (scikit-learn, 2018). Brał on udział także w rozwijaniu omawianego powyżej pakietu „SciPy" (Pedregosa, 2018). Pedregosa uzyskał tytuł doktora w dziedzinie informatyki, pracując nad neuroobrazowaniem (Pedregosa, 2015). W pracy naukowej zajmował się zbliżonymi zagadnieniami (Pedregosa, Bach, Gramfort, 2014; Pedregosa, Leblond, Lacoste-Julien, 2017). Jest pracownikiem działu badań w Google (Pedregosa, 2018).

Dorobek polskich naukowców związanych z DS zaczynam od przedstawienia prac Ryszarda Tadeusiewicza. Jest on profesorem nauk technicznych związanym

4 Wysokopoziomowy język programowania ogólnego zastosowania to m.in. język Python.

z Akademią Górniczo-Hutniczą w Krakowie. Zajmuje się systemami wizyjnymi robotów przemysłowych, systemami sensorycznymi; sieciami neuronowymi i biocybernetyką (Tadeusiewicz, 2018). Jego pierwsza praca poświęcona sieciom neuronowym ukazała się pod koniec lat siedemdziesiątych (Tadeusiewicz, Mikrut, 1978). Tadeusiewicz w mediach prezentuje stanowisko optymistyczne wobec rozwoju tzw. AI lub big data, a wizję przejęcia władzy nad ludźmi przez roboty uważa za nierealną (Burda, 2018: 67).

Naukowcem, którego w przeprowadzonych w ramach niniejszej pracy wywiadach swobodnych z uczestnikami świata społecznego DS określano „polskim Hadleyem Wickhamem", jest Przemysław Biecek. Jest on profesorem Politechniki Warszawskiej i Uniwersytetu Warszawskiego. Współpracował m.in. z Rossem Ihaką, jednym z dwóch autorów języka R (Ihaka, Gentleman, 1996; Biecek, 2018b). Jest autorem narzędzi – pakietów dla języka R (Biecek, Kosiński, 2017; Caro, Biecek, 2017; Biecek, 2018c; Molnar, 2018; Staniak, Biecek, 2018), w tym pakietu przygotowanego na potrzeby prowadzonego przez siebie kursu MOOC „Pogromcy Danych" (Biecek, 2015c). Jest to jedyny bezpłatny polskojęzyczny kurs MOOC z podstaw analizy danych w języku R (w ramach badań etnograficznych ukończyłem obie jego części) (Biecek, 2015a; 2015b). Biecek jest autorem podręczników do nauki analizy danych z programem R. W 2008 roku wydał pierwszą w Polsce książkę poświęconą w całości nauce R (Biecek, 2017). Pracuje także w komercyjnym DS jako Principal Data Scientist w firmie Samsung SRPOL (Warsaw AI, 2020). Jest aktywnym członkiem społeczności DS w Polsce, uczestniczy w wydarzeniach tego świata społecznego i w trakcie badań terenowych miałem okazję go spotkać.

Istnieją prace polskich autorów o charakterze technicznym czy podręcznikowym. W starszych publikacjach mówi się o eksploracji danych czy data miningu (Lasek, 2002; 2007; Morzy, 2013; Osowski, 2013; Szupiluk, 2013), sieciach neuronowych, ML lub AI (Witkowska, 2002; Krawiec, 2003; Białko, 2005; Osowski, 2006), w nowszych o analizie danych lub DS, z użyciem języków R lub Python (Gągolewski, 2016; Gągolewski, Bartoszuk, Cena, 2016; Szeliga, 2017).

W zbiorze esejów *Granice sztucznej inteligencji* autorzy zdecydowanie obstają przy stanowisku, że AI inteligenta na miarę człowieka jest i będzie niemożliwa (Szumakowicz, 2000). Profesor statystyki Mirosław Szreder wielokrotnie wypowiedział się o społecznych konsekwencjach stosowania zaawansowanej analityki dużych zbiorów danych zarówno w periodykach naukowych (Szreder, 2015a; 2018a), jak i prasie (Szreder, 2015b; 2016; 2018b). Konsekwencje społeczne omawianych zjawisk są także głównym tematem przeglądowej pracy Jerzego Surmy (2017). Autor zajmuje się AI od końca lat osiemdziesiątych i jego zdaniem:

> […] świadomość zagrożeń z tym [big data – przyp. R.Ż.] związanych jest tak niska, że w poczuciu elementarnej obywatelskiej odpowiedzialności postanowiłem wprost napisać o wszystkich znanych mi reperkusjach tego zjawiska w odniesieniu do publicznie dostępnych badań naukowych (Surma, 2017: 7).

Istnieje polska praca Grażyny Trzpiot, definiująca DS w odniesieniu do statystyki: „Data science jest procesem zamieniania/przekształcania danych w konkretne działania" (Trzpiot, 2017: 9). Statystyka ma zastosowanie w jednym z czterech obszarów działań DS. Są to: „uzyskaj" – pozyskanie danych; „przygotuj" – przygotowanie danych; „analizuj" – tutaj wymagana jest znajomość statystyki i innych metod analizy danych; „działaj" – wdrożenie wyników (Trzpiot, 2017: 11–15).

2.2. Nauki społeczne i humanistyczne

Prace naukowe związane z DS dzieli się na cztery kategorie, wyróżnione z uwagi na problematykę:

1) epistemologiczną i metodologiczną;
2) włączającą charakterystyczne dla DS metody do badań własnych;
3) dotyczącą konsekwencji społecznych stosowania DS/big data/AI (nazewnictwo jest niespójne);
4) badającą środowisko ludzi zajmujących się DS jako wycinek rzeczywistości społecznej.

2.2.1. Krytyka epistemologiczno-metodologiczna

Działalność związana z DS zawiera elementy działalności poznawczej i jest działalnością paranaukową. Zgadzam się ze stanowiskiem Danah Boyd i Kate Crawford:

> Era Big Data właśnie się zaczęła, ale już teraz ważne jest, byśmy zaczęli kwestionować założenia, wartości i obciążenia (*biases*) tej nowej fali działalności badawczej. Jako naukowcy zajmujemy się wytwarzaniem wiedzy, zatem takie dociekania są najważniejszym elementem naszej działalności (Boyd, Crawford, 2011: 13).

Omawiane problemy podejmowano w obszarach socjologii i filozofii wiedzy w odpowiedzi na popularne wśród wczesnych entuzjastów big data podejście „liczby/dane mówią same za siebie". W jednej z pierwszych w tej kategorii prac socjologicznych (Boyd, Crawford, 2012) przywołano koronny przykład tego podejścia:

> Jest to [epoka petabajtów (*The Petabyte Age*) – przyp. R.Ż.] świat, w którym ogromne ilości danych i matematyki stosowanej zastępują każde inne narzędzie, które można wykorzystać. Każdą teorię ludzkiego zachowania, od językoznawstwa po socjologię. Zapomnijmy o taksonomii, ontologii i psychologii. Kto wie, dlaczego ludzie robią to, co robią? Chodzi o to, że oni to robią i możemy śledzić i mierzyć to z niespotykaną wiernością. **Przy wystarczającej ilości danych liczby mówią same za siebie** [podkr. R.Ż.] (Anderson, 2008).

W badaniach społecznych poświęconych big data w latach 2011–2015 ten tekst Chrisa Andersona był najczęściej cytowaną pracą (Youtie, Porter, Huang, 2016: 8). Boyd i Crawford nie zgadzają się z tezą o mówiących za siebie liczbach, podkreślając, że takie podejście uznaje za nieważną każdą teorię z każdej dziedziny nauki. To „aroganckie" podejście entuzjastów big data marginalizuje inne sposoby zdobywania wiedzy (Boyd, Crawford, 2012: 666). Susan Halford i Mike Savage (2017: 11) piszą: „liczby nie mówią same za siebie. To my umożliwiamy te wypowiedzi – za pomocą metod, które stosujemy i interpretacji, które czynimy". Entuzjaści mówią o końcu epoki ekspertów dziedzinowych dysponujących teoretyczną wiedzą substancjalną, ponieważ w myśl charakterystycznej dla epoki petabajtów zasady „co? zamiast dlaczego?" interesujące są jedynie relacje między zmiennymi, a nie teorie wyjaśniające te relacje (Żulicki, 2017). Wiedza substancjalna jest jakoby więcej niż zbędna: ona przeszkadza.

> Najważniejszym efektem *big data* będzie to, że decyzje oparte na danych ulepszą jakość ocen dokonywanych przez ludzi lub sprawią, że całkowicie stracą one na znaczeniu. […] Ekspert czy specjalista w danej dziedzinie straci część swojego znaczenia na rzeczy statystyka czy analityka danych, którzy są nieskrępowani starymi metodami rozwiązywania problemów i pozwalają przemawiać danym (Cukier, Mayer-Schönberger, 2014: 185).

Koniec ery ekspertów ilustruje żart inżynierów zajmujących się tłumaczeniem maszynowym w Microsoft: „jakość przekładu rośnie za każdym razem, gdy z ich zespołu odejdzie jeden lingwista" (Cukier, Mayer-Schönberger, 2014: 186). Tym samym DS jest przedstawiana jako nowa nauka uniwersalna (niem. *Leitwissenschaft*) (Rath, 2018). Boyd i Crawford, nie zgadzając się z podobnymi tezami, wskazują, że big data/DS zmieniają sposób, w jaki myśli się o badaniach w ogóle, i jest to zmiana o charakterze epistemologicznym i etycznym jednocześnie. Przeramowaniu podlegają kluczowe pytania o to, jaka jest konstytucja wiedzy, proces badania, obcowanie badacza z informacjami czy natura rzeczywistości. Akty pomiaru – czyli tutaj zapisywania dużych ilości danych w czasie rzeczywistym – kształtują mierzony świat (Boyd, Crawford, 2012: 665).

Big data/DS mają wartość poznawczą dla wielu dziedzin nauki, ale nie bez odwołania do teorii dziedzinowych. Mówienie o zmianie paradygmatów, końcu ery ekspertów i tym podobne entuzjastyczne stwierdzenia są bezzasadne, a także niebezpieczne w sensie poznawczym. Rob Kitchin (2014: 4) wskazuje cztery epistemologiczne obietnice big data:

1) pozwala ująć całość problemu (w myśl strategii N = all) i zapewnić pełne wsparcie decyzjom;
2) do uzyskania wartościowych wyników niepotrzebne są przedmiotowe teorie ani stawianie hipotez;
3) ponieważ dane mówią same za siebie, nieobciążone zbędą teorią, to wyniki analiz są znaczące i zgodne z prawdą o świecie;
4) wyniki analiz, niezależnie od przedmiotu, może interpretować każdy mający rozeznanie w statystyce.

Kitchin przedstawia dwie potencjalne ścieżki rozwoju współczesnej nauki. Tę bardziej pożądaną nazywa nauką opartą na danych – ma być to przeformułowanie sposobu jej uprawiania, w którym zmieszają się abdukcja, dedukcja i indukcja. Niepokojącą ścieżką rozwoju nauki może być skrajny empiryzm, czyli „dane mówią same za siebie" jako zasada główna (Kitchin, 2014: 10). Wystosowana przez Kitchina krytyka takiego podejścia (Kitchin 2014: 5–6) odnosi się do wymienionych czterech epistemologicznych obietnic.

Big data/DS to inny, ale nie zawsze lepszy sposób dostarczania kwantyfikowalnej wiedzy niż badania ilościowe wywodzące się z tradycji nauk przyrodniczych (np. sondaże). W tym kontekście mówi się o dwóch kulturach modelowania zjawisk czy dwóch podejściach do nauki w ogóle (Manhart, 1996; Breiman, 2001; Ceri, 2018). Naukę opartą na teoriach przeciwstawia się nauce opartej na danych. Pierwsza ma polegać na wyjściu od teorii, postawieniu hipotez i testowaniu ich za pomocą danych zebranych w badaniu, druga zaś na analizie danych bez wcześniejszych pytań i założeń (przy czym jedno podejście nie wyklucza drugiego) (Ceri, 2018). Tym samym DS to inne narzędzia służące poznawaniu świata, lecz nie „tylko" narzędzia, a „aż" narzędzia. Przypomina to słowa, którymi zreferowano jedną z tez Marshalla McLuhana: „kształtujemy nasze narzędzia, a następnie one kształtują nas" (Culkin, 1967: 70). Zgadzam się z tezą, iż nowe narzędzia zmieniają sposób myślenia o badaniach społecznych i o teoriach społecznych (Boyd, Crawford, 2011: 3). Ten nowy sposób myślenia określano mianem „uwodzicielskiego pseudopozytywizmu" (Dalton, Taylor, Thatcher, 2016: 6) czy łagodniej „wywodzącego się z idei pozytywistycznej" (Kitchin, 2014: 7), a także „odnowionego naturalizmu", w którym społeczeństwo traktowane jest jak fenomeny przyrody (Törnberg, Törnberg, 2018: 2). Przyjmujący naturalistyczny sposób myślenia badacze zapominają o dyskusjach toczonych w naukach społecznych we wczesnej fazie rozwoju internetu. Mówiono wówczas, że cyfryzacja życia jest nieodłączną częścią procesu przejścia od modernizmu do postmodernizmu, co czyni życie społeczne bardziej otwartym, płynnym czy mozaikowym. Tym samym cyfryzacja może powodować, że życie społeczne jest coraz mniej, a nie coraz bardziej uchwytne, szczególnie za pomocą ilościowej analizy dużych ilości danych (Törnberg, Törnberg, 2018). Kazimierz Krzysztofek nazywa to zjawisko „fragteracją", czyli równoczesnym rozpadaniem się, ale i integrowaniem świata społecznego (Krzysztofek, 2010). Przedstawiłem te epistemologiczno-metodologiczne pułapki myślowe w osobnej pracy (Żulicki, 2019).

2.2.2. Data science jako strategia badań w naukach społecznych i humanistyce

Na pograniczu humanistyki, nauk społecznych i DS wyrastają dziedziny aplikujące charakterystyczne dla DS metody do badań własnych: humanistyka cyfrowa (*digital humanities*) i obliczeniowe nauki społeczne (*computational social sciences*).

Mówi się o zwrocie obliczeniowym (*computational turn*) dotyczącym myślenia i prowadzenia badań we współczesnej nauce. Boyd i Crawford porównują ten zwrot do zmiany społecznej, która zaczęła się od wprowadzenia produkcji masowej w fabrykach Forda (Boyd, Crawford, 2011: 3; 2012: 665).

Surma (2017: 23) manifestem obliczeniowych nauk społecznych nazywa pracę Davida Lazera z zespołem (2009). Nowe możliwości zbierania i analizy danych dotyczących ludzi są już przedmiotem pracy badawczej firm (wymieniono Google i Yahoo) i agencji rządowych (wskazano U.S. NSA[5]), tak więc jeżeli akademia nie chce pozostawić innym podmiotom tego rodzaju prac – musi działać (Lazer i in., 2009: 721). Artykuł rozpoczyna się słowami „żyjemy w sieci" (Lazer i in., 2009: 721). Badania nad sieciami społecznymi (*social network analysis* – SNA) w naukach społecznych sięgają lat trzydziestych ubiegłego wieku. Linton C. Freeman wskazuje, że początki metodycznego stosowania metafory sieci w naukach społecznych to socjometria Jacoba L. Moreno i Helen Jennings (Freeman, 2014: 26).

Badania nastawione na analizę sieci kontynuowali m.in. Robert K. Merton czy Claude Lévi-Strauss. W latach dziewięćdziesiątych zagadnieniem zainteresowali się m.in. Albert-Laszlo Barábasi (fizyk) i Réka Albert (biolożka). Barábasi jest jednym z autorów manifestu obliczeniowych nauk społecznych (Lazer i in., 2009). Jego badanie jest przywołane w entuzjastycznej pracy *Big Data. Rewolucja*…, gdzie autorzy uzasadniają podejście „*N* = all" (Cukier, Mayer-Schönberger, 2014: 49–50). Sam autor powołuje się zarówno na Leonarda Eulera, twórcę podstaw teorii grafów[6], jak i Stanleya Milgrama z problemem „małego świata" (Barabási, 2002).

Ośrodkiem rozwoju obliczeniowych nauk społecznych w początkach XXI w. był Massachusetts Institute of Technology. Jedne z pierwszych prac koncentrowały się na analizach behawioralnych z uwzględnieniem sieci społecznych i lokalizacji osób, początkowo dzięki urządzeniu do noszenia na ciele (Choudhury, Pentl, Pentland, 2002), później z użyciem telefonów komórkowych i sensorów (Eagle, 2005; Eagle, Pentland, 2006).

W Polsce pionierem podobnych badań jest Dominik Batorski, który w ramach rozprawy doktorskiej analizował dane z komunikatora Gadu Gadu z użyciem GNU R (Batorski, 2004). Jest on organizatorem spotkań, tzw. meetupów, grupy Data Science Warsaw (Data Science Warsaw, 2018b), był członkiem rady programowej „największego wydarzenia data science w Polsce" (Data Science Warsaw, 2018a), współorganizował lub promował liczne imprezy tego świata społecznego (ChallengeRocket.com, 2018; Kaggle, 2018b; PyData, 2018; WDI18, 2018). Jest jednym z założycieli i Chief Scientist w firmie Sotrender, oferującej narzędzia do analizy marketingu w mediach społecznościowych (Sotrender, 2018).

---

5 Warto przypomnieć, że w 2013 roku Edward Snowden ujawnił tajne dokumenty dotyczące zbierania cyfrowych danych o ludziach właśnie przez NSA (van Dijck, 2014).

6 Pojęcia wierzchołków (*nodes*) i krawędzi (*edges*) z matematycznej teorii grafów pojawiają się także w kontekście architektur baz danych czy systemów rozpraszania obliczeń.

Cyfrowa humanistyka (*digital humanities*) posługuje się często metodami ilościowej analizy tekstów, w tym ilościową eksploracją tekstów (*text mining*). Teksty są jednym z typów danych nieustrukturyzowanych. Zdaniem niektórych autorów big data to właśnie dane zarówno duże, jak i nieustrukturyzowane. Mogą to być np. wpisy na portalach społecznościowych, komentarze na forach internetowych, treści stron WWW czy blogów, zdigitalizowane bądź wydawane wyłącznie w formie cyfrowej artykuły, książki, dokumenty. *Text mining* to komputerowa analiza tekstów traktowanych jako dane. Używany jest także termin „obliczeniowa analiza tekstu" (*computational text analysis*) (O'Connor, Bamman, Smith, 2011). *Text mining* to analiza ilościowa umożliwiająca pomiary tekstów/dokumentów. Służy ona odkrywaniu wzorów użycia słów i wzorów powiązań między słowami lub dokumentami (O'Connor, Bamman, Smith, 2011). Metody text miningu wywodzą się z eksploracji danych ustrukturyzowanych i dziedziny NLP (*natural language processing*), której przedmiotem jest przetwarzanie informacji zawartej w języku naturalnym – np. w celu umożliwienia sterowania urządzeniem za pomocą głosu (Dzieciątko, Spinczyk, 2016). Text mining znajduje zastosowanie np. w identyfikacji słów kluczowych, kategoryzacji dokumentów według wzorca albo grupowaniu bezwzorcowym (wyestymowaniu kategorii), wykrywaniu określonych treści w dokumentach (Dzieciątko, Spinczyk, 2016).

Polscy humaniści i badacze społeczni pracują w różnych obszarach humanistyki cyfrowej czy obliczeniowych nauk społecznych. Radosław Bomba raczej nie korzysta z metod obliczeniowych, ale z pewnością z cyfrowych i prowadzi analizy cyfrowej kultury – jest zarówno badaczem gier komputerowych (Bomba, 2015), jak i propagatorem humanistyki cyfrowej (Radomski, Bomba, 2013; Bomba, 2017). Podobnie można scharakteryzować dorobek Magdaleny Szpunar, która sama siebie nazywa socjolożką internetu, jest zatem „cyfrowa" w sensie przedmiotu badań, a jej prace dotyczą np. nierówności technologicznych, w tym algorytmicznych (Szpunar, 2013; 2018; 2019). Maciej Eder, filolog, zajmuje się ilościową analizą literatury, m.in. do zadań atrybucji autorskiej i stylometrii (Rybicki, Eder, 2011; Eder, 2013; 2015; 2017), jest on autorem pakietu „stylo" w GNU R (Eder, Rybicki, Kestemont, 2016). Był prelegentem na ogólnopolskich konferencjach użytkowników R w 2017 i 2018 roku (Why R? Foundation, 2018). Marek Troszczyński wykorzystuje metody ML i text mining do wykrywania mowy nienawiści na forach internetowych (Troszczyński, Wawer, 2017). Krzysztof Tomanek i Grzegorz Bryda (2015) stosują podobne narzędzia do badania postaw nauczycieli akademickich w opiniach studentów, piszą o metodyce prowadzenia podobnych badań (Tomanek, Bryda, 2017), natomiast Agnieszka Kwiatkowska (2017) używa text miningu do analizy polskich debat parlamentarnych. Metody ilościowej analizy tekstu zostały zauważone przez polskich badaczy z kręgu socjologii jakościowej i są wdrażane do praktyki badawczej obok oprogramowania CAQDAS. Tomanek i Bryda piszą o zastosowaniach text miningu w książce *Metody i techniki odkrywania wiedzy. Narzędzia CAQDAS w procesie analizy danych jakościowych* pod redakcją Jakuba Niedbalskiego (2014). W roku 2016 na XVI zjeździe Polskiego Towarzystwa

Socjologicznego problematyce tej poświęcono panel „Big Data, CAQDAS i nowe technologie w polu socjologii jakościowej" (PTS) z udziałem wyżej wymienionych socjologów. W roku 2017 ukazał się „Przegląd Socjologii Jakościowej" (t. XII, nr 2) pod tytułem „Big Data i CAQDAS w badaniach jakościowych". Artykuł wprowadzający redaktorów tomu zawiera rozważania na temat stosowania metod oraz specyfiki epistemologicznej tego rodzaju analiz (ateoretyczność – zmianę pytania będącego podstawą budowania modeli z „dlaczego?" na „co?" i „dane mówią same za siebie", analizowanie całej populacji zamiast próby) (Brosz, Bryda, Siuda, 2017). W „Studiach Socjologicznych" pojawiły się tematycznie pokrewne teksty pokazujące możliwości korzystania z nowych źródeł danych (Rodak, 2017; Turner, Zieliński, Słomczyński, 2018). Badanie przekazywania informacji między ludźmi za pomocą Twittera realizowane jest przez fizyków z Politechniki Warszawskiej (RENOIR, 2018). Wyniki dotyczą np. mierzenia entropii w dynamice komunikacji międzyludzkiej (Kulisiewicz i in., 2018).

Badania z użyciem metod ilościowych, dotyczące AI, prowadzi Aleksandra Przegalińska. Zajmuje się ona interakcją między człowiekiem a chatbotem (Ciechanowski i in., 2018) czy możliwością mierzenia uwagi człowieka za pomocą ubieralnego gadżetu (Przegalińska i in., 2018). Przegalińska często wypowiada się w mediach i pracach popularnonaukowych o przyszłości ludzi i AI (Przegalińska, 2014; 2015; 2016a; 2016b; 2016c; Brzezińska, Przegalińska 2015), była też jedną z głównych prelegentek na konferencji branżowej DS (PyData, 2018).

W pracach obliczeniowych nauk społecznych widoczny jest entuzjazm wobec stosowanej metodologii i epistemologii. Pojawiają się pretensje do prawdziwości, obiektywności, pewności, słuszności i użyteczności uzyskiwanych wyników. Pisano o „eksploracji rzeczywistości" (Eagle, Pentland, 2006), „sygnałach prawdy" jako „twardych miarach" zachowań społecznych (Pentland, Heibeck, 2008; Arena, Pentland, Price, 2010), analizach o bezprecedensowym zasięgu, głębi i skali (Lazer i in., 2009: 722), inżynierii społecznej dla lepszego świata (Eagle, Greene, 2014), uzyskaniu „punktu widzenia Boga na nas samych" (Pentland, 2009: 80). Barabási – jeden z twórców NSN – wyrażał nadzieję na odkrycie ścisłych, matematycznych praw opisujących i pozwalających prognozować ludzkie zachowanie dzięki analizie obiektywnych danych (Krzysztofek, 2011: 126). Prezentowałem krytyków tego rodzaju pretensji, należy jednak zauważyć, że naukowcy używający metod obliczeniowych w badaniach społecznych także podawali w wątpliwość własne założenia i metody. Pisano o sprzeciwie wobec głosów o końcu teorii i danych mówiących same za siebie (Chang, Kauffman, Kwon, 2014) oraz o pułapce pozornego obiektywizmu (Eder, 2014).

Powstają prace postulujące włączanie big data/DS do arsenału badawczego nauk społecznych, obok innych metod i technik (Jemielniak, 2020). Mówi się o integracji wielkich danych i grubych danych. Autorką określenia „grube dane" (*thick data*) jest Tricia Wang, socjolożka stosująca metody etnograficzne w badaniach marketingowych (Wang, 2013; 2016). Oznacza ono materiały zbierane za pomocą

metod i technik badań jakościowych. Łączenie wglądów, jakie daje analiza danych ilościowych typu big data i analiza materiałów jakościowych, ma prowadzić do osadzenia wyników analizy ilościowej w kontekście, a więc ułatwić ich wyjaśnianie i rozumienie (Bornakke, Due, 2018: 1–4). Podobne badania wykonywano wcześniej, niż wszedł do użycia termin „data science" (Anderson i in., 2009). Powstała monografia o tytule *Ethnography for a data-saturated world*. Przy współpracy data scientistów i badaczy społecznych zawarto w niej porady metodologiczne i praktyczne oraz przykłady badań, ale i próby badań etnograficznych środowiska data scientistów (Knox, Nafus, 2018).

„Bogactwo analityki danych [ilościowych – przyp. R.Ż.] zwiększa, a nie zmniejsza zapotrzebowanie na uzupełnianie ich o wyniki badań etnograficznych" (Jemielniak, 2018: 17). Dariusz Jemielniak w pierwszej w Polsce monografii *Socjologia internetu* (2019) prezentuje liczne metody ilościowe i jakościowe dla tego medium i obszaru badań. Także Kitchin postuluje ostrożne korzystanie z możliwości, jakie daje big data/DS, obok tradycyjnego podejścia do badań ilościowych i jakościowych w naukach społecznych. Mówi on o możliwości przyjęcia nowej, epistemologicznej strategii, która miałaby łączyć w sobie indukcję, dedukcję i abdukcję (Kitchin, 2014: 5–6). Bazując m.in. na jego myśli, proponuję włączanie big data/DS do badań społecznych w ramach triangulacji metod i technik (Żulicki, 2017: 200). Inni autorzy, zainspirowani m.in. omawianym artykułem Kitchina, postulują – podobnie jak on – abdukcyjne podejście, upatrując w nim przyszłość tych nauk (Savage, Halford, 2017: 1145).

Ważnym i słabo zagospodarowanym polem współpracy DS z przedstawicielami nauk społecznych jest wyjaśnianie działania gotowych modeli (Miller, 2019). Nie chodzi o uczenie ludzi teorii stojącej za modelowaniem ani umiejętności modelowania. Chodzi o wyjaśnianie człowiekowi, skąd biorą się konkretne odpowiedzi modelu, ważności zmiennych w modelu itd., szczególnie dla metod o bardzo wysokiej złożoności (jak *deep learning*). Ten obszar badań nazywany jest XAI (*eXplainable AI*). W Polsce zajmuje się nim m.in. Biecek.

Źródłem zwrotu obliczeniowego (*computational turn*) jest nie tylko akademia. Firmy i rządy dążą do wydobywania z danych maksymalnej użyteczności, w postaci np. profilowania reklamy, projektowania produktów, zarządzania ruchem ulicznym czy przeciwdziałania przestępczości (Boyd, Crawford, 2011: 13). Zarówno rządy, biznes, jak i nauka traktują dane jak Świętego Graala, prawdziwe odzwierciedlenie ludzkiego zachowania, zebrane za pomocą przezroczystych termometrów, a więc najbardziej wiarygodne, a co za tym idzie – użyteczne źródło informacji (van Dijck, 2014: 199). To „logika cywilizacji numerycznej" – istnieje to, co jest kwantyfikowalne, a musi być takie przynajmniej po to, aby możliwa była wycena w pieniądzach (Krzysztofek, 2012: 225–226). Inni autorzy mówią o kulturze algorytmicznej (Striphas, 2015). Z powodu nastawienia na użyteczność nazywam w tej pracy DS działalnością paranaukową.

> Myślenie współczesne jest pod przemożnym wpływem nauki, a naukowość – rzeczywista lub pozorowana – traktowana jest jako najwyższa nobilitacja idei. […] Obserwujemy dziś rozrost sektora **paranaukowego** [podkr. R.Ż.], zwłaszcza w tych dziedzinach, w których nauka odpowiada na jakieś palące ludzkie potrzeby i nadzieje (Sztompka, 2007: 298).

To nastawienie na użyteczność niesie wiele konsekwencji etycznych i społecznych, które obok krytyki metodologiczno-epistemologicznej są przedmiotem zainteresowania tzw. krytycznych studiów danych (Dalton, Taylor, Thatcher, 2016).

Istnieje jeden tekst, w którym autorzy postulują łączenie nauk społecznych i DS, ale nie w sensie opisywanej wyżej triangulacji. Chodzi o włączanie do badań i praktyki DS krytyki metodologiczno-epistemologicznej i tej dotyczącej konsekwencji społecznych (Neff i in., 2017). Artykuł przygotowano częściowo na podstawie badań etnograficznych przeprowadzonych w zajmującym się DS środowisku akademickim w USA, jednak nie opisuję tego jako przykładu badań społecznych poświęconych DS, ponieważ tekst ma charakter interwencyjny – jest wezwaniem do działania na rzecz bardziej etycznego DS.

2.2.3. Konsekwencje społeczne data science/AI/big data

Crawford i Boyd[7] oraz związane z nimi instytuty AI Now oraz Data & Society (AI Now Institute, 2018; Data & Society, 2018) są jednymi z najaktywniejszych na polu krytycznych studiów danych. Działalność Boyd i jej grupy została pozytywnie oceniona przez osoby zajmujące się DS w USA (Gutierrez, 2014: 320). Badaczki pracują w Microsoft na stanowiskach dyrektorek badań (Microsoft Research, 2019a; 2019b). Choć krytyka praktyk z zakresu zbierania informacji o ludziach, inwigilacji czy nierówności związanych z technologią jest starsza niż DS, big data czy tzw. AI, to niniejszy podtemat dotyczy krytyki autorstwa Crawford i Boyd oraz osób z nimi współpracujących.

AI Now Institute opublikował kilka raportów poświęconych konsekwencjom społecznym w omawianych dziedzinach. Koncentruję się na raporcie z przełomowego 2018 roku – roku afery Cambridge Analytica i nie tylko. Autorzy raportu posługują się terminem „AI", obejmujacym całą gamę technologii, od bazujących na zaprogramowanych regułach, przez korzystające ze statystyki czy ML, do technik DL. Chodzi o zadania rozpoznawania mowy, tłumaczenia, rozpoznawania obrazów, prognozowania, podejmowania decyzji (Crawford i in., 2018: 44). W centrum skandali, jakie zaszły w AI w 2018 roku, jest pytanie o odpowiedzialność/rozliczalność (*accountability*) (Crawford i in., 2018: 7) – „Kto jest odpowiedzialny za to, że system AI robi człowiekowi krzywdę?". Autorzy posługują się nierzadko publicystyczną retoryką, ale warto przyjrzeć się publikacji z uwagi na przykłady tzw. skandali z systemami AI w roli głównej.

7 Właściwie danah boyd, zgodnie z objaśnieniem na stronie internetowej socjolożki (Boyd, b.d.).

AI Now Institute twierdzi, że w 2018 roku systemy AI gwałtownie rozwijały się w kolejnych dziedzinach życia społecznego, stwarzając ryzyko dla coraz większej liczby ludzi. Choć AI wciąż oferuje warte rozważenia obietnice, to szybkie wdrażanie tych systemów bez odpowiedniej oceny, odpowiedzialności i kontroli może powodować poważne zagrożenia. Według autorów potrzebne są niezwłoczne regulacje prawne systemów AI, sektor po sektorze, ze szczególną uwagą zwróconą na technologie rozpoznawania twarzy i emocji, oraz rygorystyczne badania na potrzeby wsparcia tych regulacji. Pytanie nie brzmi „Czy w systemach AI są obciążenia (*biases*) i czy są one krzywdzące?" – dyskusja jest zamknięta, w wyniku wydarzeń roku 2018 nie ma bowiem żadnych wątpliwości, że tak właśnie się dzieje. Następnym zadaniem jest zaadresowanie tych krzywd, co jest szczególnie pilne z uwagi na skalę działania systemów AI, ich funkcjonowanie centralizujące wiedzę i władzę w rękach nielicznych, oraz powiększającą się nierówność dystrybucji kosztów i benefitów, która tej centralizacji towarzyszy (Crawford i in., 2018: 42). Słowo „bias" jest popularnym określeniem technicznym, także w polskim świecie społecznym DS, a rozwiązania mające adresować ów problem w systemach ML proponowali naukowcy i aktywiści, są one również oferowane komercyjnie – firmę audytującą algorytmy prowadzi np. C. O'Neil (por. Courtland, 2018).

Wraz ze wzrostem wszechobecności, złożoności i skali systemów AI coraz bardziej palącym problemem jest brak znaczącej odpowiedzialności/rozliczalności i nadzoru, na który mają się składać podstawowe zabezpieczenie odpowiedzialności, odpowiedzialność prawna i właściwy proces. Wskazano pięć obszarów problemowych (Crawford i in., 2018: 7):

1) rosnąca przepaść odpowiedzialności (*accountability gap*) w AI, gdzie faworyzowani są ludzie tworzący i wdrażający te technologie, a obciążani ci najbardziej dotykani ich skutkami;
2) używanie AI do zwiększania i wzmacniania inwigilacji, szczególnie w połączeniu z maszynowym rozpoznawaniem twarzy i emocji, powiększa możliwości scentralizowanej kontroli i opresji;
3) zwiększanie rządowego użycia automatycznych systemów decyzyjnych, które bezpośrednio wpływają na jednostki ludzkie i społeczności, przy braku ustalonych struktur odpowiedzialności/rozliczalności (*accountability*);
4) nieregulowane i niemonitorowane eksperymenty z użyciem AI na populacjach ludzkich;
5) ograniczenia rozwiązań technologicznych w problemach dotyczących uczciwości, obciążeń (*bias*) i dyskryminacji ludzi przez systemy oparte na AI.

W raporcie Crawford i zespołu (2018) wymieniono liczne przykłady negatywnych konsekwencji społeczny wdrażania systemów AI. Powtarzane w wielu tekstach i dobrze znane także w społecznym świecie DS w Polsce obejmują one m.in. chiński system krajowej oceny obywateli, opisywany przez zachodnie media jako ziszczenie wizji Orwella o roku 1984 (Associated Press, 2015; Creemers, 2015; Hatton, 2015; Denyer, 2016; Nguyen, 2016; Wong, Chin, 2016; Botsman, 2017; Gostkiewicz, 2017; Kędzierski, 2018), model przewidywania prawdopodobieństwa

recydywy COMPAS, wdrożony do amerykańskiego wymiaru sprawiedliwości, a „uprzedzony" wobec osób o innym niż biały kolorze skóry (Northpointe Inc., 2012; Angwin i in., 2016; Larson i in., 2016; Chouldechova, 2017; Norén, 2018; Taddeo, Floridi, 2018; Richardson, Schultz, Crawford, 2019; Saltz, 2019), otagowanie portretu czarnoskórej pary przez aplikację Google Photos etykietą „goryle" (Grush, 2015; Burrell, 2016), aferę Cambridge Analytica dotyczącą nieuprawnionego wykorzystania danych z Facebooka do targetowania reklamy politycznej w kampaniach Donalda Trumpa i probrexit oraz podobne, krytykowane jako zagrażające demokracji (Ćwiklak, 2017; Lapowsky, 2017; 2019; Schwartz, 2017; Szymielewicz, 2017; Batorski, 2018; Cadwalladr, 2018; Cadwalladr, Graham-Harrison, 2018; Czubkowska, 2018; Deptuła, 2018; Dwoskin, Harwell, Timberg, 2018; Franceschi-Bicchierai, 2018; Gersz, 2018; Grewal, 2018; Isaak, Hanna, 2018; Mac, 2018; Mims, 2018; Obem, 2018; Wąsowski, 2018; Wong, 2018), model selekcji wdrożony do rekrutacji w firmie Amazon, odrzucający aplikacje kobiet (van den Broek, 2019; Crawford, West, Whittaker, 2019), śmiertelne potrącenia pieszych przez tzw. samochody autonomiczne firm Uber i Tesla (Awad i in., 2018; 2020; Crawford i in., 2018: 23).

Mało znanym przykładem polskim jest nieprzejrzysty i wadliwie skonstruowany model profilowania bezrobotnych w urzędach pracy, wycofany z użycia wyrokiem Trybunału Konstytucyjnego, m.in. staraniami Fundacji Panoptykon (Sztandar-Sztanderska, Kotnarowski, Zieleńska, 2021).

Te same i podobne przykłady oraz zbliżoną w zakresie i treści krytykę negatywnych konsekwencji społecznych wdrażania omawianych technologii odnaleźć można w wielu popularnych publikacjach, pisanych nie tylko przez przedstawicieli nauk społecznych (O'Neil, 2017; Surma, 2017; Galloway, 2018; Harari, 2018; Vaidhyanathan, 2018; Lorenzi, Berrebi, 2019; Sumpter, 2019; Crawford, 2021).

Na wyższym poziomie abstrakcji przedmiotowa krytyka i rozważania w naukach społecznych przebiegały m.in. od problematyki sprawczości jednostek w konfrontacji z technologiami (Iwasiński, 2017; Krzysztofek, 2017; 2018; Kowalski, Biele, Krzysztofek, 2019), przez przemiany postrzegania prywatności w związku z tzw. danetyzacją (Ożóg, 2009; 2018; Dopierała, 2013; van Dijck, 2014; Iwasiński, 2016), krytykę nowych form kapitalizmu (Zuboff, 2015; 2019) czy zarzuty o kolonialny charakter big data/DS/AI (Mohamed, Png, Isaac, 2020).

W raporcie AI Now Institute nie ma przykładów pozytywnych konsekwencji rozwoju big data/DS/AI. Stwierdza się jedynie, że systemy bazujące na AI „wciąż mają wiele do zaoferowania" (Crawford i in., 2018: 42). Na potencjał pozytywnych konsekwencji AI wskazują Mariarosaria Taddeo i Luciano Floridi, jednak ich praca jest apelem o etyczność w projektowaniu, użyciu i regulacjach prawnych sztucznej inteligencji, a nie *stricte* analizą konsekwencji. Wyrażają oni pragmatycznie pojęte korzyści: obniżanie kosztów, zmniejszanie ryzyka, poprawę zwięzłości i niezawodności oraz umożliwienie nowych sposobów rozwiązania złożonych problemów. Takie korzyści mogą jakoby następować pod warunkiem wykonywania

nastawionego na etyczność skoordynowanego wysiłku społeczeństwa obywatelskiego, polityków, biznesu i akademii (Taddeo, Floridi, 2018). Nie znalazłem również innych prac z dziedziny nauk społecznych, które poświęcone byłyby pozytywnym konsekwencjom. W popkulturze obrazy zagłady ludzkości spowodowanej przez AI zdecydowanie dominują nad innymi wizjami relacji człowieka ze sztuczną inteligencją (Knapik, 2018: 11). W naukach społecznych i humanistyce jest, moim zdaniem, podobnie. Może dominuje następujące podejście:

> Ponieważ na ogół korporacje i przedsiębiorcy, którzy przodują w technicznej rewolucji, sami naturalnie wychwalają własne wytwory, zadanie bicia na alarm i wyjaśniania wszelkich możliwych czarnych scenariuszy przypada w udziale socjologom, filozofom i historykom – takim, jak ja (Harari, 2018: 10).

Wiele pozytywnych konsekwencji społecznych widocznych jest w pracach entuzjastów rozwoju omawianych dziedzin. Mówi się o możliwości uzyskania obiektywnej wiedzy, która pomoże rozwiązać najbardziej palące problemy ludzkości. Zauważalna jest nadzieja podmiotów tworzących, wdrażających i kupujących systemy oparte na AI na pragmatycznie pojęte pozytywne konsekwencje, a właściwie korzyści: zwiększenie swojej skuteczności, efektywności, skali, szybkości działania, redukcji kosztów, wygody. W analizie kodeksów etycznych w dziedzinach zajmujących się AI wskazano, że dominuje tam myślenie o ogromnym potencjale sprawczym. Wyraża się ono w przekonaniu, że zarówno pozytywny, jak i negatywny wpływ AI jest powszechnym, uniwersalnym zmartwieniem (Greene, Hoffman, Stark, 2019).

Z moich badań wynika, że w świecie społecznym DS podzielane jest takie podejście. Rozwój technologiczny uważa się za nieunikniony, jest on imperatywem, dotknie każdej dziedziny życia, a coraz większe znaczenie będzie miała AI. Technologie, w tym tzw. AI czy – technicznie rzecz ujmując – ML, postrzegane są raczej jako neutralne narzędzia w ludzkich rękach. Może to akcentować ludzką odpowiedzialność za cel używania technologii:

> Technologia jest **niewinna** [podkr. oryg.]. To data scientist ustawia dane wejściowe i zadaje modelowi zmienną celu. Pojawią się wzory, a niektóre z nich mogą być dla wielu ludzi krzywdzące. Musimy być świadomi ostatecznego celu, jak w każdej innej technologii (Casas, 2018).

2.2.4. Data science jako podmiot badań

Powstały jedynie cztery jakościowo zorientowane artykuły dotyczące DS jako badanego wycinka rzeczywistości społecznej. Choć krytyka epistemologiczna, metodologiczna czy dotycząca konsekwencji społecznych rozwoju technologii związanych z DS zaczęła się około roku 2010 i wciąż trwa (Desai i in., 2022), to do 2017 roku nie znalazłem żadnych opublikowanych badań empirycznych skoncentrowanych na środowisku ludzi zajmujących się DS.

Mowa o czterech tekstach:

1) na problemie traktowania algorytmów jako fetyszy koncentruje się tekst bazujący na częściowo ustrukturyzowanych wywiadach z osobami zawodowo zajmującymi się wizją komputerową (*computer vision*) w USA i Azji oraz na trzyletnich badaniach etnograficznych osób należących do społeczności zajmującej się monitorowaniem i poprawą parametrów własnego ciała (*quantified self* – QS) (Thomas, Nafus, Sherman, 2018);
2) inny tekst mówi o praktykach odróżniania DS od statystyki publicznej, której DS ma być odłamem, oraz włączaniu praktyk DS do statystyki publicznej; bazowano w nim na badaniach etnograficznych urzędów i organizacji statystyki publicznej w kilku krajach europejskich (Grommé, Ruppert, Cakici, 2018);
3) w tej samej książce znajduje się najbardziej interesujący mnie tekst poświęcony stawaniu się prawdziwym data scientistą, przygotowany na podstawie badań etnograficznych środowiska data scientistów w Rosji, związanych z nowo powstałą jednostką DS na jednej z uczelni wyższych, we współpracy z firmą Yandex (Lowrie, 2018);
4) na podstawie tych samych badań Ian Lowrie przygotował też tekst bliższy socjologii nauki i techniki, poświęcony relacjom osób zajmujących się DS z algorytmami i tworzeniu tzw. algorytmicznej racjonalności (Lowrie, 2017).

W pracach, które tutaj przedstawiam, deklarowano podejście etnograficzne. Niemniej nigdzie nie znalazły się dokładniejsze wyjaśnienia dotyczące metodologii ani podstawowej ramy teoretycznej. W żadnej nie ma odniesień do teorii społecznych światów, niniejsza praca może być zatem pierwszym zastosowaniem tej ramy do badania DS.

Suzanne L. Thomas, Dawn Nafus i Jamie Sherman (2018) podkreślają asymetrię pomiędzy osobami tworzącymi algorytmy (u nich są to specjaliści wizji komputerowej) a użytkującymi algorytmy (społeczność QS). Mowa o asymetriach społecznych, kulturowych oraz ekonomicznych, różna jest bowiem rola twórców i użytkowników wobec algorytmów i danych jako zmaterializowanych artefaktów (czyli tego, za co można otrzymać pieniądze) (Thomas, Nafus, Sherman, 2018: 9). Asymetrie występują także po stronie twórców całych systemów zawierających algorytmy – pomiędzy specjalistami tworzącymi same algorytmy a twórcami oprogramowania, „obudowy" dla algorytmów, a także pomiędzy osobami zajmującymi się rutynowymi zadaniami przygotowania danych do dalszych czynności a specjalistami tworzącymi algorytmy (Thomas, Nafus, Sherman, 2018: 5–6). Chociaż określenie „twórcy algorytmów" odnosi się jedynie do luźno traktowanego podzbioru tego, co nazywam światem społecznym DS, to zgadzam się z odróżnieniem osób zajmujących się DS od informatyków.

Autorzy nazywają algorytmy fetyszami, ponieważ ludzie (zarówno twórcy, jak i użytkownicy) przyznają algorytmom możliwości niewynikające ani z podstaw matematycznych, ani z zapisu w kodzie. Algorytmiczna obietnica jest znacznie większa niż ich możliwości. Ten sprawczy, demoniczno-boski status algorytmów

przekłada się na praktyki wysokiego wyceniania i doceniania pracy twórców algorytmów, przy niedostrzeganiu pracy będących jakoby w opozycji użytkowników (Thomas, Nafus, Sherman, 2018: 9). Ta opozycja jest fałszywa, ponieważ algorytmy działają i są ulepszane dzięki danym tzw. użytkowników końcowych.

W drugim z tekstów znajdują się licznie odwołania do Pierre'a Bourdieu. Tworzenie się figury data scientisty opisane jest jako tworzenie się pola profesji. Powstawanie tego pola Francisca Grommé, Evelyn Ruppert i Baki Cakici wiążą ze zmianami sposobów oceny metod, technologii, ekspertyzy, umiejętności, wykształcenia i doświadczenia, szczególnie w porównaniu do pola statystyki publicznej. Trwa walka o ramowanie (*framing*) dyskusji – „Czy DS zagraża istnieniu statystyki publicznej?". Wytworzenie się nazw „data science" i „big data" autorzy uważają za formę przemocy symbolicznej, demonstrującej współzawodnictwo metod określania prawdy (Grommé, Ruppert, Cakici, 2018: 37–38).

Statystycy poświęcają wiele wysiłku na definiowanie profesji data scientistów i swojej relacji do nich, co prowadzi do redefiniowania profesji statystyka i obiektów jej działania, czyli statystyki publicznej/oficjalnej. Grommé, Ruppert i Cakici mówią o podobnych zagadnieniach, np. w wymiarze edukacji, habitusie statystyków i data scientistów, ale różnych w sensie kapitału kulturowego. Wskazano, że data scientiści pracują z mniej uporządkowanymi danymi niż statystycy (np. danymi z mediów społecznościowych), używają raczej metod algorytmicznych niż statystycznych, bardziej cenią wizualizację danych i poświęcają jej więcej uwagi. W komunikacji wyników data scientiści mówią o wartości biznesowej, eksperymentowaniu oraz pracy metodą prób i błędów. Statystycy koncentrują się na odpowiedzialności zawodowej, ponieważ dostarczają „oficjalnych" wyników dla władzy i opinii publicznej. Istnieje jednak trend polegający na włączaniu do statystyki publicznej źródeł danych, tzw. big data, metod algorytmicznych i narzędzi charakterystycznych dla DS. Autorzy uważają, że jest to także redefiniowanie profesji statystyka publicznego, szczególnie poprzez włączanie elementów habitusu DS do habitusu statystyka (Grommé, Ruppert, Cakici, 2018: 45–50).

Wśród statystyków publicznych podkreślana jest rola źródeł danych, zwanych big data. Trwają spory, czy i w jakim zakresie te źródła danych mogą wzbogacić lub zastąpić spisy powszechne bądź sondaże. Moim zdaniem Grommé z zespołem przejęli tę perspektywę od uczestników swoich badań. Doprowadziło ich to do stwierdzenia, że data scientiści to profesja uważana za realizującą potencjał big data, czyli nowych źródeł danych (Grommé, Ruppert, Cakici, 2018: 33).

Lowrie, w odróżnieniu od wyżej wymienionych autorów, sytuuje swoje badanie w Moskwie (Lowrie, 2018: 63) jako konkretnym czasie, miejscu i kulturze. Dużo w jego tekście mowy o matematyce. Jest ona uznawana za podstawę dla innych umiejętności data scientistów, podstawę niezmienną (w odróżnieniu od konkretnych rozwiązań technologicznych), za to, co uczy ścisłości myślenia i właściwego rozwiązywania problemów niezależnie od dziedziny, za uniwersalny język porozumiewania się data scientistów. Przygotowanie matematyczne oraz znajomość najnowszych technologii do pracy z danymi są konieczne, by zajmować się DS, ale

to nie czyni prawdziwego data scientisty. Prawdziwy data scientista to człowiek bystry, który ma dobre podejście do rozwiązywania problemów (Lowrie, 2018: 63). Innym kryterium bycia prawdziwym data scientistą jest rozwiązywanie prawdziwych problemów – to odróżnia data scientistów od matematyków i informatyków (Lowrie, 2018: 71–72). Niemniej prawdziwy data scientista może zarówno pracować komercyjnie, jak i akademicko (Lowrie, 2018: 79).

Wyniki prezentowane w omawianym tekście są częściowo zbieżne z wynikami moich badań, m.in. w kwestii przynależności do świata społecznego DS zarówno uczestników pracujących komercyjnie, jak i naukowo. Lowrie zwraca uwagę na uznawanie uczenia się przez całe życie za element określający prawdziwego data scientistę. Ja proponuję ujęcie tego problemu w kategoriach samorozwoju jako wartości badanego świata. Zapoznałem się z tekstami Lowriego (i innymi tutaj omawianymi) już po zakończeniu badań i analiz własnych.

Inny tekst tego autora bazuje na powyższej tezie o braku zasadniczych różnic między data scientistami w komercji i akademii. Zbiorowa praktyka intelektualna data scientistów tworzy spójną formę racjonalności algorytmicznej, nieredukowalną do konglomeratów praktyk lub skrawków teorii zapożyczonych z matematyki bądź informatyki. Ta racjonalność ma być związana z pojawieniem się infrastruktury dużych zbiorów danych i głęboko zakorzeniona w „obliczeniowej nowoczesności" (Lowrie, 2017: 1). Oddzielenie nauki od technologii w DS nie jest możliwe, bo algorytmy można badać tylko pośrednio, jako część systemów technologicznych. Algorytmy, w odróżnieniu od innych części matematyki, nie są badane i rozwijane za pomocą dowodów. Można badać je tylko pośrednio, za pomocą miar efektywności, i do tego muszą one być zaimplementowane w języku programowania, na hardwarze z odpowiednią mocą obliczeniową oraz uruchomione na zbiorze danych. Algorytm bez konkretnego zapisu w postaci kodu w języku programowania i bez konkretnego zbioru danych, na którym jest walidowany, pozostaje neutralny (*inert*). Koncentracja na redukowaniu kosztów i uzyskaniu wyniku prowadzi do pozornie poprawnej konkluzji, że DS to dziedzina techniki czy inżynierii, a nie nauki. Lowrie, powołując się na Jean-Françoisa Lyotarda, wskazuje, że w inżynierii dominuje kryterium efektywności, w nauce zaś kryterium prawdy. Ma to pasować do DS w tym sensie, że działanie jest w DS oceniane jako dobre, kiedy daje lepszy wyniki w sensie utylitarnym. Nie ma mowy o ocenianiu wyniku w kategoriach prawdy. Data science zmieniło technologiczne kryterium efektywności w standard epistemologiczny, w służbie nowej formy badań naukowych nad algorytmami. Połączenie nauki i inżynierii pod hasłem logiki efektywności jest jakoby najsilniej widoczne w biznesowych zastosowaniach DS. Optymalizacja działania algorytmu to optymalizacja biznesu – zmniejszenie odpływu klientów czy zwiększenie wartości koszyków zakupowych (Lowrie, 2017: 3–9).

Za cenne w powyższym tekście uważam zwrócenie uwagi na kategorię efektywności, którą uznaję za centralną wartość w badanym świecie społecznym. Niemniej nie zgadzam się z ujęciem Lowriego w tym wymiarze, że traktuje on DS jako coś homogenicznego i mówi o podejściu data scientistów jako takich do kolejnych

interesujących go zagadnień. Każde z tych zagadnień może być areną sporu subświatów w DS i światów przecinających się z DS.

Zdecydowanie zgadzam się jednak ze stwierdzeniem, że:

> […] choć od 2013 r. **wiele pisze się w humanistyce i naukach społecznych o algorytmach i danych, to bardzo mało uwagi poświęcono osobom, które budują te epistemologiczne ramy** [podkr. R.Ż.] poprzez rozwój i wdrażanie określonych technologii (Lowrie, 2017: 2).

Te osoby to data scientiści.

Rozdział 3
Rama teoretyczna i metodologia badań własnych

Za najważniejsze teorie socjologiczne dostarczające sposobów konceptualizacji wybranego obszaru badania i aparatu pojęciowego do badania takiego rodzaju całości społecznej uważam teorie społecznych światów w ujęciach Tamotsu Shibutaniego, Anselma L. Straussa, Howarda S. Beckera, a przede wszystkim teorię społecznych światów/aren Adele E. Clarke. Koncepcja Clarke jest przeze mnie traktowana, zgodnie z propozycją tej autorki, jako część pakietu teorii/metod badań.

Konsekwencją usytuowania pracy w ramie teoretyczno-metodologicznej społecznych światów/aren jest jej osadzenie w perspektywie symbolicznego interakcjonizmu oraz paradygmatu interpretatywnego socjologii (Krzemiński, 1986; Piotrowski, 1998; Konecki, 2000; Kuhn, 2001; Hałas, 2006; Silverman, 2010).

Głównych inspiracji metodologicznych dostarcza mi podejście badawcze Clarke, które nazywa ona analizą sytuacyjną, oraz strategia ugruntowanego teoretyzowania (Clarke, 2003; 2005; Clarke, Friese, 2007; Charmaz, Clarke, 2013; Clarke, Friese, Washburn, 2017). Decyzja o takim wyborze jest motywowana trojako. Po pierwsze, autorka proponuje „ugruntowane teoretyzowanie" zamiast „teorii ugruntowanej", co oznacza prowadzenie analizy nastawionej na tzw. sytuację, zrozumienie złożoności i heterogeniczności badanej rzeczywistości oraz uwzględnianie zanurzenia w lokalnych kontekstach, za immanentną cechę badanej rzeczywistości, przyjmując nieład, rozmycie czy pogmatwanie. Uznaję to za adekwatne podejście do tak dynamicznie zmieniającej się dziedziny jak DS. Po drugie, Clarke uznaje aktorów pozaludzkich za pełnoprawne elementy badanego kontekstu, co w przypadku pracy dotyczącej DS jest niezbędne. Trzecim powodem jest to, że autorka stosuje owo podejście do studiów nad nauką i techniką (*science and technology studies* – STS), które są mi bliskie z uwagi na teorie, metody i badany podmiot. Argumentem nadrzędnym jest to, że taka metodologia jest konsekwencją przyjęcia ramy teoretycznej społecznych światów, z naciskiem na społeczne światy/areny Clarke.

Cel tej książki – opis etnograficzny świata społecznego DS w Polsce – jest zgodny z podstawowym zadaniem etnografii: „opisowym objaśnieniem świata życia badanych" (Konecki, 2010: 20).

Etnografię jako metodę rozumiem następująco (Angrosino, 2010: 45–46):

1) to praca badacza w terenie, w naturalnym środowisku życia badanych, nie w laboratorium;
2) badacz i badani pozostają w bezpośrednim kontakcie, a badacz jest jednocześnie uczestnikiem i obserwatorem;
3) badanie jest wieloczynnikowe, „wielostanowiskowe" (Marcus, 1995), używa się wielu technik zbierania danych;
4) zaangażowanie badacza w przebywanie i kontakt z badanymi jest długotrwałe;
5) podejście do generowania wiedzy jest indukcyjne, brak weryfikowania hipotez;
6) prowadzony jest dialog z badanymi, komentarze do wniosków i interpretacji badacza uzyskiwane są na bieżąco;
7) przedstawia holistyczny obraz badanego środowiska.

Podejście etnograficzne, szczególnie wielostanowiskowe, jest wykorzystywane w badaniach społecznych światów i STS (Clarke, Friese, Washburn, 2015: 46). Etnografia wielostanowiskowa to metoda analizy sytuacyjnej (Clarke, 2005: xxxiii). Marcin Zaród wskazuje, że „koncepcja etnografii wielostanowiskowej Marcusa oparta była na badaniach etnograficznych prowadzonych w ramach STS", i że stosował on takie podejście w swoich badaniach hakerów z uwagi na „latourowskie postulaty podążania za aktorami i rozpraszania tego, co lokalne" (Zaród, 2018: 156).

Niniejsze badania to również etnografia wielostanowiskowa. Zawierają one także elementy autoetnografii, netnografii i etnografii opartej na współpracy.

3.1. Społeczne światy/areny

Rama teoretyczna społecznych światów ma swoje korzenie w tzw. szkole chicagowskiej, w której socjologowie, bazując na myśli George'a H. Meada i Johna Deweya, praktykowali badania terenowe niewielkich grup społecznych lub przestrzeni miejskich. Badania takich „całości społecznych" nastawione były na wgląd w interakcje, w praktyki wspólnego wytwarzania znaczenia i działania na podstawie tego znaczenia, także wgląd w tożsamości. Te „całości społeczne" badano w ich naturalnym środowisku, w konkretnym miejscu i czasie.

Od końca lat pięćdziesiątych do końca osiemdziesiątych ubiegłego wieku teorię społecznych światów rozwijali przedstawiciele szkoły chicagowskiej: Anselm Strauss, Tamotsu Shibutani, Howard S. Becker. Do lat osiemdziesiątych przedstawiciele symbolicznego interakcjonizmu, przyjmujący perspektywę społecznych światów, koncentrowali się na badaniach problemów społecznych, sztuki,

medycyny, pracy, zawodów. Później wzrosło zainteresowanie badaniem nauki i techniki, a jedną z pierwszych pracy były badania uczennicy Straussa – A.E. Clarke z 1987 roku. W końcu XX wieku badania tego rodzaju zaczęły uwzględniać infrastrukturę materialną i narzędzia świata społecznego, nazywane aktorami pozaludzkimi (*nonhuman*) (Clarke, Star, 2008: 113–115).

Zastosowanie teorii społecznych światów/aren, analizy sytuacyjnej i strategii ugruntowanego teoretyzowania jest właściwe dla badania DS. Data science stanowi chaotyczne i dynamiczne połączenie m.in. nauki, techniki, pracy, zawodu. Jest przedmiotem sporów i dyskusji wewnątrz i na zewnątrz „środowiska". Jest światem nierozerwalnie związanym z infrastrukturą techniczną i narzędziami pracy, stąd zależy mi na ujęciu aktorów pozaludzkich.

Zdaniem Straussa (1978: 122) najważniejsza cecha definicyjna świata społecznego to istnienie jednej uderzająco ewidentnej działalności podstawowej (*primary activity*). Ta działalność jest oczywista i staje się kryterium wyodrębnienia świata społecznego (Kacperczyk, 2016: 31). Skoro zatem działalność podstawowa data scientistów jest dość oczywista – polega na komputerowych operacjach na danych cyfrowych[1] – oraz skoro niezwykle trudne jest wykazanie innych kryteriów pozwalających na wyróżnienie DS jako całości społecznej, to przyjmuję założenie o możliwości wydzielenia takiej całości społecznej jak DS na podstawie jej działania podstawowego. Tym samym **zakładam, że DS jest społecznym światem** w rozumieniu wyżej wymienionych autorów.

Praktyki autodefiniowania DS jako całości – „dziecka", wyrastającego z „rodziców", tzn. bardziej dojrzałych dziedzin nauki i praktyki, lub całości znajdującej się na przecięciu tych dziedziny (statystyki i matematyki, informatyki, biznesu) uważam za markery do zastosowania pojęcia społecznego świata.

Zagadnieniem kluczowym dla teorii społecznych światów jest działanie podstawowe:

> Serce świata społecznego stanowi podstawowe działanie (*primary activity*). Wokół niego koncentruje się cała energia uczestników. Stanowi ono także kryterium wyodrębnienia społecznego świata jako pewnej całości, bo jego uczestnikami są aktorzy podejmujący takie działanie (Kacperczyk, 2016: 34).

Działaniu podstawowemu towarzyszą działania wspomagające w skali makro i mikro. „W perspektywie makrospołecznej podstawowe działanie obrasta całą siecią działań pomocniczych, służących przetrwaniu i ekspansji społecznego świata jako całości" (Kacperczyk, 2016: 35). Strauss wyróżnia czternaście subprocesów pomocniczych wobec działania podstawowego, m.in. finansowanie, reklamowanie, budowanie organizacji. Skala mikro to działania wspomagające działanie podstawowe pojedynczego aktora społecznego. Jego „działanie podstawowe zanurzone jest w innych podtrzymujących i umożliwiających je działaniach" (Strauss, Corbin, 1990: 236; Kacperczyk, 2016: 35).

1 Wskażę, że to określenie działania podstawowego społecznego świata DS nie jest zupełnie poprawne.

Za istotną cechę definicyjną społecznego świata uznawana jest jego technologia, czyli:

> […] wykorzystanie określonych środków technicznych, umożliwiających specyficzny sposób wykonania działania. […] zawsze jest uwikłana w budowanie i podtrzymywanie społecznego świata, wiedzę o stosownych miejscach i okolicznościach, w których można podejmować dane działanie, oraz o specjalistycznym sprzęcie, jakim „należy" dysponować, aby wykonywać je prawidłowo (Kacperczyk, 2016: 36–37).

Na technologie społecznego świata „powołują się" procesy legitymizacji i zaświadczania o autentyczności. Rozwój technologii społecznego świata może prowadzić do procesu segmentacji czy jej „odmiany" – profesjonalizacji pewnych subświatów, o czym za chwilę. Za pomocą opisu zmian technologii w czasie można dokonać także opisu tzw. areny świata społecznego. Zmiany technologii ujawniają interpretacyjne zmagania nad znaczeniami (Clarke, Casper, 1996: 615).

Brak wyraźnych granic społecznych światów jest uznawany za najpoważniejszy problem związany z ich konceptualizacją. Strauss (1993) uważa, że problem podnoszą badacze przyzwyczajeni do ujmowania całości społecznych jako odgraniczonych czy wydzielonych od innych. W przypadku światów społecznych nie da się takich granic dookreślić. Relatywna płynność granic jest jedną z podstawowych właściwości światów społecznych (Strauss, 1993: 212–213). Świat społeczny nie jest sprowadzalny do grupy czy organizacji (Kacperczyk, 2016: 38).

Działania we wszystkich społecznych światach i arenach obejmują ustanawianie i utrzymywanie granic między społecznymi światami oraz uzyskanie legitymizacji dla samego świata (Clarke, Star, 2008: 118).

Arena jest „polem działań i interakcji pomiędzy potencjalnie nieskończoną ilością rozmaitych zbiorowych całości" (Clarke, 1991: 128). Jest polem, na którym toczy się istotny dla świata społecznego problem, i oznacza niezgodę na kierunek działania podejmowany przez świat społeczny lub jego segment (Strauss, 1993: 227). W świecie opieki paliatywnej w Polsce arena skoncentruje się wokół obiektu granicznego – morfiny i podobnych leków opioidowych (Kacperczyk, 2006). Obiekt graniczny „staje się punktem zapalnym, jądrem krystalizacji dla areny, a pośrednio dla autodefinicji uczestników zamieszanych w dyskusję światów" (Kacperczyk, 2016: 41). Zrozumienie, o co spierają się uczestnicy społecznego świata/społecznych światów, jest niezbędne dla zrozumienia społecznego świata w ogóle, np. zachodzących w nim procesów i jego wartości.

Pojęcie wartości w koncepcjach światów społecznych nabiera znaczenia w odniesieniu do działania podstawowego:

> […] w społecznym świecie działanie nigdy nie jest działaniem dowolnym. Obwarowane jest regułami, wyobrażeniami, jak powinno przebiegać, a przede wszystkim z jakich pobudek powinno wypływać. […] O granicach pomiędzy subświatami decydują ostatecznie zasadnicze różnice w pojmowaniu istoty podstawowego działania, a te są tak naprawdę różnicami w sferze wartości, które są tutaj rozumiane jako punkty orientacyjne, pozwalające dokonać oceny działania, a jednocześnie wskazujące, jak ma ono wyglądać i w jakich granicach przebiegać (Kacperczyk, 2016: 44).

W uniwersum elementów społecznego świata znajduje się również pojęcie tożsamości. Tożsamość uznaje się za sprzężoną/splecioną z wartościami. „Jeśli tożsamość widzimy jako zobowiązanie jednostki wobec jej grupy odniesienia" – przypomnieć należy pojawiające się już u Howarda Beckera (1974; 1986) pojęcie *commitment*, rozumiane jako konstytuujące świat społeczny zobowiązanie do podejmowania działania podstawowego – „oznacza to, że jednostka jest związana ze swoją »grupą« wartościami rozumianymi jako kryteria oceny czy perspektywy poznawcze" (Kacperczyk, 2016: 48). Te perspektywy i kryteria są nieustannie poddawane „testowi rzeczywistości" (Shibutani, 1955: 569).

Wśród podstawowych procesów zachodzących w społecznym świecie nieustannie dochodzi do legitymizacji i zaświadczania o autentyczności. Legitymizacja wiąże się ze wskazaniem na wyjątkowy sposób wykonywania działania podstawowego przez uczestników społecznego świata oraz na konkretną technologię, jaką dysponuje dany świat lub segment świata. W skali makro zachodzi proces „wyznaczania granic prawidłowej działalności". W skali mikro istnieje wiele objaśnień, uzasadnień, wymówek pozwalających pojedynczym aktorom kontynuować działanie podstawowe (Kacperczyk, 2016: 36).

Odwołując się do Straussa (1993: 215) oraz artykułu Roba Klinga i Elihu M. Gersona (1978), Anna Kacperczyk twierdzi, że „jedną z najważniejszych cech społecznego świata jest jego nieuchronna segmentacja (*segmentation*), prowadząca do wewnętrznego różnicowania się i wydzielania w jego ramach subświatów", za przyczynę segmentacji przyjmuje zaś specjalizację zainteresowań i różnicowanie się interesów pewnych podgrup w świecie społecznym (Kacperczyk, 2016: 49). Segmentacja, zdaniem Straussa (1984), ma trzy zasadnicze drogi: pączkowanie, odszczepienie i przecinanie się światów.

Adele E. Clarke (1991) używa metafory mozaiki, mówiąc o wielości światów i subświatów społecznych, w których równocześnie i równolegle uczestniczą aktorzy. Określa świat społeczny w odwołaniu do pojęcia środowiska społecznego, „świata czegoś", czyli jako „grupy wspólnie zaangażowane w pewną działalność", „grupy mające wspólne kompetencje do wykonywania pewnych działań, dzielące różnego rodzaju zasoby, by osiągać swoje cele i budujące wspólne ideologie dotyczące tego, jak realizować własne interesy". Charakter działania podstawowego, wokół którego koncentruje się świat społeczny, może być różnorodny i związany np. ze sprawami artystycznymi, religijnymi, zawodowymi, rekreacyjnymi, komercyjnymi (Clarke, 1991: 131, za: Kacperczyk, 2016: 32). Takie mozaikowe, akceptujące chaos widzenie społecznych światów Clarke przełożyła na strategie badawcze i analityczne. Mówi ona o renowacji i regeneracji klasycznej teorii ugruntowanej na rzecz ugruntowanego teoretyzowania (Clarke, 2003: 558). Ugruntowane teoretyzowanie dokonywane jest na podstawie analizy sytuacyjnej. Tę ramę ujmowania społecznych światów/aren nazwano pakietem[2] teorii/metod badań, który

2 Warto przypomnieć o pojęciu pakietu w świecie społecznym DS – pakiet to narzędzie używane w języku programowania; jest zbiorem funkcji wykonujących określone zadania.

jest zakorzeniony w konstruktywizmie społecznym, zawiera w sobie symboliczny interakcjonizm i filozofię pragmatyczną (Clarke, Star, 2008: 12).

Podejście Clarke cechuje holizm metodologiczny oraz (neo)pragmatyzm (Diaz-Bone, 2013). Procesy zachodzące w społecznych światach badacz może zrozumieć tylko wtedy, kiedy założy, że nieład stanowi immanentną cechę obserwowanej przez niego rzeczywistości społecznej (Kacperczyk, 2016: 568). Rzeczywistość społeczna jest uważana za pełną konfliktu i nieprzezwyciężalnych sprzeczności. Znajduje to wyraz w tzw. arenie sporu. W podejściu Clarke areny są tożsame ze społecznymi światami. Nie istnieje badanie świata społecznego bez badania jego aren. Clarke wyraża to, posługując się określeniami „teoria społecznych światów/aren" lub „analiza społecznych światów/aren" (Kacperczyk, 2016: 572).

Areny występują na różnym poziomie rzeczywistości społecznej. Arena jako przestrzeń sporu może dotyczyć jednego lub kilku subświatów, jednego lub kilku światów społecznych, a nawet tzw. problemów publicznych, czyli zagadnień dziejących się pomiędzy wieloma światami społecznymi. Strauss podkreśla, że pomiędzy arenami na każdym z poziomów występują przecięcia. Szczególnie w przypadku aren obejmujących wiele światów należy badać uwikłanie reprezentantów (tj. zabierających głos przedstawicieli światów biorących udział w sporze) w spory wewnątrz własnych światów czy subświatów (Strauss, 1978: 124–25).

Analiza sytuacyjna, będąca metodologiczną konsekwencją przyjęcia omówionych wyżej założeń, jest dla badaczy „zaproszeniem do uwzględnienia chaotyczności licznych elementów sytuacji: ludzi, elementów pozaludzkich (*nonhuman*), dyskursów, technologii i innych" (Morrison, 2016: 519). Kacperczyk, powołując się na Straussa (1978) i Beckera (1986), uznaje, że w badaniu społecznych światów/aren jednostką analizy nie są pojedynczy ludzie, a właśnie społeczny świat/arena. Traktowana jest ona jako zbiorowość (*social unit*) wytwarzająca znaczenia i podejmująca kolektywne działania oraz jako uniwersum dyskursu (Kacperczyk, 2016: 573). To samo mówią Adele E. Clarke i Susan Leigh Star. Społeczny świat/arena jest jednostką analizy szczególnie w badaniach nauki i techniki; może być to badanie kontrowersji lub badanie dyscypliny (Clarke, Star, 2008: 123–124):

> Jeśli i kiedy liczba światów społecznych staje się duża i poprzecinana konfliktami, różnymi rodzajami kariery, różnymi punktami widzenia, źródłami finansowania itd., to ta całość jest analizowana jako arena. Arena składa się zatem z wielu społecznych światów zorganizowanych ekologicznie wokół zagadnień, będących przedmiotem wspólnego zainteresowania i zaangażowania (*commitment*) w działanie [tłum. R.Ż.] (Clarke, Star, 2008: 113).

Perspektywa społecznych światów/aren jest kluczowa dla niniejszej książki i właściwa dla badanego podmiotu, czyli DS.

Poczynając od pracy Clarke poświęconej naukom reprodukcyjnym w USA z 1987 roku, w badaniach nauki i techniki stosowano teorię społecznych światów/aren. Poza licznymi pracami Clarke i Star tego rodzaju badania dotyczyły m.in. wyłonienia się dziedziny genetyki środowiskowej, chirurgii płodów i postrzegania

płodów jako pacjentów, zmian składów w zespołach inżynierów oprogramowania (Clarke, Star, 2008: 124–126).

Polską pracą socjologiczną w nurcie STS jest rozprawa doktorska Marcina Zaroda *Aktorzy-sieci w kolektywach hakerskich w Polsce* (2018). Uważa on, że mogą występować przecięcia pomiędzy światem DS a hakerami. Choć w pracy pojawiają się stosowane przez Clarke, Star i ich naśladowców terminy – np. drukarka 3D w charakterze obiektu granicznego (Clarke, Star, 2018: 307–313) – to nie ma odniesień do społecznych światów. Autor pracuje w ramie Teorii Aktora-Sieci (ANT).

Teoria Aktora-Sieci jest koncepcją odmienną od światów społecznych. Dostarcza ujęcia wiedzy – konstruowania sieci jako procesu wiedzotwórczego. Niezbędne jest włączenie w proces aktorów pozaludzkich (*nonhumans*) (Abriszewski, 2010). Możliwość uwzględnienia aktorów, takich jak: źródła danych, hardware umożliwiający składowanie danych lub wykonywanie obliczeń, software i koncepcje użyte do magazynowania danych czy do ich przetwarzania/eksploracji/modelowania/wizualizacji, jest użyteczna w opisie świata DS (por. Żulicki, 2016: 25).

Koncepcje społecznych światów i ANT różnią się też kulturowo:

> Porównując ANT Latoura ze społecznymi światami/arenami, mówi się, że scentralizowany charakter władzy w ANT jest bardziej francuski, podczas gdy pluralizm perspektyw w koncepcji światów społecznych jest charakterystyczny dla Amerykanów. Bowker i Latour (1987) stworzyli nieco podobny argument, porównując francuskie i angloamerykańskie STS. Argumentowali oni, że oś racjonalności/władzy jest tak naturalna dla francuskiej technokracji (i zbadanej między innymi u Foucaulta), że właśnie to musi być uwzględniane w pracach angloamerykańskich, w kontekście angloamerykańskim zwykle bowiem „zakłada się" (*assumed*), że nie ma związku między tymi dwoma [władzą i racjonalnością – przyp. R.Ż.] [tłum. R.Ż.] (Clarke, Star, 2008: 123).

Koncepcja światów społecznych jest bardziej odpowiednia do pracy nad opisem DS niż ANT, także dlatego, że DS ma angloamerykański rodowód. Zgodnie z zakorzenioną w szkole chicagowskiej strategią badania społecznych światów/aren – analizą sytuacyjną – osadzam badania w konkretnym miejscu i czasie. Data science w Polsce może mieć swój specyficzny charakter także dlatego, że idea technokracji historycznie odgrywała tu znikomą rolę (Kurczewska, 1997: XII). Od początków technokracji we Francji XIX wieku, kiedy to Henri de Saint-Simon przedstawił pomysł administrowania ludźmi tak jak rzeczami, poprzez kontynuatorów amerykańskich w okresie po I wojnie światowej – Howarda Scotta, Williama H. Smytha, Thorsteina Veblena – była ona ideą wyrosłą na odmiennej niż dominująca w ówczesnej Polsce filozofii życiowej, politycznej czy etyce pracy. Jak wskazuje Joanna Kurczewska:

> […] przedstawiciele polskich elit intelektualnych częściej myśleli o Polsce i świecie w kategoriach politycznych czy narodowo-kulturowych aniżeli w kategoriach cywilizacyjno-gospodarczych czy też w takich, które politykę bezpośrednio uzależniały od kondycji cywilizacyjnej społeczeństwa (Kurczewska, 1997: XIII).

Nie znaczy to, że nie było w polskiej myśli społecznej tradycji przygotowującej do recepcji zachodnioeuropejskich i amerykańskich idei technokratycznych. Autorka mówi o tradycji pozytywistów warszawskich „ceniących sobie naukę i technikę jako dźwignie postępu społecznego" jako o nurcie konkurencyjnym i jednocześnie znacznie mniej popularnym niż romantyzm emancypacyjno-narodowy (Kurczewska, 1997: XII). Elementy idei technokratycznych można odnaleźć w świecie społecznym DS.

3.2. Analiza sytuacyjna i ugruntowane teoretyzowanie

Adele E. Clarke przekłada mozaikowe, akceptujące chaos i sprzeczności widzenie społecznych światów/aren na strategie badawcze i analityczne. Mówi ona o renowacji i regeneracji (*renovate and regenerate*) teorii ugruntowanej na rzecz ugruntowanego teoretyzowania, dokonywanego w ramach analizy sytuacyjnej (Clarke, 2003: 558). Jest to tzw. pakiet teorii/metod badań.

Zdaniem Kathy Charmaz (2009: 232): „usytuowanie teorii ugruntowanych w ich społecznych, historycznych, lokalnych oraz interakcyjnych kontekstach wzmacnia je". Choć to odejście od tradycyjnej metodologii teorii ugruntowanej (MTU), to Clarke widzi rzecz inaczej. Nie chodzi jej o uwzględnienie kontekstu, ponieważ „**otoczenie (*conditions*) sytuacji jest sytuacją** [podkr. oryg.]. Nie ma czegoś takiego jak kontekst" (Clarke, 2005: 71). Badacz ma przy tym wykazać się jak największą samoświadomością wobec konstruowanego przez siebie wyjaśnienia. Teoria ugruntowana ma być elastyczną strategią heurystyczną. Zmierza się w kierunku namysłu nad drogą, która doprowadziła do uzyskania końcowego efektu analiz, i to właśnie ta droga nazwana jest ugruntowanym teoretyzowaniem (Kacperczyk, 2007: 9). Wykorzystywana jest koncepcja „teoretyzowania" zamiast koncepcji „teorii", ponieważ najważniejszym celem analizy sytuacyjnej jest otwarcie danych i sytuacji na niekończące się (*ongoing*) negocjacje, a nie promowanie obiektywnej prawdy (Uri, 2015: 146). Jednostką analizy nie jest dla Clarke wybrany proces czy zjawisko, a sytuacja. Jest ona konstruowana za pomocą trzech rodzajów map (sytuacyjnych, społecznych światów/aren i pozycyjnych) i poprzez pracę analityczną z tymi mapami. Odróżnia to analizę sytuacyjną od MTU, dążącej do ukazania ludzkiego działania jako podstawowego procesu społecznego (Clarke, Friese, Washburn, 2015: 12), co jest wynikiem rozumienia tego, czym jest sytuacja. Nie jest ona uwzględnieniem otoczenia, a jednocześnie (i nie tylko) jest tym, co tradycyjnie nazwalibyśmy podstawowym procesem i jego otoczeniem. Clarke pokazuje, co składa się na sytuację w formie matrycy sytuacyjnej (*situational matrix*), gdzie między elementami nie ma żadnej hierarchii (Clarke, 2005: 73, za Kacperczyk, 2007: 22), a mianowicie:

1) elementy czasoprzestrzenne;
2) elementy ludzkie (*human*), indywidualne i kolektywne;
3) elementy pozaludzkie (*nonhuman*);
4) elementy polityczno-ekonomiczne;
5) dyskursywne konstrukcje aktorów;
6) elementy organizacyjne/instytucjonalne;
7) główne kwestie kontestowane;
8) elementy lokalne i globalne;
9) elementy socjokulturowe;
10) elementy symboliczne;
11) dyskurs popularny i inne dyskursy;
12) pozostałe elementy empiryczne.

Elementy te, tworzące „sytuację działania" (*the situation of action*), uznaje się za sprzężone i wpływające na siebie. Wizualizacja matrycy sytuacyjnej, przypominająca dziecięcy rysunek słońca, gdzie w centrum – w miejscu tarczy – umieszczono tytuł „sytuacja działania", elementy sytuacji ułożono zaś jak promienie, umieszczona została na okładce pierwszej pracy w całości poświęconej analizie sytuacyjnej (Clarke, 2005). Autorka wskazuje, że np. Anselm L. Strauss i Juliet Corbin we wcześniejszych tego rodzaju matrycach (*conditional matrix*) w centrum stawiali działanie (*action*), a następnie pokazywali hierarchię warunków – od m.in. interakcyjnych przez organizacyjne i dalej narodowe, graficznie ujęte w formie kolejno oddalających się od centrum kręgów. Analiza sytuacyjna zakłada, że elementy otoczenia sytuacji (*conditional elements*) muszą być określone w samej sytuacji, ponieważ są dla niej konstytutywne. One są sytuacją, a nie tylko otaczają ją, kształtują czy przyczyniają się do niej (Clarke, 2015: 96–98). Podejście Clarke jest zatem, podobnie jak Charmaz, konstruktywistyczne, ale Clarke podkreśla traktowanie sytuacji jako jednostki analizy. Tym samym odchodzi ona od znanego z innych nurtów jakościowych nastawienia na jednostkę ludzką jako podmiot, w kierunku nastawienia na uchwycenie pełnej sytuacji badania (Kacperczyk, 2007: 10). Celem badania dla Clarke jest „zrozumienie złożoności i heterogeniczności indywidualnych i kolektywnych sytuacji, dyskursów i interpretacji sytuacji" (Clarke, 2005: xxv, za Kacperczyk, 2007: 10). Jej zdaniem jest to zgodne z dwiema ideami Normana Denzina: „sytuacyjnej interpretacji" (*situating interpretation*) (Clarke, 2005: xxviii) i włączenia do interakcjonizmu myśli postmodernistycznej i poststrukturalistycznej Rolanda Barthesa, Jacquesa Derridy, Michela Foucaulta, Jeana Baudillarda (Clarke, 2015: 90).

Co do unikania nadmiernych uproszczeń, to widać to w postulacie nagłaśniania głosów grup czy pozycji słabo słyszalnych (*helping silences speak*), przemilczanych, nieobecnych w społecznym świecie, a także w konstruowanym przez badacza wyjaśnieniu (Clarke, 2003: 561, 569; 2005: 85; 2015: 108; Clarke, Star, 2008: 119, 124, 129). „Metody badań mają umożliwiać i zachęcać analityka do naświetlenia poprzednio marginalizowanych i nielegitymizowanych perspektyw wiedzy o życiu społecznym" (Kacperczyk, 2007: 10).

Zgadzam się z Kacperczyk, która uważa, że Clarke nie dokonuje rewitalizacji czy ożywienia MTU, tylko proponuje nowe podejście, „ponowne wcielenie".

> Jest to kompletne zerwanie z tradycyjną, *pure Grounded Theory – pushing GT around the Postmodern Turn* [podkr. oryg.] oznacza w istocie wypchnięcie jej poza obszar jej zakorzenienia, i to już nie jest ta sama GT (Kacperczyk, 2007: 25).

Postmodernistyczne podejście Clarke załamuje wizję generowania teorii z danych (Kacperczyk, 2007). Proponowany jest szereg – mówiąc językiem MTU – kodów teoretycznych: od samego pojęcia społecznych światów/aren, przez wszelkie pojęcia i procesy w takich światach, elementy sytuacji działania. Proponowane jest także zadawanie konkretnych, w żadnym razie nieugruntowanych w danych terenowych, pytań o tak skonceptualizowany świat społeczny/arenę (Clarke, 2005: 115–116) i stosowanie konkretnych strategii analitycznych z użyciem map – wizualnych i konceptualnych szablonów (Clarke, 2005: 83–144), z pewnością zatem „prekonceptualizacja" jest silna. Podejście to jest konstruktywistyczne i jako takie je przyjmuję.

Jak napisałem wyżej, **zakładam, że DS jest społecznym światem**. **Tym samym dokonywany w tej pracy opis świata społecznego jest przeze mnie konstruowany, nie jest odkrywany.**

3.3. Techniki badawcze i analityczne

Zastosowałem typową dla badania społecznych światów, a szczególnie dla analizy sytuacyjnej, etnografię wielostanowiskową (Marcus, 1995; Clarke, 2005: xxxiii; Clarke, Friese, Washburn, 2015: 46). Źródłem danych jakościowych były wywiady swobodne, obserwacje uczestniczące, a także elementy autoetnografii, netnografii i etnografii opartej na współpracy. Jest to dominujący komponent moich badań empirycznych. Próba dobrana została w sposób nieprobabilistyczny, celowy, zgodnie ze strategią próbkowania teoretycznego. Materiał jakościowy był analizowany za pomocą kodowania, kilku rodzajów not i map.

Do technik etnograficznych dołączyłem ilościową analizę (na poziomie opisu i eksploracji, a nie wnioskowania statystycznego) danych zastanych dwojakiego rodzaju. Po pierwsze, wykonana została wtórna analiza wewnętrznych sondaży „środowiska" DS; sondaże te uznaję za akty samowiedzy badanego świata. Po drugie, przeanalizowałem dane z serwisu meetup.com, popularnej platformy służącej organizacji spotkań, m.in. osób zainteresowanych DS. Celem tej części badań było przede wszystkim ujęcie pełnej sytuacji badania oraz, dodatkowo, enkulturacja badacza w działaniu zbliżonym do działania podstawowego badanego świata społecznego, a w konsekwencji głębsze zrozumienie kategorii jakościowych.

Najważniejszym źródłem danych jakościowych były indywidualne **wywiady swobodne ukierunkowane** (Lutyński, 1974; za Przybyłowska, 1978: 62–63; por. Konecki, 2000: 170), prowadzone w duchu wywiadu intensywnego (Charmaz, 2009: 39).

Obserwacja uczestnicząca to technika (a szerzej – rola etnografa), która służyła mi w dużej mierze za źródło pytań badawczych do wywiadów (por. Zaród, 2018: 171). Czasami było odwrotnie – informację uzyskaną w wywiadzie byłem w stanie zinterpretować w sposób bliski punktowi widzenia uczestników badanego świata dopiero dzięki kolejnym obserwacjom uczestniczącym. Obserwacje uczestniczące były niezwykle ważne dla mojej enkulturacji w społecznym świecie DS (Jemielniak, 2018: 20). Obserwacje, w których spotykałem na żywo uczestników osadzonych w świecie DS, były też pomocne w nawiązywaniu kontaktów z potencjalnymi rozmówczyniami/rozmówcami.

Obserwacje łączyły się z **autoetnografią**, gdyż notatki sporządzane w przerwach lub po zrealizowaniu obserwacji w terenie przybierały autoetnograficzny charakter, tzn. dotyczyły własnych odczuć, stanów wewnętrznych, pojawiających się w trakcie obserwacji myśli oraz własnych działań i interakcji z innymi ludźmi w miejscu badania. To technika otrzymywania danych jakościowych, w której badacz tworzy autonarracyjne notatki, bazując na autorefleksji i autoanalityce (Kacperczyk, 2014: 37–38).

Celem autoetnografii było zwiększenie samoświadomości badacza i odsłonięcie informacji składających się na jego intelektualne tło – zgodnie z postulowaną zmianą stanowiska badacza „z »wszechwiedzącego analityka« do »samoświadomego uczestnika« w procesie produkcji zawsze częściowej wiedzy" (Clarke, 2003: 555–556). Starałem się o „jednoczesne bycie outsiderem i insiderem podczas przeprowadzania swojego badawczego projektu" (Darmach, 2017: 95).

Skoro **netnografia** jest badaniem etnograficznym społeczności w internecie (Kozinets, 2003), to badania świata DS nie można wykonać bez tego elementu. Nie można uczestniczyć w tym świecie bez korzystania z platform, stron i narzędzi internetowych. Świadomy braku systematyczności pojęć dotyczących badań jakościowych z użyciem internetu (Jemielniak, 2018: 11–13) nazywam użyte techniki netnografią w rozumieniu: „odmiana etnografii zaadaptowana do badania kultur i społeczności wirtualnych oraz do badań nad wirtualnymi aspektami życia społecznego" (Kacperczyk, 2016: 639). Moje badania netnograficzne były nastawione na analizę treści, a nie analizę zdarzeń i interakcji (Kacperczyk, 2016).

Etnografia oparta na współpracy została użyta w niewielkim stopniu i w sposób niepełny, niemniej dla wyników badania okazała się bezcenna. Współpraca z uczestniczkami i uczestnikami badań pozwoliła m.in. na przygotowanie ostatecznej, kompletnie innej od pierwotnej, wersji opisu działania podstawowego badanego świata. W założeniu Luke'a Erica Lassitera:

> […] etnografię opartą na współpracy (*collaborative ethnography*) możemy określić ogólnie jako podejście do badań etnograficznych, które w sposób zamierzony i wyraźny ujawnia oraz podkreśla elementy współpracy na każdym etapie procesu badawczego – od konceptualizacji projektu, poprzez pracę terenową, do ważnego w tym kontekście etapu pisania (Lassiter, 2005: 16).

W niniejszej pracy chodzi jedynie o współpracę badaczy z osobami na różnych etapach badań i jej pisania. Konsultowałem się w sposób nieformalny z osobami osadzonymi w badanym świecie. Najczęściej były to rozmowy po wywiadach, kilkakrotnie przesłanie zainteresowanym pierwszej wersji fragmentu tekstu, najrzadziej były to konsultacje teoretycznego pomysłu przed napisaniem właściwego fragmentu.

Terenowe badania jakościowe były realizowane od 21 października 2016 do 14 czerwca 2019 roku.

Ponieważ „dobre badanie etnograficzne łączy dane z obserwacji, wywiadów oraz analiz danych zastanych" (Angrosino, 2010: 102), w czerwcu i lipcu 2019 roku wykonana została wtórna analiza danych ilościowych:

1) pochodzących z wybranych sondaży „środowiska" DS; autorzy ankiet udostępnili publicznie zebrane dane;
2) z serwisu meetup.com grup zlokalizowanych w Polsce i deklarujących zajmowanie się DS; dane zostały pobrane legalnie z użyciem API.

Jeżeli chodzi o **dane zebrane w środowiskowych sondażach**, skorzystałem z sześciu zbiorów ułożonych przez cztery podmioty, wybrane przeze mnie jako znaczące w badanym świecie społecznym DS na podstawie własnych badań etnograficznych. Wybrane zmienne zharmonizowałem *ex-post*, tzn. połączyłem w spójny sposób dane, które zostały zebrane osobno i bez założenia, że będą porównywane (Dubrow, Tomescu-Dubrow, 2016). Starałem się korzystać z możliwie najszerszego horyzontu czasowego i zawężać zbiory danych do respondentów mieszkających w Polsce.

Najstarszym źródłem danych ilościowych był formularz zgłoszeniowy pierwszej w Polsce konferencji użytkowników języka programowania R o nazwie „WhyR?" z 2017 roku. Organizatorzy udostępnili zbiór danych w serwisie GitHub[3]. Zawęziłem zbiór (n = 246) do respondentów deklarujących zamieszkiwanie w Polsce, a więc analizowana próba n = 202.

Analizowałem także dane z formularza zgłoszeniowego drugiej w Polsce konferencji „PyData 2018". Zbiór danych również udostępniono na GitHubie[4]. Po zawężeniu zbioru (n = 435) jak wyżej analizowana próba to n = 284.

Innym źródłem danych były ankiety Kaggle ML & DS Survey z 2017 i 2018 roku. Kaggle to będąca częścią Google LLC platforma internetowa, na której ludzie mogą brać udział w konkursach na najlepsze modele predykcyjne i opisywanie zestawów danych, przesyłanych przez firmy i użytkowników oraz wygrywać nagrody (Kaggle, 2018b). Platforma ta jest popularna w badanym świecie społecznym i służy np. kreowaniu własnego wizerunku. Analizowane zbiory danych ML & DS Survey także były przedmiotem owych konkursów, można je pobrać po

3 https://raw.githubusercontent.com/WhyR2017/konkursy/master/dane_z_formularza_rejestracyjnego.csv (dostęp: 19.07.2019).

4 https://raw.githubusercontent.com/stared/random_data_explorations/master/201811_pydatawaw2018/pdwc2018_anonym.csv (dostęp: 19.07.2019).

zalogowaniu się na konto użytkownika platformy[5]. Zbiór (Kaggle, 2017a) z 2017 r. to odpowiedzi z pierwszego w historii sondażu użytkowników Kaggle. Wzięło w nim udział 16 716 osób z całego świata, ja analizowałem próbę n = 130 deklarujących zamieszkanie w Polsce. Liczba zmiennych to 288. Zbiór z roku 2018 (Kaggle, 2018a) zawiera odpowiedzi 23 860 ochotników ujęte w 395 zmiennych, „polska" próba n = 252.

Źródła te były dla mnie znaczące także w sensie jakościowym, a proponowane ujęcie działania podstawowego wzorowane jest na definicji z pierwszego sondażu Kaggle (2017b) *The State of Data Science and Machine Learning*. W czasie wykonywania moich analiz dane z sondażu 2018 były najnowszymi z dostępnych.

Do najnowszych źródeł danych należały badania Stack Overflow Annual Developer Survey 2018 i 2019. Stack Overflow (SO) to forum internetowe przeznaczone raczej do pomocy technicznej dotyczącej różnych języków programowania do wszelkich zastosowań, nie tylko DS. Jest częścią portalu Stack Exchange, w skład którego wchodzi m.in. forum datascience.stackexchange.com, przeznaczone do pomocy merytorycznej w dziedzinie DS. Posługiwanie się SO należy do podstawowych umiejętności uczestników świata DS i ogólnie osób piszących kody. W ankiecie z 2018 roku wzięło udział 98 855 ochotników zrekrutowanych spośród użytkowników forum SO, odpowiedzi ujęto w 129 zmiennych. Respondenci z Polski, deklarujący jednocześnie przynależność do kategorii programistów (*DevType*) o nazwie „Data scientist or machine learning specialist", stanowili analizowaną przeze mnie grupę n = 121. Dane organizatorzy upublicznili przez Dysk Google[6]. W badaniu z roku 2019[7] uczestniczyło 88 883 ochotników, odpowiedzi zapisano za pomocą 85 zmiennych. Analizowana przeze mnie próba n = 111 została odfiltrowana w ten sam co w starszym zbiorze z SO sposób.

Przeanalizowałem także dane pozyskane z serwisu meetup.com, gdzie tworzone są grupy osób o podobnych zainteresowaniach, spotykające się na żywo. Meetup to popularna forma organizacji spotkań osób zainteresowanych DS, nie tylko w Polsce. W dniu 4 czerwca 2019 roku za pomocą API serwisu pobrałem dane o wszystkich 86 ówczesnych grupach meetupowych zlokalizowanych w Polsce. Informacje o nich dostępne są publicznie w internecie na stronach grup, jednak pobranie ich masowo poprzez API możliwe jest jedynie dla użytkowników posiadających konto[8] w serwisie i wymaga wygenerowania unikalnego tokenu autoryzującego

5 Utworzyłem własne konto na Kaggle 6 lutego 2017 roku w ramach badań etnograficznych, nie uczestniczyłem w konkursach.

6 https://drive.google.com/uc?export=download&id=1_9On2-nsBQIw3JiY43sWbrF8EjrqrR4U (dostęp: 19.07.2019).

7 https://drive.google.com/file/d/1QOmVDpd8hcVYqqUXDXf68UMDWQZP0wQV/ (dostęp: 19.07.2019).

8 Konto na meetup.com założyłem 26 września 2016 roku na potrzeby prowadzonych badań etnograficznych, uczestniczył w spotkaniach grup w różnych miastach, w momencie pobierania danych przez API byłem zapisany jednocześnie do ośmiu grup zajmujących się DS.

operację (Meetup b.d.). Dane pobrałem w plikach JSON[9], a następnie wykonałem parsowanie plików JSON na tabelę płaską, zawierającą 86 jednostek analizy (czyli grup meetupowych) opisanych za pomocą 31 zmiennych.

Pobranie danych przez API wykonałem w języku Python, a wszystkie inne czynności związane z pracą na danych z meetup.com oraz z wyżej opisanych sondaży środowiskowych w języku R. Kod umożliwiający powtórzenie całości analiz, wraz ze wszystkimi plikami wejściowymi i wyjściowymi, upubliczniłem na swoim koncie GitHub[10], w duchu powtarzalnych badań ilościowych (*reproducible science*) (Peng, 2011a; Leek, 2016). Podobne (i znacznie bardziej zaawansowane) projekty dotyczące przetwarzania, analizy i modelowania danych zastanych za pomocą języków R i Python są zbliżone do tego, co uznaję za działanie podstawowe badanego świata DS. Takie projekty to typowa dla uczestników badanego świata interakcja z aktorami pozaludzkimi (np. danymi, kodem, środowiskiem programistycznym i innymi narzędziami cyfrowymi). Były one dla mnie ważnym elementem enkulturacji w świecie społecznym DS i bez wątpienia pomogły w zrozumieniu specyfiki działania podstawowego badanego świata.

Łącznie wykonałem 26 wywiadów swobodnych ukierunkowanych (30 godz. i 44 min nagrań), 46 aktów obserwacji uczestniczącej (zarówno online, jak i offline, łącznie około 295 godz. obserwacji), sporządziłem 12 not autoetnograficznych. Badań zrealizowanych z użyciem netnografii i etnografii opartej na współpracy ani analiz ilościowych danych zastanych nie ujmuję w ten sumaryczny sposób z uwagi na ich inny charakter, choć terminy realizacji analiz ilościowych zostały utrwalone (w repozytorium GitHub). Spis zrealizowanych badań znajduje się w aneksie.

Wszystkie wywiady były rejestrowane na dyktafonie cyfrowym po uzyskaniu świadomej zgody. Wykonano transkrypcje *verbatim*. Rozmówczyniom/rozmówcom zagwarantowałem poufność, gdzie „badacz może zidentyfikować autora wypowiedzi, ale jednoznacznie obiecuje tej informacji nie ujawniać" (Babbie, 2006: 518).

3.4. Usytuowanie badacza

„Zacznę, à la Strauss, od odrobiny mojej intelektualnej autobiografii w sensie tego, jak stałam się tym, kim jestem teraz i jak znalazłam się w obecnym miejscu" (Clarke, 2015: 85).

9 *JavaScript Object Notation* – ustrukturyzowany format wymiany danych, może być odczytywany np. przez przeglądarki internetowe czy inne aplikacje.

10 Analiza meetupów: https://github.com/zremek/meetup-harvesting, sondaży: https://github.com/zremek/survey-polish-data-science. Założyłem konto na GitHubie 3 czerwca 2018 roku na potrzeby badań etnograficznych, później używałem go także do innych projektów analitycznych (por. Żulicki, Żytomirski, 2020).

Własne usytuowanie ująłem w notatce autoetnograficznej:

> Mam ochotę napisać – „jaki jest mój background", bo tak mówi się w świecie DS i chyba dość mocno nasiąknąłem tym językiem. Zatem mój background jest taki, że jako dzieciak miałem trochę zacięcia do matematyki i w szkole średniej poszedłem do klasy matematyczno-informatycznej. Nie interesowały mnie komputery, tylko zagadki. Jednak pod koniec szkoły (2005–2006) zacząłem się coraz mocniej interesować polityką, filozofią, literaturą, i doszedłem do wniosku, że chciałbym zajmować się sondażami wyborczymi oraz politycznymi diagnozami i będę studiował socjologię. […] Po lekturze książki Kennetha Cukiera i Victora Mayera-Schönbergera, 2014, *Big Data. Rewolucja, która zmieni nasze myślenie, pracę i życie*, Warszawa: MT Biznes Ltd. – i akceptacji opiekuna naukowego w 2015 zacząłem intensywne poszukiwanie literatury. […] Po pierwsze – i o tym powiem jeszcze nie raz – na początku badań trochę bardziej chciałem sam zostać data scientistą niż pisać pracę socjologiczną o DS. Nazywam to „byciem wannabesem". Po drugie, uważam swoje dawne zbliżenie z matematyką za zaletę w badaniu tego obszaru. Choć nauczyłem się bardzo niewiele programowania, tyle samo analizy danych i jeszcze mniej uczenia maszynowego, to nie bałem się tych tematów (a nawet niekiedy zajmowanie się nimi było przyjemne). Na podstawie obserwacji potocznych sądzę, że wśród zdecydowanej większości moich koleżanek i kolegów humanistów występuje w różnych konfiguracjach strach, niechęć i brak zainteresowania podobnymi tematami (nawet analizą danych ankietowych w SPSS-ie; nawet zapoznaniem się z tabelami/wykresami sporządzonymi przez kogoś innego). Po trzecie, osobiście ważna jest dla mnie prywatność. Jestem introwertykiem i nigdy nie miałem ochoty na silną obecność w mediach społecznościowych, niemniej bez wątpienia lektury dotyczące zagrożeń prywatności i innych konsekwencji społecznych związanych ze zbieraniem i analizą danych o ludziach sprawiły, że zmieniłem swoje zwyczaje i narzędzia cyfrowe. W rodzinie i wśród znajomych jestem znany jako zachęcający do skasowania konta na Facebooku, używania innych niż Google wyszukiwarek, sprawdzania uprawnień instalowanych aplikacji, używania szyfrowanych email, stosowania adblocków, dbania o bezpieczeństwo haseł i pinów. {autoetnografia 12}

Opisywaną wyżej chęć wejścia do świata DS w sensie marzeń o podjęciu pracy zarobkowej w tym obszarze zdiagnozowałem u siebie pod koniec 2018 roku – wtedy, gdy już osłabła {autoetnografia 8}.

Moje usytuowanie jako osoby niebojącej się matematyki i jednocześnie humanisty (socjologa jakościowego) prowadziło do tego, że poświęciłem dużo uwagi technologiom badanego świata i starałem się udowodnić swoje – nietypowe dla humanistów – rozeznanie w tej materii. Rozeznanie w stosowanych technologiach jest niezbędne do zrozumienia świata społecznego DS. Początkowo szukałem potwierdzenia swojego rozeznania, np. pamiętałem zadowolenie z ukończenia pierwszego kursu języka R {obserwacja 3}, ze zdania testu umiejętności numerycznych {obserwacja 7}, z ukończenia studiów podyplomowych „Analiza danych i data mining" na Wydziale Matematyki i Informatyki Uniwersytetu Łódzkiego.

To usytuowanie przyczyniło się do głębszego zrozumienia dwóch aspektów świata DS – perspektywy osób chcących wejść do świata DS (tzw. wannabesów) lub osób początkujących w DS oraz roli tzw. zdolności matematycznych w praktykach legitymizacji i zaświadczania o autentyczności.

Utrudnieniem w zrozumieniu sytuacji kobiet w badanym świecie było to, że jestem mężczyzną i wszystkie wywiady oraz obserwacje prowadziłem osobiście. Płeć jest czynnikiem wpływającym na interakcje w czasie badań jakościowych (Konecki, 2000: 173–174; Charmaz, 2009: 42–43). Pomimo rozpoznania sytuacji kobiet w DS jako areny początkowo nie potrafiłem rozmawiać o tym z kobietami osadzonymi w świecie DS. Udało mi się uzyskać narracje rozmówczyń/rozmówców dopiero po konsultacjach z bardziej doświadczonymi socjolożkami i gruntownym przebudowaniu pytań dotyczących sytuacji kobiet. Starałem się opracować tę arenę, wychodząc poza męską perspektywę oraz unikając łatwej i pozornie „szlachetnej" roli obrońcy uciśnionych. Możliwe, że w tych poszukiwaniach motywowały mnie narodziny córki w 2017 roku.

Innym aspektem jest moja większa styczność z językiem R niż z Python. Te służące do wykonywania działania podstawowego i najbardziej popularne w badanym świecie języki programowania skryptowego wyznaczają dwa przenikające się subświaty społecznego świata DS w Polsce, a spór o przewadze jednego narzędzia nad drugim opisuję jako arenę. Decyzja o koncentracji na jednym języku – czyli R – nie była podjęta świadomie. Na etapie wczesnych poszukiwań w 2015 roku natrafiłem na kurs „Pogromcy Danych" (Biecek, 2015a; 2015b), który później ukończyłem w ramach badań {obserwacja 3}. W tym czasie język R był w sensie ilościowym bardziej popularny niż Python, trwająca przewaga popularności Pythona na świecie i w Polsce rozpoczęła się około 2018 roku (Nunns, 2017; Piatetsky, 2017; 2018c; Strong, 2018). Po zrozumieniu, że zajmuję się przede wszystkim subświatem DS posługującym się R, a także że społeczność użytkowników R jest znacznie szersza i nierzadko rozłączna ze światem DS, starałem się zwrócić w kierunku subświata użytkowników Pythona {obserwacje 30, 31, 39, 43}. Przyczyniło się to do lepszego zrozumienia aren technologicznych w świecie DS.

3.5. Kwestie etyczne w badaniach własnych

Badania etnograficzne, a szczególnie obserwacja uczestnicząca ukryta, są, z etycznego punktu widzenia, problematyczną techniką badawczą (Babbie, 2006: 336; Li, 2008; Angrosino, 2010: 121–125; Kacperczyk, 2016: 668–672). Znane są problemy braku świadomej zgody osób obserwowanych na udział w badaniach oraz posługiwania się fałszywą tożsamością przez wykonujących obserwacje badaczy.

Co do świadomej zgody, to w żadnym z aktów obserwacji nie ogłaszałem na forum faktu prowadzenia badań, a więc nie pytałem o taką zgodę ani w obserwacjach offline (np. na meetupie), ani online (np. biorąc udział w kursie). Obserwowałem wyłącznie to, co ma charakter ogólnodostępny. Nie zatrudniałem się, nie wstępowałem do żadnej formalnej ani nieformalnej organizacji, nie prowadziłem

obserwacji na spotkaniach prywatnych. Nie wszystkie obserwacje realizowano w miejscach publicznych (np. nie można uznać za publiczne warsztatów online z ograniczoną liczbą miejsc lub konferencji z udziałem płatnym), jednak miały one charakter ogólnodostępny (nie były miejscami prywatnymi ani z dostępem ograniczonym wyłącznie dla członków organizacji), a udział w nich był niezbędny do pozyskania danych o badanym wycinku rzeczywistości społecznej:

> Metody badań niejawnych pozostają w sprzeczności z zasadami świadomej zgody i mogą naruszać prywatność badanych. Do obserwacji uczestniczącej i nieuczestniczącej w sferze niepublicznej bez wiedzy badanych […] należy uciekać się wyłącznie wtedy, gdy inne metody nie wystarczają do pozyskania podstawowych danych (Walne Zgromadzenie Delegatów Polskiego Towarzystwa Socjologicznego, 2012, pkt 17).

Nie korzystałem w trakcie obserwacji z nagrań audio ani wideo. Brak świadomej zgody w moich obserwacjach uczestniczących nie zaszkodził osobom obserwowanym.

Zgadzam się ze stanowiskiem, że „ostatecznie decyzja, czy tworzyć autoetnografię analityczną czy ewokatywną jest decyzją o charakterze etycznym" (Kacperczyk, 2017: 141) i opowiadam się za typem analitycznym. Muszę wyjaśnić moje stanowisko. Kieruję się tym, co Kacperczyk nazwała etyką scjentystyczną, zakładającą, że:

> […] badacz ma prawo użyć instrumentalnie każdego źródła danych, by zgłębić analizowany temat i wydrzeć mu jego tajemnice. […] Analiza jest zawsze cięciem i kawałkowaniem, enzymatycznym procesem toczącym się na „danych", którymi od początku do końca zarządza badacz (Kacperczyk, 2017: 141).

Stąd wąskie zastosowanie etnografii opartej na współpracy – konsultowałem się z osobami uczestniczącymi w społecznym świecie DS, ale samodzielnie nadawałem ostateczny kształt konsultowanym fragmentom. Jest to z jednej strony zgodne z przyjętym przeze mnie stanowiskiem konstruktywistycznym – zakładam, że dokonywany w tej pracy opis świata społecznego jest **konstruowany** przez badacza, a nie **odkrywany** – a z drugiej strony zgodne z podejściem etycznym badanego przeze mnie świata społecznego DS.

Rozdział 4
Elementy społecznego świata data science

4.1. Data science w Polsce – ujęcie ilościowe

Bazując na badaniu przeprowadzonym przez Fundację Digital Poland w październiku 2018 roku na grupie ponad 160 firm[1] zajmujących się sztuczną inteligencją[2] (Borowiecki, Mieczkowski, 2019), informacjach o studiach i badaniach naukowych (Czarnowski i in., 2018; Ministerstwo Cyfryzacji RP, 2018b: 111–113; Borowiecki, Mieczkowski, 2019: 52; Gontarz, 2019: 34–35), aktach samowiedzy – wynikach sondaży wewnątrz świata DS (Migdał, 2017; Prokulski, 2017) oraz własnych analizach dotyczących wydarzeń w polskim świecie DS, stwierdzam, że **badany świat koncentruje się w największych ośrodkach miejskich, przy dominacji Warszawy**.

We wszystkich analizowanych wymiarach w Warszawie zlokalizowana jest co najmniej jedna czwarta (wymiar: liczba grup meetupowych) działalności w Polsce. W innych wymiarach (liczba członków grup meetupowych, liczba kierunków studiów, liczba firm) w Warszawie zlokalizowane jest co najmniej dwie piąte działalności[3]. Wśród pozostałych ośrodków miejskich nie ma jednoznacznych różnic w udziale w działalności w czterech analizowanych wymiarach.

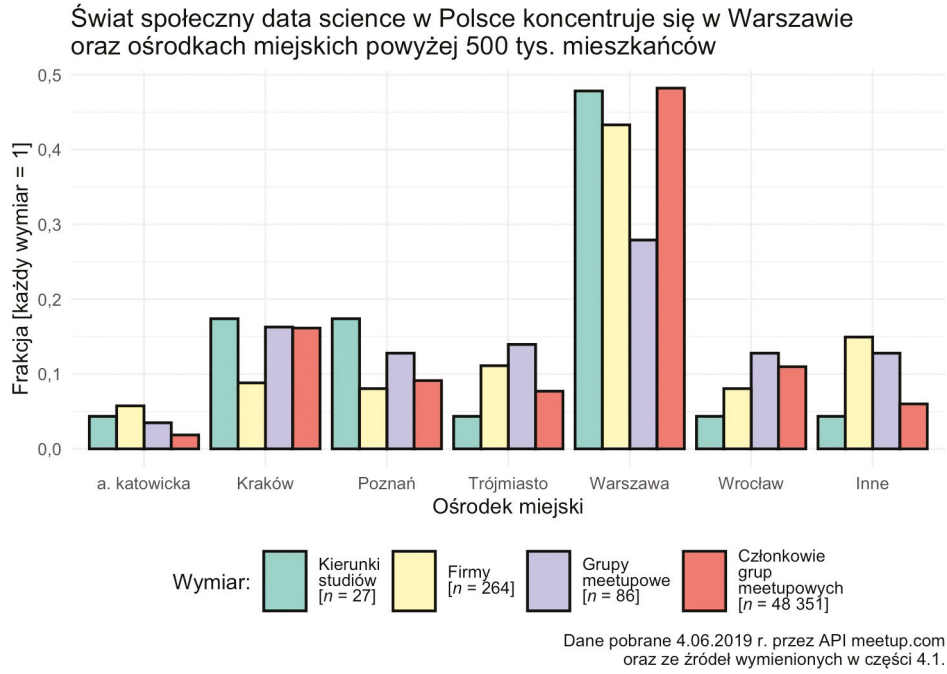
---

1 „Ponad 160 małych i dużych firm wzięło udział w naszym sondażu. Nasze badania wskazują, że w Polsce w obszarze AI działa ponad 260 firm" (Borowiecki, Mieczkowski, 2019: 10). Z materiałów dołączonych do raportu wynika, że są to 264 podmioty.

2 Zdefiniowano to następująco: „firmy rozwijające lub oferujące usługi lub produkty AI (*companies which develop or offer AI services or products*)" (Borowiecki, Mieczkowski, 2019: 14).

3 W przypadku firm i kierunków studiów nie porównuję liczby osób (pracowników, studentów), a tylko podmioty. W przypadku firm Fundacja Digital Poland zbierała informacje o ich wielkości, ale w raporcie *Map of the Polish AI* nie pokazano tej zmiennej w podziale na ośrodki miejskie. Fundacja odmówiła mi dostępu do kompletu danych źródłowych, powołując się na RODO i zgody udzielone przez podmioty uczestniczące w sondażu fundacji. Na rysunku 4.2 podaję liczbę 264 firm, zaznaczając, że jest to jedyna znana liczba firm AI w Polsce ogółem, wynikająca z materiałów dołączonych do raportu fundacji. Faktycznie w sondażu

Poniżej (rys. 4.1) lokalizacje studiów i meetupów zgrupowano do sześciu ośrodków miejskich w sposób zaproponowany przez autorów badania firm (Borowiecki, Mieczkowski, 2019). Dysponuję bardziej szczegółowymi informacjami, głównie jeżeli chodzi o meetupy. Prezentuję sumy liczby członków grup meetupowych we wszystkich polskich miastach, w jakich grupy istnieją, oraz liczbę członków na jedną grupę, także w podziale na miasta. Najmniejsza jest społeczność meetupowa w Rzeszowie, przy 135 osobach w jednej grupie, największa zaś w Warszawie – 23 306 członków w 24 grupach. W pięciu województwach – zachodniopomorskim, warmińsko-mazurskim, lubuskim, opolskim i świętokrzyskim – nie ma żadnej grupy meetupowej o temacie „data science". Najmniej osób na jedną grupę przypada w Lublinie (około 97 osób, przy sumie 387 członków w czterech grupach w tym mieście), a najwięcej – 971 osób na grupę – w Warszawie.

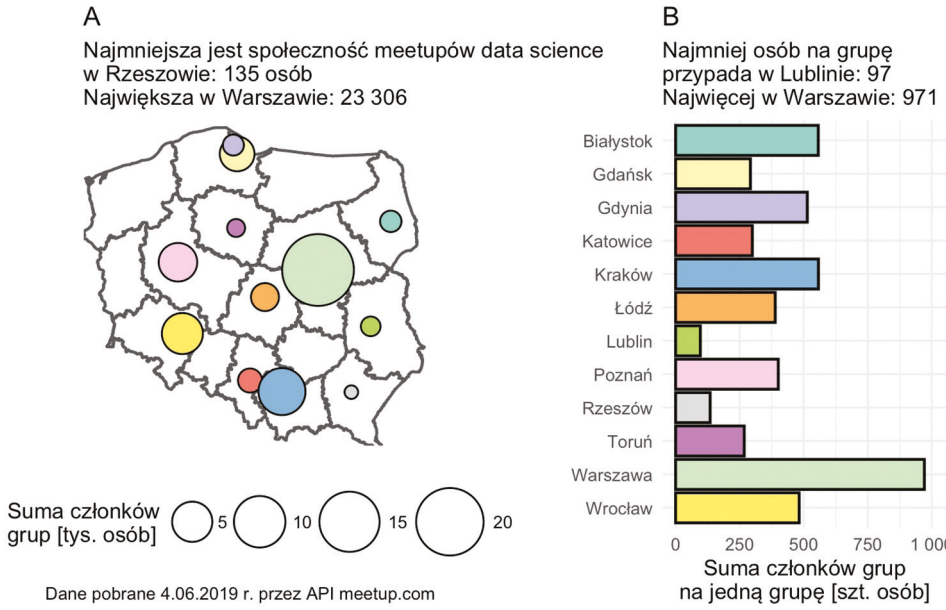


Rysunek 4.1. Lokalizacja działalności świata społecznego data science w Polsce – firmy, studia, meetupy

Źródło: opracowanie własne za pomocą GNU R.

udział wzięło „ponad 160 firm" (por. Borowiecki, Mieczkowski, 2019: 10). W przypadku kierunków studiów odstąpiłem od pracochłonnego, a mającego znikomą wartość do celów pracy, zadania wystosowywania próśb o podanie liczby studentów do każdej z uczelni lub odpowiedniej jej jednostki administracyjnej.

Liczebność grup meetupowych jest zróżnicowana w kraju i w obrębie miast. Ogółem minimalna liczebność to 45 osób, a maksymalna 5339. Średnia liczebność wynosi około 562 osób, mediana jest równa 296,5. Oznacza to, że liczebność grup ma rozkład niesymetryczny, prawoskośny – wiele jest grup nielicznych, a mało bardzo dużych. Prawoskośność tego rozkładu wskazuje wysoka, dodatnia wartość współczynnika skośności $A = 3{,}6$. Podobny rozkład występuje w miastach z największą liczbą grup[4]. Suma członków grup w mieście jest silnie dodatnio skorelowana z wielkością populacji miasta, współczynnik korelacji rang Spearmana $r_s = 0{,}91$. Przyjąłem stan populacji na 31 grudnia 2018 roku (System Wspomagania Analiz i Decyzji GUS, 2019).



Rysunek 4.2. Suma członków grup meetupowych data science i liczba członków na grupę w podziale na dwanaście polskich miast

Źródło: opracowanie własne za pomocą GNU R.

Prezentowane dane o grupach meetupowych mają cztery cechy, które trzeba wziąć pod uwagę, interpretując wyniki analizy. Po pierwsze, liczba członków grup jest zdecydowanie niższa niż liczba osób biorących udział w spotkaniach[5].

4 Warszawa: 24 grupy, A = 2,1; Kraków 14 grup, A = 1,7; po 11 grup: Gdańsk, A = 0,95; Poznań, A = 0,4; Wrocław A = 1,5.

5 Wśród analizowanych 86 grup w dniu pobrania przeze mnie danych 17 miało zaplanowane kolejne spotkanie. Maksymalna liczba kliknięć „RSVP", oznaczających potwierdzenie uczestnictwa w spotkaniu przez członków, wynosi 45% liczby wszystkich członków grupy (meetup Pearson Poznań AI & Tech, 224 uczestników, 101 potwierdzeń). W przypadku

Po drugie, te same osoby są zapisane do wielu grup o tej samej lub zbliżonej tematyce[6]. Liczbę członków należy uważać za wyraz zainteresowania grupą, nie można wnioskować na jej podstawie o wielkości świata społecznego. Po trzecie, fakt, że meetup istnieje, nie oznacza, że jest aktywny, tzn. organizuje spotkania. Po czwarte, wybrałem prostą technicznie metodę wyszukania wszystkich meetupów z zapisanym tematem „data science". Nie wszystkie te meetupy są postrzegane jako część świata społecznego DS w Polsce.

Demografię badanego świata próbuję przedstawić w wymiarach struktury płci, wieku i wykształcenia uczestników. **Ogółem występuje zdecydowana przewaga mężczyzn[7], dominacja osób w wieku 25–34 lat[8], mających wykształcenie magisterskie[9] i wykształcenie w kierunkach informatycznych[10].** Jest tak również dla skrzyżowania tych cech. Mężczyźni w wieku 25–34 lata z wykształceniem magisterskim w kierunku informatycznym stanowią nieco ponad 13% respondentów[11] i są najczęściej występującą grupą[12].

Korzystam przede wszystkim z analiz własnych. Wykonałem je na publicznie dostępnych danych pochodzących z międzynarodowych ankiet przeprowadzonych wśród użytkowników Stack Overflow (w 2018 i 2019 roku) i Kaggle (w roku 2017 i 2018) oraz z formularzy rejestracyjnych odbywających się w Polsce konferencji „WhyR?" w 2017 i „PyData" w 2018 (por. podrozdział 3.3).

W świecie DS zdecydowanie najwięcej jest mężczyzn. Spośród analizowanych źródeł najniższą proporcję – około dwóch mężczyzn na jedną kobietę – ma konferencja „WhyR? 2017". Oznacza to, że mężczyźni stanowili prawie 70% ankietowanych przy nieco ponad 30% kobiet. Najwyższa proporcja 13:1 (około 65% mężczyzn do 5% kobiet) wystąpiła w ankiecie Stack Overflow (SO) 2018. Kategoria

najliczniejszych grup odsetek potwierdzeń jest znacznie niższy, np. dla PyData Warsaw – 7% (261 potwierdzeń).

6 Jestem członkiem ośmiu z analizowanych grup meetupowych.

7 Dla każdego z analizowanych źródeł danych „mężczyźni" stanowią co najmniej 65% wszystkich odpowiedzi na pytanie o płeć, łącznie z brakami/odmowami odpowiedzi.

8 Grupa wiekowa 25–34 lata to co najmniej 25% respondentów dla źródeł zawierających informację o tej zmiennej, dla każdego z analizowanych źródeł, łącznie z brakami i odmowami odpowiedzi.

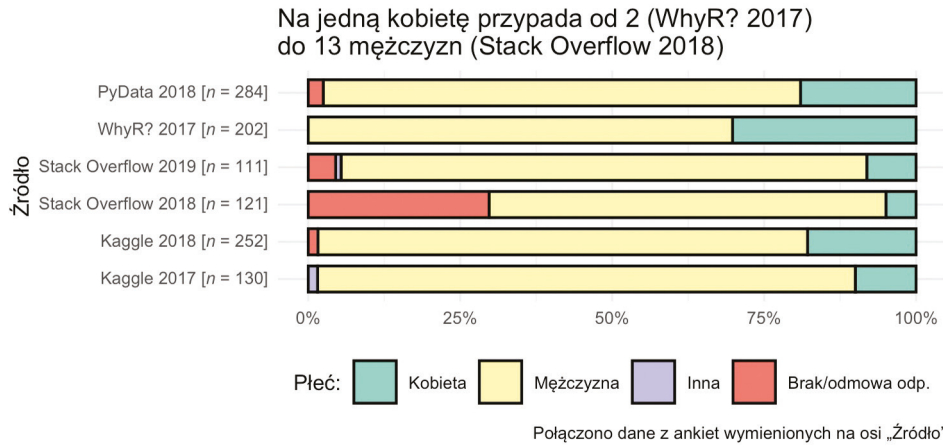
9 Wykształcenie magisterskie deklarowało co najmniej 40% badanych, dla każdego z analizowanych źródeł, łącznie z brakami i odmowami odpowiedzi.

10 Ukończenie kierunku informatycznego wskazywało co najmniej 28% uczestników badań, dla każdego z analizowanych źródeł, łącznie z brakami i odmowami odpowiedzi.

11 Przy połączeniu źródeł danych ankietowych Kaggle (2017b i 2018b) i Stack Overflow (2018 i 2019) oraz usunięciu braków i odmów odpowiedzi $n = 493$ osoby. Wymienione źródła danych omawiam dalej.

12 Druga najczęściej występująca grupa różni się od pierwszej tylko kierunkiem wykształcenia – matematycznym lub statystycznym i stanowi niecałe 8% z 493 respondentów zdefiniowanych, dla każdego z analizowanych źródeł, łącznie z brakami i odmowami odpowiedzi. Trzecia to mężczyźni w wieku 18–24 lata z wykształceniem informatycznym na poziomie licencjata lub inżyniera (około 7%), czwarta to mężczyźni w wieku 35–44 lata z wykształceniem informatyczny na poziomie magistra (około 4,5%).

płci „Inna" została uwzględniona w ankietach międzynarodowych[13], natomiast w polskich nie. Odsetek wskazań „Inne" nie przekroczył 2% dla żadnego źródła. W SO 2019 taką kategorię wskazała jedna osoba, w Kaggle 2017 dwie. Odsetek kobiet jest wyższy dla późniejszego badania międzynarodowego w ramach tego samego realizatora[14]:



Rysunek 4.3. Płeć uczestników według ankiet związanych z polskim światem społecznym data science

Źródło: opracowanie własne za pomocą GNU R.

W każdym ze źródeł najczęściej wskazywano wiek od 25 do 34 lat. Dla czterech źródeł drugą najczęściej wskazywaną kategorią był przedział wiekowy 18–24 lata. Dla „WhyR?" drugim najczęściej wskazywanym był przedział od 35 do 44 lat (około 6% odpowiedzi), a 18–24 był trzecim pod względem częstości przedziałem (2% wskazań). W żadnym ze źródeł respondenci nie wskazali wieku wyższego niż 69 lat. Osoby w wieku 55–69 stanowiły nie więcej niż 2% w ankietach Kaggle, w innych odsetek był jeszcze niższy.

W każdym ze źródeł najczęściej podawano wykształcenie magisterskie. Najrzadziej tę kategorię wskazywano w „WhyR? 2017" (około dwóch piątych odpowiedzi), ale wysoki był udział braków/odmów odpowiedzi[15]. Osoby z wykształceniem magisterskim stanowiły około 62% respondentów Kaggle 2018 i około połowy

13 Na przykład w badaniu Kaggle 2017 kategorię nazwano „[osoba] niebinarna, genderqueer lub gender-niekonformistyczna" (*non-binary, genderqueer, or gender non-conforming*).

14 Dla Kaggle 2018 jest to wzrost o około 8 p.p. względem 2017 roku (odpowiednio 18% i 10%), dla Stack Overflow 2019 o 3 p.p. względem roku 2018 (odpowiednio 8% i 5%).

15 Pytanie, na podstawie którego prezentuję poziom wykształcenia dla „WhyR? 2017" brzmiało: „Stopień naukowy do umieszczenia na certyfikacie uczestnictwa w konferencji (o ile dotyczy)". Poziom wykształcenia podały zatem tylko osoby, które chciały otrzymać potwierdzenie uczestnictwa w konferencji.

w innych źródłach. Udział osób z wykształceniem pierwszego stopnia (licencjat lub inżynier) wynosił od 8% („WhyR? 2017") do 31% („Stack Overflow 2018"). Dla każdego źródła była to druga najczęściej wskazywana kategoria. Udział osób mających doktorat sięgał jednej szóstej dla „WhyR? 2017" i Kaggle (2018). Odsetek osób z doktoratem był – podobnie jak w przypadku odsetka kobiet – wyższy dla późniejszego badania międzynarodowego w ramach tego samego realizatora[16].

Ankiety międzynarodowe zawierają także informację o kierunku wykształcenia badanych. Kierunki określano na każdym poziomie wykształcenia. W każdym ze źródeł międzynarodowych najczęściej wskazywano wykształcenie informatyczne. Najniższy odsetek tego kierunku wykształcenia występuje w Kaggle 2017 (nieco ponad 28%), najwyższy zaś w Stack Overflow 2018 (około 54%). Drugim najczęściej wymienianym było, w przypadku trzech źródeł, wykształcenie matematyczne lub statystyczne (około 12% dla Stack Overflow 2018 i po około 18% dla obu ankiet Kaggle). Drugi najczęściej wskazywany kierunek był inny dla Stack Overflow 2019 – tam wykształcenie medyczne lub przyrodnicze wybierano nieco częściej (około 12%) niż matematyczne/statystyczne (10% odpowiedzi). Nauki medyczne/przyrodnicze wskazywano nieznacznie częściej niż techniczne. Jedynie w przypadku Kaggle pierwszy z wymienionych obszarów wybrało 7%, a drugi niecałe 8% badanych. Autorzy ankiet uwzględnili możliwość podania wykształcenia biznesowego/ekonomicznego w badaniach późniejszych, tzn. 2018 dla Kaggle i 2019 dla Stack Overflow. Ten kierunek wykształcenia wyróżnia się dziesięcioprocentowym udziałem wskazań w badaniu Kaggle 2018. Wykształcenie społeczne wskazywano rzadziej – było to od 2% odpowiedzi dla Stack Overflow 2019 do około 4% w pozostałych źródłach. Obszar humanistyczny nie przekraczał 2,5% wskazań (Stack Overflow 2018). Odsetek odpowiedzi „Inne" był najwyższy w ankiecie Kaggle 2017 (około 3%), w tym źródle najczęściej odnotowano odmowę/brak odpowiedzi (około 30%).

Przyjrzyjmy się charakterystyce ekonomicznej podmiotów zajmujących się AI oraz rynkowi pracy dla data scientistów w Polsce. W pierwszym z obszarów bazuję na raporcie Fundacji Digital Poland:

1) około **jednej trzeciej badanych podmiotów uzyskuje dochód głównie** – w co najmniej 70% – **ze zleceń zagranicznych** (Borowiecki, Mieczkowski, 2019: 6, 20);
2) ponad **20% badanych firm nie uzyskało dotychczas dochodu z produktów i usług bazujących na AI** (Borowiecki, Mieczkowski, 2019: 19);
3) około **dwóch trzecich badanych firm finansuje swoje działanie całkowicie samodzielnie** (Borowiecki, Mieczkowski, 2019: 15);
4) ponad **jedna trzecia firm jest przynajmniej częściowo finansowana przez kapitał zagraniczny**; należą do tej grupy także podmioty wymienione

16 Dla Kaggle (2018) jest to wzrost o około 6 p.p. względem 2017 roku (odpowiednio 16% i 10%), dla „Stack Overflow 2019" o 3 p.p. względem roku 2018 (odpowiednio 7% i 4%).

w punkcie 3 jako finansujące swoje działania całkowicie samodzielne (Borowiecki, Mieczkowski, 2019: 15);
5) firmy **najczęściej świadczą usługi w zakresie analityki, big data i business intelligence (43%), najrzadziej zaś w obszarach wojsko i obronność (3%)** (Borowiecki, Mieczkowski, 2019: 23–24);
6) zadania, jakie wykonywane są w ramach oferowanych usług, to **najczęściej przetwarzanie i rozpoznawanie obrazów (62%)** (Borowiecki, Mieczkowski, 2019: 27–28);
7) pomimo że wielkość badanych firm zajmujących się AI jest zróżnicowana, to **zespoły techniczne zazwyczaj są nie większe niż pięć osób (58%)** (Borowiecki, Mieczkowski, 2019: 32);
8) około **77% firm współpracuje z uczelniami wyższymi**; **prawie dwie piąte firm opublikowało artykuł w czasopiśmie naukowym** (Borowiecki, Mieczkowski, 2019: 44–46).

Co do rynku pracy dla data scientistów w Polsce, to warto odnotować ogromny entuzjazm wobec tego zajęcia. Pojawiło się wiele tekstów dotyczących np. data scientisty jako zawodu przyszłości (Kodołamacz.pl i Centrum Zarządzania Innowacjami i Transferem Technologii Politechniki Warszawskiej, 2017; Nurczyk, Ramza, 2017) czy specjalistów od big data, którzy są „na wagę złota" (Błaszczak, 2016). Światowe Forum Ekonomiczne wskazuje, że zawód Data Analysts and Scientists w latach 2018–2022 rozwijał się we wszystkich branżach i regionach geograficznych (Centre for the New Economy and Society, 2018: 43–65, 68–125). Ministerstwo Cyfryzacji RP twierdzi, że w Unii Europejskiej brakuje 400 tys. pracowników na „rynku danych" i luka ta będzie się powiększać (Borowik i in., 2018).

Istnieje jedna publikacja dotycząca polskiego rynku pracy dla zawodów związanych z danymi cyfrowymi – nie tylko DS – przygotowana przez portal Enterprise Software Review (Gontarz, 2019). Opracowanie to otrzymałem, uczestnicząc w konferencji Big Data Technology Summit {obserwacja 37}. Wstęp pokazuje entuzjazm wobec rosnącego popytu na pracę w tych zawodach. W publikacji podkreślono wysokie zarobki, elitarność umiejętności i kompetencji umysłowych oraz zagrożenie polskiego rynku „wysysaniem" pracowników za granicę (Gontarz, 2019: 4). Podano przedziały miesięcznych zarobków brutto data scientistów w Polsce, w podziale na poziom doświadczenia, bazując na danych z raportu płacowego dla branży IT firmy Sedlak & Sedlak. Podane przedziały należy chyba rozumieć jako minima i maksima, tzw. widełki wynagrodzenia. Te minima i maksima rosną – widełki poszerzają się wraz ze wzrostem poziomu doświadczenia. Dla młodszych specjalistów wynoszą one 5264–6362 zł, dla doświadczonych 6709–8825 zł, natomiast dla starszych od 9812 zł do ponad 14 000 zł (Gontarz, 2019: 25, wykres 1).

O zarobki i poziom doświadczenia pytano także w międzynarodowych ankietach Kaggle 2017 i 2018 oraz Stack Overflow 2018 i 2019. W ankiecie Kaggle z 2017 roku poproszono respondentów o wpisanie dokładnej kwoty. Nie wskazano,

czy chodzi o zarobki brutto, czy netto. Pytanie o doświadczenie miało pięć kategorii – przedziałów różnej szerokości. Zależność widoczna w przywołanym wyżej raporcie (Gontarz, 2019: 25) zachodzi tutaj dla trzech środkowych kategorii doświadczenia zawodowego: 1–2 lata, 3–5 lat i 6–10 lat. Ogółem minima i maksima są odpowiednio znacznie niższe i wyższe w ankiecie Kaggle 2017 niż w polskim raporcie. W przypadku kategorii dotyczącej najdłuższego doświadczenia zawodowego – powyżej 10 lat – część statystyk (Q1, Q3) jest niższa niż dla kategorii 6–10 lat. Minima i maksima są jednak wyższe, a więc najwyższe w opisywanym przekroju. Osoby z najdłuższym doświadczeniem rozpoczynały karierę, gdy nie używano terminu „data scientist". Jeżeli obecnie nie posługują się tym terminem, ich praca może być uważana za mniej prestiżową, a więc nieco niżej opłacaną niż tych, którzy zbudowali swoją karierę jako data scientiści.

W ankiecie Kaggle 2018 respondenci wybierali przedział zarobków. W niej także nie ma informacji, czy podano kwoty brutto, czy netto. Przeliczyłem roczne kwoty zarobków w USD na kwoty miesięczne w PLN. Tak jak w Kaggle 2017 zależność wzrostu zarobków wraz z doświadczeniem zawodowym nie zachodzi dla najwyższej kategorii doświadczenia.

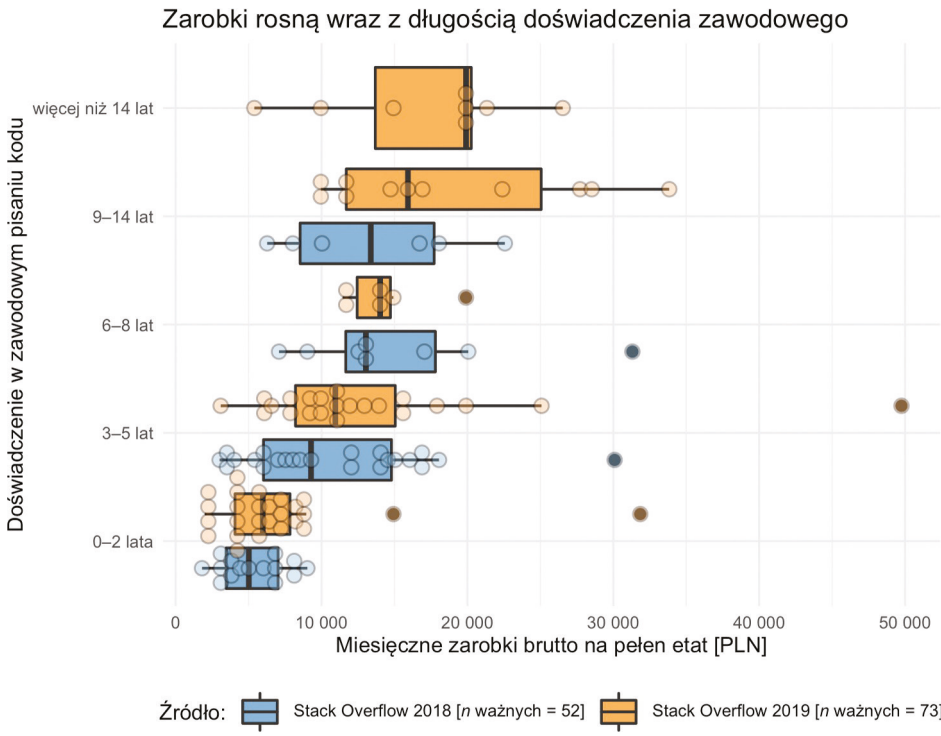
W ankietach Stack Overflow 2018 i 2019 poproszono respondentów o wpisywanie dokładnej kwoty zarobków brutto i przeliczono to na zarobki na pełen etat. Zarobki dla tych samych kategorii doświadczenia zawodowego wzrosły w czasie.

Występują wartości odstające zarobków – rzędu 30–50 tys. złotych brutto, podczas gdy w polskim raporcie Sedlak & Sedlak wskazywano pułap około 14 tys. jako górną granicę tzw. widełek wynagrodzenia dla najbardziej doświadczonych data scientistów. Jak wynika z badań własnych oraz np. z omawianego wyżej tekstu (Gontarz, 2019: 4), te najwyższe wynagrodzenia mogą uzyskiwać osoby pracujące zdalnie, z Polski, w ramach kontraktów, np. dla USA.

Oferty dotyczące przetwarzania i analizy danych pojawiają się na portalu Pracuj.pl najczęściej w branży „Badania i rozwój", a następnie w „IT – rozwój oprogramowania". Łukasz Prokulski podzielił stanowiska na trzy grupy: **data scientist, data engineer i data analyst**. **Te stanowiska to łącznie niecałe 0,35% (162 oferty) spośród 46 536 ofert pracy na terenie Polski** zebranych przez niego z Pracuj.pl w dniach 8–12 lutego 2019 roku. Najrzadziej występowały stanowiska data engineer (28 ofert, około 17% z 162), więcej było ofert dla **data scientistów (46 ofert, prawie 28,5%)**, najczęściej oferowano stanowiska analityków danych (88 ofert, ponad 54%). Stanowiska analityków występowały równie często w kategorii „Finanse/Ekonomia" co w branży rozwoju oprogramowania. Łącznie omawiane stanowiska pojawiły się w czternastu różnych branżach (Prokulski, 2019).

Pracuj.pl jest serwisem ogólnobranżowym. Oferty pracy dotyczące konkretnych branż mogą pojawiać się na innych, wyspecjalizowanych serwisach. Prokulski (2019) mówi o odpływie ofert branży IT z Pracuj.pl. Uzyskana z tego portalu liczba ofert pracy dotyczących przetwarzania i analizy danych jest około dwukrotnie

zaniżona[17], ale rozkład liczby ofert dla grup stanowisk data analyst, data scientist i data engineer uważam za oddający postrzeganie tych stanowisk w badanym świecie społecznym.



Rysunek 4.4. Miesięczne zarobki brutto data scientistów w Polsce w podziale na liczbę lat doświadczenia

Źródło: opracowanie własne za pomocą GNU R na podstawie Stack Overflow 2018 i 2019.

Choć w Polsce **już w latach dziewięćdziesiątych XX wieku stosowano komercyjnie to, co obecnie bywa nazywane AI, to DS jako nazwa pojawia się od 2014 r., natomiast rozkwit zainteresowania DS widoczny jest od roku 2017**.

W raporcie Fundacji Digital Poland wskazano, że 3% badanych podmiotów zaczęło stosować AI w latach dziewięćdziesiątych, a połowa w latach 2017–2018, przy

17 Oferty publikowane są np. na stronach WWW firm i udostępniane na grupie facebookowej Data Science PL, na LinkedIn, na portalach nofluffjobs.com, pl.indeed.com, careerjet.pl, glassdoor.com, upwork.com (portal służy raczej do oferowania i przyjmowania zleceń, nie do zatrudniania). Działania związane z oferowaniem pracy to także to, co dzieje się w kontaktach twarzą w twarz: targi pracy w formule stoisk odbywały się np. na konferencji Data Science Summit w 2019 roku, a przedstawiciele firm mówią bądź prezentują wizualnie (na slajdach, roll-upach) co najmniej informację o fakcie poszukiwania kandydatów do pracy podczas niemalże każdej konferencji czy na meetupach.

czym badanie zrealizowano w październiku 2018 roku (Borowiecki, Mieczkowski, 2019: 13). Z moich analiz grup meetupowych wynika, że najstarszą z działających grup o tematyce DS była DataKRK (dawniej Cracow Hadoop User Group) z Krakowa, założona w styczniu 2012 roku. W roku 2013 nie powstały żadne istniejące grupy, a w 2014 założono ich sześć[18]. W kolejnych latach liczba powstających grup rosła, poza rokiem 2016. W 2017 i 2018 powstały odpowiednio 21 i 23 grupy. Do dnia pobrania danych w 2019 roku powstało 10 grup, co może wskazywać na podobną liczbę pojawiających się grup w trzecim roku z rzędu. Analizuję dane dotyczące grup istniejących na dzień pobrania danych. Nie mogę na tej podstawie powiedzieć nic o grupach, które działały wcześniej, ale rozwiązano je przed 4 czerwca 2019 roku.

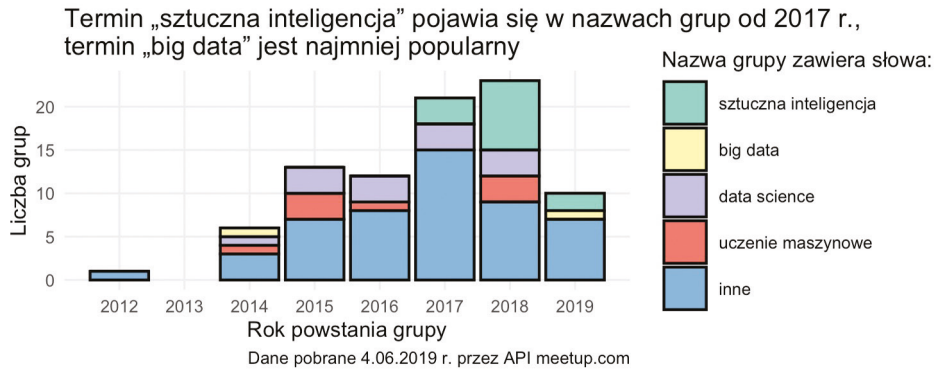
Od 19 lipca 2015 roku funkcjonuje najważniejsza platforma komunikacji polskiego świata społecznego DS, czyli grupa facebookowa Data Science PL[19]**.** Piotr Migdał planował założenie grupy facebookowej dla grona znajomych zajmujących się DS, którego liczebność oceniał wówczas na około 50 osób. Na 25 czerwca 2019 roku grupa ta liczyła 9047 osób, z czego w dniach 29 maja – 25 czerwca aktywnych (co najmniej jeden post, komentarz, reakcja) członków było 8089 (około 90%). Facebook umożliwia administratorowi grupy śledzenie liczby jej członków od lipca 2018 roku, nie mogę zatem pokazać całej historii zmian tej liczby w czasie. Wiem jedynie, że w lipcu 2018 roku było to około 5 tys. osób. Do 25 czerwca 2019 roku liczba ta wzrosła do około 9 tys. osób, a więc o ponad 80%.

Kończąc próbę przedstawienia badanego świata w czasie, prezentuję wybrane terminy w nazwach analizowanych grup. Koncentruję się na czterech kluczowych pojęciach. Rozpoczęcie stosowania terminu „sztuczna inteligencja" w nazwach meetupów zaczęło się w 2017 roku i było kontynuowane przez kolejne lata. **Rok 2017 to moment początkowy trwającego wciąż rozkwitu zainteresowania DS w Polsce.** Termin „AI" jest w świecie społecznym DS postrzegany jako termin nietechniczny, a co za tym idzie – nieprecyzyjny. Stosowany jest w charakterze hasła marketingowego, mającego zaciekawiać i wzbudzać emocje. Terminu tego używają także osoby niebędące częścią świata społecznego DS, a przede wszystkim stosowany jest on w komunikacji skierowanej do osób spoza badanego świata. Używanie terminu „AI" w nazwach grup meetupowych może być elementem strategii rekrutowania członków grup wśród osób na bardzo wczesnym etapie zainteresowania problematyką przetwarzania i analizy danych. Zbliżoną funkcję pełnił wcześniej termin „big data", jednak w nazwach obecnie istniejących grup występuje on najrzadziej z wybranych terminów (dwie na 86 nazw). Obecnie termin „big data" stracił swą niedawną siłę oddziaływania marketingowego i jest uznawany przede wszystkim za termin techniczny, dotyczący metod i technologii

18 Były to grupy: 1) Data Science Warsaw; Big Data, Warsaw; Spotkania Entuzjastów R-Warsaw RUG Meetup & R-Ladies Warsaw w Warszawie; 2) Rka/Cracow R Users Group Kraków; INIME na żywo Kraków w Krakowie; 3) TriCity Machine Learning Meetup Gdynia w Gdyni.

19 Informacje w tym akapicie pochodzą z nieformalnych rozmów z założycielem i administratorem grupy, Piotrem Migdałem.

składowania i przetwarzania zbiorów danych (np. za pomocą narzędzi Hadoop i Spark), nie dotyczy analizy danych ani ML. Termin ten w sensie technicznym należy do informatyki, ewentualnie świata przenikającego się z DS – inżynierów danych. Brak jest trendów, jeżeli chodzi o posługiwanie się określeniami „data science" i „uczenie maszynowe" w nazwach meetupów (rys. 4.5).



Rysunek 4.5. Wybrane terminy w nazwach 86 grup meetupowych data science według roku powstania grupy (styczeń 2012 – czerwiec 2019)

Źródło: opracowanie własne za pomocą GNU R.

4.2. Sposoby definiowania działania podstawowego w data science

„Serce świata społecznego stanowi *podstawowe działanie* (*primary activity*). Wokół niego koncentruje się cała energia uczestników. Stanowi ono także kryterium wyodrębnienia społecznego świata jako pewnej całości, bo jego uczestnikami są aktorzy podejmujący takie działanie" (Kacperczyk, 2016: 34–35).

Co właściwie robią ludzie zajmujący się DS? **Działania podstawowego badanego świata nie można ująć jako „analizy danych".** Analizą danych – w sensie czynności, a nie działania podstawowego – zajmuje się co najmniej kilkanaście środowisk, które nie są tożsame z DS (Granville, 2014). Dotyczy to także naukowców – akademików, którzy przecież również analizują dane (Azam, 2014).

Nie można ująć działania podstawowego świata DS jako analizy danych, ponieważ nie stanowi to kryterium wyodrębnienia całości społecznej. Z drugiej strony jest to ujęcie zbyt wąskie, bo świat DS po części zajmuje się analizą danych, ale nie tylko tym. Ponadto istnieje nazwa stanowiska pracy „analityk danych" (*data analyst*), a ludzie wykonujący pracę analityka, traktowaną często jako coś, co robiło się za pomocą arkuszy kalkulacyjnych i zapytań do baz danych w języku SQL już trzydzieści lat temu, niekoniecznie uważani są za uczestników świata społecznego DS.

4.2.1. Działanie podstawowe data science

Działanie podstawowe społecznego świata DS w Polsce określam jako pisanie kodu do przetwarzania, analizy i modelowania danych.

Pisanie kodu oznacza czynność zapisywania – za pomocą klawiatury komputerowej – serii instrukcji do wykonania przez komputer. Chodzi o własnoręczne pisanie instrukcji w formie tekstu, tzw. linijek kodu, przez człowieka w języku programowania[20] zrozumiałym dla komputera (Nowosad, 2019). Nie używam słowa „programowanie", żeby zwrócić uwagę na **fizyczną czynność**[21] **pisania kodu** oraz by zaznaczyć, że **DS jest światem odrębnym od świata programowania** – inżynierii oprogramowania (*software engineering*) i tworzenia oprogramowania (*software development*). Między tymi światami występują przecięcia, ale te światy nie są tożsame ani nie pozostają w relacji hierarchicznej – DS nie jest podzbiorem, podgrupą czy częścią „wewnątrz" świata programowania. Rozróżnienie na kodowanie i programowanie występuje też w żargonie osób umiejących programować, o czym dalej.

Programowanie oznacza zarówno umiejętność (osoba potrafi programować w sensie znajomości składni języka programowania i koncepcji programowania[22]), jak i proces umysłowy (osoba konceptualnie opracowuje, co ma zostać wykonane, tłumaczy wymyśloną czynność na język programowania), samą czynność pisania (tutaj słowa „programowanie" i „kodowanie" oznaczają to samo) oraz tworzenie oprogramowania (czyli np. aplikacji, stron internetowych – tym nie zajmuje się świat DS). **Uczestnicy świata społecznego DS potrafią zatem programować, wykonują proces umysłowy przetłumaczenia ludzkiego pomysłu na język programowania, ale programując, piszą kod, a nie tworzą oprogramowanie.**

Zapisany w języku programowania zbiór instrukcji wykonujący określone zadania określa się mianem skryptu:

---

20 Najpopularniejsze języki programowania w DS to Python i R.

21 Jest to zatem element namacalny (*palpable*) (Strauss, 1978: 121). Tę namacalność, czyli pisanie kodu jako czynność fizyczną na klawiaturze komputera przy patrzeniu w ekran, podkreśla się w komunikatach skierowanych do osób uczących się programowania. Porównuje się pisanie kodu do gry na instrumencie muzycznym, mówiąc o konieczności regularnego ćwiczenia i stopniowego nabierania wprawy w początkowo niewykonalnych czynnościach. „Musisz **napisać na klawiaturze** (*type*) [podkr. oryg.] każdy z tych przykładów, własnymi rękoma. Jeśli kopiujesz i wklejasz [przykłady z książki – przyp. R.Ż.], możesz równie dobrze wcale tego nie robić. Te przykłady mają ćwiczyć Twoje ręce, mózg (*brain*) i umysł (*mind*) w czytaniu, pisaniu i dostrzeganiu (*see*) kodu. Jeśli kopiujesz i wklejasz, oszukujesz siebie" (Shaw, 2014: 2–3).

22 Na podstawie mojego niewielkiego doświadczenia z językiem R wymieniam np.: pojęcia obiektów prostych (wektory, macierze) i złożonych (listy, ramki danych), indeksowanie, zmienne i przypisywanie zmiennych, typy danych, operatory logiczne, funkcje, pakiety, pętle, instrukcje warunkowe (Biecek, 2015a; Wickham, Grolemund, 2017; Nowosad, 2019).

> *Badacz:* […] *ciężko mi jest zrozumieć taką różnicę, niektórzy mówią, że to jest jakaś duża różnica, a niektórzy, że nie, różnica pomiędzy data science a programowaniem, tak w ogóle, nie?* […].
> Rozmówczyni/rozmówca: […] Więc na pewno data scientiści programują, aczkolwiek też to no można, można to próbować rozgraniczyć w ten sposób, że jeżeli ktoś pisze skrypt, który coś robi, no, powiedzmy, bierze jakieś dane, je przetwarza w jakiś sposób i wypluwa, nie wiem, tabelkę na koniec, no to można powiedzieć, że to nie jest, chociaż to też jest taki **żargon** trochę, że to nie jest programowanie, tylko **kodowanie** albo **pisanie skryptów**, podczas gdy programowaniem można by nazwać faktycznie właśnie jakąś aplikację albo program, który byłby przygotowany do robienia czegoś **ciągle**, […] data science zajmuje się właśnie, no, tą węższą częścią wyciągania wniosków z danych, przygotowywaniem modeli, a niekoniecznie właśnie tworzeniem programu, który będzie ten model, nie wiem, obsługiwał. Jakoś tak, coś takiego. {wywiad 9}[23]

Data science to przede wszystkim zajmowanie się danymi, a nie tworzenie oprogramowania czy automatyzacja operacji. Prosty zbiór instrukcji wykonujący operacje na danych w języku R wygląda następująco (por. Biecek, 2015a):

```
sekwencja <- seq(0, 10, 0.1)
poziom <- exp(sekwencja)
plot(x = sekwencja, y = poziom, xlab = "czas spędzony z R",
  ylab = "poziom Data Science we krwi")
```

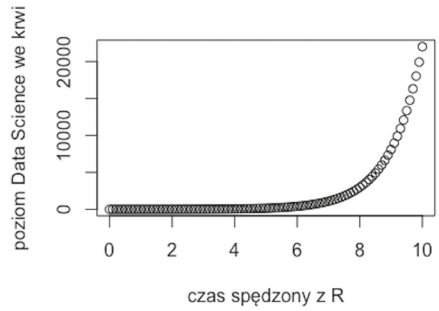
Instrukcje to kolejno utworzenie sekwencji liczb funkcją `seq()` i przypisanie jej do zmiennej *sekwencja*, dalej obliczenie eksponenty funkcją `exp()` dla każdej liczby w *sekwencji* i przypisanie wyniku do zmiennej *poziom*, na koniec narysowanie wykresu funkcją `plot()`, dla której podawane są wartości argumentów – dwie przypisane we wcześniejszych liniach zmienne i dwie wartości tekstowe. Ten zbiór instrukcji zapisany do pliku *wykres_ds_we_krwi.R* to właśnie skrypt.

Po uruchomieniu skryptu instrukcją:

```
source("wykres_ds_we_krwi.R")
```

otrzyma się wynik w postaci wykresu:

23 W cytowanych fragmentach wywiadów moje wypowiedzi oznaczam kursywą i skrótem B (badacz), a wypowiedzi uczestników zwykłym krojem pisma i skrótem R (rozmówczyni/rozmówca). Wykonałem transkrypcje dosłowne, tak samo dosłowne są cytaty. Pogrubieniem zaznaczam słowa wypowiadane z naciskiem, w nawiasach kwadratowych elementy niewerbalne [np. pauza, śmiech lub R pokazuje zdjęcie na telefonie] lub wyjaśnienia.



Rysunek 4.6. Poziom data science we krwi według czasu spędzonego z GNU R

Źródło: Biecek, 2015a.

Na wykresie widać efekt działania skryptu. Pozycję każdego narysowanego punktu określają liczby zapisane w zmiennych *sekwencja* (oś x) i *poziom* (oś y), które podano jako wartości argumentów `x` i `y` funkcji `plot()`, a osie opisane są wartościami argumentów `xlab` i `ylab` tej funkcji. Nim odniosę się do treści powyższego wykresu, chcę pokazać, czym jest skrypt wykonujący operacje na danych. Choć jest on w podanym przykładzie bardzo prosty, a dane fikcyjne, to wykonywana operacja – wizualizacja danych na wykresie – jest jednym z typowych zadań data scientistów. Skryptu nie uznaje się za oprogramowanie (*software*). Posłużę się przejaskrawionym porównaniem powyższego skryptu ze znanym w naukach społecznych oprogramowaniem analitycznym SPSS. Ten skrypt można uruchomić w środowisku R, a następnie zobaczyć wynik jego działania, tj. wykres. Oprogramowanie SPSS można także uruchomić przez wiersz poleceń[24], ale typową strategią użytkownika SPSSa jest raczej dwukrotne kliknięcie w odpowiednią ikonę na pulpicie. Po uruchomieniu aplikacji użytkownik ma możliwość klikania w menu, ewentualnie pisania kodu wewnątrz aplikacji i wykonania wielu działań – od załadowania danych, przez operacje na nich, do zapisania plików wynikowych w wybranym formacie. Omawiany skrypt nie daje możliwości ani załadowania danych, ani wyboru operacji, ani zapisu wyniku, a przede wszystkim nie daje możliwości klikania w cokolwiek, bo nie ma interfejsu graficznego. Nie jest to „program do użytku przez ludzi" {wywiad 9}. Ten skrypt i skrypty w ogóle nie są przeznaczone do interakcji z użytkownikiem nieumiejącym programować[25]. Osoba umiejąca programować może ręcznie zmienić kod zapisany w skrypcie i uzyskać inny wynik. W powyższym przykładzie może być to zmiana wartości argumentów funkcji `seq()`, co spowoduje wykonanie tych samych operacji na innych

24 Zwany też konsolą, terminalem czy powłoką (*shell*), o czym więcej w części poświęconej technologiom.

25 Chyba że skrypt jest jednym z wielu elementów gotowej aplikacji, niemniej użytkownik końcowy nie ma z takimi skryptami bezpośredniej interakcji, ewentualnie interakcję zapośredniczoną przez interfejs graficzny.

danych. Skrypt po modyfikacji, lub tylko jego element, można zatem wykorzystać ponownie do wykonania podobnej operacji. Skrypt niezmodyfikowany umożliwia prześledzenie i sprawdzenie operacji wykonywanych na konkretnym zestawie danych oraz powinien zapewniać powtarzalność wyniku.

Pisany kod służy **do przetwarzania, analizy i modelowania danych**. Uczestnicy świata społecznego DS mogą pisać kod niewykonujący operacji na danych w ramach działań wspomagających, ale nie jako działanie podstawowe. **Działanie podstawowe jest nakierowane wyłącznie na dane**, zawsze są one w formie cyfrowej i zawsze są traktowane ilościowo. To nakierowanie wyłącznie na dane odzwierciedla sama nazwa DS. W Polsce mało znane i marginalnie używane są nazwy *business science* czy *decision science*, choć DS może wspomagać podejmowanie decyzji w organizacjach czy szeroko pojęty biznes. Zaznaczam, że **moje ujęcie działania podstawowego DS obejmuje zarówno działalność komercyjną, jak i akademicką oraz hobbystyczną**, co wyjaśnię dalej. Przyjrzyjmy się, jak jeden z rozmówców opisał, co robią ludzie zajmujący się DS:

> R: Gdybym miał podać swoją definicję, to powiedziałbym, że ludzie zajmujący się data science zdobywają dane, poznają dane, przetwarzają dane, wydobywają z danych użyteczne wzorce, wreszcie komunikują dane.
> W ramach powyższych zadań wydzieliłbym:
> – zdobywanie danych:
> • parsowanie publicznych danych,
> • łączenie danych z różnych repozytoriów,
> • czyszczenie danych,
> – poznawanie danych:
> • ekstrakcja cech,
> • wejście w domenę,
> • eksploracja analityczna danych,
> – przetwarzanie danych:
> • inżynieria cech,
> • czyszczenie i transformacja,
> • redukcja wymiarowości,
> – wydobywanie wzorców:
> • klasyfikacja,
> • regresja,
> • klastrowanie,
> – komunikowanie danych:
> • wizualizacja,
> • business intelligence,
> • inteligentne raportowanie. {konsultacja e-mailowa z R w sprawie działania podstawowego – wywiad 23}

Rozmówca posłużył się terminami przetwarzania, analizy i modelowania danych nieco inaczej, niż proponuję, ale te terminy ani w języku polskim, ani w angielskim nie są dookreślone i uczestnicy badanego świata doprecyzowują je przez podawanie przykładów. Przybliżę, co oznaczają dla mnie operacje przetwarzania,

analizy i modelowania danych. Wzoruję się na ujęciu Hadleya Wickhama i Garretta Grolemunda (2017), adaptując zaproponowany przez nich schemat ujęcia tych operacji jako procesu.

Przetwarzanie danych (*data wrangling*) jest pierwszym etapem całego procesu. Składa się na nie kolejno zaimportowanie danych, ich porządkowanie i wykonanie transformacji danych. Dane mogą zostać zaimportowane z różnych źródeł, ale najbardziej typowym scenariuszem w zastosowaniach komercyjnych jest import z bazy danych. Takie dane mają już zaprojektowany sposób dostępu i pewną strukturę[26], jest to zatem scenariusz uznawany za prostszy. Może też istnieć potrzeba pozyskania danych, np. ze źródeł publicznie dostępnych (dane rządowe, naukowe, do celów szkoleniowych czy demonstracyjnych, np. z pakietów R/Python, dane z konkursów, np. Kaggle), ze źródeł wewnętrznych (np. firmowa poczta e-mail czy repozytorium plików w formatach biurowych) czy zewnętrznych (np. media społecznościowe, strony WWW, co składa się na tzw. *web scraping*, ewentualnie zakup danych, ale to nie jest element działania podstawowego), co uznawane jest za scenariusz trudniejszy. Te czynności w powyższym cytacie mój rozmówca nazwał parsowaniem danych[27]. Dodać należy, że w zastosowaniach naukowych czy szkoleniowych dane mogą być fikcyjne, jak w pokazanym wcześniej prostym skrypcie. Podstawową czynnością jest **import** danych do środowiska programowania, czyli – jak mawiają uczestnicy badanego świata: pobranie, wciągnięcie, wrzucenie, załadowanie danych do wykonywania na nich dalszych czynności. Pobranie z bazy danych czy za pomocą metod web scrapingu przeważnie odbywa się też w danym środowisku programowania. Dalej wykonywane jest **porządkowanie** (*tidy*) danych, na które, ogólnie rzecz ujmując, może składać się np. łączenie danych z różnych źródeł i ujednolicanie sposobu ich zapisu, definiowanie sposobu zapisu danych[28], definiowanie, co jest jednostką obserwacji, a co zmienną. Mówię dla uproszczenia o danych w formie tabelarycznej, ale trzeba przypomnieć, że opisywany proces co do zasady odnosi się też do innych typów danych, np. fotografii, dźwięku czy tekstów. W Polsce używane jest określenie **czyszczenia danych**. Czyszczenie i w ogóle czynności przetwarzania danych są uznawane za najbardziej czasochłonne, a także żmudne i nudne części procesu DS. Chodzi o doprowadzenie danych do stanu, który będzie umożliwiał przejście

26 Szczególnie w przypadku danych w bazach relacyjnych z dostępem przez SQL. Istnieją także bazy nierelacyjne, tzw. *data lake* (jezioro danych), gdzie dane są słabiej ustrukturyzowane.

27 To słowo pochodzi z informatyki (*computer science*), jest używane w różnych kontekstach w znaczeniu przeanalizowania, zrozumienia treści i wykonania czynności. Mówi się na przykład, że w Pythonie czy R instrukcje po wpisaniu w konsolę są parsowane i od razu widoczny jest wynik działania instrukcji (języki te działają interaktywnie, nie wymagają kompilowania jak np. język Java). Mówi się także o parsowaniu w sensie zaimportowania ciągów znakowych typu „20190605" jako daty 5 czerwca 2019 r. Spotkałem się także z określeniami typu „nie sparsowałem tego, co powiedziałeś", co oznacza niezrozumienie słów drugiej osoby.

28 Jak w przypadku tzw. parsowania ciągu znaków na datę, może być to parsowanie np. na zapis czasu, na cechę ilościową czy kategorialną.

do iteracyjnego procesu transformowania, wizualizacji i modelowania danych[29]. **Transformacje danych** to np. tworzenie podzbiorów danych za pomocą filtrowania, wybierania zmiennych czy próbkowania. Może być wykonywane grupowanie i podsumowanie danych, np. z poziomu jednej obserwacji na sekundę do średniej tych obserwacji na minutę. Transformacje danych to także tworzenie nowych zmiennych. Na przykładzie szkoleniowym (Wickham, Grolemund, 2017) dla danych o lotach samolotów pasażerskich może być to utworzenie nowej zmiennej *prędkość lotu* w wyniku podzielenia zmiennej *dystans* przez *czas w powietrzu*. Kolejna nowa zmienna może wyrażać tę prędkość w m/s zamiast km/godz., inna może być logarytmem tej prędkości. Tworzenie nowych zmiennych, selekcja zmiennych czy bardziej zaawansowane matematycznie metody, np. analizy składowych głównych (*principal component analysis* – PCA), są ogólnie metodami na tzw. redukcję wymiaru danych. Chodzi o zmniejszenie liczby zmiennych na rzecz ich jakości[30], przed modelowaniem. Za transformacje lub czyszczenie uznaje się także znane naukowcom zadania identyfikacji i zaadresowania braków danych czy wartości błędnych, nietypowych i odstających (*outliers*). Mogą być także stosowane podsumowania danych w formie np. statystyk opisowych. Opisywane czynności są częścią zarówno etapu przetwarzania, jak i analizy danych, na którą składają się transformacje i wizualizacja danych.

Polskie słowo „przetwarzanie" nie oddaje emocjonalnego nacechowania określenia *data wrangling* u Wickhama i Grolemunda – *wrangle* dosłownie oznacza kłótnię, spór. Doprowadzenie danych do formy nadającej się do pracy często sprawia u osoby piszącej kod wrażenie walki (Wickham, Grolemund, 2017). „Czyszczenie" akcentuje bardziej żmudny i nudny charakter tych czynności. Mówi się także o preprocessingu danych (*data preprocessing*). Niektóre operacje w fazie przetwarzania danych mogą być wykonywane przez osoby, które nie są uznawane za zajmujące się DS. Chodzi o zadania ręcznego etykietowania danych, np. fotografii, na potrzeby późniejszego użycia metod nadzorowanego ML. Thomas, Nafus i Sherman (2018: 5–6) twierdzą, że choć osoby uważające się za rozwijające algorytmy lub oprogramowanie (nie używają określenia *data scientist*) czasami wykonują czynność przetwarzania danych, to kojarzy się im ona z praktykantami i osobami zatrudnionymi zewnętrznie do powtarzalnych prac, tzw. klikaczami (*clickworkers*) (Crawford i in., 2018: 33–34). W świecie DS ma to częściowo zastosowanie, jeżeli chodzi o etykietowanie danych. Nie mogę zgodzić się z tą tezą, jeżeli mowa o wielu innych czynnościach, które są częścią przetwarzania danych, a zapisuje się je

29 W konkursach (np. Kaggle) czy w nauce na kursach lub studiach uczestnikom udostępniane są dane już uporządkowane/wyczyszczone. Choć trzeba je zaimportować do środowiska programistycznego, to do rozpoczęcia transformacji/wizualizacji/modelowania potrzeba niewiele pracy uznawanej za czasochłonną, żmudną i nudną.

30 Jest to np. usunięcie zmiennych silnie ze sobą skorelowanych. W naukach społecznych stosowane jest np. wyliczenie indeksu przez sumowanie kilku zmiennych, których wartości reprezentują odpowiedzi na pytanie oparte na skali Likerta – to także redukcja wymiaru danych.

w kodzie. Także etykietowanie danych może być uznawane za wartościową część pracy data scientisty, pozwalającą lepiej poznać zbiór danych i uzyskać lepszy model {obserwacje 40 i 46} – powstają narzędzia mające wspomagać data scientistów w etykietowaniu (Ratner i in., 2017). Niemniej cały etap przygotowania danych jest raczej postrzegany jako nudny i odtwórczy, choć niezbędny, modelowanie zaś jako etap najciekawszy, najbardziej satysfakcjonujący – „to, po co tam jesteś [jako data scientista – przyp. R.Ż.]":

> *B:* […] *w każdym projekcie bardzo ważne jest przygotowanie danych, a modelowanie jest później, iii ludzie mówią mi różnie, że, no, niektórzy się zajmują obiema rzeczami, niektórzy nie. Jak to jest w twoim przypadku?*
> R: Ja się zajmuje obiema rzeczami. I właściwie nie kojarzę, żeby ktoś się zajmował tylko jedną z tych rzeczy, a drugą nie, dlatego że to jest trochę **brudna robota** z tym przygotowywaniem danych i nie wyobrażam sobie, żeby ktoś miał taką rolę, że robi tylko brudną robotę i nie ma na koniec tego fajnego. No i w drugą stronę tak samo. […]
> *B: Ale dlaczego nazywasz to brudną robotą?*
> R: No bo to jest coś powtarzalnego maksymalnie, tu nie ma tej sztuki, nie ma tej nauki, nie ma tego rozkminiania, tylko bardziej coś w stylu powtarzalnych czynności, które doprowadzają do tego, żeby dane były czyste. Czasem to, czasem w tym jest sztuka, ale zasadniczo nie ma tak do końca, zazwyczaj to jest jednak przygotowanie, żeby to się poprawnie ładowało, żeby się dobrze zapisywało do plików, czyli to już jest taka deweloperska robota, taka, co właśnie od tego chcieliśmy odejść jako data scientiści, mam wrażenie. Od tego, co ciągle trzeba robić, podobne sekwencje, czynności, nie? Chociaż to też nie jest tak na 100%, no, to też jest w miarę ciekawe, to przygotowanie tam [niezrozumiałe] Po prostu w pewnym momencie to się kończy i wtedy sobie dopiero przypominasz, po co tam jesteś [z uśmiechem]. {wywiad 10}

Czynności przetwarzania danych za pomocą kodu, także tzw. czyszczenie, są w badanym świecie uważane za „brudną robotę" w podwójnym sensie. Po pierwsze – jest to praca żmudna, nieciekawa, nieprzyjemna, łatwa i rutynowa, a po drugie – prawdziwa, ciężka, trudna i ważna dla całego projektu. Pojawia się określenie *get your hands dirty* (dosł. 'ubrudź sobie ręce'), np. na warsztatach czy kursach, kiedy prowadzący chcą podkreślić, że uczestnicy zrobią coś naprawdę, nie na szkolnym, przygotowanym specjalnie na te potrzeby zbiorze danych.

Drugim etapem procesu jest analiza i modelowanie danych. Składa się na nie **iteracyjny proces transformowania, wizualizacji (te dwie części nazywam analizą) i modelowania danych**. O ile transformacje mogą inicjować proces, a modelowanie kończyć go, to **te czynności nie mają jednoznacznie określonej kolejności**. Najbardziej typowy będzie proces od transformacji, przez wizualizację, gdzie wzrokowo ocenia się wyniki transformacji w odniesieniu do relacji między zmiennymi, do modelowania, kiedy następuje próba ilościowego podsumowania tejże relacji. Wyniki miar modelu mogą prowadzić do pomysłów na kolejne transformacje w celu poprawy wybranych miar i tak dalej, ale należy powtórzyć, że nie ma jednoznacznej kolejności tych czynności. Iteracje pomiędzy transformacjami a wizualizacją danych nazywam analizą, Wickham i Grolemund (2017)

używają zaś określenia „eksploracja danych". Stosują oni to określenie, ponieważ ich książka dotyczy właśnie tego rodzaju analiz, ale wskazują, że istnieją także analizy konfirmacyjne i wnioskowanie formalne (np. statystyczne metody testowania hipotez). Podejście eksploracyjne jest wykorzystywane w DS częściej. Dla danych tabelarycznych może to być podsumowywanie danych w różnych przekrojach z użyciem wykresów lub tabel, za pomocą statystyk.

Mówię o danych w formie tabelarycznej, pokażę więc przykład analizy dla innego rodzaju danych. W przypadku obrazów modelowanych za pomocą konwolucyjnej sieci neuronowej (*convolutional neural network* – CNNs) stosuje się metody pomnażania danych (*data augmentation*). Wynikiem pomnażania danych obrazowych jest większa liczba bardziej zróżnicowanych obrazów, co ma zapobiegać zjawisku nadmiernego dopasowania, tzw. przeuczenia[31] (*overfitting*) modelu. Pomnażanie polega na przekształcaniu obrazu oryginalnego na różne sposoby, np. przycinaniu, obracaniu, odbijaniu lustrzanym, przesuwaniu, dodaniu szumu, zamianie kolorów, dodaniu zakłóceń (mających symulować np. opady śniegu), zmianie perspektywy. Możliwe są kombinacje tych zmian. Obrazy zmienione dołączane są do oryginalnego zbioru danych, zbiór wynikowy jest zatem większy i bardziej różnorodny niż zbiór oryginalny (por. Jung, 2019). Pomnażanie jest rodzajem transformacji. Wizualizacja polega na wyświetlaniu obrazu oryginalnego w porównaniu do tych przekształconych (Jung, 2019). Wizualizacje na etapie analizy są tworzone na potrzeby osób wykonujących tę analizę lub ich najbliższych współpracowników i przełożonych. Wizualizacja końcowa, służąca komunikowaniu wyników całego procesu, jest przygotowywana z myślą o innym odbiorcy – docelowym adresacie.

Czym jest **modelowanie**, w sensie modeli ML, wspominałem w tej książce wielokrotnie. „Modele są narzędziem do wyodrębniania wzorów z danych" (Wickham, Grolemund, 2017). W języku nadzorowanego ML, tj. takiego, w którym znane są wartości zmiennej zależnej, można określić modelowanie jako znajdowanie funkcji łączących zmienne niezależne z zależną. Metodą tego rodzaju jest regresja – liniowa czy logistyczna. „Regresja jest babcią modeli nadzorowanego ML – naukowcy pracowali nad nią już na początku XIX w." (Foreman, 2017: 229). W regresji liniowej człowiek zakłada występowanie liniowej zależności między zmiennymi, nie zakłada z góry wartości współczynników kierunkowych ani wyrazu wolnego – są one ustalane na podstawie danych za pomocą metody najmniejszych kwadratów. Inne modele ML – czy ich podgrupa bazująca na sieciach neuronowych, nazywana DL – nie wymagają założenia o charakterze zależności. Model zatem „uczy się przez doświadczenie", dzięki czemu może wykonywać klasyfikację czy predykcję nie na podstawie zadanych z góry reguł, a na podstawie wzorów wyodrębnionych ze zbioru danych.

31 Przeuczenie oznacza sytuację, w której model „nauczył się" nie tylko sygnału, ale i szumu z danych treningowych, w związku z czym nie jest zdolny do poprawnej generalizacji zależności między zmiennymi (Larose, 2005: 91–93). O przeuczeniu mówi się, że „model nauczył się danych na pamięć".

Mówi się o podziale zbioru danych na treningowy (*training set*), testowy (*test*) i kontrolny (*validation*) (Szeliga, 2017: 111–12). Pierwszy zbiór służy do iteracyjnego przygotowania, zwanego „uczeniem" lub „trenowaniem" modelu, drugi do dostrojenia parametrów, a za pomocą trzeciego, wcześniej nieużywanego zbioru danych sprawdzana jest skuteczność działania modelu:

> *B:* […] *co to w ogóle jest modelowanie?*
> R: Takie lepienie z plasteliny, układanek [śmiech]
> *B: OK* [z uśmiechem]. *Brzmi ciekawie.*
> R: No, weźmy sobie na przykład sieć neuronową.
> *B: No.*
> R: Sieć może mieć różną budowę. Może mieć jeden neuron, może mieć dwa neurony, może mieć dwieście neuronów. Może mieć jedną warstwę, może mieć pięć warstw, może mieć dwieście warstw różnych. Do tego jeszcze dochodzą inne parametry między, w międzyczasie. Więc modelowanie polega na tym, żeby ten model był optymalny, i to się robi naaa podstawie, ym, nauczenia modelu, bo uczysz model na jednym zbiorze danych i on tam sobie wykminia swoje reguły, jakieś tam wzory matematyczne. I na przykład może mieć precyzję na tą 99%, tak, ale musisz go później przetestować na zbiorze walidacyjnym, czyli na takim, którego w ogóle nie widział. I dopiero wtedy patrzysz, jaka tak naprawdę jest ta precyzja, jak dobrze się nauczył. Bo może tak być, że on się po prostu wykuł na pamięć. I jak zobaczy nowe jakieś dane, to na koniec sobie nie potrafi z nimi poradzić w ogóle. Więc modelowanie polega na tym, żeby znaleźć ten balans, żeby błąd na jednym zbiorze i na drugim zbiorze był jak najbardziej zbliżony do siebie i jak najmniejszy oczywiście.
> *B: Mhm, no dobra, a czy modelowanie to jest najważniejsza część w data science?*
> R: Najważniejsze to jest przygotowanie danych, bo jak masz dobry model, ale masz chujowe dane, to ci nic nie wyjdzie tak naprawdę. Także wyczyszczenie danych i dobre przygotowanie danych to jest kluczowe. {wywiad 8}

Rozmówczyni podkreśliła, że najważniejszą częścią projektu DS jest faza pierwsza, tzn. przygotowanie danych. W myśl powiedzenia *garbage in – garbage out* (śmieci na wejściu to śmieci na wyjściu) żaden model nie zrekompensuje braku przygotowania danych. To samo mówi Dominik Batorski:

> […] większość kursów kładzie nacisk na przegląd modeli i metod machine learningowych. Tymczasem w praktyce dużo większe znaczenie ma tzw. feature engineering, przygotowanie danych, preprocesing, zapewnianie jakości danych. To, czy zastosuje się takie lub inne metody – lasy, regresje czy sieci – ma o wiele mniejsze znaczenie. A feature engineeringu nie można się nauczyć na kursach. Dlatego z punktu widzenia zatrudniających kluczowe znaczenie ma rozróżnienie na ludzi z wiedzą teoretyczną i praktyków (Gontarz, 2019: 19).

W powyższym fragmencie Batorski twierdzi, że na modelowanie kładzie się nacisk, ucząc zagadnień związanych z DS, w praktyce komercyjnej ważniejsze są zaś przygotowanie oraz analiza danych. Szczególnie analiza danych, z powodu swojej czasochłonności, uznawana jest za wąskie gardło (*bottleneck*) w procesie DS (Staniak, Biecek, 2019). Powstaje wiele pakietów, które mają przyspieszać realizację

analizy danych. Od roku 2018 cieszą się one coraz większą popularnością (Staniak, Biecek, 2019).

Modelowanie jest uznawane za najprzyjemniejszy, najciekawszy, najbardziej emocjonujący i najtrudniejszy element projektów DS. To model jest jakoby „inteligentny", wokół możliwości modeli narosła zatem popkulturowa, eksploatowana przez marketing otoczka tzw. AI. To modele wdrożone do większych systemów informatycznych czy jakichkolwiek urządzeń automatyzują zadania w czasie rzeczywistym. To dzięki modelom roboty mogą zastąpić ludzi i zabrać im pracę. Takie postrzeganie modelowania jest charakterystyczne dla laików i początkujących uczestników badanego świata. Jest ono także charakterystyczne dla komunikatów zaadresowanych do laików (np. marketingu usług/produktów zawierających tzw. AI) i do początkujących (np. marketingu studiów/kursów DS). Jednak **do kanonu wiedzy uczestników z doświadczeniem należy to, że modelowanie zajmuje od 10% do 40% czasu**[32] **w projekcie DS**, przygotowanie i analiza danych są bowiem najbardziej czasochłonne, a zarazem niezbędne dla sensowności modelowania:

> R: […] to wcale nie jest tak, że data scientist to spija śmietankę i cały czas sobie buduje te modele i w ogóle to ma tak świetnie. […] tak jak w tym projekcie związanym z wykrywaniem wyłudzeń, po prostu [pauza] **musieliśmy siedzieć** i **sprawdzać cecha po cesze**. […]. To jest żadna przyjemność i spora część projektów taka jest. Jak projekt data science jest źle poprowadzony, to wtedy te rzeczy **przygotowawcze** zżerają prawie cały czas projektu i to, co jest najfajniejsze, o czym się mówi [pauza] **modelowanie**, właśnie ta **zabawa**, z dużymi komputerami, zabawa z GPU, wiele procesorów, to na to zostaje 10% czasu. […] ludzie od wielu, wielu lat myślą, że machine learning i sieci neuronowe to są takie magiczne skrzynki. Że niektórzy umieją te skrzynki obsługiwać, że do tych skrzynek, mmm, **wrzuca** się dane [pauza] i z tych skrzynek wychodzi usprawnienie biznesu. [pauza] A to jest **kompletna bzdura**, ludzie myślą, że to się dzieje **samo**. Że żeby być data scientistą, wystarczy umieć kręcić śrubkami w tych magicznych skrzyneczkach i to załatwia całość. I dlatego się wydaje, że to jest takie super, świetne i że te modele będzie się budować cały czas. To jest właściwie największy **mit** i dlatego ludzie chcą sztuczną inteligencję, dlatego ludzie uważają, że w projektach data science tak cały czas robi się takie super ekstra rzeczy. {wywiad 5}

To, że doświadczeni uczestnicy wiedzą, iż modelowanie to niewielka część projektów i że uznają etap przygotowania danych za kluczowy, nie oznacza, że modelowanie nie jest jednym z najbardziej lubianych elementów pracy – także dla osób z dużym doświadczeniem. Jeden z rozmówców najpierw żartował, a następnie mówił o swojej radości z działania modelu:

---

32 Popularne jest przekonanie o podziale czasu 80/20 (przygotowanie danych/modelowanie), nawiązującym do tzw. zasady Pareto (por. Gulipalli, 2019). Badania ankietowe data scientistów w USA pokazują, że 53% czasu zajmuje respondentom „zbieranie, etykietowanie, czyszczenie i organizowanie danych", a „modelowanie danych" 19% (CrowdFlower, 2017: 5). Przy tym 60% respondentów wskazało, że najmniej lubi zajmować się „czyszczeniem i organizowaniem danych", a 78% najbardziej lubi zajmować się „modelowaniem danych" (CrowdFlower, 2017: 5).

B: OK. A co jest tak w ogóle w pana pracy najfajniejsze?
R: Obiad.
B i R: [śmiech]
R: Nie no serio [ze śmiechem]
B: To tak jak u mnie [ze śmiechem]
R: Najpierw jest ta walka z głodem do obiadu, a potem walka z sennością po obiedzie [z uśmiechem]
B: Ehe [ze śmiechem] [pauza]
R: Nie wiem. Każdy ma pewnie, eee, swoje, co mu się najbardziej podoba [pauza]
B: Mhm?
R: Ja odczuwam niesamowitą radość, kiedy model jest w stanie robić coś rozsądnego. Tak, czyli na przykład bierzemy sobie właśnie jakieś aaa, jakieś zagadnienie rozpoznawania czegoś na obrazkach, zaczynamy uczyć ten model i on zaczyna **rzeczywiście** wykazywać takie, po wynikach widzimy, że on coś rozpoznaje, potem jeszcze robimy parę różnych takich tricków, żeby zobaczyć, co powoduje, że on rozpoznaje dany obiekt i patrzymy, i mówimy: no kurczę, to ja bym zrobił tak samo. Nagle się człowiek orientuje, że yyy, że to **coś** to już jeeest takie coś bardzo **mądrego** [powoli], co udało mu się stworzyć. I ja jeszcze mam ten background, eee, informatyczny i ja wiem na przykład, że narzędziami informatycznymi bym tego nie zrobił. {wywiad 2}

Wickham i Grolemund (2017) całość etapu drugiego, na który składają się analiza i modelowanie danych, nazywają rozumieniem danych. Rozumienie danych nie oznacza jednak znajomości domeny, z której dane pochodzą. Przy tym **nie znajduję w badanym świecie tak ostro krytykowanych poglądów w rodzaju „dane mówią same za siebie"**, w projektach komercyjnych **odnajduję za to przekonanie o konieczności współpracy data scientisty z ekspertem domenowym** (Alekseichenko, 2019a). Jednocześnie – pozostając przy projektach komercyjnych – data scientista nie wie, jak rozwiązać problem, z którym przychodzi do niego klient. Nie jest znawcą domeny (choć znajomość domeny jest uważana za zaletę), jest umiejącym programować ekspertem od danych. Problem jest rozwiązywany (a przynajmniej dokonywana jest próba rozwiązania) z użyciem modelu, który wyodrębnia wzorce z danych. Rolą data scientistów jest przygotowanie środowiska do modelowania:

R: Nasza robota polega na tym, że tworzymy takie [pauza] **pole**, taką przestrzeń, w której **modelujemy** różne zjawiska i to modelowanie to jest tak naprawdę wyciąganie wniosków i my nie robimy tego wyciągania wniosków, one się **same** wyciągają, my tylko jesteśmy w stanie zbadać, czy one są poprawne, czy są logiczne, czy **czasami** mamy takie, taki sposób zaglądania do modelu, gdzie stwierdzamy, że OK, robi coś rozsądnego, a czasami nawet tego nie mamy i tylko po efektach działania modelu stwierdzamy, czy jest dobrze, czy jest źle. Nasza praca polega na [pauza] **pomaganiu** modelowi, żeby **dobrze** działał. Aczkolwiek ta różnica między nami a normalnymi programistami czy w ogóle normalnymi inżynierami jest taka, że my w zasadzie nie wiemy, jak rozwiązać problem. My tylko tworzymy takie środowisko, w którym problem, powiem, **sam** się rozwiązuje. {wywiad 2}

Wickham i Grolemund (2017) za **ostatni element procesu DS uważają komunikowanie** uzyskanych wyników. **Te elementy etapu komunikowania, które są realizowane w postaci kodu, są traktowane jako część działania podstawowego badanego świata**, inne zaś jako działania wspomagające. W kodzie realizowane są

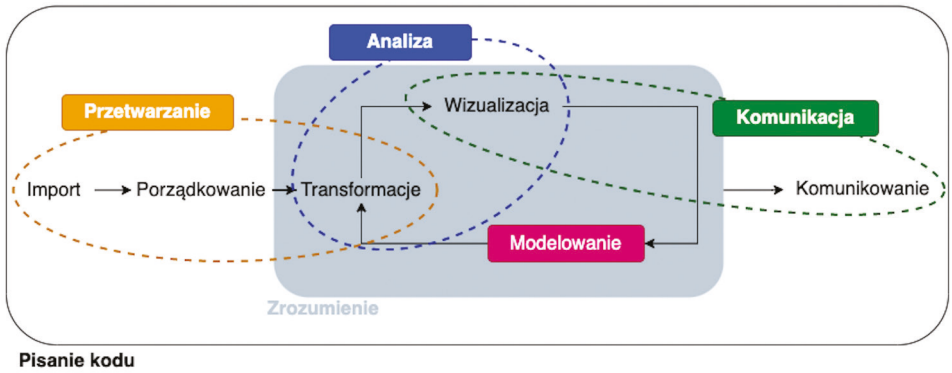
wizualizacje danych – od wykresów na potrzeby osób analizujących dane do wizualizacji wyników także dla osób spoza badanego świata. Są to zarówno wykresy obrazujące zależności między zmiennymi, jak i wizualizacje objaśniające działanie modeli ML (Biecek, 2018a; 2018c). Świat DS uważa za wartościowe tworzenie wizualizacji danych w sposób dostarczający wielowymiarowej, złożonej, jak najpełniejszej informacji. To wszystko w sposób czytelny, rzetelny i atrakcyjny wizualnie, z uwzględnianiem tego, jak ludzkie oko i mózg przetwarzają obraz (por. Biecek, 2016). Za działanie wspomagające uznawane będzie komunikowanie wyników w formie np. spotkania z klientem (dla projektów komercyjnych), opublikowania artykułu (dla projektów naukowych), zamieszczenia wpisu na blogu (dla realizacji hobbystycznych). Granica jest jednak płynna i rozmyta, w Pythonie i R istnieją bowiem narzędzia do publikowania wyników w postaci np. slajdów, notatników, plików .pdf czy .html, publikacji w serwisie GitHub (Pérez, Granger, 2007; 2015; Xie, Allaire, Grolemund, 2018; Allaire i in., 2019), a nawet prowadzenia strony internetowej, bloga i pisania dokumentacji lub książek (Xie, 2016; Xie, Thomas, Hill, 2022). Komunikacja między uczestnikami badanego świata odbywa się także poprzez sam kod. Zarówno Python, jak i R bywają określane, niekoniecznie przez uczestników badanego świata, jako *lingua franca* DS (Barry, 2016; Nunns, 2017; ActiveState, 2018; Thieme, 2018).

Istnieje kilka innych niż ten autorstwa Wickhama i Grolemunda (2017) sposobów określenia procesu, który realizuje projekt DS. Właściwie nazwa „data science" nie jest tutaj zasadna, chodzi jednak bez wątpienia o projekty bazujące na analizie i modelowaniu danych cyfrowych. Wciąż popularny jest powstały pod koniec lat dziewięćdziesiątych CRISP-DM (*cross-industry standard process for data mining* – międzysektorowy standard procesu data mining) (Azevedo, Santos, 2008; Biecek, 2019a). Poza omówionymi (przygotowaniem, analizą, modelowaniem danych i komunikacją wyników) wymieniane są: zrozumienie lub dookreślenie problemu (co jest równoznaczne ze zdefiniowaniem celu projektu), testowanie lub ewaluacja uzyskanych wyników, wdrożenie wyników (Azevedo, Santos, 2008; Biecek, 2019a). Nie ująłem testowania czy ewaluacji wyników, ponieważ w moim rozumieniu jest to integralną częścią każdego z omawianych elementów procesu. Na podstawie takich ocen podejmowane są decyzje o kierunku iteracji albo o przejściu do kolejnego etapu. Nie uwzględniłem określania celu projektu ani wdrożenia wyników, ponieważ niekoniecznie są one uznawane za to, czym zajmuje się świat DS. Ani definiowanie celów, ani wdrożenie wyników nie są składowymi działania podstawowego.

Podsumowując to ujęcie, **w ramach działania podstawowego świata społecznego DS w Polsce realizowany jest proces, na który składają się**:

1) w fazie pierwszej kolejno operacje importu, porządkowania i transformowania danych, co określam jako **przetwarzanie danych**;
2) w fazie drugiej iteracyjnie i niekoniecznie następujące po kolei operacje transformowania i wizualizacji danych na potrzeby analityczne, co określam jako **analizę danych oraz modelowanie danych**;

3) w fazie trzeciej **komunikowanie wyników** uzyskanych w fazach pierwszej i drugiej, **może ono być częściowo uznawane za działanie podstawowe, częściowo za działania wspomagające**.



Rysunek 4.7. Operacje wykonywane na danych z użyciem programowania jako proces

Źródło: opracowanie własne na podstawie badań etnograficznych oraz tzw. R4DS (Wickham, Grolemund, 2017) za pomocą draw.io.

Całość procesu jest zapisywana w kodzie – ramka „pisanie kodu"[33] otacza wszystkie elementy procesu. Stąd moje ujęcie działania podstawowego badanego świata jako **pisania kodu do przetwarzania, analizy i modelowania danych**.

4.2.2. Data science – komercyjne, akademickie, hobbystyczne – zapisany w kodzie proces operacji na danych

W polskim świecie społecznym DS osoba może zostać uznana za uczestnika tego świata przez innych uczestników tylko, jeżeli wykaże się doświadczeniem w pisaniu kodu do przetwarzania, analizy i modelowania danych, bez względu na to, czy jest to doświadczenie komercyjne, akademickie czy hobbystyczne. Chodzi o pisanie kodu w ramach wszystkich trzech wyróżnionych obszarów – przetwarzania, analizy i modelowania – w jednym procesie, by osiągnąć postawiony cel. Jest to określane jako przeprowadzenie projektu od początku do końca.

Jeżeli osoba nie przeprowadziła przynajmniej jednego projektu w całości, zgodnie z wyżej omówionym sensem, ale wykonywała części działania podstawowego, może być uznana za uczestnika jednego ze światów przecinających się z DS. Przykładowo: dla osób koncentrujących się na przetwarzaniu i przechowywaniu danych będzie to świat inżynierii danych (*data engineering*), dla koncentrujących się na

33 Oryginalnie użyto słowa „programowanie" (*programming*) (Wickham, Grolemund, 2017), ale pozostaję przy określeniu „pisanie kodu", by zaznaczyć odrębność od programowania w sensie budowy oprogramowania (*software development*).

analizie – świat analityki danych, dla koncentrujących się na modelowaniu – świat nauki akademickiej (chodzi przede wszystkim o rozwijanie metod ML). Istnieje także grupa tzw. wannabesów, skoncentrowanych na uczeniu się DS, a szczególnie modelowania – to o nich mówiła rozmówczyni: „bardzo często hobbystyczne zajmowanie się tym, eee, wiąże się po prostu z, eee, takim etapem bardzo wstępnym, nie?" {wywiad 21}. W grupie wannabesów można wyróżnić nierozłączne grupy studentów, tzw. graczy w Kaggle, lub kursantów MOOC, których ogólnie traktuje się jako aspirujących do uczestnictwa w badanym świecie.

Nie każdy uczestnik świata społecznego DS uznaje się sam lub jest uznawany przez innych uczestników świata za data scientistę. Dotychczas używałem tych określeń zamiennie, należy jednak to doprecyzować. Z moich badań wynika, że **określeniem „data scientist" są bez wątpliwości nazywane osoby realizujące komercyjnie opisane wyżej działanie podstawowe**. „Data scientist" to w Polsce raczej nazwa stanowiska pracy lub zakresu obowiązków zawodowych niż nazwa określająca każdego uczestnika świata społecznego DS. Stwierdzenie „zajmuję się data science, ale nie jestem data scientistką/scientistą" jest dopuszczalne. Dzieje się tak również w przypadku osób, które realizują działanie podstawowe komercyjnie. Nazwy stanowisk pracy takich ludzi mogą być różne od „data scientist", osoby te mogą także z różnych powodów unikać określania siebie mianem data scientisty. Określenie to może być postrzegane jako tytuł nic nieznaczący, przereklamowany lub zdezawuowany.

Działanie podstawowe może być realizowane przez tę samą osobę w ramach działalności komercyjnej, akademickiej, hobbystycznej równolegle lub jednocześnie, możliwe są także wszystkie inne połączenia tych sfer działalności w ramach działania jednej osoby. Koncentracja działania osoby na jednym z elementów działania podstawowego – przetwarzaniu, analizie lub modelowaniu danych – może wskazywać na jej przynależność do świata/subświata przecinającego się ze światem DS (np. inżynieria danych, analityka danych, nauka akademicka, wannabesi).

Mówię także o DS hobbystycznym, jednak w toku badań **nie spotkałem się z osobą zajmującą się DS wyłącznie hobbystycznie**. W międzynarodowym środowisku DS istnieje nietłumaczone pojęcie *citizen data scientist*, które oznacza jednocześnie hobbystę i podejmowanie działań w ważnej sprawie obywatelskiej, niemniej w Polsce nie spotkałem osoby definiującej się w ten sposób. Brak dowodu nie jest dowodem braku, mogłem po prostu nie dotrzeć do takich ludzi. Działanie podstawowe świata DS jako hobby podejmowane jest raczej równolegle bądź zamiennie z tym samym działaniem jako pracą akademicką lub komercyjną. Osoby, które w czasie wolnym próbują swoich sił w różnych elementach działania podstawowego świata DS, są w Polsce raczej określane mianem *wannabe* czy uważane za w ogóle niemające wiele wspólnego z badanym światem – szczególnie kiedy nie biorą udziału w dyskursie tego świata.

Moje ujęcie działania podstawowego wzorowane jest na definicji z pierwszego sondażu Kaggle (2017b): „data scientist to ktoś, kto używa kodu do analizy danych

(*we define 'data scientist' as someone who uses code to analyze data*)". Podstawą tego ujęcia jest całość badań własnych, od końca 2016 do połowy 2019 roku (por. aneks), przy znaczącej roli konsultacji z rozmówczyniami/rozmówcami.

4.2.3. Przykłady zastosowania data science

Teoretyczne ujęcie działania podstawowego ilustruję przykładami zastosowania DS w różnych sferach życia. Pomoże to **zrozumieć, co robią (a czego nie) osoby zajmujące się DS**. Prezentuję przykłady ikoniczne, wyraziste, najbardziej znane nie tylko w badanym świecie, ale także poza nim. Prześledzenie ich pokazuje nie tylko, co robią (a czego nie) uczestnicy badanego świata, ale i **jak to robią** – jakie są kryteria wartościowania określonych działań i powiązanych z tymi działaniami tożsamości. Takie paralegendarne historie mają znaczenie dla „konstruowania wzorcowych tożsamości, bohaterów społecznego świata lub areny" (Kacperczyk, 2016: 46).

Za każdym razem **prezentuję rozwiązanie budowane na bazie danych cyfrowych i operacji ich przetwarzania, analizy i modelowania**. **Ta część bazowa jest obudowywana zestawem innych technologii** (np. gdy gotowy model służy do poprawy obrazu z aparatu fotograficznego zainstalowanego w smartfonie).

> Ds team [tzn. zespół data science – przyp. R.Ż.] robi tylko małą część systemu, który dostaje klient. {notatka terenowa, obserwacja 36}

Tą **obudową badany świat nie zajmuje się w ramach działania podstawowego lub nie zajmuje się nią wcale**. Przykłady dzielę na trzy kategorie: wgląd (*insight*), model, produkt.

W kategorii wglądu przykłady sięgają czasów popularności określenia „data mining", kiedy to mówiono o odkrywaniu wiedzy z baz danych. To „odkrywanie" mam na myśli, mówiąc o wglądzie (*insight*) – odnalezieniu w danych informacji, która nie była wcześniej znana, a może być użyteczna. Stosowane mogą być różnego rodzaju metody analizy i modelowania, także te najprostsze. Szeroko powtarzany był przykład, w którym amerykańska sieć sklepów wielobranżowych Target identyfikowała ciężarne klientki w celu skierowania do nich oferty promocyjnej. Identyfikacji służyły takie informacje, jak kupowanie dużych ilości bezzapachowych produktów do nawilżania skóry, bezzapachowego mydła, dużych opakowań kulek bawełnianych czy suplementów cynku, magnezu. Andrew Pole, „statystyk" firmy Target, wskazywał, że można oszacować datę rozwiązania ciąży dla każdej ciężarnej klientki i kierować do niej ofertę dostosowaną do miesiąca. Media powtarzały historię, w której ojciec nastolatki interweniował w jednym ze sklepów, pokazując kierownikowi placówki papierową gazetkę z promocjami produktów dla ciężarnych, zaadresowaną do jego córki. Wściekły mężczyzna miał wykrzyczeć, że Target zachęca nastolatkę do zajścia w ciążę. Po kilku dniach telefonicznie

przepraszał on kierownika sklepu za zajście – w jego domu działy się rzeczy, o których nie wiedział. To „Target wiedział o ciąży nastolatki wcześniej niż jej ojciec" (Hill, 2012). Odniesienia do tej historii znajdują się np. w materiałach do nauki analizy i modelowania danych (Foreman, 2017: 229–312), natomiast w amerykańskim środowisku DS bywała ona przykładem działalności wątpliwej etycznie (Brooks, 2014: 139; Gutierrez, 2014: 176, 321). Wgląd jako rezultat pracy data scientistów czy statystyków to informacja, z której skorzystanie zależy od innych ról w organizacji – np. działu marketing przygotowującego ofertę, działu sprzedaży decydującego o promocyjnych cenach, działu obsługi klienta rozsyłającego listy itp. Sam wgląd bez dalszych działań nie ma wpływu na wyniki finansowe firmy.

Na odkrywaniu połączeń pomiędzy produktami bazuje opatentowany system rekomendacji firmy Amazon (Linden, Jacobi, Benson, 2001), początkowo księgarni internetowej. Historia systemu posłużyła za przykład zastosowania big data i wsparcia tezy, że dane są lepsze od ekspertów (Cukier, Mayer-Schönberger, 2014). W latach dziewięćdziesiątych polecenia książek w Amazon były recenzjami pisanymi przez zespół krytyków literackich. Zwiększały one sprzedaż konkretnych książek, dążono jednak do spersonalizowania rekomendacji i sięgnięto do danych o transakcjach online. Początkowo badano wylosowaną, reprezentatywną próbę danych w celu wyłonienia kategorii klientów i dostosowania do nich rekomendacji. Okazało się, że rekomendacje te były bardzo powierzchowne i w efekcie wizyty w sklepie internetowym przypominały klientowi „zakupy w towarzystwie wiejskiego głupka". Dokonano więc zmian. Zaczęto badać wszystkie zgromadzone dane, co zwane jest podejściem N = all, szukano zaś powiązań nie pomiędzy klientami a produktami – badano relacje ilościowe pomiędzy wszystkimi książkami, bez względu na autora czy gatunek literacki. Uzyskany model powiązań działał w czasie rzeczywistym, wyświetlając klientowi rekomendacje na stronie internetowej sklepu. Okazało się to metodą generującą sprzedaż większą niż recenzje krytyków. Kalkulacja kosztów doprowadziła do decyzji o zwolnieniu recenzentów i przejściu na rekomendacje na podstawie danych. System ten generuje zakupy stanowiące ponad 30% zysków sprzedażowych Amazona, choć informacji firma nie potwierdziła oficjalnie (Cukier, Mayer-Schönberger, 2014). Z perspektywy użytkownika:

> Dzięki mojej sumiennej, choć mimowolnej współpracy serwery Amazona znają dziś moje preferencje i zainteresowania lepiej niż ja. Nie uznaję już ich sugestii za reklamę; postrzegam je raczej jako przyjacielską pomoc, ułatwiającą wędrówkę przez dżunglę, jaką stał się rynek książki. I jestem im za to wdzięczny. A z każdym nowym zakupem uaktualniam swoje preferencje w bazach danych i niechybnie tworzę listy swoich przyszłych zakupów (Bauman, Lyon, 2013: 177–178).

To model działający „na produkcji" – klient wchodzi w interakcje z systemem informatycznym sklepu (czyli tzw. systemem produkcyjnym), rekomendacja za pomocą modelu wykonywana jest automatycznie, bez udziału pracowników. Surma zaznacza, że jest to zasadnicza jakościowa różnica w porównaniu do znanych od dziesięcioleci systemów tzw. *business intelligence* (BI). *Business intelligence* daje

wgląd w dane historyczne, jak np. zestawienia sprzedaży w podziale na kategorie produktów i regiony, co ma wspomagać proces decyzyjny ludzi. W omawianym przykładzie systemu rekomendacyjnego, a także w dalszych przykładach, widoczne jest:

> […] wyjście poza paradygmat klasycznych systemów BI w kierunku użycia tzw. złotej pętli decyzyjnej, gdzie wygenerowana propozycja nie trafia do menadżera, ale jest wykonywana automatycznie (Surma, 2017: 43).

Systemy rekomendacji znalazły zastosowanie w wielu sklepach internetowych czy platformach handlu internetowego (jak Allegro lub Ebay), a także w serwisach oferujących treści. Na modelach ML bazują rekomendacje np. należącej do Google'a platformy wideo YouTube (Davidson i in., 2010), serwisów muzycznych iTunes i Spotify (van den Oord, Dieleman, Schrauwen, 2013) czy wypożyczalni seriali i filmów Netflix (Amatriain, Basilico, 2012; Netflix, 2019).

Przypadek Netflixa jest znany w świecie społecznym DS także dlatego, że algorytmy będące bazą rekomendacji powstały w rezultacie otwartego konkursu ogłoszonego przez tę firmę w 2006 roku. Kombinacja zwycięskich modeli została (jak mawiają uczestnicy badanego świata, po polsku i po angielsku) wdrożona „na produkcję" (*put into production*, ewentualnie *deployed* lub *shipped*), czyli działała w czasie rzeczywistym w serwisie, klienci Netflixa widzieli zatem rekomendacje bazujące na działaniu tej kombinacji (Amatriain, Basilico, 2012). Około 75% nowych zamówień generowanych jest dzięki rekomendacjom (Cukier, Mayer-Schönberger, 2014).

Omawiane systemy rekomendacji traktuję jak przykład modeli „na produkcji". Wytrenowana na danych, uzyskana w kolejnych iteracjach ostateczna wersja modelu ML jest zapisywana do pliku, np. w formacie .h5, który przechowuje architekturę modelu i wytrenowane wagi. Mówiąc potocznie, taki plik sam z siebie niczego nie zrobi. Żeby za pomocą tego modelu automatycznie np. zarekomendować film klientowi logującemu się na swoje konto do serwisu internetowego, taki plik trzeba wmontować w infrastrukturę informatyczną serwisu – tak, by działał ciągle, dla każdego klienta, dla wielu klientów równocześnie, odpowiednio szybko, w akceptowalnym przedziale kosztów utrzymania go itp.

Za jedno z największych osiągnięć tzw. AI uznawane są sytuacje, w których model wytrenowany do grania w grę wygrywa z mistrzem-człowiekiem (por. Elish, Boyd, 2018: 59–63). Już w 1997 roku komputer IBM Deep Blue wygrał pojedynek szachowy z Garrim Kasparowem, jednak nie używano tam ML. Zastosowano obliczanie i przeszukiwanie wszystkich możliwych kombinacji ruchów (tzw. rozwiązanie siłowe – *brute force*), co nie jest korzystaniem z danych i niekoniecznie jest nazywane AI (Press, 2018). W roku 2010 inne dzieło IBM, Watson, zwyciężyło w teleturnieju „Jeopardy!" (Ferrucci i in., 2010). Serią osiągnięć szczyci się zespół DeepMind, należący do tej samej co Google spółki Alphabet: w 2015 roku ich model AlphaGo wygrał w chińską grę Go z mistrzem Fanem Hui. Jest to rozwiązanie

niebazujące na zaprogramowanych regułach i w małym stopniu na przeszukiwaniu możliwości ruchów, a składa się z sieci neuronowych częściowo trenowanych na danych historycznych, częściowo za pomocą uczenia ze wzmocnieniem (*reinforcement learning*) – model gra w grę przeciwko sobie. Później na bazie tego rozwiązania zespół przygotował bardziej ogólny model AlphaZero, który wykazywał bardzo dobre wyniki w grze w szachy, Go i w chińskie shogi (Silver i in., 2016; 2017; 2018).

Omawianie kategorii produktów z zastosowaniem DS zacznę od poprawiania zdjęć z aparatu w smartfonie. Typowym zastosowaniem jest rozpoznawanie scenerii w kadrze, zanim człowiek wciśnie spust migawki. Jest to rozszerzenie znanej dawno przed smartfonami funkcji automatycznego ustawiania (auto) w aparatach fotograficznych, np. ISO, balansu bieli. Model ML klasyfikuje obraz z aparatu jako jedną ze zdefiniowanych scenerii (portret, krajobraz, zbliżenie itd.) i w połączeniu z informacjami z innych czujników automatycznie dostosowywane są wszystkie ustawienia aparatu. Upraszczając, wytrenowany model zapisany do pliku jest w tym przypadku raczej instalowany na smartfonach, choć stosowane są też rozwiązania, w których model wykonuje klasyfikację na serwerach dostawcy, a komunikacja ze smartfonem odbywa się przez internet. To pierwsze rozwiązanie jest szybsze, co uznano za ważne dla użytkownika w przypadku robienia zdjęć. Producenci smartfonów, jak np. Huawei, podkreślają, że ich urządzenia są wyposażone w przeznaczone do „sztucznej inteligencji" procesory (*AI chip* lub *neural engine*), które mają przyspieszać proces działania modelu (Torres, 2018). Zdjęcia już zrobione także mogą być poprawiane z użyciem modeli. Może być to np. wygładzanie skóry, zmiana kolorów. Część z tych elementów działa także dla nagrywania wideo. Tzw. aparaty ze sztuczną inteligencją (*AI camera*) montowane są przeważnie we flagowych, czyli najdroższych i najlepiej wyposażonych smartfonach firm Samsung, Google, Apple czy Huawei. Istnieją także aplikacje mobilne, które umożliwiają korzystanie z omawianych funkcji na urządzeniach nieposiadających ich fabrycznie (Gogoi, 2019).

Z różnych funkcji na smartfonach z systemem Android można korzystać głosowo, tzn. nie wpisując poleceń na ekranie urządzenia, a mówiąc do mikrofonu w języku naturalnym, za pomocą Asystenta Google. Tego rodzaju asystentów oferują także: Samsung – Bixby, Apple – Siri, a dla urządzeń innych niż smartfony Microsoft – Cortana i Amazon – Alexa. Usługi te pozwalają na wykonywanie wielu czynności w środowisku cyfrowym (np. obsługa skrzynki e-mail, mediów społecznościowych, uzyskiwanie informacji o sporcie/polityce/pogodzie itd., planowanie trasy dojazdu, odtwarzanie muzyki). Mogą działać na laptopach, jak Cortana i Siri, ale także na tzw. inteligentnych głośnikach (np. Sonos One, Google Home, Amazon Echo) czy inteligentnych wyświetlaczach (np. Amazon Echo Show, Google Nest Hub, Lenovo Smart Display), stając się częścią tzw. inteligentnego domu. Możliwe jest zatem głosowe sterowanie np. ogrzewaniem, oświetleniem, alarmem itd. (Dunn, 2016; López, Quesada, Guerrero, 2018; Van Camp, 2019).

Operacje na danych i rezultaty działania modeli – czyli to, za co odpowiada DS – są niewielką częścią tych rozwiązań. Składa się na nie cały ekosystem – od infrastruktury informatycznej, przez elektronikę i obudowę samego urządzenia, zasoby naturalne zużywane przez centra danych „w chmurze" oraz na jego wykonanie, transport, przechowywanie, sprzedaż, po pracę ludzką na każdym poziomie wykwalifikowania i każdym elemencie łańcucha dostaw (Joler, Crawford, 2018).

Na koniec przykład uważany za historię sukcesu firmy Google. Jest inny niż powyższe, bo dotyczy laika, który zbudował model ML na własne potrzeby. Mowa o Japończyku Makoto Koike, byłym inżynierze z branży samochodowej, który zaczął pracować w gospodarstwie rolnym swoich rodziców. Sortowanie ogórków okazało się trudnym i czasochłonnym zadaniem, tak więc Koike, który słyszał o możliwościach ML na przykładzie AlphaGo, chciał zatrudnić do jego realizacji komputer. Dzięki temu, że firma Google właśnie udostępniła bibliotekę Tensor Flow, mógł wykonać pierwszy prototyp. Wysoki poziom dopasowania modelu już na początkowym etapie prototypu miał dać Koike pewność siebie i poczucie, że jest on w stanie rozwiązać problem. Inżynier zbudował maszynę do sortowania ogórków: człowiek kładzie warzywo na półkę, miniaturowy komputer Raspbery Pi 3 wykonuje trzy cyfrowe zdjęcia i pierwszy, mniejszy model zainstalowany na komputerze klasyfikuje warzywo jako ogórek lub coś innego. Jeżeli to ogórek, zdjęcia są przesyłane przez sieć na serwer Google Cloud, gdzie na wirtualnej maszynie z systemem Linux działa drugi, większy model, który klasyfikuje ogórka do jednej z dziewięciu grup. Wynik klasyfikacji jest przesłany na Raspbery Pi 3 – ten komputer steruje taśmą, po której jedzie ogórek i ramieniem, które wrzuca go do jednego z dziewięciu pudełek. Trenowanie modelu na domowym komputerze PC z systemem Windows zajęło Koike trzy dni, a dysponował on zaledwie siedmioma tysiącami własnoręcznie zrobionych i zaetykietowanych zdjęć ogórków, które były celowo zmniejszone. Użycie większych, dokładniejszych zdjęć poprawiłoby najpewniej działanie modelu – obecnie dokładność klasyfikacji na produkcji to około 70% – ale wymaga to znacznie większej mocy obliczeniowej niż domowy PC. Tu z pomocą spieszy Google. Serwer Google Cloud oferuje nową usługę Cloud Machine Learning, dzięki której można skorzystać z mocy wirtualnych maszyn w celu trenowania swoich modeli z użyciem TensorFlow. Usługa jest rzecz jasna niedroga, a użytkownik płaci tylko za realnie wykorzystany czas obliczeń. Koike nie mógł się już doczekać, aż ją wypróbuje (Sato, 2016).

4.3. Technologie społecznego świata

Technologia umożliwiająca specyficzny sposób wykonania działania podstawowego jest dla uczestników czynnikiem decyzji co do granic społecznego świata i subświatów. Specjalizowanie się w używaniu określonych technologii prowadzi

do tworzenia kolejnych światów lub subświatów, czego przykładem są stanowiska pracy wyspecjalizowanej w częściach procesu DS. Technologie są nieodłączną składową: procesów stawania się uczestnikiem świata społecznego DS, tożsamości uczestnika (a więc także praktyk zaświadczania o jego autentyczności), procesów legitymizacji działań badanego świata. Technologie jako „zamrożone dyskursy" (Clarke, Star, 2008: 115) odzwierciedlają wartości społecznego świata. Są areną sporów o wartości, co wyraża się m.in. w dyskusjach nad sposobami wykonywania działania podstawowego.

Ma słuszność Natalia Hatalska (2021: 49), pisząc: „żeby móc podejmować decyzje dotyczące sposobów korzystania z technologii, musimy mieć świadomość mechanizmów jej działania". Trzeba rozumieć najważniejsze szczegóły techniczne tych mechanizmów, jeżeli chce się pojąć świat społeczny DS i uczestniczyć w sporze o sposoby korzystania z technologii – w charakterze naukowców, konsumentów i obywateli.

4.3.1. Języki programowania

Skoro działanie podstawowe społecznego świata DS w Polsce to pisanie kodu (do przetwarzania, analizy i modelowania danych), to **najważniejszymi technologiami do realizacji tego działania są języki programowania skryptowego**. Najpopularniejsze obecnie języki to Python i R. Przez lata trwały dyskusje o ich zaletach i wadach w porównaniu do siebie (DeZyre, 2015; Junco, 2017; Kromme, 2017; Matloff, 2017; Olszewski, 2017; Dar, 2018; Piatetsky, 2018c; Sekercioglu, 2018; Sharp Sight, 2018; Silaparasetty, 2018; Watson, 2018; Muenchen, 2019), co opisuję jako arenę.

Obecnie zarówno na świecie, jak i w Polsce **najczęściej używany jest Python** (Nunns, 2017; Piatetsky, 2017; 2018c; Strong, 2018). Około 87% badanych w Polsce firm AI deklarowało, w odpowiedzi na pytanie wielokrotnego wyboru, używanie Pythona, a 38% potwierdzało używanie GNU R (Borowiecki, Mieczkowski, 2019: 36–37). Są zatem osoby używające obu języków programowania – około 12% badanych w międzynarodowej ankiecie KDnugetss (Piatetsky, 2017), natomiast zgodnie z danymi Fundacji Digital Poland w Polsce odsetek ten wynosi co najmniej 25%[34].

Oba języki to otwarte oprogramowanie (*open source*) (Kotuła, 2014: 13–86), choć są dystrybuowane na różnych licencjach[35]. Nie zauważam, by rozróżnienie na oprogramowanie otwarte (Python) i wolne (GNU R) było elementem

34 Zakładając, że 100% używa Pythona lub R, odsetek używających obu języków wynosi 25% (100% – (87% + 38%)). Mowa o firmach, a nie osobach – w zespole jedna osoba może używać Pythona, inna R, a kolejna obu i jeszcze kilku języków programowania.

35 Dla Pythona od 2001 roku do dziś to licencja kompatybilna z GPL (*General Public License*), a dla R to przez cały czas licencje GNU GPL (kilka wersji dla różnych komponentów).

dyskutowanym w polskim świecie społecznym DS. **Widoczne jest za to rozróżnienie na oprogramowanie otwarte** (*open source* czy FLOOS) **i oprogramowanie zamknięte** (*proprietary software*). Oprogramowanie zamknięte jest używane zdecydowanie rzadziej niż otwarte, a jeśli już, to obok rozwiązań *open source*. W badaniu Fundacji Digital Poland wskazywano wyłącznie na oprogramowanie MATLAB (12% wskazań w pytaniu wielokrotnego wyboru) oraz SAS (6%). Pierwsze jest stosowany na uczelniach, drugie w korporacjach, oba umożliwiają pisanie kodu w zamkniętych językach (Borowiecki, Mieczkowski, 2019: 36–37). Podobnie jest w przypadku IBM SPSS – spotkałem się z jego marginalnym używaniem. Piotr Migdał wskazał, że narzędzia *open source* są tak popularne z uwagi na ich elastyczność i „zwinność", pożądaną w DS ze względu na szybki rozwój dziedziny (Borowiecki, Mieczkowski, 2019: 41).

Inne wymienione w raporcie języki programowania do operacji na danych to Scala (14% wskazań), Julia (6%), Octave (5%), Lua (3%) (Borowiecki, Mieczkowski, 2019: 36–37). Wiele języków programowania ma zastosowanie przede wszystkim do działań wspomagających lub działań niezwiązanych ze światem DS, w tym do wdrożeń produkcyjnych modeli ML. Połowa badanych firm wskazała na używanie języków C/C++ lub C#, 38% używa JavaScript, 30% języka Java, 6% korzysta z Ruby, wymieniono także Prolog (2%) i Lisp (2%) (Borowiecki, Mieczkowski, 2019: 36–37). Vladimir Alekseichenko, znany w badanym świecie data scientista, konsultant, przedsiębiorca i popularyzator DS, tak podsumował wyżej prezentowane wyniki:

> **Python i R to języki używane zarówno do tworzenia prototypów, jak i do wdrażania rozwiązań AI. Python jest prosty**, pozwala użytkownikom zbudować funkcjonalność za pomocą kilku linii kodu i w efekcie **umożliwia skupienie się na rzeczywistym przypadku biznesowym**. Dlatego biznes lubi ten właśnie język. **Wielcy gracze, jak Google, Facebook czy Uber**, stworzyli wokół siebie społeczność użytkowników i z tego powodu **łatwo jest rozpocząć swoją podróż w dziedzinie uczenia maszynowego w języku Python. Język R** ma swoje zalety i wielu fanów, ale także pewne **wyzwania, które obejmują standaryzację kodu**. Ponadto w R może być **trudno wdrożyć uczenie głębokie** i jest do tych zastosowań **niewiele dobrych bibliotek** (przy czym nikt naprawdę nie pracuje nad rozwiązaniem tego problemu). Z perspektywy data scientisty pracuje się z Pythonem lub R, a jeśli istnieje potrzeba większej wydajności, to **tylko niektóre części kodu są przepisywane w C/C++ lub Javie** [podkr. oryg.]. Inne języki są właściwie nieważne (Borowiecki, Mieczkowski, 2019: 37–38).

W Pythonie czy R ilość pisanego przez człowieka kodu do wykonania danego zadania jest znacznie mniejsza niż np. w Javie lub C++. W Pythonie i R więcej czynności jest obsługiwanych domyślnie i człowiek nie potrzebuje już myśleć np. o zaplanowaniu i zwolnieniu pamięci komputera na wykonywane operacje czy o podawaniu typu danych dla każdej przypisywanej zmiennej – a tego typu czynności muszą być zapisane przez człowieka np. w Javie. Wypisanie przez komputer frazy początkujących „Witaj, świecie! (*Hello, world!*)" w Pythonie i R (a także w JavaScript, Lisp, Lua, Ruby) to jedna linia kodu. W Javie i C++ to sześć lub siedem linii (Scriptol.com, 2006). **Dla tego samego zadania kod w Pythonie zajmuje około**

10–20% linii potrzebnych na kod w Javie (Paulson, 2007: 12). To jedna z różnic pomiędzy językami kompilowanymi a interpretowanymi, a jak zaraz wskażę – także między statycznymi a dynamicznymi.

Kompilacja i interpretacja są jak dwa sposoby czytania tekstu w nieznanym języku. Kompilacja jest jak całkowite przetłumaczenie artykułu z chińskiego na polski przez tłumacza, a następnie przeczytanie całego artykułu. Interpretacja jest jak czytanie z tłumaczeniem sobie każdego słowa, używając chińsko-polskiego słownika (Chen, 2010: 23).

Python i R to języki interpretowane. Są to także języki dynamiczne, w przeciwieństwie do statycznych, np. Javy czy C#. Oznacza to, że różnią się m.in. sposobem zapisywania typu danych. W R czy Pythonie człowiek nie musi zapisywać ręcznie typu danych, a sam kod może być napisany w ten sposób, że typ danych może się zmieniać. Dodać należy, że językiem dynamicznym jest także Lisp (Paulson, 2007: 12–13). Przykładowo: w R funkcja wykonująca obliczenie może być napisana w ten sposób, że przyjmuje za argumenty wartości o różnych typach danych – liczby całkowite (*integer*), jak np. 2, i zmiennoprzecinkowe (*floating-point*), jak 2,0001 – tak więc typ danych może się zmieniać już po zapisaniu kodu. Wykonanie obliczenia zarówno dla 2, jak i 2,0001 zapewnia nie człowiek, a komputer – a dokładniej interpreter kodu, który sprawdza, jaki jest typ danych dla konkretnej wartości w momencie wykonania kawałka kodu (Wickham, 2014a).

Wymienianych jest wiele zalet takich języków, jak Python i R, czyli języków skryptowych, dynamicznych, interpretowanych. W tej sprawie wypowiadał się m.in. twórca Pythona, Guido van Rossum, mówiąc o (Paulson, 2007):

1) elastyczności – kod może być napisany w ten sposób, że obsłuży różne typy danych, poza tym Python i R działają na różnych systemach operacyjnych (Windows, macOS, dystrybucje Linuxa);
2) efektywności – osiąganej dzięki mniejszej niż dla języków typu Java ilości kodu do wykonania zadania oraz interaktywnej pracy metodą iteracji, a co za tym idzie – szybkości dostarczenia działającego rozwiązania;
3) kreatywności rozwiązań – dzięki odciążeniu człowieka z najbardziej szczegółowych i powtarzalnych zadań.

Python i R są językami uznawanymi za proste lub łatwe i odpowiednimi do pisania kodu przez osoby niemające wykształcenia informatycznego.

Co do różnic historycznych, R został stworzony przez statystyków dla statystyków. Jego nazwa pochodzi od imion twórców Rossa Ihaka i Roberta Gentlemana oraz jest nawiązaniem do nazwy języka S, na którym był wzorowany (Ihaka, Gentleman, 1996; Fox, 2009; Thieme, 2018).

> Od początku język R był tworzony i rozwijany z myślą o osobach zajmujących się analizą danych. Z tego też powodu przez informatyków często nie był traktowany jak pełnowartościowy język, ale raczej jak język domenowy (DSL, ang. *domain-specific language*). Z czasem jednak możliwości R rosły, pojawiło się coraz więcej rozwiązań wykraczających poza analizę danych i dziś R jest jednym z popularniejszych języków programowania (Biecek, 2017: 3).

Python jest językiem ogólnego przeznaczenia. Jego nazwa pochodzi od grupy komików Monthy Python. Python był wzorowany na języku ABC przeznaczonym do nauki programowania dla początkujących i jest uznawany za jeden z najlepszych języków do rozpoczęcia nauki programowania oraz jeden z najlepszych dla ludzi używających kodu od czasu do czasu (*casual coders*) (van Rossum, 1997; Hughes, 1999; Nunns, 2017; ActiveState, 2018). Jednocześnie mówi się o „stromej krzywej uczenia (*steep learning curve*)" programowania w tych językach, ale odnosi się to do porównania wykonywania operacji na danych za pomocą programowania skryptowego z wykonywaniem ich, klikając w interfejsie graficznym (Fox, 2009: 10; Muenchen, 2014; Biecek, 2015a), a nie do porównania do wykonania takich operacji w innych językach programowania. Przy tym ta względna trudność w korzystaniu z R w porównaniu z klikanymi pakietami analitycznymi (jak SPSS) bywa uwazana za to, co wzmacnia społeczność użytkowników (Muenchen, 2014).

W DS raczej pisze się skrypty, niż tworzy oprogramowanie, więc korzysta się głównie z języków skryptowych, dostępnych i odpowiednich dla nieprogramistów. Osoby zajmujące się DS to zatem ludzie „umiejący programować", a nie „programiści" (Nowosad, 2019).

Prostota czy łatwość korzystania jako cechy języka programowania są związane ze **społecznością** ludzi korzystających z języka programowania. Społeczność (*community*, w Polsce terminy te używane są zamiennie) jest uznawana w badanym świecie za ważną cechę czyniącą korzystanie z narzędzi prostszym. Chodzi o **możliwość łatwego dostępu do rozwiązań wypracowanych przez innych członków społeczności dla własnych celów**. Zarówno Python, jak i R są uznawane za języki, wokół których zbudowała się silna społeczność i łatwo uzyskać wsparcie w korzystaniu z nich. Dostępne są dla nich liczne rozszerzenia – pakiety zawierające funkcje, rozwiązania technologiczne ułatwiające korzystanie z tych języków, duża baza napisanego kodu, liczne materiały edukacyjne, wiele odpowiedzi na pytania, możliwe są spotkania twarzą w twarz z osobami korzystającymi z tych języków.

Zwrócenie uwagi na społeczność skupioną wokół danej technologii przy wyborze własnych narzędzi pracy jest typową poradą dla początkujących koderów, nie tylko w DS: „**używaj tego, czego używają Twoi znajomi**" (Norvig, 2001).

Pakiet jest zbiorem funkcji wykonujących określone zadania: „funkcjami możemy analizować dane, rysować dane, odsłuchiwać dane, wykonywać obliczenia, wysyłać maile, robić najróżniejsze rzeczy" (Biecek, 2015a). Pakiet rozszerza wachlarz dostępnych w języku funkcji, co ułatwia wykonywanie zadań. Człowiek potrzebuje wykonać pewną operację na danych i zamiast pracochłonnego pisania kodu na niższym poziomie abstrakcji szuka w pakietach gotowej funkcji, która zrealizuje tę operację. Znani w DS twórcy narzędzi podejmowali wysiłek tworzenia takich funkcji z powodu braku dostępnych rozwiązań, początkowo sami dla siebie.

Dla języka R największymi repozytoriami pakietów są CRAN (*The Comprehensive R Archive Network*), Bioconductor i GitHub (Fox, Leanage, 2016: 10). Do innych należy np. MRAN firmy Microsoft. CRAN to repozytorium ogólne, najstarsze, pod opieką

R Foundation. Tam pobiera się także podstawową dystrybucję[36] GNU R. Bioconductor powstał w 2001 roku i jest repozytorium pakietów dla bioinformatyki. GitHub to publiczne repozytorium kodu dla dowolnych języków programowania, praktyka publikowania pakietów w tym miejscu jest najnowszą, pakiety nie są poddawane kontroli jakości jak w pozostałych repozytoriach (Fox, Leanage, 2016: 10). Pakiety z GitHuba są traktowane jako „nieoficjalne". Istnieje też praktyka publikowania tam wczesnych lub innych wersji pakietów, które są dostępne „oficjalnie" na CRAN. Na przykładzie CRAN można zaobserwować rozwój liczby pakietów R: repozytorium powstało w 1997 roku, pierwsze pakiety pojawiły się tam w 2001 roku, a w 2016 było ich prawie 8 tys. (Fox, Leanage, 2016: 3–5). W dniu 16 lipca 2019 roku CRAN zawierał ponad 14,5 tys. pakietów (R Foundation, 2019). Każdy z nich to, tak jak R, *open source*. Dla porównania SAS – najpopularniejsze na świecie analityczne oprogramowanie zamknięte (*proprietary*) – w 2015 roku miał około 1,2 tys. funkcji. Na CRAN w samym 2015 roku zamieszczono ponad 1,3 tys. pakietów, co oznacza około 27,6 tys. funkcji w jednym roku. Społeczność R stworzyła więcej funkcji przez rok niż firma SAS Institute przez cały swoje istnienie od lat siedemdziesiątych (Muenchen, 2019).

W przypadku Pythona koncentruję się na pakietach związanych z operacjami na danych – jest to język ogólnego przeznaczenia i ma pakiety do innych zastosowań. Największe i najpopularniejsze jest repozytorium PyPI (*The Python Package Index*), z którego instaluje się pakiety komendą *pip*. Dla dziedziny „Nauka/badania"[37] w 2019 roku w tym repozytorium było ponad 10 tys. pakietów – dokładnej liczby nie podano (Python Software Foundation, 2019). Przede wszystkim dla Pythona rozwijane są pakiety do modnego obecnie DL.

Skorzystanie[38] z pakietu może sprowadzać się do wykonania dwóch linii kodu dla każdego z omawianych języków. Dla podstawowych w DS pakietów `dplyr` i `NumPy`:

```
# R
install.packages("dplyr")
library(dplyr)
# Python
pip install numpy
import numpy as np
```

36 Sam język programowania jest abstrakcją, dystrybucja zaś to coś, co pozwala używać języka na komputerze (Wickham, 2014a). Język R ma także dystrybucje oferowane przez Microsoft (2019) i Anacondę (2019b).

37 Wybrano filtr „Intended Audience :: Science/Research".

38 Upraszczając, dla przypadków, kiedy pakiet jest dostępny z CRAN dla R i PyPI dla Pythona. Dla obu języków pierwsza linia kodu instalująca pakiet potrzebna jest tylko wtedy, kiedy pakiet nie jest ściągnięty na komputer użytkownika, a więc przed pierwszym użyciem. Przy kolejnych użyciach należy jedynie załadować pakiet drugą linią. Część `as np` dla Pythona nie jest niezbędna, ale zwyczajowa, podyktowana m.in. wygodą. Dzięki temu elementowi korzysta się później z pakietu `numpy`, pisząc krótsze `np`.

Linie rozpoczynające się znakiem # to tzw. komentarze i nie są one wykonywane przez komputer. Pierwsze linie dla każdego języka to pobranie pakietu z repozytorium przez internet na komputer kodera i zainstalowanie pakietu w odpowiednim miejscu. Drugie linie to uruchomienie pakietu, tzw. załadowanie go do przestrzeni roboczej. Pakiety ładuje się przy każdej nowej sesji, tzn. przy kolejnym uruchomieniu środowiska, w którym pisze się kod, ponowna instalacja standardowo nie będzie potrzebna.

To, że pakiety są publikowane w otwartym dostępie, nie oznacza, że są one budowane tylko oddolnie, przez osoby fizyczne czy nieformalne grupy. Wszystkie używane w DS technologie *open source* mogą być budowane oddolnie, jak również w firmach i na uczelniach, we współpracy firm i uczelni, a także przy wsparciu finansowym z różnego rodzaju grantów. Poważną rolę odgrywają wspierający rozwój *open source* giganci technologiczni (Google, Facebook, Microsoft, Amazon), nie tylko jeżeli chodzi o pakiety, ale o wiele innych narzędzi świata DS.

Istnieją tzw. ekosystemy pakietów: dla Pythona np. SciPy, dla R np. tidyverse. W ich skład wchodzą pakiety zawierające funkcje do wykonania każdego z etapów procesu DS (przetwarzania danych, analizy, modelowania, komunikacji), choć raczej nie wystarczają one do wykonania każdego projektu w całości (Wickham, Grolemund, 2017). Ekosystem nie jest tylko kolekcją pakietów – pakiety te mają podobną konwencję pisania kodu, podobne sposoby zapisywania obiektów (Bryan, Wickham, 2017: 785), są zintegrowane i dobrze ze sobą współgrają. Ma to pomagać w rozwiązywaniu złożonych problemów za pomocą prostych elementów, małych kroków, umożliwiających łatwe rekonfiguracje, a więc iteracyjną pracę z danymi (Wickham, 2018d). Korzystanie z pakietów w tych dwóch ekosystemach – SciPy i tidyverse – jest w badanym świecie uznawane za elementarz umiejętności koderskich.

Inne powszechnie znane pakiety to TensorFlow (TF) i PyTorch oraz tzw. nakładka Keras. W Polsce na używanie TF wskazało 69% badanych firm, na Keras 49%, a na PyTorch/Torch 38% (Borowiecki, Mieczkowski, 2019: 39–40). Korzysta się z nich zdecydowanie najczęściej w Pythonie, bardzo rzadko w R lub innych językach. Keras i TF wraz z językiem Python to trzy składniki najpopularniejszego, według ankiet portalu Kdnuggets, sześcioelementowego zestawu technologii (nazwanego też ekosystemem) *open source* dla DS (Piatetsky, 2018d).

Pakiet TF pozwala na wygodne i szybkie eksperymentowanie z różnymi parametrami modeli, a także szybkie działanie modeli, w tym modeli „na produkcji". Obliczenia mogą być wykonywane „na wielu maszynach w klastrze oraz w maszynie na wielu urządzeniach obliczeniowych" (Abadi i in., 2016). Maszyna to komputer. Klaster to zestaw komputerów działających razem. Urządzenia obliczeniowe w maszynie to procesor ogólnego zastosowania CPU, procesor graficzny GPU, ewentualnie TPU – wyspecjalizowany hardware o zmniejszonej precyzji obliczeń, dostępny wyłącznie na maszynach Google'a, nie jest w sprzedaży. Obliczenia mogą

być zatem wykonywane z użyciem TF szybko i na dużych danych, bo można[39] użyć procesorów GPU lub TPU (urządzeń szybszych niż CPU do obliczeń w ML), a także wielu procesorów w wielu komputerach jednocześnie. PyTorch działa podobnie – także pozwala na korzystanie z szybkości GPU (ale już nie TPU, bo należy on do Google'a, a PyTorch to pakiet wspierany przez firmę Facebook) (Ketkar, 2017; Yegulalp, 2017). PyTorch jest pythonowym portem do biblioteki Torch, napisanej w języku C z dostępem za pomocą języka Lua (Yegulalp, 2017; Collobert i in., 2019; Ierusalimschy, Celes, de Figueiredo, 2019). Zaletą PyTorch jest dobra integracja z popularnymi pakietami[40] Pythona i językiem Python w ogóle (Paszke i in., 2017; Sopyła, 2019). Spory o wady i zalety TF i PyTorcha (por. Sopyła, 2019) lub Keras + TF i PyTorch (Misal, 2018; Agarwal, 2019) opisuję jako element areny w badanym świecie.

Publikowanie w otwartym dostępie omawianych pakietów przez Google'a i Facebooka to element gry tych firm o pozycję w globalnej branży związanej z DS i ML (Arakelyan, 2017), co staram się ująć w pełnej sytuacji badania (*full situation of inquiry*, por. Clarke, 2005: 73).

Keras jest narzędziem wyższego poziomu (*high-level API*) niż TF – nie można wytrenować modelu, korzystając wyłącznie z Kerasa, potrzebne jest zaplecze (*backend*) w postaci np. TF. Można skorzystać z innych backendów, z których każdy to *open source*: Theano (pakiet autorstwa LISA Lab Université de Montréal), CNTK (autorstwa Microsoft), TF (Google) (Keras, 2019). Na PyTorch nie ma takiej nakładki. Zarówno PyTorch, jak i Keras są stosowane w szkoleniu początkujących (Migdał, Jakubanis, 2018; Sopyła, 2019). Uczestniczyłem w warsztatach dla początkujących prowadzonych właśnie z użyciem Kerasa i TF z językami R i Python {obserwacje 26, 35, 39, 43}.

Rozwiązania technologiczne ułatwiające korzystanie z omawianych języków to m.in. dla R środowisko programistyczne, tzw. IDE (*integrated development environment*) o nazwie R Studio (RStudio Team, 2016), a dla Pythona dystrybucja języka o nazwie Anaconda (Anaconda, 2018) oraz notatnik programistyczny Jupyter Notebook (Pérez, Granger, 2015). Oba te rozwiązania umożliwiają także pracę z R. Dla Pythona istnieje wiele IDE i innych edytorów kodu (Reitz, 2018: 26–31), ale nie ma w tym zakresie, tak jak w przypadku RStudio, dominującego[41]

---

39 Za pomocą innych rozwiązań też jest to możliwe. Jednak za zaletę prezentowanych pakietów uznawane jest to, że omawiane czynności można wykonać relatywnie łatwo, szybko i wygodnie w stosunku do innych rozwiązań.

40 Przede wszystkim NumPy z uwagi na korzystanie z podobnych obiektów zwanych tensorami, czyli rodzajem wielowymiarowej tabeli (*multidimensional array*).

41 Dla języka R istnieją także inne IDE, ale są one bardzo rzadko używane lub w ogóle nie są używane. Z pewnością nie można także w badanym świecie wskazać na podziały wśród użytkowników R z uwagi na używane IDE. Z języka R można korzystać także za pomocą notatnika Jupyter (nazwa pochodzi od języków programowania Julia, Python, R), jego szerszego odpowiednika Jupyter Lab, edytora RIDE (także dla Pythona i innych), platformy Radiant, rozszerzenia R Tools dla Visual Studio firmy Microsoft, IDE Architect, IDE Atom (dla wielu

dla DS wyboru (Piatetsky, 2018b). Dla języka Python używanie notatnika Jupyter (w pytaniu wielokrotnego wyboru) wskazało 57% badanych (ponad 1900 respondentów z różnych kontynentów), z IDE PyCharm korzystało 35%, z IDE Spyder 27%[42] (Piatetsky, 2018b). Nie ma badań dotyczących wyboru IDE dla języka R, co wskazuje na dominację RStudio.

Środowisko programistyczne IDE ma zapewniać narzędzia do automatyzacji typowych, powtarzalnych czynności. Ma to na celu zwiększenie szybkości pracy osoby piszącej kod i zmniejszenie liczby popełnianych przez nią błędów (Muûlu i in., 2012: 669). Jest to oprogramowanie, za pomocą którego pisze się kod w danym języku czy kilku językach. Środowisko to ma funkcje pomocne w samym pisaniu kodu – jak autouzupełnianie – a także funkcje pomocne w dodatkowych czynnościach, jak korzystanie z kontroli wersji, instalacja pakietów itp.

Na rysunku 4.8 widoczne są trzy okna: terminal języka R w systemie Windows – Rterm, interfejs domyślnego edytora kodu podstawowej dystrybucji języka R – RGui, IDE – RStudio. Posługuję się tym samym przykładem kodu generującego wykres, co w opisie działania podstawowego. Oprócz kodu, rozpoczynającego się tzw. znakiem zachęty > widać komunikat powitalny używanej wersji języka R. Widoczny jest on zawsze po uruchomieniu konsoli.

Konsola jest podstawowym sposobem pisania i uruchamiania kodu – widać ją, gdy korzysta się z języka programowania na każdy z trzech sposobów. Dla Rterm jest to tylko konsola, dla RGui konsola jest już wewnątrz okna, które ma swoje menu, dla RStudio konsola to jeden z wielu elementów aplikacji. Wszystko, co nie jest konsolą, pełni funkcje pomocnicze. W konsoli możliwe jest pisanie i natychmiastowe uruchamianie linii kodu. Zauważyć należy także, jak wyświetlany jest wynik działania kodu. Przy pracy w Rterm wykres rysowany jest w osobnym oknie urządzenia graficznego, dlatego nie ujęto go na rysunku 4.8. W RGui takie samo okno otwiera się wewnątrz okna, w RStudio wykres można zobaczyć w jednej z kilku kart (otwarta „Plots") panelu w prawym, dolnym rogu. W konsoli praca z kodem jest interaktywna – pisze się linię i po wciśnięciu *Enter* jest ona wykonywana. Jest to sposób używany raczej do szybkiego sprawdzania działania pojedynczych linii czy funkcji, a nie do realizacji całego ciągu operacji na danych. Typową strategią jest pisanie kodu w edytorze skryptów. Taki edytor można otworzyć także w RGui, czego nie ujęto na rysunku 4.8. Podczas pisania skryptu można pisać wiele

---

języków) za pomocą rozszerzenia rbox, również za pomocą rozszerzenia z edytora Vim (dla wielu języków). Dostępne są także graficzne interfejsy (GUI), które pozwalają wyklikać pewne operacje, korzystając z R jako backendu: Rattle, RKWard, Tinn-R, BlueSky, Exploratory, Displayr, Stagraph, Deducer, The R Commander, R MLstudio, esquisse, ggraptR, R AnalyticFlow (karupakalas, 2018; dimlakgorkehgz, 2019).

42 Wymieniono także: Visual Studio (21% wskazań), Sublime Text (12%), Atom (10%), Vim (8,5%), inne IDE (3,1%), Eclipse (3%), inny edytor (2,5%), Emacs (2,2%), gedit (1,3%), Rodeo (1,2%). W podziale na kategorię miejsca pracy ani w podziale na lokalizację geograficzną nie widać wyraźnych różnic w rozkładzie odpowiedzi – wszędzie dominuje notatnik Jupyter, a na drugim i raz na trzecim miejscu znajduje się PyCharm (Piatetsky, 2018b).

linii jedna pod drugą, bez natychmiastowego uruchamiania, a także swobodnie nawigować po liniach, kopiować i wklejać elementy. W konsoli nie jest to możliwe. Można jedynie wyświetlić historię uruchomionych linii klawiszami strzałek „góra" i „dół". W edytorze można także wygodnie zapisać tworzony skrypt – w przykładzie widać, że skrypt nie jest zapisany, co sygnalizuje czerwony kolor czcionki wyświetlającej domyślną nazwę „Untitled1". Linie kodu uruchamiane ze skryptu są przesyłane do konsoli i tam wykonywane. Zauważmy także dostępną wyłącznie w RStudio część interfejsu z kartą „Środowisko (*Environment*)". Wyświetlają się m.in. przypisane wcześniej zmienne – w naszym przykładzie *sekwencja* i *poziom* – oraz informacje o nich. W innych sposobach pracy można zobaczyć listę przypisanych zmiennych w konsoli, wywołując ją w kodzie odpowiednią funkcją.

W Rterm kod nie odróżnia się kolorem od innego tekstu, w RGui kod w konsoli wyróżnia jednolity kolor czerwony; kod w skrypcie miałby jednolity kolor czarny, czego nie prezentuję. Niemniej jest tutaj zaprojektowana różnica w wyglądzie kodu skryptu względem kodu w konsoli (co nie ma sensu w Rterm, gdyż nie ma narzędzia do pisania skryptów). W RStudio ta różnica także występuje, a dodatkowo w skrypcie automatycznie kolorowana jest składnia, np. kolorem niebieskim wyróżniono liczby, kolorem czerwonym ciąg znaków. Silnie odczuwaną we własnoręcznym pisaniu kodu funkcją jest autouzupełnianie. W Rterm i RGui autouzupełnianie jest ograniczone. Można zobaczyć, jakie funkcje zaczynają się od liter, np. „se", wpisując te litery i wciskając klawisz *Tab*. Lista funkcji zostanie wypisana w konsoli. W przypadku gdy możliwa jest tylko jedna sytuacja, np. wpisanie liter „len", to nazwa funkcji zostanie automatycznie uzupełniona do „length" (ale przy wpisaniu „le" otrzyma się listę nazw w konsoli). W RStudio przy wpisaniu nawet jednej litery po wciśnięciu *Tab* otrzymuje się rozwijaną listę podpowiedzi, wyświetlaną w oknie pomocniczym przy wpisanej literze, wraz z informacją, z jakiego pakietu pochodzi funkcja. Po tej rozwijanej liście można nawigować klawiszami strzałek, wyświetlają się automatycznie krótkie opisy funkcji i ich argumentów, a po wciśnięciu *F1* wyświetla się pomoc w panelu w karcie *Help*. Jeśli po wciśnięciu *Tab* widać w podpowiedziach szukaną funkcję, nie trzeba niczego przepisywać, a jedynie podświetlić nazwę funkcji za pomocą strzałek i wcisnąć *Tab* ponownie. Ludzie, pisząc kod, używają autouzupełniania nie tylko, by przypomnieć coś sobie, ale także by zmniejszyć liczbę uderzeń palcami w klawisze. Przykładowo funkcja `install.packages()` dzięki autouzupełnianiu jest wpisywana za pomocą sześciu uderzeń: `inst` + dwukrotnie *Tab*. Ręczne jej wpisanie to 18 uderzeń w klawiaturę. W RStudio autouzupełnianie polega także na domykaniu nawiasów i cudzysłowów. Przykładowo po wpisaniu `inst` + dwa razy *Tab* widać wpisane `install.packages()`, a kursor tekstu znajduje się pomiędzy nawiasami, człowiek przechodzi zatem od razu do wpisania argumentów funkcji, nie myśląc już o konieczności wpisania znaku `)` na koniec ani nie musząc cofać kursora do środka nawiasu. Przy bardziej złożonym i kilkakrotnie poprawianym kodzie pomocna jest funkcja domyślnego

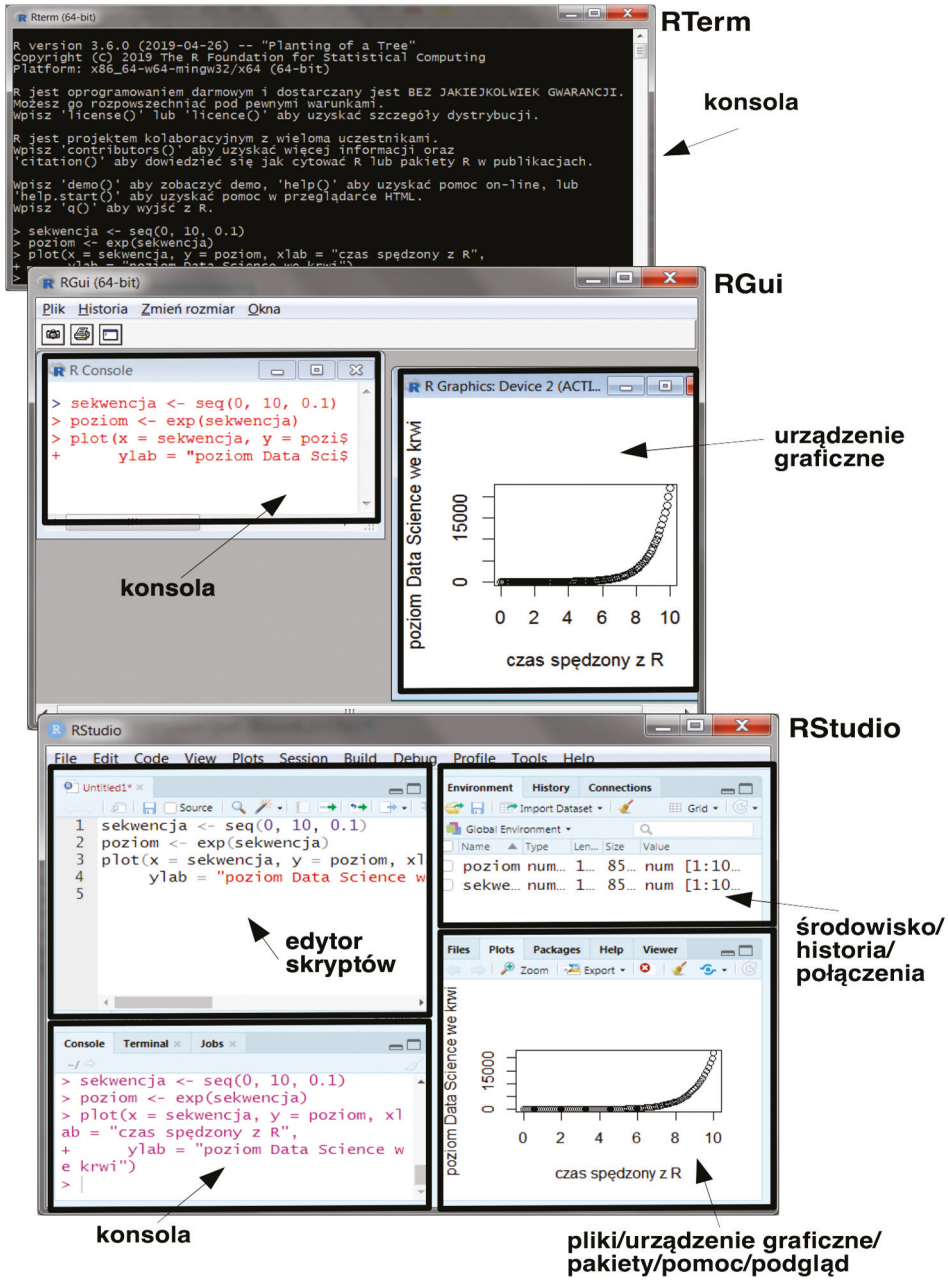
podświetlania początku/końca nawiasu, co służy ewentualnemu ręcznemu domknięciu – bez domkniętych nawiasów kod nie zadziała. RStudio ma wiele wbudowanych skrótów klawiaturowych, np. niewygodny do wprowadzania (będący pozostałością po niewystępującym obecnie na klawiaturach symbolu) symbol przypisania `<-` w RStudio można wprowadzić skrótem klawiaturowym *Alt* + -. Symbol wprowadzany jest wraz z poprzedzającą go i następującą po nim spacją. Przykłady można by mnożyć: RStudio oferuje m.in. narzędzia do wyłapywania i poprawiania błędów w kodzie (*debugging*), połączenia z systemem publikacji wyników do plików tekstowych lub interaktywnych, połączenia z systemami kontroli wersji Git i SVN, menadżera pobierania i aktualizowania pakietów. Wiele operacji pomagających pracować z kodem można dzięki RStudio znaleźć w jednym miejscu, a nawet wyklikać w interfejsie graficznym. Wiele funkcji tego IDE opisanych jest w oficjalnej ściągawce (*cheat sheet*) (RStudio Team, 2017). Tę ściągawkę można także wyklikać z poziomu RStudio pod przyciskiem *Help*.

Jeśli pisanie kodu w Rterm porównać do jazdy drezyną kolejową, to praca w RGui jest kierowaniem lokomotywą spalinową, a w RStudio nowoczesnym składem międzynarodowego ekspresu. Zawsze jednak jedzie się po szynach – języku programowania, a dokładniej zainstalowanej na komputerze dystrybucji języka. Niemniej „drezyna", czyli praca w terminalu, ma swoje szerokie zastosowania, nie tylko do korzystania z języków R lub Python, ale w ogóle do korzystania z komputera (szczególnie na unixowych systemach operacyjnych – macOS i dystrybucjach Linuxa).

Anaconda to nie IDE, a tzw. dystrybucja, która – zdaniem twórców – jest:

> […] najłatwiejszym sposobem na realizację data science i uczenia maszynowego w Pythonie/R na systemach Linux, Windows i Mac OS X. Z ponad 15 milionami użytkowników na całym świecie, jest to przemysłowy standard dla programowania, testowania i trenowania na pojedynczej maszynie, umożliwiający *indywidualnym data scientistom* [podkr. oryg.]: [1] szybkie pobranie ponad 1500 pakietów dla data science w Pythonie/R; [2] zarządzanie bibliotekami (*libraries*), zależnościami (*dependencies*) i środowiskami (*environments*) za pomocą Conda; [3] programowanie i trenowanie modeli uczenia maszynowego i uczenia głębokiego za pomocą scikit-learn, TensorFlow i Theano; [4] skalowalne i wydajne analizowanie danych za pomocą Dask, NumPy, pandas i Numba; [5] wizualizację wyników z użyciem Matplotlib, Bokeh, Datashader, Holoviews (Anaconda, 2018).

Słowo „dystrybucja" używane jest w odniesieniu do Anacondy jako całości, ale właściwie chodzi o dystrybucje języków Python i R. Dodatkowo na Anacondę składa się menadżer pakietów i środowisk Conda oraz kolekcja ponad 1500 pakietów dla DS. Około 200 z nich jest już domyślnie zainstalowanych, nie trzeba zatem robić tego samodzielnie. Inne pakiety można doinstalować z kolekcji pakietów, a jeszcze inne z ogólnego repozytorium PyPI. Podczas instalacji z kolekcji pakietów używane jest polecenie `conda install`, a nie `pip install`, jak dla repozytorium PyPI. To polecenie menadżera środowisk i pakietów Conda.



Rysunek 4.8. IDE jako narzędzie ułatwiające pracę z kodem na przykładzie RStudio w porównaniu do RGui i Rterm

Źródło: opracowanie własne na Windows 7.

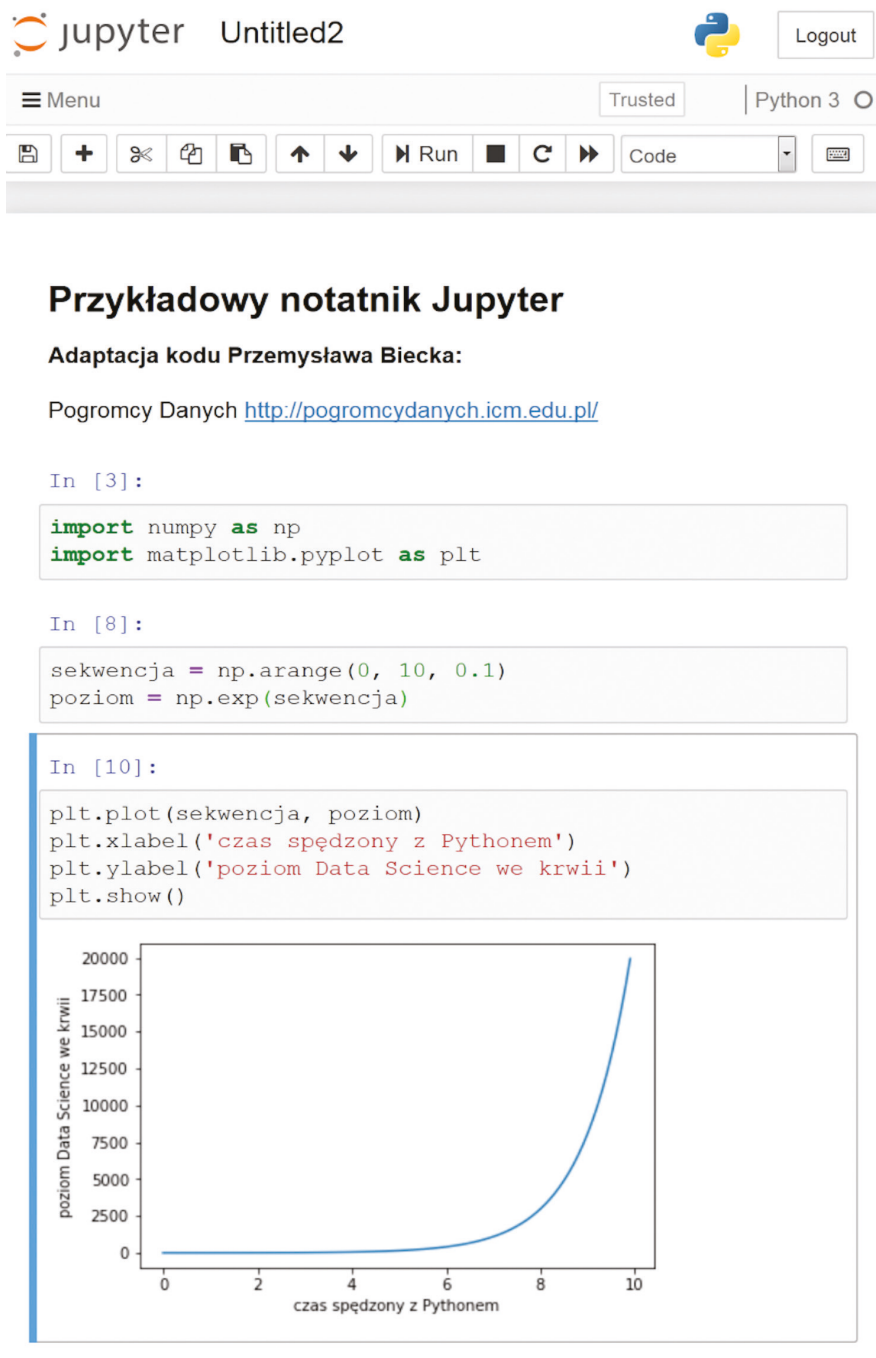
Menadżer Conda jest uznawany za ułatwienie pracy dlatego, że przy korzystaniu z narzędzi *open source* rozwijanych przez międzynarodową społeczność pojawiają się różne wersje samych języków programowania, pakietów i tzw. zależności (*dependencies*). Różne podmioty rozwijają różne części tej rozdrobnionej „galaktyki" technologii, działając trochę chaotycznie na zasadzie tzw. bazaru (Raymond, 2001). Powstają nowe lub inne wersje narzędzi, a w konsekwencji różne elementy stają się niekompatybilne lub działają inaczej, niż działały wcześniej. Jest to uznawane za wysoce problematyczne, ponieważ jedną z podstawowych zalet zapisywania operacji na danych za pomocą kodu ma być powtarzalność (*reproducibility*) czynności (Pérez, Granger, 2015; Wickham, 2018d). Dla języka Python widać tę różnorodność – używane są wersje 2 i 3 tego języka. Sama podstawa kodu może być dla tych dwóch wersji niekompatybilna, tym bardziej zatem dotyczy to pakietów. Menadżer Conda ułatwia tworzenie środowiska z konkretnymi wersjami technologii do realizacji konkretnych projektów. Jeden data scientista może za jego pomocą utworzyć wiele takich środowisk na jednym komputerze, tzw. maszynie, do różnych projektów. Conda ułatwia zarządzanie nimi, kontrolowanie wersji narzędzi składających się na środowiska, ewentualne aktualizowanie narzędzi i sprawia, że nie ma konfliktów pomiędzy tymi wersjami. Innymi narzędziami umożliwiającymi „zamrożenie" całego środowiska w sensie wszystkich wersji narzędzi w danej chwili są tzw. kontenery, jak np. Docker (Avram, 2013) i technologia tzw. orkiestracji kontenerów Kubernetes (Lardinois, 2015). Nie są to części Anacondy, są one technologiami używanymi głównie poza DS, m.in. w inżynierii danych.

Conda ułatwia także instalowanie pakietów za pomocą polecenia `conda install` i instalowanie pakietów w odpowiednich wersjach dla wybranych środowisk (Anaconda, 2019a). Zaletą Anacondy i będącego jej częścią menadżera Conda jest to, że mogą one działać dla różnych języków programowania. Są one jednak kojarzone głównie z językiem Python poprzez związek z PyData i organizacją NumFOCUS, ale umożliwiają pracę z R, a także zarządzanie innymi zależnościami (*dependencies*), np. narzędziem szyfrującym OpenSSL (Vanderplas, 2016). Z moich badań wynika, że Anacondą posługują się wyłącznie osoby używające Pythona. Mogą one używać także R i wielu innych narzędzi, ale nie ma praktyki korzystania z Anacondy przez ludzi w ogóle niestosujących języka Python dla DS. Anaconda upraszcza instalowanie innych narzędzi ułatwiających pracę z Pythonem, takich jak IDE Spyder i notatnik Jupyter. Omawiana dystrybucja pozwala użytkownikowi na obsługę komendami z poziomu terminala/wiersza poleceń, ale i na wyklikanie wielu operacji (jak właśnie instalacja IDE) w interfejsie graficznym Anaconda Navigator.

Zarówno RStudio, jak i Anaconda są dostępne od 2012 roku (Allaire, 2012; Vanderplas, 2016). Oba rozwiązania są rozwijane przez komercyjne podmioty, które oferują także ich płatne, rozszerzone wersje. Wersje te mają udogodnienia do pracy zespołowej i na dużą skalę (RStudio, 2018b; Anaconda, 2019b).

Jupyter to interaktywne środowisko obliczeniowe w formie notatnika. Działa w przeglądarce internetowej. Pozwala na edytowanie i uruchamianie dokumentów analitycznych, które są przystosowane do czytania przez ludzi. Twórcy chcieli narzędzia do zespołowego tworzenia obliczeniowych narracji (*computational narratives*), które składają się z danych, kodu oraz równań, tekstu i wizualizacji (Pérez, Granger, 2015). Określili oni rozwiązany za pomocą tego notatnika problem jako „wspólne tworzenie powtarzalnych narracji obliczeniowych, które można wykorzystać w zakresie różnych odbiorców i różnych kontekstów" (Pérez, Granger, 2015). Notatniki można także w łatwy sposób publikować w formie interaktywnej i statycznej w różnych formatach. Choć zamknięte aplikacje komercyjne – MATLAB, Mathematica, SAS, SPSS i MS Excel – oferują zbliżone funkcjonalności, to fakt, że są zamknięte i kosztowne, czyni je nieatrakcyjnymi dla otwartej i powtarzalnej nauki oraz dla DS (Pérez, Granger, 2015). Logo Jupytera jest wzorowane na rysunku księżyców Jowisza autorstwa Galileusza. Autorzy porównali swoją pracę do Galileusza, który sam zbudował teleskop (Somers, 2018).

Notatnik składa się z komórek (*cells*) – tekstowych i z kodem. Na rysunku 4.9 pierwsza komórka zawiera tekst „Przykładowy notatnik..." sformatowany w języku znaczników Markdown. Widoczny adres strony WWW jest działającym hiperłączem. Kolejne komórki zawierają kod w języku Python wersji 3 (numer wersji widoczny jest w prawej górnej części, poniżej przycisku *Logout*). Rezultaty działania kodu są widoczne – wykres rysowany za pomocą kodu w komórce czwartej, ostatniej. W komórce drugiej kod ładuje odpowiednie pakiety, a w trzeciej wykonuje obliczenia i przypisuje je do zmiennych, więc nie ma wypisywanego wyniku. Liczby widoczne przy komórkach, np. `In [3]:` wskazują kolejność uruchamiania komórek. Jeżeli każdą komórkę uruchomiono by tylko raz, w odpowiedniej kolejności, to widoczne byłyby cyfry 1, 2, 3. Byłoby tak dla w całości bezbłędnie napisanego kodu. Ostatnie uruchomienie komórki ma numer 10. Podczas tworzenia tego przykładu elementy kodu nie działały zgodnie z planem, wprowadzałem więc poprawki i modyfikacje, po czym interaktywnie sprawdzałem rezultat działania. Jest to typowa strategia osób zajmujących się DS (choć tak prosty przykład bez trudu napisze w całości osoba posługująca się Pythonem na co dzień). Notatnik Jupyter ma dużo funkcji, wspiera łącznie około czterdziestu języków programowania. Wiele poleceń można wyklikać, można także korzystać ze skrótów klawiaturowych. Istnieje wiele tzw. magicznych komend, poprzedzonych znakiem **%** dla linii i dla komórek (*line magics*, *cell magics*). Pozwalają one np. na pisanie kodu w HTML-u lub JavaScriptcie, na korzystanie z wiersza poleceń, przekazywanie zmiennych pomiędzy notatnikami, uruchamianie gotowych skryptów Pythona czy notatników Jupyter z poziomu aktywnego notatnika (por. Rogozhnikov, 2016).



Rysunek 4.9. Notatnik Jupyter – interfejs użytkownika z przykładowym tekstem, hiperlinkiem, kodem w języku Python i wizualizacją danych

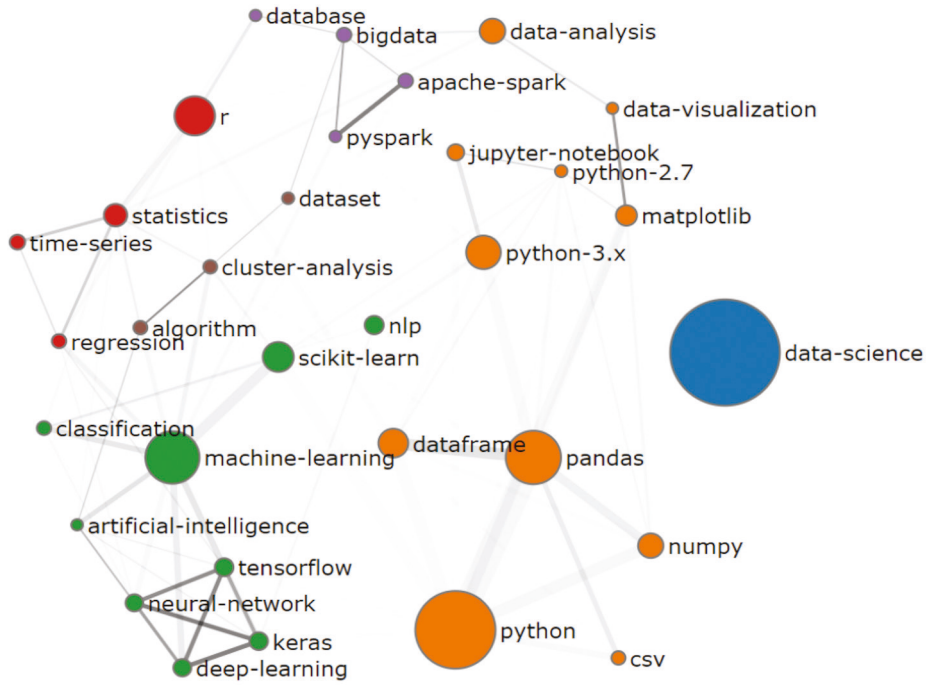
Źródło: opracowanie własne na systemie Windows 7.

Notatnik Jupyter może być **uruchomiony lokalnie**, tzn. na własnym komputerze lub na serwerze wewnętrznym (maszynie w sieci lokalnej, bez połączenia przez internet), a także **w chmurze**, czyli na serwerze zewnętrznym (maszynie położonej gdziekolwiek na świecie, z połączeniem przez internet). Na bazie Jupytera zbudowany jest notatnik chmurowy Google Colaboratory, zwany Colab. Według oficjalnego stanowiska notatnik ten ma pomagać rozpowszechniać studia i badania w dziedzinie ML (Google, 2019). Jest to narzędzie zbliżone do notatnika Jupyter i mające wiele z jego funkcji, a także dodatkowe ułatwienia – przykładowo: jeżeli uruchomiona komórka z kodem zwraca błąd, uaktywnia się przycisk do wyszukania odpowiedzi na ten błąd na Stack Overflow. Całość rozwiązania jest bezpłatna (Carneiro i in., 2018). Spotkałem się z używaniem Colab podczas warsztatów {obserwacje 30, 39, 43}, jest to raczej narzędzie dydaktyczne.

Przestrzenią zadawania pytań, odpowiadania na nie, a przede wszystkim wyszukiwania odpowiedzi jest forum Stack Overflow (SO). Jest ono przeznaczone do pomocy technicznej dotyczącej języków programowania do wszelkich zastosowań. Jest częścią portalu Stack Exchange, w skład którego wchodzą różne fora, także niezwiązane z technologiami. Istnieje tam forum datascience.stackexchange.com, przeznaczone do pomocy merytorycznej. Na forach Stack Exchange każde pytanie ma od jednego do pięciu tagów. Na SO istnieje także tag „data-science".

Na rysunku 4.10 przedstawiam mapę sieci tagów współwystępujących z tagiem „data-science" na forum Stack Overflow. Wierzchołki sieci to 32 najczęściej występujące tagi. Powierzchnia wierzchołków – okręgów – reprezentuje liczbę tagów na SO. Łącznie jest ich około 2,9 tys., co przedstawia powierzchnia niebieskiego okręgu dla tagu „data-science". Wierzchołki mogą być połączone krawędziami. Szerokość krawędzi reprezentuje liczbę pytań oznaczonych oboma połączonymi tagami. Cień krawędzi reprezentuje siłę związku pomiędzy tagami – im ciemniejsza krawędź, tym większa siła związku. Obliczono ją jako stosunek współwystąpień obserwowanych do oczekiwanych (*observed to expected ratio* – O/E)[43] (Czarnocka-Cieciura, Migdał, 2015). Kolor wierzchołków zależy od siły związku między tagami i pokazuje gęściej połączone podsieci, co autorzy nazywają wykrywaniem społeczności. Bazuje to na wynikach badań przedstawionych w pracy doktorskiej Piotra Migdała (2014). Tag „data-science" nie jest połączony z innymi, ponieważ współwystępuje z każdym z nich i posłużył za filtr.

43 TagOverflow rysuje krawędź łączącą dwa tagi tylko wtedy, kiedy O/E jest większe od 1, czyli tagi współwystępują razem częściej niż przypadkowo (*random chance*). *Observed to expected ratio* obliczono jako (liczbę wszystkich pytań × liczbę pytań z tagami A i B łącznie) / (liczbę pytań z tagiem A × liczbę pytań z tagiem B); O/E może być mniejsze od 1, co oznaczałoby, że tagi współwystępują rzadziej niż przypadkowo.



Rysunek 4.10. Mapa sieci tagów współwystępujących z tagiem „data-science" na Stack Overflow

Źródło: TagOverflow, stan na 23.07.2019 r., https://p.migdal.pl/tagoverflow/?site=stackoverflow&size=32&tag=data-science (dostęp: 23.07.2019); Czarnocka-Cieciura, Migdał, 2015.

Widoczne jest pięć grup tagów. Kolorem pomarańczowym zaznaczono te dotyczące języka Python i podstawowych pythonowych narzędzi do analizy i wizualizacji danych. Są to tagi:

1) „python" – jako język programowania ogółem;
2) „python-3.x" i „python-2.7" – pytania dotyczące różnych wersji języka;
3) „jupyter-notebook" – omawiany wyżej notatnik do obliczeniowych narracji;
4) „pandas" – pakiet Pythona dostarczający struktur danych i narzędzi do ich analizy (The pandas project, 2018);
5) „numpy" – pakiet Pythona do pracy z danymi w formie tabelek oraz obliczeń naukowych i inżynieryjnych (Oliphant, 2006);
6) „csv" – popularny format pliku tekstowego z danymi tabelarycznymi, dosłownie oznacza wartości rozdzielane przecinkami (*comma separated values*);
7) „dataframe" – dosłownie ramka danych, jest to struktura danych w formie tabelki, podstawowa dla pakietu „pandas"; ta sama nazwa oznacza podobną strukturę w języku R (na rysunku 4.10 widoczna krawędź łącząca tagi „dataframe" i „r");

8) „data-analysis" – ogólnie analiza danych jako czynność;
9) „data-visualization" – ogólnie wizualizacja danych;
10) „matplotlib" – pakiet do wizualizacji danych w języku Python (Hunter, 2007; Matplotlib, 2018).

Kolorem czerwonym zaznaczono tagi związane z GNU R i jego typowymi zastosowaniami:

1) „r" – język R ogółem;
2) „statistics" – statystyka ogółem;
3) „time-series" – szeregi czasowe, tzn. dane, dla których kolejne obserwacje reprezentują kolejne momenty w czasie, przeważnie w stałym interwale;
4) „regression" – regresja jako rodzaj tzw. nadzorowanego ML; mogą być użyte różne metody, np. od regresji liniowej, przez regresyjne drzewa decyzyjne, po sieci neuronowe (Larose, 2005: 12–13); zauważyć można krawędź łączącą ten tag z tagiem „machine learning".

Kolor zielony ma grupa dotycząca ML:

1) „machine-learning" – ML jako temat;
2) „deep-learning" – uczenie głębokie (DL) jako temat;
3) „neural-network" – sieć neuronowa jako metoda;
4) „artificial-intelligence" – AI jako temat;
5) „tensorflow" – omawiany wyżej pakiet do ML firmy Google;
6) „keras" – omawiana nakładka ułatwiająca korzystanie z TF i Theano;
7) „scikit-learn" – pakiet języka Python dla metod ML (Pedregosa i in., 2011; scikit-learn, 2018);
8) „nlp" – skrót przetwarzania języka naturalnego (*natural language processing*);
9) „classification" – podobnie jak w przypadku regresji różnica to kategorialna zmienna zależna (Larose, 2005: 14–15).

Brązowy kolor oznacza najmniej spójną grupę, do której należą tagi:

1) „dataset" – zbiór danych;
2) „algorithm" – algorytm;
3) „cluster-analysis" – w odróżnieniu od regresji i klasyfikacji jest to rodzaj nienadzorowanego ML; chodzi o znalezienie struktury w danych, podział na klasy wyestymowane z danych na zasadzie podobieństwa; metody to np. k-średnich, sieci Kohonena (Larose, 2005: 16–17).

Fioletowy kolor to grupa tagów dotyczących pracy z dużą ilością danych, czyli big data w technicznym sensie:

1) „bigdata" – jako temat;
2) „database" – baza danych;
3) „apache-spark" – technologia *open source* rozwijana przez Apache Software Foundation; jest to zunifikowany silnik do rozproszonego przetwarzania danych; rozwiązanie do pracy z dużą ilością danych na wielu komputerach, szczególnie jeśli dla takich danych ma być wykonane ML (Zaharia i in., 2016: 56–57);
4) „pyspark" – to rozwiązanie pozwalające korzystać ze Sparka za pomocą języka Python (Drabas, Lee, Karau, 2017).

Większość wymienionych technologii omawiam dalej.

Dzięki powyższej mapie tagów w ogólnym obrazie podobieństw pomiędzy językami Python i R (oba *open source*, skryptowe, interpretowane, relatywnie proste, z silną społecznością) chcę zarysować różnice.

Choć języki Python i R są w części zadań konkurencyjne, to w innej są komplementarne, a w jeszcze innej rozłączne. Część zadań można wykonać w obu językach w dość zbliżony (w sensie wartościującym z punktu widzenia badanego świata) sposób i tutaj trwają spory – które można opisać jako arenę – dotyczące tego, który język będzie „lepszy".

Podsumowując, **języki programowania są uznawane w badanym świecie za narzędzia niezbędne do prawidłowego wykonywania działania podstawowego**. Nie mogą być to jednak dowolnie dobrane języki. Dominuje wykorzystanie języka Python. O ile do 2017 roku użycie Pythona i R było zbliżone, to później popularność Pythona wzrosła wraz z modą na używanie DL. Uczenie głębokie w małym stopniu jest rozwijane w języku R. Oba języki mają wiele cech technicznych uznawanych za zalety w badanym świecie i najczęściej służą do pisania skryptów, a nie do tworzenia oprogramowania. Oba są uznawane za takie, wokół których wytworzyły się silne społeczności. Te społeczności są czymś innym niż świat społeczny DS, osoby korzystające z tych języków mogą zatem nie uważać się za część badanego świata. Język R uznaję za technologię odziedziczoną ze statystyki akademickiej, natomiast Pythona za język zaadaptowany z szeroko pojętego IT. Te historyczne różnice znajdują wyraz w odmiennościach dotyczących cech technicznych obu języków i w społeczności osób korzystających z nich w DS.

4.3.2. Bazy danych

Działanie podstawowe świata społecznego DS jest zawsze nastawione na dane w formie cyfrowej. Co badany świat uznaje za dane? **Osoby zajmujące się DS mogą traktować jako dane wszystko, co zapisane jest cyfrowo** (Lowrie, 2017: 9). Typologię źródeł danych przedstawiam w tabeli 4.1, za zespołem zadaniowym HLG UNECE[44] (UNECE, 2014), pracującym nad możliwościami wykorzystania big data w statystyce publicznej. Nie jest to typologia wypracowana przez świat DS, ale pomaga w zrozumieniu różnych struktur danych.

44 High-Level Group for the Modernisation of Statistical Production and Services (HLG), United Nations Economic Commission (UNCE).

Tabela 4.1. Typologia źródeł danych w podziale na strukturę, pochodzenie i tzw. 4Vs of big data

Cecha	Dane generowane przez użytkowników	Dane z systemów biznesowych	Dane generowane przez maszyny
Struktura	Nieustrukturyzowane	Ustrukturyzowane	Ustrukturyzowane
Pochodzenie	Ludzie	Systemy informacyjne	Urządzenia/sensory
Szybkość pojawiania się (*velocity*)	Szybko (różnice w zależności od zjawiska)	Jak dane z tradycyjnych źródeł	Bardzo szybko
Rozmiar (*volume*)	Duży	Mniejszy	Bardzo duży
Różnorodność (*variety*)	Wysoka	Niska	Bardzo wysoka
Wiarygodność (*veracity*)	Dostawcy/firmy i instytucje	Firmy	Technologia/dostawcy

Źródło: opracowanie własne na podstawie UNECE, 2014.

Dane nieustrukturyzowane scharakteryzowano w kolumnie „Dane generowane przez użytkowników". Są to teksty, fotografie, dźwięki i materiały wideo. Ich źródłem są ludzie, a dane przechowywane są w wielu miejscach: od dysków komputerów osobistych czy smartfonów po serwery różnych usług internetowych, np. mediów społecznościowych. Ich struktura jest „luźna", dane raczej nie są zapisywane w uporządkowany sposób (tabelarycznie). Do tego typu informacji przyporządkowano takie źródła, jak: portale społecznościowe (np. Facebook, Twitter, LinkedIn), blogi i komentarze, prywatne dokumenty elektroniczne, portale z fotografiami (np. Instagram, Flickr, Picasa), portale z wideo (np. YouTube, Netflix, TikTok), wyszukiwarki internetowe, urządzenia mobilne, wiadomości tekstowe, portale umożliwiające generowanie map przez użytkowników oraz pocztę elektroniczną. Inaczej jest z danymi, których charakterystykę pokazano w kolumnie drugiej i trzeciej tabeli 4.1 – tam dane są silniej ustrukturyzowane. Dane o charakterze nieustrukturyzowanym są treściami generowanymi przez ludzi, a pozostałe, czyli ustrukturyzowane, nie, choć mogą być rejestracją bądź pomiarem działań ludzkich. Wpisy i zdjęcia utworzone przez użytkowników portalu społecznościowego nie są przechowywane w formie relacyjnej bazy danych z określonymi atrybutami (zmiennymi). Dane pozyskane z systemów śledzących użytkowników internetu, np. z Google Analytics (śledzenie zachowania na stronach internetowych bazujące na pliku w przeglądarce internetowej, tzw. ciasteczku – *cookie-based tracking*) czy Facebook Pixel (śledzenie bazujące na koncie konkretnego użytkownika Facebooka – *people-based tracking*), dane o transakcjach, ruchu kursora na stronie czy kliknięciach przybierają taką ustrukturyzowaną formę.

Dane ustrukturyzowane przechowywane są od kilkudziesięciu lat w formie relacyjnych baz danych z dostępem przez SQL, są zatem uznawane za źródło tradycyjne. Zanim jednak więcej o nich, dodam, że w tzw. big data (w sensie technicznym)

również przy danych ustrukturyzowanych stosowane są inne, nierelacyjne architektury baz danych, m.in. z uwagi na problem tzw. skalowalności[45].

Wyjaśnienia wymagają jeszcze dwie sprawy: skąd się biorą dane, zanim zostaną gdzieś przechowane oraz czy dane zawsze są przechowywane w bazach danych.

W pytaniu wielokrotnego wyboru 78% badanych podmiotów wskazało, że **dane na potrzeby projektu zapewnia im klient**, 77% – że **dane zapewniają oni sami – firma AI**, 73% wskazało na **korzystanie z danych darmowych w wolnym dostępie** (*open access*), a najmniej – 31% – deklarowało **zakup danych** (Borowiecki, Mieczkowski, 2019: 42–43). Autorzy podsumowali te wyniki stwierdzeniem, że polskie firmy AI korzystają z każdego dostępnego im źródła danych.

W przypadku kiedy **dane zapewnia klient**, raczej będą to dane z bazy danych lub dane w plikach. Inaczej będzie, kiedy chodzi o zespół DS pracujący w ramach większej firmy, która jest tzw. klientem wewnętrznym tego zespołu, a inaczej przy pracy zespołu data scientistów dla tzw. klienta zewnętrznego. W pierwszym przypadku zespół DS raczej samodzielnie korzysta z wewnętrznych, firmowych baz danych na potrzeby realizowanych projektów i eksperymentów. Członkowie zespołu DS mają nadane przez administratorów uprawnienia dostępu do danych. Uprawnienia mogą być różne, w zależności od ról w zespole DS. Niemniej data scientiści pobierają i łączą dane z różnych źródeł w firmie raczej samodzielnie, czyli pojedyncza osoba na swoim komputerze może napisać kod (np. w języku SQL), który wykonuje takie operacje pobrania czy łączenia danych z baz firmy, jakie ta osoba uzna za stosowne w danym momencie (w ramach swoich uprawnień). Udział zespołów IT, administratorów baz danych czy osób zajmujących się bezpieczeństwem danych, prawników, osób decyzyjnych jest w tym przypadku sporadyczny w codziennej pracy pojedynczego data scientisty. Przy pracy dla klienta zewnętrznego zespół DS może uzyskać zbliżony samodzielny dostęp do baz danych na potrzeby realizacji konkretnego projektu, jednak wymaga to wielu działań technicznych, prawnych i organizacyjnych.

Należy tu zwrócić uwagę na **dychotomię – dane są raczej zamknięte** (*proprietary*), **uważane za tajemnicę nawet wewnątrz jednej firmy**. **Technologie, kod i stosowane w DS metody są zaś raczej otwarte** (*open*) (Gutierrez, 2014: 217). Technologie, kod i metody są szczegółowo prezentowane na konferencjach/meetupach/warsztatach, udostępniane w formie pakietów, publikowane na blogach/stronach WWW czy w czasopismach w wolnym dostępie. Natomiast nie spotkałem się z prezentowaniem danych z projektu komercyjnego – często nie podaje się nawet nazwy firmy, dla której był on realizowany, ograniczając się do określenia w rodzaju „dla jednego z największych europejskich banków", „dla naszego klienta z branży FMCG". W wywiadach rozmówczynie/rozmówcy mówiły/mówili, że obowiązują ich formalne zobowiązania tajemnicy (np. w formie *non-disclosure*

45 Ogólnie rozumianej jako możliwość rozbudowy wielkości i jednoczesnego zachowania odpowiedniej wydajności systemu, co może odnosić się zarówno do technologii, jak i np. do całej organizacji.

agreement – NDA) i nie mogą o czymś powiedzieć, tak więc uważałem podanie nazw klientów i podobne szczegóły za wyraz okazanego mi zaufania. Umowy typu NDA to ukonstytuowane zobowiązanie (*commitment*) uczestników świata DS wobec organizacji i osób, które płacą za usługę. Niemniej sprawa otwartości technologii czy metod nie jest jednoznaczna. Przedstawiciele nauk społecznych czy prawnicy krytykują przecież same metody ML jako tzw. czarne skrzynki. Z pozycji DS lub biznesowych rzeczników świata DS szczegóły techniczne i metodologiczne przedstawia się klientom jako magię – a właściwie nie przedstawia.

Inne rozwiązanie, kiedy dane zapewnia klient, to przekazanie zespołowi DS plików z danymi. Ten scenariusz może zachodzić także przy pracy dla klienta wewnętrznego, w sytuacji, gdy jakaś część firmy z różnych powodów nie chce lub nie może dać zespołowi DS dostępu do baz. Wtedy zespół DS może być „odciążony" z samodzielnego pozyskiwania danych z baz i otrzymać wstępnie przygotowany pojedynczy plik z danymi do dalszej pracy. Może to być uznawane przez data scientisów za zaletę, bo przygotowanie danych jest uważane za nudne. Z drugiej strony, jeśli jakość danych w pliku nie pozwala na analizę lub dla tych danych modele nie działają w zadowalający sposób, brak możliwości korzystania z bazy danych *ad hoc* może być postrzegany jako spowolnienie i utrudnienie pracy.

Brak dostępu do baz, niezależnie od tego, czy to klient wewnętrzny, czy zewnętrzny, może być spowodowany także tym, że klient nie ma danych w bazie, a właśnie w plikach. Może tak być w przypadku danych w formie tekstów, obrazu lub dźwięku, co raczej nie budzi zastrzeżeń uczestników badanego świata. Inaczej jest w przypadku danych w formacie tabel, zapisanych w arkuszach kalkulacyjnych. Taki format danych może być z powodzeniem przechowywany w tzw. tradycyjnych, relacyjnych bazach danych i takie przechowywanie jest w świecie DS uznawane za właściwe. Dane tabelaryczne w plikach mają potencjał bycia o wiele bardziej niespójnymi, nieuporządkowanymi, niezdefiniowanymi i zasadniczo trudniejszymi w pracy niż dane w relacyjnej bazie danych. Brak korzystania z relacyjnych baz danych – technologii z kilkudziesięcioletnią historią – dla danych o charakterze tabelarycznym uznawany jest za wyraz tzw. **niskiej kultury danych**. Określenie odnosi się też do innych aspektów korzystania z danych w organizacji, m.in. tego, czy ogólnie są one używane do wsparcia decyzji (tzw. bycie *data-driven*), sposobu przechowywania danych (także w bazach, bo zarówno bazy relacyjne, jak i nierelacyjne mogą być niespójne, chaotyczne, źle zaprojektowane, niekompletne), odpowiedzialności za dane, znajomości posiadanych danych, dostępu do danych, stosowanych narzędzi do pracy z danymi. Tak więc projekty DS czasami mają bardzo długą fazę przygotowania danych, ponieważ osiągnięcie produkcyjnego wdrożenia modeli wymaga zmiany w organizacji – poprawy kultury danych w firmie.

W przypadku **gdy dane potrzebne do projektu zapewnia sama „firma AI"**, chodzi o zapewnienie dostępu do danych zastanych, np. zawartości mediów społecznościowych. Stosowane są m.in. metody tzw. webscrapingu (dosł. 'zdrapywanie'

lub 'zeskrobywanie' danych z internetu). Zbliżoną metodą jest tzw. web harvesting (dosł. 'żniwa w sieci' lub 'zbieranie plonów'), mówi się także o skryptach realizujących scraping/harvesting jako tzw. pajączkach czy crawlerach (dosł. 'pełzaczach'). Są to metody pobierania danych ze środowiska cyfrowego, w których nie korzysta się z tzw. wtyczki, czyli API zapewnionego po stronie źródła danych, a „zdrapuje" się dane bezpośrednio ze środowiska cyfrowego. Za pomocą komputera można pobrać np. ceny książek ze strony Amazon, a nie przepisywać te ceny ręcznie jedna po drugiej (Jemielniak, 2019).

Ponieważ scraping jest pobieraniem danych bezpośrednio ze środowiska cyfrowego, odbywa się zatem bez zgody i wiedzy podmiotów, które administrują tym środowiskiem, są jego właścicielami czy twórcami treści – może to budzić wątpliwości etyczne lub prawne. Trudno czasem określić, czy konkretny element w sieci jest publiczny i można uważać go za zawartość mediów, czy prywatny (Shiab, 2015). Spotkałem się z opinią, że będąc data scientistą nie należy uczyć innych ludzi pisania tzw. pajączków, ponieważ naraża to nauczyciela na konsekwencje prawne {obserwacja 35}.

Firmy AI ankietowane przez Fundację Digital Poland wskazywały także na korzystanie z danych w wolnym dostępie. Mogą to być np. dane z sondaży (tu Kaggle i Stack Overflow oraz z konferencji PyData i WhyR) i właściwie z dowolnych badań dostępnych w publicznych repozytoriach. Jest też wiele zbiorów danych w wolnym dostępie, np. na stronie Wydziału Informatyki Uniwersytetu w Toronto. Zbiory te[46] wykorzystywane są w nauczaniu metod analizy danych lub ML, a także w badaniach nad metodami/algorytmami – służą za punkt odniesienia do tzw. benchmarku. Źródłami zbiorów danych mogą być: Kaggle (publikuje dane do konkursów), strony rządowe czy statystyki publiczne, znana w badanym świecie jest wyszukiwarka do zbiorów danych[47] firmy Google.

Najrzadziej wskazywano na **zakup danych** – każdy inny scenariusz uzyskał ponad 70% wskazań, zakup danych deklarowało zaś 31% ankietowanych firm (Borowiecki, Mieczkowski, 2019: 43). Zakup danych oferują różne podmioty, zarówno komercyjne (np. Acxiom, Experian, ChoisePoint – por. Crain, 2018: 88), jak i publiczne (Główny Urząd Statystyczny, 2019). Po pierwsze, zakup danych to dodatkowy koszt w projekcie, a to przeważnie będzie uznawane za wadę. Po drugie, wśród osób zajmujących się DS może panować przekonanie, że są w stanie samodzielnie poradzić sobie z pozyskaniem potrzebnych danych, dzięki własnej

46 Między innymi: MINST (obrazy – odręcznie pisane cyfry, stosowane do zadań klasyfikacji), CIFAR-10 (obrazy – obiekty w dziesięciu kategoriach, także do zadań klasyfikacji; jest też np. CIFAR-100), Titanic Survivor (tabela – dane o każdym pasażerze Titanica wraz z informacją „przeżył"/„nie przeżył", zadania klasyfikacji binarnej), Boston Housing (tabela – ceny mieszkań w Bostonie, zadania regresji), a przede wszystkim Iris (tabela – pomiary kwiatów trzech gatunków irysa, zadania klasyfikacji) Ronalda Fishera z 1936 roku. Dane te łatwo pobrać z internetu, powołuje się na nie wiele kursów/tutoriali, a także artykułów amatorskich i naukowych, dostępne są w popularnych pakietach Pythona i R.

47 https://toolbox.google.com/datasetsearch (dostęp: 7.04.2020).

dociekliwości i umiejętnościom koderskim. Zakup danych może być zatem postrzegany jako pójście na łatwiznę lub zbędny wydatek. Po trzecie, działalność tzw. brokerów danych – czyli firm zbierających dane i sprzedających je odbiorcom – była przedmiotem krytyki etycznej i prawnej (por. Crain, 2018), więc korzystanie z ich usług może być uważane za nieetyczne, ryzykowne w sensie potencjalnych konsekwencji prawnych lub niekorzystne dla wizerunku firmy/pojedynczych osób zajmujących się DS. Znana jest historia Helen Stokes, której odmawiano wynajęcia mieszkania z powodu informacji o jej dawnych aresztowaniach, znajdujących się w zbiorach danych zakupionych od brokerów. Informacje te zostały dawno wykreślone z oficjalnych rejestrów, ponieważ kobieta nie została skazana. Dla brokerów nie miało to znaczenia, ponieważ „osoby takie jak Helen Stokes nie są ich klientami. Są produktem, a reagowanie na ich skargi kosztuje" (O'Neil, 2017: 208).

W jednym projekcie DS korzysta się z wielu źródeł danych różnego pochodzenia – **łączy się wiele zbiorów danych** pochodzących od klienta, zapewnionych przez firmę realizującą projekt, pozyskanych z wolnego dostępu i zakupionych.

Przechodząc do technologii **relacyjnych baz danych i języka SQL**, zaznaczyć należy, że są to narzędzia powszechnie używane w informatyce, nie tylko w związku z analizą czy modelowaniem danych. Mają one zastosowanie np. w systemach bankowych, sklepach internetowych, systemach bibliotecznych, kadrowych czy magazynowych. „Teoria relacyjnych baz danych była jak dotąd najbardziej udaną próbą opanowania danych, przechowywanych w formie elektronicznej" (Kriegel, Trukhnov, 2011: xxv). Relacyjne bazy danych i język SQL są abstrakcjami, które mają liczne dystrybucje, tzw. systemy baz danych (w Polsce używany jest skrót SZRBD – systemy zarządzania relacyjnymi bazami danych). Do popularnych należą m.in. MySQL, PostgreSQL (oba *open source*) i Oracle, Microsoft SQL Server (zamknięte, komercyjne). Istnieją aplikacje z interfejsem graficznym do zastosowań biurowych, np. Microsoft Access (por. Kriegel, Trukhnov, 2011: xxvi).

Znajomość relacyjnych baz danych i SQL jest uznawana za elementarz umiejętności osoby zajmującej się DS (*Is it a job offer for a Data Scientist?*, 2016; Ismail, Abidin, 2016; Sharp Sight, 2016; Chen, Jiang, 2018; Devlin, 2018; Muenchen, 2019), choć w DS są to technologie **pomijane, bo ich znajomość bywa uznawana za oczywistą**. Są to technologie niezbędne do pracy dla przeważającej liczby osób zajmujących się DS, szczególnie komercyjnie. Są one jednak niemodne w badanym świecie. Za modne, „seksowne" technologie bazodanowe uznaje się ewentualnie[48] te kojarzone z tzw. big data (np. Hadoop, Spark, technologie NoSQL, o czym dalej), a w przypadku modelowania – DL (Devlin, 2018).

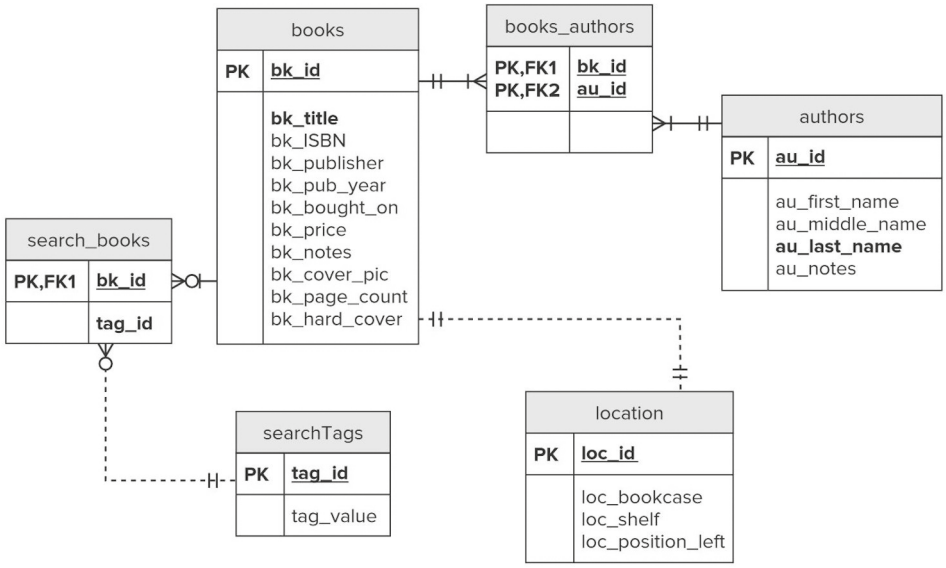
48 Ewentualnie, ponieważ świat DS raczej nie zajmuje się, a na pewno nie w ramach działania podstawowego, problemami baz danych jako takich. Istnieje inny świat inżynierii danych (*data engineering*), który współpracuje z DS w zakresie udostępniania zbiorów danych, o czym więcej w rozdziale piątym.

Devlin (2018) wskazuje, że:

1) SQL jest wszędzie – do tzw. kwerend czy „odpytywania" (*query*) baz danych używają go najbardziej znane firmy stosujące DS w swoich produktach, jak Uber, Netflix, Facebook, Google, Amazon;
2) istnieje popyt na pracowników znających SQL – powołując się na wyniki analizy danych z portalu Indeed, zawierającego oferty pracy, wskazano, że znajomość SQL-a była najczęściej wymienianą umiejętnością dla stanowisk pracy z danymi (dla różnych przekrojów wymieniano ją w około 36% do 56% ofert pracy);
3) SQL jest językiem o ustabilizowanej pozycji – wskazując na wyniki ankiet *Stack Overflow Developer Survey 2017* i 2018, stwierdzono, że popularność SQL-a jest stabilna, a ponadto wyższa niż języków R i Python; ta sytuacja występuje zarówno dla ogółu ankietowanych, jak i wyłącznie dla osób zajmujących się DS.

Koncepcja relacyjnych baz danych powstała na początku lat siedemdziesiątych (Ceri, 2018: 68–69), a jej pomysłodawcą był Edgar F. Cood z firmy IBM (Codd, 1970). Jej podstawę matematyczną stanowi teoria zbiorów. Ważnym elementem tej koncepcji jest unikanie tzw. redundancji (powtarzania informacji w wielu miejscach) (Codd, 1970: 385–386) za pomocą przechowywaniu danych nie w jednej tabeli, a w wielu tabelach połączonych ze sobą za pomocą tzw. kluczy (*keys*). Przykładowo: biblioteka potrzebuje informacji o książkach, ale jeżeli przechowywać by całość danych w jednej tzw. płaskiej tabeli (dwuwymiarowej, mającej wiersze i kolumny), prowadzi to do powtarzania informacji, np. o autorach, dla różnych egzemplarzy książki. Na rysunku 4.11 przedstawiono przykład schematu relacyjnej bazy danych dla biblioteki (Kriegel, Trukhnov, 2011).

Tabele dwuwymiarowe (zwane za Coddem relacjami – w powyższym przykładzie to: books, books_authors, authors, search_books, searchTags, location) połączone są ze sobą za pomocą kluczy, czyli identyfikatorów unikatowych wpisów (bk_id, au_id itd.). Pod nazwami tabel w pierwszym polu tabeli zapisano skróty: PK – *primary key*, klucz główny; FK – *foreign key*, klucz obcy. W bazach relacyjnych możliwe są trzy typy połączeń: jeden do jednego, jeden do wielu i wiele do wielu. Omawiane tabele dotyczą książek (*books*) i autorów (*authors*). Są one złączone na zasadzie wiele do wielu, tzn. jedna książka może mieć jednego lub więcej autorów, a autor może napisać jedną lub więcej książek. Takie połączenie wymaga zastosowania przynajmniej jednej tabeli pośredniej, łącznikowej, tutaj books_authors, w której każdy wiersz jest unikatowym wpisem, czyli połączeniem jednego autora z jedną książką (Kriegel, Trukhnov, 2011: 85–86). Można zatem przechowywać informację np. o imieniu autora tylko raz, w odpowiednim polu – czyli kolumnie au_first_name – tabeli authors, a nie przy każdym egzemplarzu książki tego autora. Ułatwia to także usuwanie, uaktualnianie, poprawianie czy dodawanie nowych wpisów, bo wykonuje się to tylko w jednym wierszu w kolumnie odpowiedniej tabeli, a nie wiele razy w wielu wierszach.



Rysunek 4.11. Schemat relacyjnej bazy danych dla biblioteki do celów dydaktycznych

Źródło: Kriegel, Trukhnov, 2011: 85.

Dane w postaci tabeli o dwóch kolumnach – nazwisku autora i tytule napisanej przez niego książki – można by „wyciągnąć" z bazy o przykładowym schemacie za pomocą zapytania w języku SQL (Kriegel, Trukhnov, 2011: 95):

```
SELECT authors.au_last_name, books.bk_title
FROM books
JOIN books_authors ON (books.bk_id = books_authors.bk_id)
JOIN authors ON (books_authors.au_id = authors.au_id);
```

Wszystkie operacje na relacjach (czyli tabelach) w języku SQL dają w wyniku zawsze relacje. Polecenie `SELECT` dotyczy tego, jakie pola (czyli kolumny) tabel mają znajdować się w tabeli wynikowej. Jest to zapisane w składni `nazwa_tabeli_źródłowej.nazwa_pola`. Klauzula `FROM` dotyczy tabeli źródłowej. `JOIN` to polecenie złączenia tabel, a `ON` dotyczy tego, za pomocą jakich kluczy ma być wykonane złączenie (Kriegel, Trukhnov, 2011: 95).

Tego rodzaju zapytania – w praktyce mogą być znacznie bardziej skomplikowane[49] – są przez osoby zajmujące się DS realizowane raczej w środowisku Pythona/R, czyli tam, gdzie wykonywane są dalsze etapy projektu. Istnieją do tego celu odpowiednie pakiety. Można używać kwerend (*query*) w języku SQL za pomocą

49 Pokazany przykład to cztery linie kodu, „prawdziwe" (*real-world*) zapytania mogą zajmować np. 45 linii (Devlin, 2018).

innych środowisk, ale w badanym świecie istnieje tendencja do realizowania całości zadań wewnątrz środowiska, w którym używa się Pythona/R, gdyż to uznawane jest za najwygodniejsze i najefektywniejsze.

Osoby, które pracują głównie z użyciem języka SQL bądź z użyciem SQL i arkuszy kalkulacyjnych, nie są uznawane za uczestników badanego świata DS. SQL-em oraz np. szerszym językiem proceduralnym PL SQL (Feuerstein, 1996) posługują się administratorzy lub architekci/projektanci baz danych. Są to dość ustabilizowane zajęcia kojarzone z informatyką, nie z DS. Takie osoby poza zapytaniami do bazy danych w większym stopniu zajmują się administracją, czyli np. nadawaniem uprawnień do dostępu do baz, lub architekturą, czyli projektowaniem baz, zarządzaniem migracją, obsługą serwerów i, ogólnie rzecz biorąc, infrastruktury technicznej umożliwiającej pracę baz itp. W odróżnieniu od nich osoby pracujące z SQL i arkuszami kalkulacyjnymi, ewentualnie także z innym zamkniętym oprogramowaniem analitycznym, nazywane są np. analitykami danych, osobami zajmującymi się raportowaniem, ewentualnie BI (*business intelligence*), choć w tym ostatnim przypadku raczej mówi się dodatkowo o narzędziach automatyzujących raporty w formie wizualizacji (np. Tableau, Power BI, Qlick View). Takie osoby używają zapytań w SQL, by wyciągnąć z bazy danych zasoby, które następnie przetwarzają i analizują za pomocą innych narzędzi. Mogą również za pomocą samego języka SQL dokonywać podsumowania danych, np. typu obliczenie średnich w podziale na grupy.

Data science bywa łączone z pracą na danych nieustrukturyzowanych, jako danych bardziej współczesnych (nieco później powstały metody składowania, przetwarzania i modelowania danych tego rodzaju), relatywnie trudniejszych w pracy i raczej kojarzonych z tzw. big data oraz modnymi metodami DL. Niemniej **nie ma w badanym świecie zgody co do tego, że za osobę zajmującą się DS można uznawać wyłącznie taką, która pracuje z danymi nieustrukturyzowanymi** (jak obraz, tekst czy dźwięk). To doświadczenie w pracy z różnymi typami danych jest uznawane za legitymizację autentyczności uczestników.

Dane nieustrukturyzowane mogą być przechowywane w formie plików na komputerach, bez wykorzystania żadnej bazy danych, ale jest to uznawane za wyraz tzw. niskiej kultury danych, niewygodne oraz nieefektywne. Dane nieustrukturyzowane, szczególnie jeżeli mają duży rozmiar, mogą być przechowywane w nierelacyjnych bazach danych (Feinberg, 2017). Używane jest określenie „NoSQL", co ma oznaczać „nie SQL" (*no SQL*), lub „nie tylko SQL" (*not only SQL*), jednak bywa to uznawane za niewłaściwe, nierelacyjne bazy danych mogą bowiem umożliwiać dostęp przez SQL (Feinberg, 2017).

Bazy nierelacyjne czy NoSQL to ogólne określenie na alternatywne do baz relacyjnych architektury przechowywania i dostępu do danych cyfrowych – kluczy-wartości (*key-value*), kolumnowe (*wide-column*, *column family* lub *column-oriented*), dokumentów (*document*), grafowe (*graph*) (Chen, Mao, Liu, 2014: 186–189; Feinberg, 2017). Umożliwiają one przechowywanie i dostęp do danych innych niż

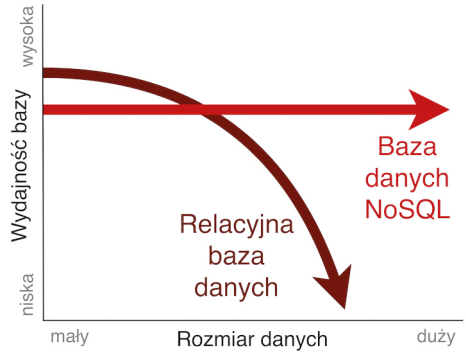
te pasujące do schematu relacyjnego, czyli przede wszystkim do danych nieustrukturyzowanych. Niemniej stosuje się je także do danych ustrukturyzowanych, ale takich, dla których podejście relacyjne uznawane jest za nieodpowiednie (Chen, Mao, Liu, 2014: 186–189; Feinberg, 2017). Przykładem mogą być dane generowane przez maszyny czy tzw. Internet Rzeczy, gdzie dane napływają w dużej ilości w bardzo małych odstępach czasu, niemniej mają formę dwuwymiarowej tabelki. Relacyjne bazy danych to zatem tylko jedno z możliwych rozwiązań do przechowywania danych ustrukturyzowanych, a właściwie konkretnej ich odmiany, która była stosowanym przez dziesięciolecia standardem.

Architektury kolumnowe i dokumentów są odmianami architektury kluczy-wartości. Idea podstawowa jest taka sama – dane przechowywane są w formie wierszy, a każdy wiersz ma unikalny klucz (*key*) oraz dowolną wartość (*value*), co tworzy unikalną parę klucz-wartość (*key-value pair*). Popularną dystrybucją bazy klucz-wartość jest Redis. Na przykładzie mediów społecznościowych wartością wiersza może być profil użytkownika, pojedynczy komentarz lub cała dyskusja. Nie ma zdefiniowanego schematu danych jak w bazach relacyjnych – pole wartości może zawierać dane różne w sensie semantycznym i technicznym. Dla baz dokumentów, np. popularnej MongoDB, wartości są dokumentami, które mogą mieć swoją wewnętrzną strukturę, zbliżoną do plików JSON. Dla baz kolumnowych, jak Cassandra, wartości są przechowywane w postaci tzw. rodziny kolumn (*column family*), w której każda kolumna stanowi drugi poziom pary klucz-wartość (Bhatt, 2013).

Architektury grafowe oparte są na teorii grafów, m.in. pojęciach krawędzi (*edges*) i wierzchołków (*nodes*), oraz na ich własnościach. Stosuje się je w przypadkach, kiedy połączenia między danymi są uznawane za tak samo ważne jak same dane. To dane dotyczące np. sieci społecznych, informacji w internecie czy połączeń między genami, białkami i enzymami (Vicknair i in., 2010; Miller, 2013: 142–143). Są to architektury o zastosowaniach węższych niż inne podejścia nierelacyjne. Do znanych dystrybucji bazy grafowej należy np. Neo4j. Architektur grafowych używają m.in. Google (Knowledge Graph), Facebook (Open Graph), Twitter (FlockDB). Największe firmy technologiczne wykorzystują i udostępniają także omawiane wcześniej typy baz nierelacyjnych – BigTable firmy Google to jeden z najwcześniejszych systemów nierelacyjnych baz danych (Chang i in., 2006), Cassandra to rozwiązanie *open source* Facebooka, Dynamo jest rozwiązaniem używanym przez Amazon, a LinkedIn używa systemu o nazwie Project Voldemort (Vicknair i in., 2010).

Nierelacyjne bazy danych mają zastosowanie w przypadku konieczności przechowywania danych w tabelach o wielu kolumnach, z których znaczna część będzie pusta, kiedy występuje dużo relacji wiele do wielu, kiedy dane mają charakter hierarchicznej struktury drzewa (*tree-like*), kiedy można spodziewać się potrzeby częstych zmian schematu danych (Vicknair i in., 2010). Tak więc, upraszczając, ideą baz nierelacyjnych jest – po pierwsze – brak zaprojektowanego schematu czy definicji

nałożonej na dane, a po drugie tzw. skalowalność, czyli możliwość niemal nieograniczonej rozbudowy wielkości przechowywanych danych czy liczby użytkowników bazy i jednoczesnego zachowania odpowiedniej wydajności systemu (Sumbul, 2014; Bugnion, 2016). Ta skalowalność jest osiągana zarówno dzięki architekturze baz, jak i rozpraszaniu przechowywania i przetwarzania danych na dużą liczbę komputerów (Sumbul, 2014; Bugnion, 2016), o czym więcej w kolejnej części, na przykładach technologii Hadoop i Spark. Pojęcie skalowalności pojawia się w badanym świecie, a także w szeroko pojętej informatyce w różnych kontekstach.



Rysunek 4.12. Porównanie skalowalności relacyjnych i nierelacyjnych baz danych – wydajność w stosunku do rozmiaru danych

Źródło: opracowanie własne na podstawie Sumbul, 2014.

Na rysunku 4.12 oś x reprezentuje rozmiar danych (*volume of data*), a y to wydajność bazy danych (*performance*). Relacyjne – zauważmy określenie „tradycyjne" w oryginalnym tytule rysunku – bazy danych (*relational*, linia ciemniejsza) są przedstawione jako nieco bardziej wydajne niż bazy nierelacyjne (*NoSQL*, linia jaśniejsza) tylko dla małego rozmiaru danych. Wraz ze wzrostem rozmiaru danych spada wydajność baz relacyjnych. Dla baz nierelacyjnych wydajność ma pozostawać taka sama, bez względu na rozmiar danych. To właśnie oznacza, że bazy nierelacyjne są skalowalne, używa się także określenia „horyzontalnie skalowalne", a potocznie mówi się, że „dobrze się skalują".

W odniesieniu do systemów, które mają zapewniać dostęp do wszystkich dostępnych w organizacji danych – ustrukturyzowanych i nieustrukturyzowanych – **używane jest określenie *data lake*** (dosł. 'jezioro danych', nie spotkałem się jednak z tłumaczeniem na język polski). Za jego twórcę uznaje się Jamesa Dixona z firmy Pentaho:

> Jeśli pomyślisz o tematycznej hurtowni danych (*data mart*), że jest ona magazynem wody butelkowanej – oczyszczonej, zapakowanej i uporządkowanej w celu łatwego jej spożycia – to jezioro danych (*data lake*) jest dużym zbiornikiem wody w bardziej naturalnym stanie. Dane płyną ze źródeł strumieniami (*stream*), aby wypełnić jezioro, a różni użytkownicy jeziora mogą przyjść, aby zbadać wodę, zanurkować w niej lub pobrać jej próbki (Dixon, 2010).

Dixon przedstawia *data lake* w odróżnieniu od *data mart*. Najpierw przybliżę, czym jest to drugie. Tematyczna hurtownia danych (*data mart*) to składowa hurtowni danych (*data warehouse*). Są to rozwiązania zorientowane na odczyt danych historycznych w celach analitycznych, stosowane dla relacyjnych baz danych. Określane są one skrótem OLAP (*online analytical processing*), a relacyjne bazy danych zorientowane na zapis i przetwarzanie w celach wykonywania bieżących operacji nazywane są OLTP (*online transaction processing*). Tak więc hurtownie danych (*data warehouse*) to bazy typu OLAP, a tematyczne hurtownie danych (*data marts*) to składowe hurtowni zawierające dane o określonym temacie czy przeznaczone dla pewnych użytkowników (Kriegel, Trukhnov, 2011: 93–94).

Krytyka hurtowni i tematycznych hurtowni danych, która stała za koncepcją *data lake*, dotyczyła po pierwsze tego, że w hurtowniach tworzą się oddzielone od siebie „silosy" z danymi, np. na poziomie odrębnych działów w firmie lub odrębnych lokalizacji firmy. W wyniku tego niemożliwe jest przeprowadzanie analizy czy modelowania danych na poziomie całej organizacji. Po drugie, dane w hurtowniach nie są w formie surowej, a są zagregowane i wybrane do celów analitycznych. Ten wybór i agregacja są podyktowane względami technicznymi i ekonomicznymi, czyli z uwagi na charakter techniczny relacyjnych baz danych uznawano za opłacalne – ze względu kosztów i wydajności – by zapewniać dostęp do celów analitycznych do danych tylko w formie pogrupowanej tematycznie, tylko wybranych części i częściowo zagregowanych. To miał na myśli Dixon, mówiąc o tematycznych hurtowniach danych jako magazynach wody butelkowanej. Takie podejście krytykowano jako mające pozwalać na udzielenie odpowiedzi wyłącznie na już znane z góry pytania (Dixon, 2010; Miloslavskaya, Tolstoy, 2016; Nicolaus i in., 2016; Feinberg, 2017).

Data lake ma zapewniać dostęp do surowej postaci wszystkich aktualnych danych w organizacji. Nie chodzi o żadną konkretną technologię, choć koncepcja *data lake* zakłada, że składowanie i przetwarzanie danych będzie rozproszone na wiele komputerów (Dixon, 2010; Miloslavskaya, Tolstoy, 2016; Nicolaus i in., 2016; Feinberg, 2017), np. za pomocą technologii Hadoop, o czym dalej. To, co uznawano za zalety *data lake* – łatwość implementacji, elastyczność względem rodzajów danych, przechowywanie danych w nieprzetworzonej postaci – nierzadko w praktyce prowadziło do powstawania chaotycznych repozytoriów, z których trudno było uzyskać dane do zastosowań analitycznych. Pojawił się termin *data swamp* (dosł. 'bagno danych'), który oznacza takie zaniedbane repozytorium (Brackenbury i in., 2018: 1–2):

> Pojawił się Hadoop, tanio można wrzucić dużo danych, jest MapReduce, więc można robić wielowątkowo. Miało być data lake, a jest data swamp i to ma teraz wiele firm. {notatka terenowa, obserwacja 13}

Koncepcja *data lake* oraz bezpośrednio powiązane z nią technologie nierelacyjnych baz danych i rozpraszania ich składowania oraz przetwarzania na wiele komputerów są tym, co w sensie technicznym określa się mianem big data.

Definicja „3Vs of big data" – wielkość danych (*volume*)[50], ich różnorodność (*variety*)[51] i szybkość pojawiania się nowych danych (*velocity*)[52] – odnosi się właśnie do takiej charakterystyki danych, dla której koncepcja *data lake* i powiązane z nią technologie zostały stworzone.

Osoba zajmująca się DS nie jest osobą zajmującą się big data. Uczestnik badanego świata może pracować z danymi o charakterze big data, choć jestem przekonany, że uczestnicy ewentualnie byliby skłonni do stosowania określenia „dane nieustrukturyzowane" czy „dane o dużym rozmiarze". Niemniej, jak wskazywałem wcześniej, **praca z tzw. big data nie jest w badanym świecie elementem koniecznym, by być uznanym za uczestnika świata społecznego DS**.

W badanym świecie posługiwanie się określeniem „big data" jest traktowane jako wskaźnik, że dana osoba nie jest uczestnikiem świata DS, a nawet więcej – że jest tzw. osobą nietechniczną. Określenie „big data" jest uznawane za niejasne, przereklamowane, a obecnie nieco przestarzałe i charakterystyczne dla dyskursu nietechnicznego: marketingowego, prawniczego, dziennikarskiego, ewentualnie nauk społecznych[53]. Z punktu widzenia osób zajmujących się DS mówi się w kontekście big data głównie o technologii Spark, bo to jest narzędzie charakterystyczne dla badanego świata.

4.3.3. Obliczenia lokalne, w chmurze, rozproszone

Wszelkie operacje na danych cyfrowych przeprowadzane są na komputerach. Słowo „komputer" ma zastosowanie do operacji wykonywanych lokalnie, w chmurze, rozproszonych i nierozproszonych.

Kiedy praca wykonywana jest **lokalnie**, oznacza to, że człowiek korzysta z hardware'u, który fizycznie znajduje się w pobliżu. Pracuje na osobistym komputerze lub na serwerze wewnętrznym (maszynie w sieci lokalnej, bez połączenia przez internet). W przypadku pracy **w chmurze** praca wykonywana jest na serwerze zewnętrznym (maszynie położonej gdziekolwiek na świecie, przy połączeniu przez internet). Praca z danymi na serwerze, bez względu na to, czy jest to serwer lokalny, czy w chmurze, wykonywana jest za pomocą komputera osobistego, tzn. laptopa lub komputera stacjonarnego. Różnica względem pracy na osobistym komputerze polega na tym, że do wykonywania operacji używane są podzespoły i moc obliczeniowa serwera. Komputer na biurku służy jedynie jako urządzenie pozwalające na dostęp do innej maszyny, a jego podzespoły i moc obliczeniowa

50 Rozmiar danych – bywa rozumiany jako zbiory większe niż 1 terabajt, a także takie, których nie da się przetwarzać za pomocą tradycyjnych narzędzi (relacyjnych baz danych i arkuszy kalkulacyjnych).

51 Zróżnicowanie – różne struktury, formaty i charakter danych.

52 Szybkość pojawiania się nowych danych – ciągły ich napływ, co powoduje, że potrzebne są metody umożliwiające wydobywanie informacji z danych w czasie rzeczywistym.

53 Moje wczesne prace zawierały, rzecz jasna, określenie „big data" już w tytule tekstu (Żulicki, 2016; 2017).

służą np. do uruchomienia przeglądarki internetowej czy terminala, a nie do operacji na danych.

W przypadku pracy lokalnej człowiek realizuje operacje na danych w całości na sprzęcie należącym do niego lub do organizacji, której jest częścią. Człowiek (organizacja) kupił sprzęt, skonfigurował go i utrzymuje jego działanie. W przypadku tzw. chmury operacje realizowane są na żądanie – sprzęt należy do innej organizacji, a podmiot korzystający z niego płaci za wykorzystane zasoby sprzętowe i czas. Jest to wypożyczanie, a nie posiadanie. Można porównać to do sytuacji, kiedy firma posiada ciężarówkę (to praca lokalna) – konkretny samochód z danego rocznika, o pewnym przebiegu, z takim, a nie innym silnikiem, z określoną przestrzenią bagażową. Firma dba o naprawy, przeglądy techniczne, płaci ewentualne mandaty itp. Kiedy firma wypożycza ciężarówkę na godziny od innej firmy, która ma ich wiele (to praca w chmurze), może dobrać samochód do jednorazowego zadania, posiadanego w tej chwili budżetu itd.

Komputer, na którym powstała większa część tej książki, ma jeden czterordzeniowy procesor CPU (procesor ogólnego zastosowania), procesor GPU (procesor graficzny), 8 gigabajtów (GB) pamięci RAM i 256 GB przestrzeni dyskowej. W świecie DS taki sprzęt byłby uznany za przeciętny zestaw do pracy biurowej. Na takim komputerze, tj. używając jego podzespołów i mocy obliczeniowej, można pracować jedynie ze zbiorami danych rzędu maksymalnie kilku GB. Dlaczego nie 100 GB? Dzieje się tak[54], ponieważ w języku R zasadniczo (a w języku Python np. dla popularnego w zastosowaniach DS pakietu „pandas") całość zbioru danych jest wczytywana do pamięci RAM i tam wykonywane są wszystkie operacje. Efektywna praca możliwa jest dla zbiorów danych rzędu 10–20% pamięci RAM konkretnej maszyny, a dla zbiorów wielkości powyżej 50% RAM praca jest niewykonalna (Kane, Emerson, Weston, 2013: 2–3). Jednym z rozwiązań jest pobranie próbki danych (*subset*) na własny komputer, innym zastosowanie pakietów, które umożliwiają bardziej efektywne wykorzystanie zasobów własnego komputera (Kane, Emerson, Weston, 2013: 2–3). Można też zastosować przetwarzanie rozproszone czy równoległe (*parallel*) (Wickham, Grolemund, 2017).

Można zadbać o mocniejszą konfigurację komputera personalnego. O ile nie stosuje się więcej niż jeden procesor CPU, a rzadko więcej niż jeden GPU, to jakość tych procesorów może być różna. Niektórzy uczestnicy świata DS inwestują w personalne komputery, np. instalują 64 GB RAM w laptopie[55], inni korzystają ze sprzętu „biurowego". Istnieje praktyka inwestowania, indywidualnie lub na poziomie organizacji, w wysokiej jakości karty graficzne (GPU)[56]. Nie mają one nic wspólnego z wielkością zbioru danych, na którym wykonywane są operacje,

54 Pomijamy to, ile przestrzeni dyskowej zajmuje system operacyjny i zainstalowane aplikacje, oraz ile używają RAM-u.

55 W takim przypadku są to przeważnie tzw. gamingowe laptopy, czyli maszyny przeznaczone dla osób grających w gry komputerowe, które do płynnego działania rozgrywek wymagają wysokiej mocy obliczeniowej (w tym wydajnej karty graficznej – GPU).

56 Przykładowa oferta komputera stacjonarnego skonfigurowanego do DS to cztery GPU firmy NVIDIA, 128 GB RAM, jeden CPU Intel Xeon W-2135, 1 TB dysk SSD, zainstalowana dystrybucja

a wyłącznie z szybkością trenowania modeli, szczególnie DL. W GPU mogą inwestować także tzw. gracze w Kaggle, którzy uczestniczą w konkursach modelarskich i – o ile granie w Kaggle to ich jedyna aktywność związana z ML – raczej nie są uznawani za uczestników badanego świata, nie podejmują bowiem działania podstawowego w sensie realizowania całego projektu DS.

Lewa strona: „Co byś zrobił/zrobiła z milionem dolarów?" [zdjęcie uśmiechniętego mężczyzny kucającego w pokoju wypełnionym sprzętem komputerowym, przed nim na pierwszym planie rozłożone na podłodze kilkaset kart graficznych]

Prawa strona: „Ale fajny projekt uczenia głębokiego (*deep learning* – DL)! Zdecydowanie powinienem spróbować go zrobić." [szczęk łańcucha] „A, tak." [napis na kuli u nogi: Brak kilku GPU].

Rysunek 4.13. GPU w modelowaniu metodami uczenia głębokiego – memy internetowe

Źródło: https://www.facebook.com/artificialintelligencememes/ (dostęp: 17.05.2019).

Na temat GPU powstają memy internetowe[57] publikowane w mediach społecznościowych, które śledzi część uczestników badanego świata. Przykładowe memy ze strony facebookowej Artificial Intelligence Memes for Artificially Intelligent Teens (dosł. 'Memy o sztucznej inteligencji dla sztucznie inteligentnych nastolatków') przedstawiono na rysunku 4.13.

Linuxa (Ubuntu 18.04 albo CentOS 7.x) oraz narzędzia, m.in. Docker, TensorFlow, PyTorch, Anaconda (Exxact, 2019).

57 Memy pokazywano nam w trakcie wywiadów, używa się ich podczas prezentacji np. na meetupach czy konferencjach. Po tym jak wykazałem się zrozumieniem fragmentów świata DS, prezentując wybrane wynik swoich badań na jednym z meetupów {obserwacja 33},

Przede wszystkim, **kiedy brakuje zasobów sprzętowych czy mocy obliczeniowej, zarówno w odniesieniu do wielkości zbioru danych, jak i trenowania modeli, można jednak skorzystać z innej, mocniejszej maszyny – serwera, lub wielu połączonych ze sobą maszyn, tzw. klastra** (*cluster*).

Serwer to komputer lub oprogramowanie przeznaczone do wykonywania usług lub udostępniania zasobów dla innych komputerów w sieci, tzw. klientów. Słowo „serwer" pochodzi od angielskiego *serve* ('służyć'). Skupiam się na serwerze jako komputerze raczej o większej niż komputer personalny mocy obliczeniowej i zasobach sprzętowych, przystosowanym do nieprzerwanej pracy (bez wyłączenia/restartu przez wiele miesięcy), mogącym posiadać wiele procesorów CPU, zazwyczaj pracującym pod kontrolą systemów operacyjnych z rodziny Linux/Unix. W przypadku pracy lokalnej osoby zajmujące się DS komercyjnie korzystają z serwera przynajmniej w zakresie importu danych z baz danych, tzn. bazy danych są przechowywane na serwerach, a data scientista na personalnym komputerze pisze zapytanie do bazy danych i importuje dane do swojego środowiska pracy (jak Python/R). Nawet w przypadku, gdy zbiór danych mieści się w pamięci RAM komputera personalnego, pracuje się czasem na serwerze. Znacznie może to skrócić czas obliczeń, szczególnie na etapie trenowania modelu ML czy porównywania wielu modeli ze sobą. W poniższym fragmencie rozmówca opowiada o wykorzystywanej infrastrukturze sprzętowej i oprogramowaniu:

> R: […] Czyli pracujemy na laptopie, na tym laptopie mamy SQL Microsoft coś tam Studio Server, czy jakkolwiek tam to się nazywa, eee, w każdym razie korzystamy z tego SQLa microsoftowego, eeem [pauza] no i z eRa[58], a jeżeli ktoś ma ochotę, to każdy ma lokalnie admina, akurat ja mam zainstalowaną też Anacondę czy jakieś tam inne frameworki analityczne w Pythonie, czy w czymkolwiek się naprawdę chce. […] I mamy też dostęp serwera do takiego analitycznego, potężnego [pauza], gdzie mamy 80 rdzeni iii [krótka pauza] jedną kartę graficzną, którą tak jakby na próbę sobie na razie kupiliśmy, yyy, jakąś tam Nividię

---

byłem testowany przez znajomego uczestnika moich badań w rozmowie prywatnej pod kątem tego, czy „też to czuję", tzn. czy śmieszy mnie żart ukuty w jednym z zespołów DS (nie śmieszył, co rozmówca skwitował porównaniem mnie do sieci neuronowej, czyli „jesteś użyteczny bez generalizacji", oznacza to mniej więcej „potrafisz powtórzyć rzeczy, których się nauczyłeś, ale nie rozumiesz naszego świata tak jak my"). Zgadzam się zatem z Jemielniakiem (2019: 132), iż „osoby uprawiające etnografię mawiają, że prawdziwe zrozumienie danej kultury zostaje ostatecznie potwierdzone, jeśli badacz lub badaczka zaczynają rozumieć żarty swoich rozmówców, a więc posiadają zbliżony do nich kapitał kulturowy". Mem ma charakter humorystyczny i satyryczny (por. Jemielniak, 2019: 139–41). Jest przerysowaną, ironiczną formą wypowiedzi, dostosowaną do zasad ekonomii uwagi (por. Harris, 2012) w mediach społecznościowych.

58 Zapisuję nazwę języka R w ten sposób, by oddać stosowaną polską wymowę, jak „w eRrze", „do eRa", „z eRem". Bardzo rzadko zdarza się wymowa anglojęzyczna „aR", „w aRze" (chyba że osoba wypowiada się w całości po angielsku, co nie jest rzadkie). Nazwę Python zapisuję za to anglojęzycznie, ponieważ stosowana jest wymowa anglojęzyczna z polską odmianą, a więc „Pajton", „w Pajtonie", „do Pajtona". Niezwykle rzadko zdarza się wymowa „Pyton" bez anglosaskiego akcentu. Moim zdaniem ma to charakter żartobliwy lub ironiczny.

> […]. Więc też mamy możliwość jakby pracy z takim sprzętem większego kalibru […] po prostu jesteś w stanie zapodać jakiś algorytm optymalizacyjny na noc i on nam sprawdzi milion wersji różnych modeli i to jest bardzo fajne akurat […] na tym serwerze, uyy, jest, postawiony **Debian**, ale wszystkie inne serwery w firmie w zasadzie chodzą na Windowsie, tak samo systemy operacyjne **nasze** są windowsowe, no więc na tym serwerze obliczeniowym to się logujemy przez tam PuTYY, yyy, m, chyba że tam zainstaluję jakiegoś basha na tym, na kompie, ale no raczej jest to po prostu takie logowanie się przez jakieś tam SSH, yyy, no, a reszta to mamy z poziomu Windowsa […] my mamy swój serwer taki, raportowy, gdzie [krótka pauza] powstaje kopia danych produkcyjnych codziennie i na podstawie tego robimy swoje obliczenia, także jest to oddzielone od systemów produkcyjnych, żeby im tam nie muliło. Kiedyś tego nie wiedzieliśmy i w jednym kraju, eem, to był ten sam serwer, tylko tam wirtualnie został podzielony, no i się okazało, że jak my tam zapodaliśmy jakąś analizę, to [krótka pauza] to pracownicy tego kraju nie mogli w ogóle nic zrobić, bo system im nie reagował na klikanie. No ale to się tylko raz zdarzyło nam takie coś. Generalnie po tym przypadku już wszystkie kraje przeszły na **podwójne** serwery. Eee no, i tak sobie właśnie liczymy, no i część z tych, część administracji tymi serwerami analitycznymi jest po naszej stronie, yy, też nasz departament analityczny zajmuje się utrzymaniem hurtowni danych, gdzie w ogóle łączymy te źródła [krótka pauza] yyy, jakby w jedno wielkie źródło, yym, iii, i obliczamy jakby, i w ogóle już z pominięciem tych serwerów raportowych obliczamy to po naszej stronie, głównie te właśnie, yyy, sprawy raportowe, emm, no i my mamy ten, my jako nasz zespół, ten analityczny, to administrujemy tym, tym serwerem, właśnie tym kolosem takim, co ma te GPU i te 80 rdzeni. {wywiad 15}

„Każdy [w zespole DS] ma lokalnie admina" {wywiad 15} oznacza, że na personalnym komputerze może instalować to, co uważa za stosowne, i korzystać np. częściowo z R, częściowo z Pythona. Każdy ma także oprogramowanie firmy Microsoft, umożliwiające import danych z bazy za pomocą zapytania w SQL-u. Dane podzielone są na serwery operacyjne i analityczne (czyli OLTP i OLAP), zbudowane tak, by zespół DS pracował na kopiach danych, nie zaburzając pracy operacyjnej firmy – zwracam uwagę na historię, gdzie „my tam zapodaliśmy jakąś analizę, to pracownicy tego kraju nie mogli w ogóle nic zrobić" {wywiad 15}. W tej firmie osobnym bytem jest hurtownia danych, czyli „jedno wielkie źródło [danych – przyp. R.Ż.]", a jeszcze innym serwer analityczny, „kolos" o 80 rdzeniach procesora CPU i z GPU kupionym „na próbę" {wywiad 15}. Serwer analityczny pracuje pod kontrolą wolnego systemu operacyjnego Debian, będącego dystrybucją Linuxa. Człowiek z laptopem pracującym pod kontrolą systemu Windows, czyli tzw. klient, łączy się z serwerem za pomocą otwartego oprogramowania PuTTY, które emuluje terminal i zapewnia połączenie z użyciem protokołu SSH (*secure shell*, jest to szyfrowany standard komunikacji klient–serwer). Człowiek na laptopie uruchamia PuTTY, widzi interfejs graficzny, wprowadza numer IP serwera, z którym chce się połączyć. Następnie widzi okno terminala, wprowadza login i hasło do serwera. Jeżeli są poprawne, połączenie zostaje ustanowione i można poprzez terminal PuTTY korzystać z serwera, używając komend systemu operacyjnego serwera. Rozmówca mówił także o „instalowaniu basha na kompie" jako alternatywie. Bash to także rodzaj terminala[59] typowego

59 Bash (*The Bourne-Again Shell*) to domyślna tzw. powłoka (*shell*) systemów Linux/Unix, a także język programowania skryptowego, którego się w niej używa. Określenia „terminal",

dla systemów Linux/Unix (w tym macOS, który jest systemem unixowym). Terminal systemu Windows nie jest kompatybilny z bashem, stąd instalowanie na Windowsie takich emulatorów, jak właśnie PuTTY czy inne, do różnych zadań (np. korzystania z kontroli wersji Git).

W powyższym fragmencie zespół DS zajmuje się utrzymaniem infrastruktury do celów analitycznych, a zespół IT infrastruktury operacyjnej. **Osoby odpowiedzialne za infrastrukturę w zespołach DS raczej nie są w badanym świecie uznawane za osoby zajmujące się DS**, ta rola określana jest przeważnie jako inżynier danych (*data engineer*).

Rozpraszanie czy tzw. równoległe operacje na danych wykonywane są na wielu maszynach – a właściwie procesorach lub dyskach – **równocześnie**. Już dekadę temu znacznie taniej można było kupić więcej słabszych komputerów niż jeden o równoważnej mocy: osiem serwerów, każdy z ośmioma CPU i 128 GB RAM było bardziej opłacalne niż jeden serwer z równoważnymi zasobami, czyli 64 procesorami CPU i 1 TB RAM (Jacobs, 2009: 43). Za wierzchołki są uważane pojedyncze maszyny (tzn. serwery/procesory), natomiast krawędzie to połączenia między nimi. Sieć współpracujących maszyn nazywana jest klastrem (*cluster*). Rozproszone może być zarówno składowanie danych, jak i operacje na nich, o ile te operacje nie wymagają komunikacji pomiędzy wierzchołkami (maszynami), czyli mogą być wykonywane niezależnie, równolegle na różnych częściach zbioru danych (Jacobs, 2009: 43). Mając bardzo wiele niezależnych problemów w dużym zbiorze danych, do wykonania obliczeń „potrzebujesz tylko systemu (takiego jak Hadoop lub Spark), który pozwala wysyłać różne zestawy danych do różnych komputerów" (Wickham, Grolemund, 2017).

Rozproszone na wiele maszyn jest także przechowywanie danych. Stosowana jest technologia HDFS. Umożliwia ona horyzontalną skalowalność baz danych, przy czym chodzi o wolumen na poziomie całego internetu (White, 2015). Problemy infrastruktury do przechowywania, a nawet przetwarzania danych (bez wątpienia w przypadku celów innych niż analityczne) nie leżą w obszarze zainteresowania świata DS. Przynależą one do szeroko pojętych ról informatyków zajmujących się big data, ewentualnie do inżynierów danych, których świat może przenikać się ze światem DS.

Spark jest technologią, która realizuje odmienne niż Hadoop MapReduce podejście do paradygmatu MapReduce i wypiera rozwiązanie hadoopowe (w zakresie operacji na danych). W sensie technicznym Spark rozszerza MapReduce o koncept *Resilient Distributed Datasets*, co prowadzi do tego, że Spark może wykonywać wiele rodzajów zadań przetwarzania rozproszonych danych (tworzenia zapytań zbliżonych do SQL-owych, ML i przetwarzania danych w reprezentacji grafowej), do czego wcześniej potrzebne było kilka innych narzędzi. Spark to „zunifikowany silnik do przetwarzania big data" (Zaharia i in., 2016: 56).

„powłoka", „bash", „wiersz poleceń" są używane nierzadko jako synonimy, zasadniczo chodzi o program do kontroli innych programów i systemu operacyjnego za pomocą interfejsu tekstowego, a nie graficznego (GUI).

Hadoop pracuje wyłącznie w trybie wsadowym (*batch*), a Spark zarówno w trybie wsadowym, jak i strumieniowym (*stream*). W trybie wsadowym przetwarzany jest zbiór danych historycznych, np. wszystkie transakcje z dużej platformy aukcji internetowych z jednego dnia, a następnie zwracany jest wynik, np. suma kwot tych transakcji w podziale na kategorie produktów. W trybie strumieniowym dane przetwarzane są w czasie rzeczywistym (*real-time*), np. każdy przelew wykonywany za pomocą bankowości elektronicznej danego banku jest oceniany jako poprawny albo oszustwo (*fraud*). W tym przypadku przetwarzanie całości danych z wybranego okresu byłoby bezsensowne z punktu widzenia bezpieczeństwa w banku. Spark realizuje przetwarzanie strumieniowe w sposób inny niż tradycyjne systemy do tego rodzaju zadań – za pomocą dzielenia danych wejściowych na bardzo małe wsady co 200 milisekund (Zaharia i in., 2016: 60). Właśnie z uwagi na możliwość przetwarzania strumieniowego Spark jest stosowany także do produkcyjnego wdrażania modeli ML.

Spark jest obecnie technologią szeroko stosowaną, szczególnie w dużych firmach. Chiński Tencent wykorzystuje Sparka do przetwarzania około 1 PB[60] danych dziennie, z użyciem klastra złożonego z 8000 wierzchołków, co stanowi największy opublikowany przypadek zastosowania tej technologii (Zaharia i in., 2016: 61). Ze Sparka można korzystać za pomocą języków Scala, Python, R i Java, dla Pythona i R istnieją odpowiednie do tego celu pakiety. Rozproszone ML z użyciem Sparka jest możliwe, ale dostępnych jest 50 typowych algorytmów (Zaharia i in., 2016: 61), co w badanym świecie może być uznawane za ograniczenie.

Spark nie jest w badanym świecie postrzegany jako narzędzie charakterystyczne dla DS, a raczej ma zastosowanie na początkowych etapach projektów (może dochodzić do współpracy data scientistów z data engineerami lub działami IT) oraz na etapach wdrożenia (może zachodzić współpraca data scientistów z machine learning engineerami lub działami IT). Niemniej **Spark, na podstawie ankiet portalu KDnuggets, został uznany za składową najpopularniejszego, sześcioelementowego ekosystemu technologii *open source* do zastosowań w DS i ML**. Ten ekosystem to (Piatetsky, 2018d):

1) Python – język programowania;
2) Anaconda – dystrybucja Pythona do zastosowań DS;
3) scikit-learn – pakiet metod tzw. tradycyjnego ML;
4) TensorFlow (TF) – pakiet stosowany głównie do metod DL;
5) Keras – tzw. nakładka ułatwiająca korzystanie z TF;
6) Apache Spark – zunifikowana technologia do rozproszonego przetwarzania danych.

60 PB, czyli petabajt, to biliard bajtów, czyli 10^{15} B. Dla porównania 1 TB to 10^{12} B, 1 GB to 10^{9} B, 1 MB to 10^{6} B, 1 KB to 10^{3} B. Kilkanaście stron tekstu i kolorowej grafiki (np. artykuł naukowy) w pliku .pdf zajmuje około 1 MB. Tak więc 1 PB to około 10^{9} = 1 000 000 000 takich zapisanych w formacie .pdf artykułów.

Tak zwana chmura obliczeniowa to wiele klastrów, których zasoby i moc obliczeniową można wypożyczyć. Niemniej nie zachodzi tutaj praktyka współdzielenia czy dzielenia się posiadanymi zasobami przez względnie równorzędnych w sensie gospodarczym partnerów – **w toku badań nie spotkałem się w świecie DS z korzystaniem z chmur dostawców innych niż największe firmy technologiczne: Microsoft Azure, Amazon Web Services (AWS) i Google Cloud Platform (GCP)**. Według raportu Fundacji Digital Poland z obliczeń w chmurze (wymieniono wyłącznie Azure, AWS, GCP) korzystało 76% badanych firm (pytanie wielokrotnego wyboru), 67% firm używało własnych komputerów personalnych, 36% własnych serwerów, 33% wykonywało obliczenia na hardwarze zapewnionym przez swoich klientów, 20% wskazało na wynajmowanie wyspecjalizowanego (*dedicated*) hardware'u, a 3% na jeszcze inny sprzęt komputerowy (Borowiecki, Mieczkowski, 2019: 43). Choć za ideą korzystania z chmury obliczeniowej stoi zdanie „nie potrzebuję mieć wiertarki, a tylko dziurę w ścianie", to **nie jest to współdzielenie**, jak wspólny przejazd prywatnym samochodem kierowcy, umówiony przez BlaBlaCar (*hitchhiking*/*carpooling* – kierowca jedzie własnym samochodem we własnej sprawie, a jeżeli inne osoby nie dołączą, to przejazd odbędzie się z pustymi miejscami dla pasażerów), **a wypożyczenie czy wynajęcie na żądanie** (*on-demand* – kurs odbywa się tylko, jeżeli jest zamówiony, jak w przypadku taksówek czy usług firm Uber, Lyft) (Frenken, Schor, 2017: 5).

Za pomocą chmury na żądanie dostępne są zasoby i moc obliczeniowa dużych klastrów komputerowych. By skorzystać z chmury po raz pierwszy, konieczne jest – oprócz utworzenia konta użytkownika, akceptacji regulaminów, dodania numeru karty kredytowej – skonfigurowanie tzw. **maszyny wirtualnej**. Od jej mocy i konfiguracji zależeć będzie koszt korzystania z chmury. Maszyna wirtualna to wynajęta „część" chmury. Użytkownik konfiguruje taki zdalny, wirtualny komputer, wybierając jego procesory, rodzaj i wielkość dysku, rodzaj i wielkość pamięci RAM, system operacyjny. Maszyna wirtualna wykorzystywana w praktyce może mieć dużą moc. Na przykładzie GCP może być to np. 80 rdzeni CPU architektury Intel Broadwell i 1,88 TB RAM (konfiguracja o nazwie n1-ultramem-80), jednak do celów dydaktycznych konfiguruje się małe, tanie maszyny, np. 2 rdzenie CPU Intel Skylake i 7,5 GB RAM (konfiguracja n1-standard-2) oraz 10 GB przestrzeni dyskowej i praca pod kontrolą systemu operacyjnego Ubuntu 16.04 LTS {obserwacja 35}. Konfiguracja n1-ultramem-80 to koszt około 8,826 USD za godzinę pracy, a w przypadku n1-standard-2 ten sam czas kosztować będzie 0,067 USD[61]. Mówię o jednej maszynie wirtualnej, zwanej pojedynczą instancją. Będzie ona raczej platformą obliczeniową do trenowania modeli ML, ewentualnie do produkcyjnego wdrożenia modelu. Możliwe jest także skonfigurowanie wielu instancji, identycznych lub różnych, i połączenie ich w klaster, a następnie używanie klastra

61 Przy tej samej przestrzeni dyskowej 10 GB i systemie Ubuntu 16.04 LTS. Stan na 28 sierpnia 2019 roku, informacja pochodzi z mojego prywatnego konta Google Cloud Platform, założonego podczas badań własnych {obserwacja 35}.

do obliczeń lub przechowywania danych w systemie rozproszonym/równoległym. Taki klaster raczej będzie służyć jako infrastruktura dla baz danych.

Podsumowując, przedstawiam fragment notatki z obserwacji uczestniczącej przeprowadzonej na warsztatach online. Prowadzący wyjaśnił krótko, czym jest chmura i dlaczego się z niej korzysta, oraz przeprowadził uczestników przez proces założenia konta i konfiguracji maszyny wirtualnej na GCP:

> […] o chmurze – pewne rzeczy nie zmieszczą się na naszym żelazie (tak na hardware mówią geekowie), więc trzeba wziąć większy komputer […] chmura to cudzy komputer i przemiana wykonała się na naszych oczach, stać teraz na niego nie tylko duże instytucje, ale praktycznie każdego i możesz mieć np. 1–2 TB RAM-u tylko wtedy, kiedy tego potrzebujesz. […] to kosztuje tak mało, że nawet student, decydując, czy zje dziś obiadek, czy wynajmie komputerek na dwie godziny, może sobie na to pozwolić – [prowadzący mówi – przyp. R.Ż.] „myślę, że decyzja tu będzie zrozumiała" […] 3 największe [chmury to – przyp. R.Ż.]: Google, Amazon, Microsoft Azure […] [prowadzący – przyp. R.Ż.] podkreśla, że nie pracuje w Google (ale byłoby mu miło), sam go używa, jest rok za darmo, można się pouczyć […] „polecam ci to, co najlepsze, wiem, jak wiesz coś lepszego, to daj mi znać" […] [by skorzystać z GCP – przyp. R.Ż.] trzeba się zarejestrować, klikać OK i akceptować regulaminy […] konfigurujemy maszynę w chmurze, „zaczyna się robić ciekawie", […] wybieramy lokalizację, [prowadzący – przyp. R.Ż.] mówi, że jak masz biznes w UE, to nie opłaca się i nie jest legalne wybieranie serwera w USA z uwagi na to, że dane są wtedy poza UE, ale dla uczenia się samodzielnie tańszy będzie ten ze Stanów i ten bierzemy […] na chmurze działamy tak – jak potrzebujemy, to włączamy, jak nie potrzebujemy, to wyłączamy. {notatka terenowa, obserwacja 35}

Dostęp do mocy obliczeniowej na żądanie jest uznawany za tak tani, że „nawet student […] może sobie na to pozwolić" {obserwacja 35}. Ponadto GCP dla nowych użytkowników zapewnia bezpłatne korzystanie z usług do wartości 300 USD przez rok. W formie memów i anegdot powtarzane są żarty typu „budzę się w nocy i boję się, że zapominałem wyłączyć maszynę na AWSie" {np. obserwacja 40}. Opłaty pobierane są za czas włączenia instancji, bez względu na to, czy maszyna wykonuje obliczenia. Włączenie i wyłączenie wykonywane jest ręcznie.

„Chmura to cudzy komputer" {obserwacja 35} to tłumaczenie powiedzenia *there is no cloud, it's just someone else's computer* (dosł. 'nie ma chmury, jest tylko cudzy komputer'). Posługiwanie się określeniem „chmura" może być w DS i w IT postrzegane jako wskaźnik bycia tzw. osobą nietechniczną. To określenie nie jest aż tak jednoznacznie kojarzone przez środowiska techniczne z nietechnicznym dyskursem jak termin „big data". Niemniej jestem przekonany, że przynajmniej przez część uczestników świata DS termin „chmura" jest kojarzony z komunikatami marketingowymi kierowanymi z badanego świata do klientów, kandydatów do pracy lub ewentualnie wannabesów. Osoby zajmujące się DS w komunikacji między sobą posługują się nazwami konkretnych usług w chmurze, np. „to było postawione na SageMakerze AWSowym". Słowo „postawione" oznacza zrobione, wdrożone, działające.

Lisa Portmess i Sara Tower (2015) zauważyły, że chmura jest jedną z retorycznych metafor związanych z tzw. big data. Chmura nie jest bezcielesna, jest cudzym, dużym komputerem. Metafora chmury kierowana jest do klientów i ma wywołać wrażenie, że chodzi o coś niesamowitego, niematerialnego, nieuchwytnego i nadzwyczajnego. Naprawdę mowa o halach z rzędami serwerów, gigantycznymi systemami chłodzącymi, licznymi zabezpieczeniami zapewniającymi energię elektryczną, strzeżonymi przez pracowników ochrony[62]. **Wrażenie braku materialności tzw. chmury wywołuje** (chyba u osób korzystających z chmury) **poczucie wyzwolenia także z ciężaru moralnego** – jak rozumiem związanego z tym, jakie dane i do jakich celów są w tej chmurze przetwarzane (Portmess, Tower, 2015: 6).

Portmess i Tower (2015: 6) zwracają także uwagę na **problem wykorzystywania zasobów naturalnych** – prądu, wody, przestrzeni fizycznej – przez to, co nazywane jest chmurą, a co one określają ogólnie centrami danych, materialnym wymiarem big data. O problemie tym pisał także AI Now Institute (2018: 6–7, 33–34). Wskazano koszty, jakie ponosi środowisko naturalne z tytułu istnienia centrów danych. Ma to stanowić, obok problemu pracy rzeszy niewykwalifikowanych osób (tzw. klikaczy – *clickworkers*) z krajów postkolonialnych i rozwijających się, wgląd w pełen łańcuch dostaw (*full stack supply chain*) systemów bazujących na AI (AI Now Institute, 2018).

Nie spotkałem się w badanym świecie z poruszaniem problematyki wykorzystywania zasobów naturalnych przez, ogólnie rzecz biorąc, infrastrukturę obliczeniową. Uczestnicy świata DS raczej nie zwracają uwagi na materialne aspekty infrastruktury obliczeniowej (poza osobami samodzielnie rozbudowującymi komputery, np. o wysokiej klasy GPU), nie należy to bowiem do ich obowiązków służbowych, raczej nie mają fizycznego kontaktu z materialnością tej infrastruktury (tzn. dotykają głównie klawiatury komputerowej, nie podłączają kabli w serwerowniach) i właściwie całość swoich zadań wykonują w środowisku cyfrowym. Z zagadnieniami kosztów energii elektrycznej mogą stykać się data scientiści na stanowiskach kierowniczych bądź właściciele firm, choć przeważnie będą oni zwracać uwagę na koszt użycia usług chmurowych, w sensie ceny za godzinę, przy uwzględnieniu specyfiki danego projektu. Z krytyki wobec zużycia zasobów naturalnych przez centra danych zdają sobie sprawę dostawcy usług chmurowych, jak Google, które chwali się swą wysoką oceną w raporcie Greenpeace'u (Cook i in., 2017: 42).

62 Zachęcam do obejrzenia przygotowanego przez Google filmu reklamowego w technologii 360°, prezentującego centrum danych GCP w miejscowości The Dalles w Oregonie – *Google Data Center 360° Tour* (b.d.).

4.3.4. Komunikacja lub współpraca

W niniejszym podrozdziale omawiam narzędzia stosowane w badanym świecie do komunikacji lub współpracy pomiędzy jego uczestnikami. Część ma zastosowanie także w komunikacji z osobami nienależącymi do tego świata. Są to:

1) systemy kontroli wersji, głównie Git;
2) repozytoria kodu bazujące na kontroli wersji – np. serwisy GitHub, BitBucket;
3) narzędzie do pracy grupowej Slack;
4) poczta elektroniczna oraz komunikatory tekstowe i głosowe, np. Skype, Google Hangouts/Meet, FaceTime, MS Teams;
5) media społecznościowe, każde ma odrębną specyfikę: Facebook, LinkedIn, Twitter;
6) portal meetup.com;
7) strony internetowe, np. KDnuggets, Towards Data Science (na medium.com);
8) blogi, szczególnie na medium.com;
9) fora internetowe, ze wskazaniem na StackExchange i jego część Stack Overflow;
10) repozytorium preprintów artykułów naukowych Arxiv.

Spośród wszystkich wymienionych wyżej narzędzi **rozmówczynie/rozmówcy wskazywały/wskazywali tylko na Git**. **Pozostałe nie są traktowane jako narzędzia osoby zajmującej się DS.** Są one powszechnie używane, a sposoby ich użytkowania to niezbędny element do opisania zachodzących w tym świecie społecznym procesów i wartości tego świata.

Osoby zajmujące się DS pracują raczej bez ściśle określonych godzin, mogą działać zdalnie, a współpraca z osobami z innych miast czy międzynarodowa nie należy do rzadkości. Stąd narzędzia do komunikacji i współpracy na odległość są przez wiele osób wykorzystywane codziennie. W artykule będącym aktem samowiedzy osób zajmujących się DS w firmie IBM podkreślano, że „data science to sport zespołowy" i praca w DS polega nie tylko na współpracy osób technicznych – za takie uważają się data scientiści – ale także technicznych z nietechnicznymi, np. menadżerami, ekspertami dziedzinowymi (Zhang i in., 2020: 2–4). W moich badaniach obecna była także kwestia komunikacji z klientami, zarówno nielubiana i unikana, jak i uznawana za bardzo ważną.

Działanie podstawowe może być realizowane przez tę samą osobę w ramach sfery komercyjnej, akademickiej, hobbystycznej – równolegle lub jednocześnie, możliwe są także wszystkie inne połączenia tych sfer działalności w ramach działania jednej osoby. Za uczestnika świata DS (ale raczej nie za data scientistę) uznana będzie zatem osoba, która jest zatrudniona na uczelni wyższej jako asystent, przygotowuje doktorat z biologii w obszarze genetyki roślin, korzystając z języków R, Python i zaawansowanych metod statystycznych, była uczestnikiem konferencji PyData w Warszawie, a w czasie wolnym współorganizuje meetup DS w miejscu zamieszkania oraz z poznanymi na konferencji ludźmi przygotowuje

projekt z zastosowaniem DL, dotyczący rozpoznawania kwiatów na fotografiach (który chce przedstawić na meetupie, na kolejnej konferencji PyData, ewentualnie w przyszłości zmonetyzować, czyli zamienić w produkt komercyjny, np. aplikację na urządzenia mobilne, gdzie użytkownik mógłby rozpoznawać żywe kwiaty za pomocą kamery w smartfonie). Ta hipotetyczna osoba będzie korzystać np. z systemu Git i portalu GitHub zarówno w swoim projekcie badawczym – będzie śledzić zmiany, które sama wprowadziła, prezentować kod i wyniki promotorom – jak i w projekcie hobbystycznym – będzie śledzić zmiany swoje i zespołu, wskazywać na błędy czy proponować poprawki do czyjegoś kodu. Na GitHubie będzie także szukać inspirujących repozytoriów innych ludzi. Będzie przeszukiwać Stack Overflow, kiedy jej kod, pisany w dowolnym celu, nie działa zgodnie z oczekiwaniami; czasami zakładać nowy wątek z prośbą o poradę, rzadziej odpowiadać na takie prośby innych użytkowników. Będzie korzystać ze Slacka, by rozmawiać ze swoim zespołem hobbystów, planować pracę, udostępniać sobie nawzajem różne treści. Inny *workplace* (dosł. 'miejsce pracy') na Slacku będzie służyć tym samym celom w działaniu zespołu organizującego meetup. Będzie korzystać z komunikatorów tekstowych i wideo do rozmów z pojedynczymi osobami. Na Facebooku osoba ta będzie uczestniczyć np. w grupie Data Science PL i śledzić strony z memami poświęconymi DS, na LinkedIn np. zamieści certyfikat ukończenia kursu MOOC lub informację o kolejnym spotkaniu swojego meetupu, na Twitterze będzie śledzić konta osób uznawanych za autorytety w dziedzinie DL, a w mediach społecznościowych wpisy z wybranych stron i blogów. Arxiv i inne repozytoria artykułów naukowych będą dla niej pomocne zarówno w szukaniu materiałów do pracy doktorskiej, jak i najbardziej skutecznej metody optymalizacji hiperparametrów w sieci neuronowej, tworzonej w projekcie hobbystycznym.

Narzędzia do komunikacji i współpracy są niezbędne w pracy komercyjnej, zarówno do kontaktu z klientem w ramach zespołu DS, jak i działu/firmy DS. Osoby zajmujące się DS komercyjnie pracują w małych zespołach, tworzonych *ad hoc* na potrzeby projektów. W takim np. trzyosobowym zespole przełożony może wyznaczyć kogoś do roli lidera projektu. Bywa tak, że pojedyncze osoby zajmują się projektem dla konkretnego klienta, są odpowiedzialne za całość jego realizacji. **Data scientiści najwięcej czasu spędzają na pracy indywidualnej przy komputerze.** Rzadziej komunikują się z członkami zespołu projektowego, dominuje komunikacja drogą elektroniczną. Jeszcze mniej czasu zajmuje komunikacja z całym działem/firmą (chodzi o małą, kilku-, kilkunastoosobową firmę zajmującą się tylko DS, tzw. start-up), co raczej odbywa się w formie krótkich spotkań (np. codziennie o godzinie 10.00 dział/firma spotyka się i każdy pracownik ma dwie minuty na wypowiedź: co robi, co zostało wykonane, a co nie, wyniki eksperymentów, ewentualne problemy techniczne/metodologiczne/organizacyjne, następne kroki). Data scientiści na takich spotkaniach konsultują się wzajemnie, co bywa nazywane „dzieleniem się wiedzą" czy „dawaniem inspiracji". Wzajemne konsultowanie odbywa się także w sytuacjach nieformalnych, jak wyjścia na kawę lub posiłek.

Część data scientistów komunikuje się z klientami, dla których realizują właśnie projekt, część nie. Drugi scenariusz występuje raczej w większych organizacjach, gdzie istnieje dział DS, lub w bardziej dojrzałych start-upach. W działach/firmach istnieją względnie stabilne, zdefiniowane role odpowiedzialne za kontakt, a czasami za pozyskanie klienta – może być to kierownik działu/zespołu, menadżer projektów, osoba na stanowisku sprzedażowym. Inny scenariusz zakłada, że z klientami pracuje, a także pozyskuje ich właściciel małej firmy. Czasami kontaktu z klientami nie mają osoby z małym doświadczeniem zawodowym w DS, na tzw. stanowiskach juniorskich, a już stanowiska średniego doświadczenia (zwane np. *regular* lub *specialist*) i wysokiego doświadczenia (tzw. *senior*) pracują z klientami. Komunikacja z klientami ma różne formy: e-mail, telefon (zaplanowane rozmowy to tzw. calle, od słowa *call*), komunikatory wideo, spotkania.

Samodzielność jest kluczową kategorią posługiwania się forum przeznaczonym do zadawania pytań i udzielania odpowiedzi związanych z kodowaniem – Stack Overflow. Nie można zatem uznać jej za charakterystyczną tylko dla DS, a dla szerokiej społeczności osób piszących kod w ogóle. Najpierw należy poprawnie przygotować pytanie (m.in. podać stosowane wersje narzędzi, system operacyjny komputera), najlepiej dołączając działający fragment kodu lub kodu i danych wraz z wyjaśnieniem, jaki jest rezultat osiągany, a jaki oczekiwany (Skeet, 2010). Istnieją pakiety wspomagające przygotowanie takich działających, powtarzalnych przykładów (*reproducible example*) na potrzeby zamieszczania na Stack Overflow, zgłaszania tzw. *issue* na GitHubie i komunikacji na Slacku (Bryan i in., 2019). W pytaniach na Stack Overflow nie należy używać zwrotów grzecznościowych, gdyż może to być uznawane za rozpraszające i niepotrzebnie zajmujące miejsce, a więc za stratę czasu – a to wbrew zasadzie efektywności. Zdecydowanie nieakceptowane są pytania na zasadzie „jak zrobić X od zera?" – akceptowana forma to „próbowałem A (działający przykład), które ma dać mi X (opis), niestety, uzyskuję Y (działający przykład), jak osiągnąć X?". Warto zauważyć przy okazji to, co jest przedstawiane jako kolejna zaleta pisania kodu do operacji na danych w porównaniu do używania GUI (czyli klikania) – kod to po prostu tekst, więc można np. skopiować go i wkleić, wyszukiwać go w internecie, można go wysłać e-mailem, czyli ogólnie łatwiej skomunikować się z innymi ludźmi w kwestii tego, co dokładnie chce się zrobić i szybciej uzyskać pomoc w rozwiązaniu swojego problemu (Wickham, 2018d).

Systemy kontroli wersji, jak najpopularniejszy Git, zostały stworzone po to, by wspomóc pracę zespołów programistów nad dużymi projektami budowy oprogramowania. Git lub inny system służy do zarządzania całym zbiorem plików cyfrowych, które składają się na projekt programistyczny, i śledzenia ich ewolucji. Można to porównać do funkcji śledzenia zmian, znanej z edytorów tekstu, jak Microsoft Word. Jednak kontrola wersji ma znacznie bardziej rozbudowane możliwości, śledzenie zmian jest bardziej rygorystyczne i może odnosić się do więcej niż jednego pliku. Kontrola wersji, a szczególnie Git, została zaadaptowana przez

osoby zajmujące się DS do ich specyfiki pracy. Jest zatem używana do zarządzania zbiorem plików, które zazwyczaj składają się na projekt DS: pliki z oczyszczonymi danymi, wykresy (pliki obrazów), raporty (dokumenty tekstowe) i kod (Jennifer Bryan, 2018: 2–3). Sam Git jest oprogramowaniem na wolnej licencji GNU GPL, dostępnym od 2005 roku. Inne, starsze, a obecnie mniej popularne rozwiązania to SVN i CVS, na tej samej licencji (Spinellis, 2005: 4).

Projekt DS zlokalizowany w konkretnym folderze na dysku komputera to, w terminologii kontroli wersji, repozytorium[63] (*repository*, w skrócie *repo*). Pod kontrolą jest właśnie takie *repo* i wszystkie zlokalizowane w nim pliki. Po wykonaniu zmiany, np. napisaniu kolejnych linii kodu, człowiek zapisuje plik z kodem, tzw. skrypt. Następnie wykonuje *commit* (dosł. 'zatwierdzenie', nie stosuje się polskiego tłumaczenia), tworząc odpowiednik „fotografii" repozytorium w konkretnym punkcie czasu. Zestaw różnic w zmienionym pliku nazywany jest *diff*. Ta funkcjonalność umożliwia dokładne prześledzenie zmian w każdym pliku i to nie tylko pomiędzy następującymi po sobie commitami, ale i pomiędzy dwoma dowolnymi (Jennifer Bryan, 2018: 10). To podejście pozwala zachować kontrolę nad kolejnymi wersjami plików nawet pojedynczej osobie, której łatwo zagubić się w nieformalnej strategii bazującej na zmianach nazw plików, jak np. *art1_final.doc*, *art1_final_poprawki.doc*, *art1_popr_2* itd. Sprawa staje się bardziej skomplikowana przy współpracy zespołu, który np. przesyła sobie kolejne wersje kolejnych części tekstu za pomocą e-maili. W kontroli wersji Git każdy commit musi mieć wiadomość, opisującą co zostało wykonane, istnieje możliwość dodania tagu do wersji pliku, którą chce się wyróżnić (Jennifer Bryan, 2018: 3–4, 11–12). Git nie jest przeznaczony do pracy z dużymi plikami binarnymi, tzn. z plikami w formatach biurowych, jak .pdf, .xlsx, .docx i odradza się korzystanie z nich (Jennifer Bryan, 2018: 15–16).

Zaletą Gita jest to, że jest on zdecentralizowaną kontrolą wersji. Każdy ze współpracowników ma pełną kopię repozytorium i historii jego zmian. Mogą oni pracować jednocześnie lub nie, a zmiany przez nich wprowadzone można prześledzić, a także automatycznie złączyć w całość (*merge*) i kontrolować kłopoty ze złączeniem (*merge conflict*). Istnieje możliwość tworzenia *branch* (dosł. 'odgałęzienie', nie stosuje się polskiego tłumaczenia), czyli części projektu, które pozostają w repozytorium, ale czasowo są rozwijane oddzielnie, a później łączone z gałęzią główną (Jennifer Bryan, 2018: 16–18, 20).

GitHub jest narzędziem zarówno ułatwiającym pracę grupową, repozytorium na GitHubie jest bowiem centralnym w – z założenia zdecentralizowanym – systemie Git (Jennifer Bryan, 2018: 17) i np. umożliwia ściągnięcie najnowszej wersji repozytorium wraz z całą historią, w przypadku dołączenia do zespołu nowej osoby czy awarii na czyimś komputerze, jak i **narzędziem służącym do upubliczniania swojej pracy**. Serwis GitHub, obecnie należący do Microsoftu, umożliwia darmowe tworzenie publicznych repozytoriów, repozytoria prywatne

63 Warto przypomnieć, że także cały GitHub nazywa się repozytorium, podobnie mówi się o repozytorium pakietów, jak np. CRAN.

są płatne. GitHub to zatem serwis internetowy zapewniający hosting dla repozytoriów Git. Można porównać go do bardziej zaawansowanej i sprzężonej z Gitem wersji tzw. dysków w chmurze, jak Dropbox czy Google Drive. Na GitHubie przechowywana jest zsynchronizowana kopia repozytorium lokalnego.

Dla użytkowników GitHuba dostępnych jest wiele mechanizmów zbliżonych do funkcjonowania mediów społecznościowych. Można np. wyróżniać repozytoria gwiazdką (*star*), a gwiazdki są uznawane za wyraz uznania dla pracy ich autorów. Można kopiować całość cudzego repozytorium i pracować nad nim niezależnie od twórców, przy czym informacja o skopiowaniu (*fork*) jest zachowywana. Powiązane jest to z funkcją *pull request*, czyli mechanizmem proponowania zmian do repozytorium. Z funkcji tej korzysta się za pomocą utworzenia brancha albo za pomocą forka. Ten drugi mechanizm – człowiek znajduje publiczne repozytorium i jest nim zainteresowany, następnie kopiuje je w całości z uznaniem autorstwa (*fork*), potem wprowadza zmiany i proponuje ich dodanie do oryginalnego repozytorium (*pull request*) – jest akceptowanym w środowisku *open source* sposobem włączania się nowych osób w rozwój otwartego oprogramowania, w tym np. wielu pakietów w R (Jennifer Bryan, 2018: 20).

GitHub jest traktowany jako część publicznego portfolio osób piszących kod, nie tylko zajmujących się DS. Publicznie dostępne są np. wykresy obrazujące wkład (*contributions*) w czasie każdego użytkownika GitHuba. Repozytoria są przeglądane przez osoby zajmujące się rekrutacją do pracy bądź poszukujące osób do współpracy. Użytkownicy GitHuba na podstawie repozytoriów mogą tworzyć strony WWW lub pokazy slajdów w przeglądarce internetowej za pomocą usługi GitHub Pages. Istnieją pakiety umożliwiające opublikowanie takiej prostej strony[64] WWW napisanej w RStudio (Kross, 2021). GitHub, nawet bez użycia GitHub Pages, w przyjazny dla oka sposób wyświetla pliki sformatowane w języku znaczników Markdown. Taki dokument przygotowany w RStudio można upublicznić na GitHubie, będzie czytany za pomocą przeglądarki internetowej jak zwykła strona WWW (Jennifer Bryan, 2018: 12–14).

Byłem ciekaw, jak uczestnicy świata DS oceniają przejęcie GitHuba przez korporację, jaką jest Microsoft, i czy nie kłóci się to z ideami *open source*. Jeden z rozmówców wskazywał, że przejęcie repozytoriom przez bogatą firmę przyczynia się do utrzymania jego najbardziej pożądanej funkcji, czyli dzielenia się otwartym kodem. Użytkownik GitHuba nie jest – jak użytkownik Facebooka czy Google'a – produktem sprzedawanym reklamodawcom. Jego dane są raczej bezpieczne, może kupić pakiet obejmujący m.in. prywatne repozytoria (w wersji podstawowej można tworzyć wyłącznie publiczne). Opieka finansowa Microsoftu może pomóc w zachowaniu niekomercyjnego, przyjaznego użytkownikom charakteru GitHuba. Według rozmówcy przykładem platformy, która zapewniała podobne funkcje, ale „zeszła na psy", jest SourceForge:

64 Utworzyłem taką stronę – https://zremek.github.io/.

> R: [rozmowa dotyczy przejęcia GitHuba przez Microsoft – przyp. R.Ż.] swoją drogą to też nie jest dla mnie jasne, że GitHub sam też byłby dobry. Byśmy na przykład stwierdzili, że im się kończy kasa, jakieś reklamy w ten czy inny sposób dają, albo twoje dane, w ten czy inny sposób. Było takie, najczęściej kiedyś stosowana platforma open source'owa Source SourceForge, yy. Było takie pierwsze, wiesz, wielkie, wielka sieć open source w ogóle, to wiesz. Zeszło na psy w ogóle. Totalnie na psy w sensie, że coraz więcej reklam, coraz więcej takich rzeczy, coraz więcej jakichś projektów z **wirusami**.
> *B: Na czymś próbują zarobić.*
> R: **Z wirusami!** Które po prostu instalują ci jakieś reklamy. Rozumiesz.
> *B: Masakra.*
> R: **Masakra**. Rozumiesz, już takie rzeczy totalnie shady[65]. […] GitHub jest **pięknym** miejscem i w ogóle takie miejsca i w ogóle takie miejsca to wiesz, ostatnim takim miejscem, które to były, to biblioteki Aleksandrii jak GitHub. {wywiad 26}

Z korzystaniem z publicznych repozytoriów, jak GitHub i podobny BitBucket, spotykałem się podczas badań na meetupach, warsztatach, w wielu prezentacjach konferencyjnych. Konfiguracja Gita i GitHuba jest jedną z pierwszych lekcji w MOOC, poświęconych narzędziom do DS {obserwacja 8}. Są to narzędzia zaadaptowane do DS z zespołów programistycznych i choć uznawane za przydatne, to raczej nie spełniają swojej funkcji, jeżeli chodzi o zadania charakterystyczne dla DS – kontrolowania wersji modeli i zbiorów danych. **Ponieważ projekty DS mają charakter iteracyjny, w wyniku eksperymentowania powstaje wiele wersji modeli i zbiorów danych.** Jedna z rozmówczyń tak mówiła o testowaniu w swoim zespole DS narzędzia, które ma służyć właśnie do kontroli wersji modeli:

> R: […] **w tej chwili**, yy, w zespole mamy tak zbudowaną infrastrukturę, że **do** samego jako repozytorium samego kodu używamy GitHuba, aa jako, jeśli chodzi o kontrolę wersji, to jeszcze jesteśmy w trakcieeeee **wybierania** narzędzia. Bo było, tak naprawdę, że tak powiem, szef nam wskazał narzędzie, ale też open source'owe, które testowaliśmy, o nazwie Acumos. Yyy, to jest takie narzędzie, które ma służyć jako marketplace dla modeli, ma też właśnie tam trzymać wersjonowanie, ma tam trzymać dokumentację, problem w tym, że to jest totalnie świeże narzędzie i tam połowa rzeczy nie działała. [śmiech] Eee, mieliśmy się z nim zapoznać, mieliśmy je potestować, a efekt jest taki, że, no, wygląda to na papierze pięknie, bardzo fajny pomysł, bo nie ma w ogóle takiego rozwiązania innego na rynku, open source'owego, ale po prostu nie działa, ale może za rok albo za dwa zacznie, albo jak wprowadzą jakąś dobrą wersję, to może ta wersja będzie działała [śmiech] {wywiad 18}

Acumos to platforma *open source*, która ma nie tylko wspomagać kontrolowanie wersji modeli, ale także ich wdrażanie. **Data scientiści projektują i testują modele w pewnego rodzaju laboratorium, odrębnym od środowiska operacyjnego/produkcyjnego firmy.** To laboratorium może być laptopem z zestawem narzędzi do DS. Acumos pozwala „zamrozić" parametry danej wersji wytrenowanego modelu i sklonować całe środowisko pracy do kontenera, który później można uruchomić. Data scientiści mają swobodę w budowaniu i trenowaniu modeli

65 Dosł. 'ciemne, szemrane interesy'.

za pomocą ulubionych narzędzi. Kontener z gotowym modelem i środowiskiem, w którym model powinien działać jak w laboratorium, jest zamieszczany w tzw. *marketplace*, gdzie mogą go użyć inne zespoły firmy, np. programiści zajmujący się wdrożeniem na produkcję. Kontener z modelem może być zatem używany przez osoby niemające nic wspólnego z ML, ponieważ jest działającą czarną skrzynką – dostaje dane na wejściu i zwraca rezultat. Jest to rozwiązanie zapewniające bardziej powtarzalne działanie modelu niż tylko skorzystanie z jego zapisu w pliku .h5, ponieważ w omawianym rozwiązaniu kontener zawiera także odpowiednie środowisko do działania modelu. Acumos oferuje także izolację na poziomie modelu. W pewnych sytuacjach data scientiści chcą trenować wiele modeli do różnych zadań na jednym zestawie danych. Dzięki tej izolacji różne zespoły mogą pracować nad tym niezależnie (Zhao i in., 2018).

Innym rozwiązaniem, z którym spotkałem się na jednej z konferencji {obserwacja 26}, jest DVC (*data science version control*). To także narzędzie *open source*, bazujące na Git i chmurowych usługach przechowywania danych. Git służy w nim standardowo do kontroli kodu (a właściwie plików z kodem – skryptów), a połączenie z chmurą zapewnia kontrolę „laboratoryjnych" zbiorów danych i plików z zapisem wytrenowanego modelu (np. typu .h5). Trzeci komponent DVC to kontrola *pipeline* (dosł. 'rurociąg', nie stosuje się polskiego tłumaczenia), czyli całego iteracyjnego procesu, na jaki składa się budowa kolejnych wersji modelu. W historii kolejnych commitów zachowywane są metryki z ewaluacji modelu, określające np. jego dopasowanie do danych (Petrov, 2017; Vazquez, 2017). *Data science version control* to zatem narzędzie lepiej niż sam Git i GitHub dostosowane do specyfiki projektów DS (Vazquez, 2017). To, że DS polega w dużej mierze na eksperymentach, najczęściej nieudanych, było podkreślane przez rozmówczynie/rozmówców wielokrotnie, nierzadko w porównaniu do pracy programistów.

O ile Git/GitHub i podobne rozwiązania są w badanym świecie powszechnie używane, a ich znajomość jest uznawana za elementarz umiejętności data scientistów, o tyle narzędzia takie jak np. DVC używane są bardzo rzadko. Zespoły DS stosują chałupnicze metody kontrolowania projektów i wyników eksperymentów. Mogą to być np. notatki w arkuszach kalkulacyjnych, wymiana e-maili czy wiadomości na Slacku lub innych komunikatorach, a w przypadku osób pracujących w komercyjnych zespołach DS codzienne, krótkie spotkania z wymianą informacji. W zależności od firmy zespoły mogą pracować tzw. metodykami zwinnymi, zbliżonymi do Scurm czy Agile, mogą mówić o inspirowaniu się podobnymi metodykami lub odcinać się od nich, uznając je za schematy odpowiednie do pracy zespołów programistycznych, a nie DS. Zdarzają się żarty z takich metodyk:

> Co jeszcze trzeba wiedzieć: Scrum, [osoba notuje w Google Docs, widać na projektorze – przy. R.Ż.] mówi „o napisałem scum[66]" – śmiech na sali. {notatka terenowa, obserwacja 38}

66 *Scum* dosł. 'szumowina', potocznie 'męt', 'ścierwo', 'syf' (nie jest to słowo wulgarne, ale poniżające).

W małych firmach, nazywanych start-upami, system organizacji pracy w ogóle – nie tylko kontrolowanie pojedynczego projektu i wyników eksperymentów w jego ramach – jest znacznie mniej sformalizowany i ustrukturyzowany niż w dużych firmach, tzw. korporacjach. Widać to np. w narracji rozmówczyni {wywiad 10}, która pracuje w start-upie DS i tę firmę uważa za „swoją", jednak w momencie wywiadu realizuje projekt dla korporacji w ten sposób, że fizycznie pracuje właśnie w korporacji, jest tymczasowym członkiem zespołu projektowego w firmie-kliencie swojego start-upu. Korporacji nie uważa ona za „swoją", a także kwestionuje to, czy obecny projekt może uznać za DS. Lowrie (2017; 2018) zaproponował postrzeganie DS jako dziedziny nierozerwalnie łączącej inżynierię i naukę. Jest to dość trafne, ale z moich badań wynika, że **uczestnicy świata DS uważają siebie bardziej za poszukujących badaczy, eksperymentatorów niż inżynierów**.

Nawet w korporacjach narzędzia typu DVC są używane rzadko, a jeśli już, to raczej we wdrożeniu lub tzw. utrzymaniu modeli „na produkcji". Utrzymaniem określa się serwis i naprawy wdrożonego modelu. Osoby zajmujące się takimi wdrożeniami i utrzymaniem nazywane są np. *machine learning engineer*, *DataOps*, *DevOps* dla DS.

Slack jest narzędziem do pracy grupowej – pojedynczą platformą do wysyłania wiadomości, plików i połączenia z innymi narzędziami (Woodgate, 2019). Jest to oprogramowanie zamknięte (*proprietary*), z którego w ograniczonej wersji można korzystać bezpłatnie. Slack nie jest narzędziem charakterystycznym tylko dla DS, używają go bardzo różne środowiska i grupy, także niezwiązane z technologią ani pracą zawodową.

Slack może służyć osobom zajmującym się DS do współpracy i komunikacji w projekcie. Bywa elementem chałupniczego systemu rejestrowania i kontrolowania eksperymentów z modelami. Slacka można połączyć w sumie z około 1500 różnych narzędzi do komunikacji i współpracy, także ze związanymi z pisaniem kodu, jak np. GitHub i Stack Overflow (Woodgate, 2019). Slack obsługuje język znaczników zbliżony do Markdown, za jego pomocą można zatem programowalnie formatować wpisywany tekst, jak również zamieszczać w wiadomościach fragmenty kodu, zachowując ich charakterystyczny dla IDE czy terminala wygląd (czcionka, wcięcia itd.) i wyraźnie odróżniając od innych części tekstu.

Twórcy Slacka sądzą, że do sukcesu narzędzia przyczyniła się jego wizualna odrębność od podobnych narzędzi do pracy grupowej, które wyglądają „jak tani kostium na bal z lat 70. – wszędzie przygaszony błękit i szarości" (Woodgate, 2019). Slacka zaprojektowano na wzór żywej kolorystyki gier wideo, użyto zaokrąglonej czcionki bezszeryfowej, przyjaznych ikon i dużej liczby emotikonów, by nie kojarzył się z aplikacją do pracy w przedsiębiorstwie (Woodgate, 2019). **Slack wygląda nieformalnie, swobodnie, a to zgadza się z dominującym i uznawanym za odpowiedni wizerunkiem w świecie społecznym DS.** W rozmowach na Slacku nie zaczyna się wiadomości od „Szanowny Panie" – rozmawia się w sposób nieformalny. „Slack" to dosłownie 'coś luźnego'.

Slack i inne komunikatory tekstowe są uznawane za wygodne i sprzyjające efektywności, ponieważ komunikując cokolwiek całemu zespołowi czy pojedynczej osobie, nie oczekuje się natychmiastowej odpowiedzi i nie postrzega się takich wiadomości tekstowych jako odrywających od właśnie wykonywanego zadania – zazwyczaj samodzielnie przy komputerze, nierzadko ze słuchawkami na uszach. Komunikacja werbalna, gdy druga osoba pracuje wpatrzona w ekran ze słuchawkami na uszach, jest raczej postrzegana w badanym świecie jako niemiła, rozpraszająca, a więc kontrproduktywna. Przeszkadza w byciu efektywnym, a to działanie niezgodne z wartościami badanego świata.

Za wadę, szczególnie w komercyjnych zespołach DS z uwagi na formalne umowy dotyczące tajemnicy projektów, bywa uważane to, że całość danych ze Slacka znajduje się na serwerach Slacka, a nie firmy DS. Dodatkowo serwery Slacka to faktycznie maszyny firmy Amazon, gdyż usługa działa w chmurze AWS, w niektórych projektach komercyjnych nie używa się zatem tego narzędzia z przyczyn prawnych.

Obszary robocze (*workspace*) mogą być zamknięte lub otwarte. Do tych pierwszych mogą dołączać jedynie osoby zaproszone, do drugich każdy posiadacz konta Slack. Istnieje wiele międzynarodowych workspace'ów, na których komunikują się osoby zainteresowane DS, ML, AI, big data itp. (Blumenkrantz, 2018; Formulated.by, 2018). Znam jeden polskojęzyczny workspace tego typu, ale jest on zamknięty. Tworzą go osoby uczestniczące w bezpłatnym kursie, tzw. wyzwaniu Data Workshop autorstwa Vladimira Alekseichenko {obserwacje 35, 39, 43}. W polskim świecie DS zdecydowanie bardziej popularne w ogólnodostępnej komunikacji cyfrowej niż Slack są GitHub, forum Stack Overflow, a komunikację polskojęzyczną zagospodarowuje grupa facebookowa Data Science PL.

Nie opisuję szerzej komunikatorów tekstowych i głosowych, np. Skype'a, Google Hangouts, FaceTime'a, MS Teams, od rozpoczęcia pandemii COVID-19 są one bowiem powszechnie znane.

Co do mediów społecznościowych, to w polskim świecie DS na **Facebooku** najważniejszą platformą ogólnopolskiej komunikacji jest grupa Data Science PL. W badanym świecie dobrze znany jest tzw. skandal Cambridge Analytica i inne kontrowersje związane z tym, co dzieje się z danymi przekazywanymi Facebookowi przez jego użytkowników. Świadomość uczestników w tych tematach postrzegam jako bardzo wysoką, rzecz jasna nie tylko w odniesieniu do działań firmy Facebook. Cześć uczestników świata DS zwraca szczególną uwagę na swoją prywatność w internecie, np. nie korzysta z mediów społecznościowych i stosuje alternatywy wobec usług Google'a (jak np. wyszukiwarka internetowa DuckDuckGo i szyfrowana poczta elektroniczna w domenie protonmail.com), niektórzy używają szyfrowania połączeń z użyciem VPN. Inna część uważa, że ich prywatność nie jest zagrożona i korzysta z różnego rodzaju usług. Rozmawiałem nieformalnie z osobą, która pokazała mi, jak wyłączyć profilowanie reklam Google'a na moim telefonie z systemem Android, dodając, że sama ma włączone profilowanie „ze względów zawodowych" – chce widzieć, jakie treści są do niej dopasowywane. Nawet u osób

szczególnie chroniących swoją prywatność w internecie może pojawiać się przekonanie, że **muszą korzystać z mediów społecznościowych**. Także ja byłem przekonywany, że na potrzeby realizacji badań do niniejszej pracy powinienem ponownie założyć, przynajmniej fikcyjne, konto na Facebooku.

Osoby zajmujące się DS lub zainteresowane DS mogą na Facebooku śledzić wiele stron i grup związanych z tą tematyką. Z uwagi na profilowanie treści przez Facebooka otrzymują one w newsfeedach różnego rodzaju reklamy i oferty związane z DS. Można wskazać na dwie kategorie takich treści, czyli oferty edukacyjne (np. kursy, szkolenia, studia podyplomowe) i oferty pracy (mam na myśli także programy stażowe i praktyki studenckie). Jedna z rozmówczyń wskazała, że obecnie pracuje w firmie, której reklama praktyk studenckich wyświetliła się jej na Facebooku {wywiad 20}.

Twitter jest platformą mało popularną w polskim świecie DS. Spotkałem się z przypadkami obserwowania przez uczestników badanego świata kont twitterowych o międzynarodowej renomie, kont najbardziej znanych w badanym świecie (przykładowo na 5 września 2019 roku z Twittera korzystali Piotr Migdał i Dominik Batorski, a nie korzystał Przemysław Biecek), także kont firmowych.

LinkedIn to medium społecznościowe przeznaczone do kontaktów zawodowych i kreowania profesjonalnego wizerunku (Paliszkiewicz, 2018: 84–85). Jest popularne w badanym świecie, a **wśród tej części jego uczestników, którzy biorą udział w konferencjach branżowych, posiadanie konta na LinkedIn jest przyjmowane za pewnik**.

Zbieranie nowych kontaktów na LinkedIn wśród osób zajmujących się DS jest uznawane za cenne dla przyszłej kariery zawodowej, ponieważ może się wiązać z nowymi propozycjami zatrudnienia, udziału w projektach czy uzyskania zlecenia i pozyskania osób do współpracy. Nie chodzi tylko o liczbę kontaktów – bo popyt na pracowników z doświadczeniem w pisaniu kodu do operacji na danych jest wysoki i są oni oblegani przez rekruterów – ale o jakość.

Inną praktyką jest potwierdzanie umiejętności innych osób, np. w zakresie posługiwania się językami programowania R/Python. Posiadanie wielu potwierdzeń, szczególnie jeśli chodzi o te języki programowania, jest uznawane za wskaźnik profesjonalizmu i osadzenia w badanym świecie. Dzieje się tak m.in. dlatego, że **w badanym świecie nie ma uznanych sposobów formalnego potwierdzenia umiejętności**. Potwierdzenia umiejętności od innych uczestników badanego świata – szczególnie tych doświadczonych, silnie osadzonych w świecie DS – są bardziej cenione niż certyfikaty ukończenia kursów/szkoleń (w tym MOOC), a także bardziej cenione niż ukończenie studiów wyższych i podyplomowych.

Na pewno meetupów jako spotkań ani samej strony meetup.com nie należy uważać za narzędzia do współpracy, ale do komunikacji już tak. Meetupy to przede wszystkim spotkania. Komunikacja na meetup.com to w większości strony prezentujące grupy meetupowe i informacje o kolejnych meetupach. Inne funkcje tego medium są mało używane. Choć istnieje możliwość komentowania konkretnych spotkań, to nie zauważyłem, by ten mechanizm był w badanym świecie popularny.

Ewentualnie prelegenci mogą w komentarzu do swoich wystąpień przesyłać linki do używanych prezentacji, kodu (często będzie to link na GitHuba), sporadycznie pojawią się lakoniczne komentarze uczestników spotkań. Istnieje możliwość prowadzenia indywidualnych rozmów tekstowych na meetup.com, niemalże wcale nieużywana. Organizatorzy meetupów przeważnie korzystają z wysyłania powiadomień i innych informacji do osób zapisanych do grup za pomocą e-maili grupowych. Tak wysyłane są zarówno informacje o meetupach, jak i konferencjach czy hackatonach, nierzadko wraz z możliwością uzyskania zniżki na biletowane imprezy dla „członków społeczności".

Niektóre meetupy są okazją do spotykania się grup znajomych i konsultowania się, tzw. inspirowania czy dzielenia się wiedzą w sposób podobny jak na spotkaniach w zawodowych zespołach DS. Różnicą jest to, że na meetupach spotykają się osoby z różnych firm, a także niepracujące w komercyjnym DS, co może być w badanym świecie uznawane za jeszcze bardziej cenną okazję do dzielenia się wiedzą. W tym zakresie wymieniane są informacje i poglądy dotyczące rozwiązań technicznych i metodologicznych. Często będzie to rozmowa o zrealizowanych lub trwających projektach, a także o wykorzystywanych pakietach w Pythonie/R. Jeden z rozmówców, organizator meetupu, wskazywał, że dla niego i „stałych bywalców" **spotkania są przyjemnością i nie można ich traktować jako dokształcania się ani rozmów o pracy**. To, że rozmowy dotyczą DS, porównywał on do wspólnego zastanawiania się nad interesującą zagadką w dziedzinie będącej hobby czy pasją dla spotykających się {wywiad 13}.

Meetupy mogą odbywać się w pubach, większość uczestników pije piwo podczas prezentacji i po nich, nierzadko piwo i pizza są kupowane przez firmę-sponsora meetupu. Rozmowy o charakterze konsultacji, a także towarzyskie obywają się raczej po prelekcjach, czasami przed. Meetupy mogą także odbywać się w uczelnianych aulach wykładowych. Na takich spotkaniach nie spożywa się alkoholu, sponsorowana pizza i piwo może być poczęstunkiem w pobliskim pubie po zakończeniu prelekcji. Przy takich meetupach konsultacje i rozmowy towarzyskie mają miejsce prawie wyłącznie po prelekcjach, czasem nawet bez wizyty w pubie, na korytarzu, schodach, w okolicy wejścia.

Jeden z rozmówców mówił także, że część osób na meetupach to tzw. wannabesi:

> R: […] jest też część ludzi, którzy być może chcieliby kiedyś robić takie rzeczy, więc sobie słuchają, jakby budują sieć kontaktów […] to są studenci, którzy po prostu jeszcze szukają odpowiedzi na pytania, co mam robić po studiach, więc się jakby orientują, szukają, też jakby, no, też chcą może zbudować sieć kontaktów, ale raczej tak przychodzą, żeby tylko posłuchać. {wywiad 13}

Od 2018 roku meetupy DS w Polsce stały się głównie miejscem spotkań wannabesów z osobami osadzonymi w badanym świecie. Osoby osadzone w badanym świecie, zatrudnione w firmie sponsorującej meetup, mogą zajmować się organizacją tych spotkań w ramach swoich obowiązków zawodowych, a celem

głównym takiego meetupu jest budowanie wizerunku firmy-pracodawcy (*employer branding*) wśród osób zainteresowanych wejściem do świata DS i rozpoczęciem kariery zawodowej w DS (czyli przeważnie wśród tzw. wannabesów). **Uczestnicy badanego świata raczej nie są zainteresowani prelekcjami na meetupach** (ewentualnie w roli prelegenta), **a jeżeli już biorą udział w spotkaniach, to będą zainteresowani rozmowami z innymi uczestnikami świata** (nie z wannabesami) **przed i po prelekcjach**. Osoby osadzone w badanym świecie może zatem przyciągnąć na meetup możliwość rozmowy z rzadko widywanymi lub nieznanymi osobiście osobami o podobnym lub głębszym stopniu osadzenia, czasami także względy towarzyskie.

Niemalże wszystkie wydarzenia – meetupy, konferencje, hackatony – w badanym świecie są sponsorowane przez firmy, które zatrudniają data scientistów. Jest to dominująca praktyka, choć spotkałem się z opiniami, że **bycie sponsorowanym nawet przez uczelnię może być ograniczające** i dla meetupu będzie lepiej, jeżeli piwo postawi organizator z prywatnych pieniędzy (szczególnie że ma wysokie dochody). Alekseichenko promuje swoją konferencję m.in. jako niemającą sponsorów, podkreślając, że jego wydarzenie – w odróżnieniu od większości obecnie organizowanych – nie jest „festiwalem roll-upów" (Alekseichenko, 2019b), czyli stojących wszędzie banerów z logami sponsorów, wszechobecnej rekrutacji i sprzedawania usług analitycznych sponsorów.

Ponadto **bycie organizatorem meetupu może być uznawane za cenny element budowania własnego wizerunku profesjonalnego i pozycji w świecie społecznym DS**. Wskazywano mi, że podanie w CV i podczas rozmowy rekrutacyjnej informacji o byciu przez kilka lat organizatorem meetupu wywarło silne, pozytywne wrażenie na osobach rekrutujących. Świadczy to o postrzeganiu połączenia umiejętności technicznych z interpersonalnymi (a bycie organizatorem meetupu może być uważane za wskaźnik tych drugich) jako cennych i unikalnych w pracy data scientisty.

Uczestnicy badanego świata śledzą strony internetowe związane z tematyką DS, np. KDnuggets, Towards Data Science, oraz różnego rodzaju blogi, raczej anglojęzyczne, szczególnie na medium.com. Właśnie na platformie medium.com znajduje się strona Towards Data Science. W Polsce blogi związane ze światem społecznym DS to m.in. „SmarterPoland"[67] prowadzony przez Przemysława Biecka, blog Łukasza Prokulskiego[68], blog „Szychta w danych"[69] autorstwa Piotra Sobczyka, blog Mateusza Grzyba[70], blog Piotra Migdała[71], „Big Data Passion"[72] Radosława Szmita i Marcina Wojtczaka, „R addict"[73] Marcina Kosińskiego, „Uczymy

67 http://smarterpoland.pl/ (dostęp: 19.04.2019).
68 https://blog.prokulski.science/ (dostęp: 19.04.2019).
69 http://szychtawdanych.pl/ (dostęp: 19.04.2019).
70 http://mateuszgrzyb.pl/ (dostęp: 19.04.2019).
71 https://p.migdal.pl/ (dostęp: 19.04.2019).
72 http://bigdatapassion.pl/ (dostęp: 19.04.2019).
73 http://r-addict.com/ (dostęp: 19.04.2019).

maszyny”[74] Konrada Łydy, a także podkasty: „Data Science po polsku”[75] Szymona Drejewicza (publikacje w latach 2017–2018, później w większym zespole pod nazwą „Data Evangelist”, obecnie nieaktywny) oraz „Biznes Myśli”[76] Vladimira Alekseichenko. Istnieją liczne blogi firmowe. Karol Majek, autor bloga „Deep Drive PL”, poświęconego częściowo pokrywającej się z data science tematyce autonomicznych samochodów, mobilnej robotyki i sieci neuronowych, sporządził spis polskich blogów o „sztucznej inteligencji i analizie danych” (Majek, 2020). Wymienił on także blogi „nietechniczne, bez kodu czy przykładów”, których autorzy nie są uznawani za uczestników świata DS – jak publicznie finansowana sztucznainteligencja.org.pl. Niektórzy uczestnicy badanego świata mogą śledzić np. strony branży IT, takie jak bulldogjob.pl czy poświęcony bezpieczeństwu niebezpiecznik.pl.

Komunikacja między uczestnikami świata DS odbywa się też poprzez kod. Zarówno Python, jak i R bywają określane, niekoniecznie przez uczestników badanego świata, jako *lingua franca* DS (Barry, 2016; Nunns, 2017; ActiveState, 2018; Thieme, 2018), tak więc są to także języki służące komunikacji pomiędzy ludźmi. Przykładowo na Stack Overflow (SO) fragmenty kodu służą objaśnianiu zarówno pytań, jak i odpowiedzi. W posługiwaniu się kodem komunikacja z komputerem jest najważniejsza – pozostając przy SO, najbardziej cenione są pytania i odpowiedzi zawierające kompletny, działający kod. Wyjątkowym narzędziem współpracy i komunikacji, zorientowanym głównie na użytkowników konkretnego języka programowania, są pakiety (*packages*). Uważam je za wyjątkowy przypadek podkreślanego w badanym świecie dzielenia się wiedzą. Te zestawy funkcji są chętnie prezentowane w badanym świecie, szczególnie na wydarzeniach nastawionych na konkretny język programowania {obserwacja 9}. Pakiety składają się nie tylko z kodu – czytelny opis, zwany winietką, jest uznawany za niezbędną część dobrego pakietu. Pakiety mogą być narzędziem ogólnodostępnym przez publiczne repozytoria, ale istnieje praktyka budowania pakietów na użytek wewnętrzny firmy, zespołu DS czy pojedynczej osoby. Takie prace mogą ewoluować w kierunku publicznych pakietów i innych narzędzi, co widać na przykładzie najbardziej znanych w badanym świecie twórców, którzy początkowo budowali narzędzia dla siebie.

4.3.5. Przemilczane technologie

Zainspirowany pojęciem wiedzy milczącej (Polanyi, 2005: 96) opisuję dwie technologie społecznego świata DS: **język angielski i tzw. terminal**. Traktuję pojęcie Michaela Polanyi'ego w sposób redukcyjny, nie pisząc tutaj o wiedzy w ogóle, a koncentrując się na tych przemilczanych technologiach badanego świata, których znajomość jest tak oczywista i powszechna, że nie werbalizuje się jej (np.

74 https://uczymymaszyny.pl/ (dostęp: 19.04.2019).
75 https://www.youtube.com/channel/UC7DGutbNdAjEFbOV-Al8aRA/featured (dostęp: 19.04.2019).
76 https://biznesmysli.pl/ (dostęp: 19.04.2019).

nie zapisuje w ofertach pracy, nie problematyzuje w materiałach edukacyjnych). W przypadku terminala znajduję rzadkie przykłady problematyzowania, z kolei język angielski jest technologią krystalicznie przezroczystą. Mówię także o zestawie przemilczanej wiedzy i umiejętności, który składa się na domyślną w badanym świecie biegłość technologiczną.

Dalej mówię o przemilczanych aspektach wykonywania działania podstawowego, wiązanych przez uczestników z tym, że DS jest w ich ocenie bliskie sztuce lub eksperymentom naukowym. Nie ma w świecie DS jednoznacznych ani sformalizowanych procedur postępowania w projekcie, a mikrodecyzje (np. dotyczące zmiany parametrów modelu) bywają przez uczestników podejmowane na podstawie intuicji czy doświadczenia. Pojawiają się odwołania do kreatywności, twórczości, myślenia nieszablonowego – to uznawane jest przez uczestników za kolejną cechę odróżniająca ich od programistów lub inżynierów w ogóle.

Język angielski jest domyślnym językiem badanego świata. W działalności komercyjnej zdecydowanie więcej firm zajmujących się tzw. AI w Polsce pracuje dla klientów zagranicznych niż dla polskich, a w komunikacji z nimi prawie zawsze będzie wykorzystywany język angielski. Wydarzenia w polskim świecie DS nierzadko odbywają się głównie w języku angielskim, co jest określane jako wydarzenie otwarte na osoby nieznające polskiego (w zespołach DS w Polsce mogą pracować osoby nieznające polskiego, ponieważ zespół pracuje w całości po angielsku).

Nie znając angielskiego, nie można wykonywać działania podstawowego badanego świata. Na języku tym bazują wszystkie wykorzystywane w DS języki programowania. Bez znajomości angielskiego dostęp do komunikacji w badanym świecie byłby ograniczony i uniemożliwiałby bycie jej uczestnikiem.

Angielski jest uznawany za międzynarodowy język technologii, nauki i biznesu, choć nie jest to rodzimy język dla około 95% światowej populacji (Guo, 2018: 1). Publicystyczne określenie „programowanie jest dla każdego, pod warunkiem że zna angielski" (McCulloch, 2019) jest w mojej opinii trafne. Na przykładzie Pythona wskazywałem, że jest czytelny jak *plain English* (dosł. 'prosty angielski', por. Orsini, 2014; Dodge, 2018; Garner, 2018; Reitz, 2018). „Czytelny" znaczy „czytelny dla osób znających angielski".

Choć elementy narzędzi programistycznych są tłumaczone na inne języki (tłumaczone, czyli źródłowo były zapisane po angielsku), to istnieje praktyka powrotu do języka angielskiego. Przykładowo: choć RStudio jest instalowane domyślnie w języku lokalnym, to korzystanie z tego ustawienia może być uznawane za niewygodne i co za tym idzie – nieefektywne, człowiek piszący kod uzyskuje bowiem komunikaty o błędach częściowo w języku lokalnym. To z kolei szukanie pomocy względem niezrozumiałego błędu, np. na Stack Overflow. Sam Wickham doradził początkującym:

> Jeśli Twoje komunikaty o błędach nie są po angielsku, odpal linijkę kodu 'Sys.setenv(LANGUAGE = „en")'; masz większe prawdopodobieństwo otrzymania pomocy dla komunikatów w języku angielskim (Wickham, Grolemund, 2017).

W toku własnych doświadczeń z pisaniem kodu nauczyłem się szukać w internecie pomocy w tym zakresie wyłącznie po angielsku.

Mówiłem o angloamerykańskim rodowodzie DS (ten termin i wiele innych nie są tłumaczone w badanym świecie). Taki rodowód mają technologie komputerowe (*computing*) w ogóle. Fakt, że języki programowania bazują na angielskim – słowach *if*, *then*, *break*, *list*, *return* – ma tylko kulturowe uzasadnienie. W sensie technicznym języki programowania mogłyby działać, bazując na dowolnym zestawie symboli, nawet niebędącym ludzkim językiem (McCulloch, 2019). Istnieje np. język programowania Whitespace (dosł. 'biały znak'), który składa się wyłącznie ze znaków niedrukowanych, takich jak znak spacji i tabulatora (McCulloch, 2019). Gretchen McCulloch (2019) słusznie uważa, że jedyne powszechnie wykorzystywane narzędzia z elementami pisania kodu, dla których istnieje akceptowana przez użytkowników praktyka tłumaczenia samego kodu na języki lokalne, to Excel i Wikipedia. Formuły Excela są przecież tłumaczone w całości, np. funkcja `AVERAGE()` to w polskiej wersji językowej `ŚREDNIA()`, choć dodajmy, że w badanym świecie formuły arkusza kalkulacyjnego raczej nie byłyby uznane za kod w ogóle. Excel jest w badanym świecie zdecydowanie uznawany za narzędzie niewłaściwe do wykonywania działania podstawowego.

Terminal to nieco mniej niż język angielski przezroczysta dla badanego świata technologia, tak więc – choć nie jest całkowicie przemilczana – **jej znajomość jest przyjmowana za pewnik i niemalże tożsama z umiejętnością posługiwania się komputerem w ogóle**. Wzmianki o niej pojawiają się sporadycznie, np. w materiałach edukacyjnych związanych z DS dla osób niemających wykształcenia informatycznego, np. studentów statystyki:

> Powłoka (*shell*) to program na twoim komputerze, którego zadaniem jest uruchamianie innych programów. Pseudo-synonimy to „terminal" (*terminal*), „linia poleceń" (*command line*) i „konsola" (*console*). Różnicom pomiędzy nimi poświęcony jest cały wątek na StackExchange[77], ale nie wydaje mi się, żeby był wybitnie pouczający. […] Wielu programistów spędza dużo czasu w powłoce, w przeciwieństwie do GUI [czyli interfejsów graficznych – przyp. R.Ż.], ponieważ powłoka jest bardzo szybka, zwięzła (*concise*) i wszechobecna w odpowiednich środowiskach komputerowych. Zanim otrzymaliśmy myszkę i GUI, w powłoce była wykonywana cała praca. Najczęściej używaną powłoką jest *bash*, którego nazwy czasami używa się jako proxy dla „powłoki", podobnie jak „Coca-cola" i „Kleenex" są proxy dla napojów typu cola i chusteczek[78] (Jenny Bryan, 2018).

77 *What is the difference between Terminal, Console, Shell, and Command Line?*, 2014.

78 W Polsce *bash* to konsola, tak jak adidasy to buty sportowe.

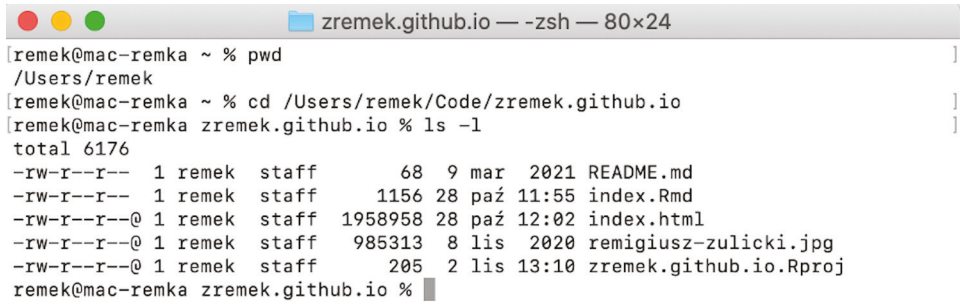
Określenia „terminal" i „konsola" pochodzą z czasów, gdy urządzenie wyposażone w monitor i klawiaturę do komunikacji człowieka z systemem operacyjnym – a więc właśnie konsola/terminal – było czymś odrębnym od komputera, czyli urządzenia wykonującego obliczenia, zajmującego oddzielne pomieszczenie (Bugalski, 2019).

Terminal w DS służy wielu celom – od uruchamiania skryptów wykonujących operacje na danych w językach Python/R, przez korzystanie z Git i technologii kontenerowych (jak Docker), do czynności, które w pracy biurowej wykonywane są za pomocą interfejsów graficznych: uruchamiania i zamykania programów, otwierania plików, tworzenia/przenoszenia/kopiowania/kasowania/nazywania/zmiany nazw plików i folderów, sprawdzania zawartości folderów/plików, instalowania/usuwania oprogramowania (w tym pakietów), aktualizowania oprogramowania i systemu operacyjnego. Jest on używany niemal zawsze, kiedy praca (w sensie faktycznego użycia sprzętu i mocy obliczeniowej) wykonywana jest na maszynie innej niż komputer personalny, czyli przy pracy na lokalnym serwerze lub w tzw. chmurze. Praca na takiej maszynie wykonywana jest w ten sposób, że na komputerze personalnym uruchamia się okno z terminalem tej maszyny i w nim wykonuje się zadania.

Korzystanie z komputera za pomocą terminala w porównaniu do używania GUI jest jak gotowanie z podstawowych produktów w porównaniu do podgrzewania gotowych potraw ze sklepu. Gotując samodzielnie (czyli korzystając z terminala), człowiek decyduje o każdym kolejnym kroku, jest to scenariusz wymagający bardziej zaawansowanych umiejętności i narzędzi, dający relatywnie większe możliwości, swobodę, kontrolę. Można przygotować większą ilość potrawy na kilka dni, półprodukty do wykorzystania na później. Odgrzewając gotową porcję (korzystając z GUI), człowiek potrzebuje mniej zaawansowanych umiejętności i narzędzi, ma relatywnie mniejsze możliwości, po prostu wygodnie korzysta z tego, co zostało przygotowane przez kogoś innego.

Interfejs terminala to jednokolorowe okno z informacją tekstową o zalogowanym do komputera użytkowniku i hoście, następnie widoczny jest znak zachęty i migający kursor. Nie ma możliwości korzystania z GUI, a więc i myszki.

Komenda `pwd` wyświetla lokalizację roboczą (*print working directory*), czyli folder, w którym użytkownik w tej chwili coś robi. Na konsolę wypisana jest lokalizacja `/Users/remek`. Następnie przez `cd` ze ścieżką `/Code/...` wykonywana jest zmiana lokalizacji roboczej (*change directory*). Nie ma wypisania na konsolę, efekt widoczny jest w kolejnej linii jako lokalizacja przed znakiem zachęty (`%`), a po informacji o użytkowniku i hoście (`remek@mac-remka`). Polecenie `ls` to wyświetlenie listy plików w lokalizacji (*list files*) z opcją `l` (długi format opisu zawartości lokalizacji, każdy w osobnej linii). To interaktywna praca za pomocą konsoli, tj. człowiek wpisuje komendę, wciska *Enter* i komenda jest wykonywana. Tak samo jak w Pythonie/R, w języku bash można zapisać skrypt. Skrypty służą głównie automatyzacji powtarzalnych czynności (Pyszczuk, 2011: 6).



Rysunek 4.14. Terminal i przykładowe polecenia

Źródło: opracowanie własne na podstawie Jenny Bryan, 2018; macOS Catalina.

Terminal nie jest narzędziem charakterystycznym wyłącznie dla DS. Niemniej jest to **narzędzie służące zwiększeniu efektywności – kluczowej wartości społecznego DS**. Korzystają z niego różnego rodzaju ludzie związani z informatyką, a także osoby, które ogólnie i nieprecyzyjnie nazywam zaawansowanymi użytkownikami komputera. **Używanie terminala ma pomagać w uczynieniu pracy na komputerze szybszą, bardziej efektywną, produktywną, płynną** (Pyszczuk, 2011; Jenny Bryan, 2018; Goldstein, 2019). Terminal czy interfejsy tekstowe w ogóle są uznawane w tej nieprecyzyjnie zdefiniowanej grupie za **narzędzia potężniejsze niż interfejsy graficzne** (Groen, 2019), dające człowiekowi możliwość bezpośredniej interakcji z komputerem i dzięki temu **większą niż przez GUI kontrolę nad komputerem** (Goldstein, 2019: 41). Osoby już biegle korzystające z domyślnej na systemach Unix/Linux powłoki bash mogą problematyzować jej „potęgę" i efektywność, a więc korzystać z innych powłok, np. zsh, fish lub rc. Albin Groen (2019) pisze także o znudzeniu starym, dobrze znanym terminalem. Zsh jest domyślną powłoką w macOS od systemu Catalina.

Charakterystyczne dla DS narzędzia ułatwiające pisanie kodu – Jupyter Notebook, Jupyter Lab, RStudio – umożliwiają korzystanie z komend bashowych bez konieczności otwierania kolejnego okna, co jest uznawane za wygodne. W notatniku Jupyter komendy wpisuje się bezpośrednio do komórki (*cell*)[79], w pozostałych dwóch narzędziach uruchamia się okno terminala otwartego w lokalizacji roboczej wewnątrz aplikacji. Te rozwiązania mają sprzyjać efektywności pracy.

Interfejs terminala jest odbierany przez osoby niedoświadczone w posługiwaniu się tym narzędziem jako nieprzyjazny i onieśmielający, gdyż w drugiej dekadzie XXI wieku użytkownik wykonujący na komputerze jedynie prace biurowe lub używający go do spraw prywatnych czy rozrywki ma styczność wyłącznie z GUI (Goldstein, 2019). **Korzystanie z terminala może być uznawane za prawdziwe użytkowanie komputera**, jak napisałem wyżej – wręcz tożsame z umiejętnością korzystania z komputera w ogóle – właśnie dlatego, że jest ono trudniejsze

79 Poprzedzając je znakiem „!" lub poleceniem zwanym *cell magic* „%%bash".

i daje więcej możliwości niż GUI. Korzystanie z terminala odróżnia użytkowników biurowo-prywatno-rozrywkowych, czyli komputerowych laików, od tych zaawansowanych. Uczestnicy świata DS to zdecydowanie ci drudzy. Pojawiają się żarty na temat popkulturowych skojarzeń z siedzącym w piwnicy nerdem lub groźnym, zakapturzonym hakerem, jakie widok okna terminala budzi wśród laików.

Nie jest jednak tak, że terminal jest w badanym świecie uznawany za jedyny właściwy sposób korzystania z komputera. „Nikt nie przyznaje odznak Git Nerd. Korzystaj z Git w taki sposób, abyś był jak najbardziej efektywny" (Jenny Bryan, 2018). Unikać myszki i dążyć – nawet kosztem efektywności – do używania terminala dla wszelkich zadań mogą początkujący uczestnicy świata DS, chcący dowieść swojej biegłości technologicznej. Z drugiej strony podczas obserwacji prelegentów (niekoniecznie osób osadzonych w świecie DS) spotykałem stylizujących tytuł slajdu o sobie na komendę bashową „$ whoami"[80]. Manifestowanie korzystania z basha jest zatem strategią autoprezentacji i legitymizowania siebie jako (co najmniej) zaawansowanego użytkownika komputera. Efektywność jest jednak w świecie DS wartością cenioną bardziej niż stosowanie konkretnego narzędzia. Jeżeli korzystanie z interfejsu tekstowego ogranicza efektywność, wpływa negatywnie na jakość pracy, należałoby użyć GUI, a także zmieniać podejścia i iteracyjnie, podobnie jak same modele ML, szukać optymalnego sposobu pracy (tzw. *workflow*).

W terminalu działa funkcja autouzupełniania wpisywanego tekstu – po wpisaniu pierwszych liter i wciśnięciu *Tab* słowo jednoznaczne będzie uzupełnione przez komputer. Autouzupełnienie nie zadziała jednak dla nazw plików czy folderów, w których istnieją spacje. Dla komputerowych laików ten potencjalny kłopot jest niedostrzegalny, ponieważ z poziomu GUI, np. w Windowsie, taki plik/katalog zadziała. Mogą pojawić się także inne problemy techniczne, stosowana jest zatem np. konwencja *snake case* `nazwa_pliku` lub *camel case* `nazwaPliku`[81]. Wśród użytkowników Linuxa pojawił się na Twitterze komentarz „przerwa (*space*) w nazwie pliku to przerwa w czyjejś duszy" (Jenny Bryan, 2018). Zasady nazywania plików są zatem elementem wiedzy milczącej badanego świata DS.

Z terminalu korzysta się inaczej na różnych systemach operacyjnych komputera. **Zasadnicza różnica występuje pomiędzy Windowsem a systemami Unix/Linux.** „Windows jest wyjątkowy… ale nie w dobrym tego słowa znaczeniu" (Jenny Bryan, 2018), ponieważ domyślnie nie ma on powłoki bash. W domyślnej powłoce Windowsa nie ma możliwości używania np. Gita czy Anacondy, podstawowe komendy niekiedy różnią się[82] od bashowych, tak więc czasami użytkownicy tego

80 Komenda pochodzi od „Who am I?" (dosł. 'kim jestem?'). Wypisuje ona w konsoli informację o aktualnie zalogowanym użytkowniku komputera.

81 Przy tym do nazw plików używane jest raczej snake_case, a przy nazwach zapisywanych w kodzie (np. zmienne, funkcje) trwają dyskusje o wadach i zaletach różnych konwencji (Sharif, Maletic, 2010; Wickham, Grolemund, 2017).

82 Przykładowo pokazane wyżej bashowe „pwd" to w powłoce Windowsa „dir".

najpopularniejszego[83] systemu operacyjnego instalują dodatkowe terminale, np. Git Bash czy będący częścią dystrybucji Anacondy dla Windowsa terminal Anaconda Prompt. Jeszcze inny terminal to Rterm, który może służyć do korzystania z technologii kontenerowych, jak Docker. **Windows może być uznawany za najmniej wygodny i najmniej efektywny system operacyjny do DS.**

Nie oznacza to, że Windows uznawany jest za nienadający się zupełnie do DS, a raczej za nie najlepszy wybór. Część uczestników świata DS korzysta z Windowsa, szczególnie na komputerach służbowych, a także po to, by uprościć sobie komunikację z klientami korzystającymi z tego systemu i aplikacji Microsoft Office. Wiele z tych osób instaluje obok Windowsa dodatkowo (tzw. *dual boot*) dystrybucję Linuxa. Część uczestników badanego świata nie przykłada większej wagi do systemu operacyjnego na komputerze personalnym, część może preferować system macOC i komputery Apple, a jeszcze inni różne dystrybucje Linuxa.

Komputery Apple są to faktycznie maszyny z systemem unixowym macOS, tak więc domyślnie można korzystać z funkcjonalności powłoki zsh/bash, a opinia o nich jako niezawodnych, intuicyjnych w obsłudze oraz drogich urządzeniach sprawia, że **mogą być w badanym świecie uznawane za symbol statusu**. O takim charakterze marki pisano wiele (Campbell, Pastina, 2010; Krzysztofek, 2011). O ile komputery z systemem macOS to około 13% globalnego rynku, to w USA jest to ponad 18%, a w Polsce niecałe 4% (StatCounter, 2019). MacBooki występują w Polsce stosunkowo rzadko i mogą być kojarzone z – jakkolwiek archaicznie to brzmi – „Zachodem" czy „zagranicą". W Polsce zajmujący się DS pracownicy firm, np. Pearson oraz iDash, występując na wydarzeniach w rolach prowadzących/prelegentów, korzystają z komputerów Apple {obserwacje 9, 26, 30, 38}. Z takim laptopem występuje publicznie także Piotr Migdał, a z grona globalnych celebrytów badanego świata np. Hadley Wickham.

Część badanego świata może preferować pracę na komputerach personalnych pod kontrolą dystrybucji GNU Linuxa, uznając maszyny z macOS za zbędny wydatek lub wyłącznie symbol statusu, a Windowsa za system niewygodny i nieefektywny. Takie osoby mogą korzystać z tzw. laptopów gamingowych lub serii ThinkPad firmy Lenovo, lub innych, tzw. biznesowych linii, ponieważ takie urządzenia mają opinię wydajnych i niezawodnych (szczególnie ThinkPady). Nie zdecydowałem się na opisywanie tych różnych pozycji dotyczących preferowanego systemu operacyjnego ani marki urządzenia jako areny badanego świata, ponieważ nie zachodzi tutaj spór. Każdy system operacyjny czy urządzenie będzie akceptowalne, o ile sprzyja efektywności i wygodzie w wykonywaniu działania podstawowego i działań wspomagających. Niemal wszystkie ważne technologie cyfrowe badanego świata działają na każdej z opisywanych platform: Windowsie, macOS i Linuxie. Natomiast **doświadczenie w korzystaniu wyłącznie z systemu operacyjnego Windows jest dla uczestników świata DS wskaźnikiem bycia co**

83 W sierpniu 2019 roku pod kontrolą Windowsa pracowało około 78% komputerów na całym świecie, drugi w kolejności macOS to już zaledwie 13%, a Linux niecałe 2% (StatCounter, 2019).

najwyżej uczestnikiem początkującym. Osoba, która nie korzystała nigdy z systemu Unix/Linux, bez wątpienia nie pracowała nigdy na serwerze ani maszynie w tzw. chmurze, a takie maszyny są powszechnie wykorzystywane, szczególnie w komercyjnym DS, pracują pod kontrolą różnych dystrybucji Linuxa i interakcja człowieka z nimi przebiega poprzez interfejs tekstowy, czyli terminal. Umiejętność korzystania z terminala i minimalna znajomość systemów operacyjnych Unix/Linux są niemal tożsame i traktuję je jako element wiedzy milczącej badanego świata. Umiejętność korzystania z terminala i różnych systemów operacyjnych nie jest jednak sama w sobie tym, co świadczy o byciu uczestnikiem świata DS. Wskazuje wyłącznie na bycie zaawansowanym użytkownikiem komputera, ewentualnie na związek z szeroko pojętą informatyką.

W repertuarze wiedzy milczącej zaawansowanego użytkownika komputera jest także m.in. znajomość podstaw technologii *front end* (dosł. 'przedni koniec') – np. języka znaczników HTML, CSS (kaskadowych arkuszy stylów) oraz języka JavaScript, czyli takich, które odpowiadają za wygląd i działanie stron WWW oraz innych aplikacji internetowych. Wiedza ta będzie przydatna szczególnie dla data scientistów zajmujących się tworzeniem interaktywnych wizualizacji danych dla klientów, ma także znaczenie przy zadaniach webscrapingu. Będzie to także wiedza o tym, w jaki sposób komputer zapisuje i wyświetla obraz, czyli znajomość pojęć, np. kanałów RGB i przypisanej każdemu z nich wartości od 0 do 255 – wiedza niezbędna każdemu, kto używa ML dla danych w formie obrazów. Będzie to również (silnie związany z technologiami *front end*) zestaw wiedzy i umiejętności dotyczących podstaw działania internetu, przeglądarek internetowych, wyszukiwarek internetowych, pojęć takich jak „URL", „domena", „hosting" itp., szczególnie do celów efektywnego wyszukiwania informacji np. na Stack Overflow, ale także przy wykonywaniu webscrapingu. Zaawansowany użytkownik posługuje się także np. pojęciem tzw. wagi plików/aplikacji, czyli ich rozmiarów wyrażonych w bajtach, rozumie różnicę rzędu wielkości między 10 KB a 10 TB. Przydatna na etapie przygotowania danych będzie znajomość wyrażeń regularnych (*regular expressions*, *regex*), czyli wzorców znaków pozwalających wyszukiwać np. słowa (a właściwie dowolne ciągi znaków) m.in. na podstawie tego, jak się one zaczynają, kończą, czy zawierają w sobie konkretny wzór. Przykłady można by mnożyć; także znajomość SQL-a może być uznawana za coś oczywistego, aż do poziomu przemilczenia.

Część uczestników świata DS to osoby, które już w wieku kilku, kilkunastu lat interesowały się komputerami. Niektórzy mogą identyfikować się jako nerdzi lub geekowie, inni jako gamerzy/e-sportowcy, część jako entuzjaści nowoczesnych technologii, istnieje szerokie i niebudzące popkulturowych skojarzeń określenie *tech-savvy*, które oznacza biegłość technologiczną[84]. Autodefiniowanie się jako

84 Badacze społeczni od około 2000 roku używali określeń „tech-savvy", „GenTech", a także „digital natives" (tłumaczonego jako „cyfrowi tubylcy") do nazywania pokoleń ludzi korzystających z komputerów i internetu od najmłodszych lat, nieznających świata bez tych technologii (Mallan, Singh, Giardina, 2010; Pavlicek, Sudzina, Malinova, 2017; Jain, 2018). Choć

osoby od dziecka zainteresowanej technologią lub matematyką opisuję w charakterze praktyki indywidualnej legitymizacji działania uczestników badanego świata. Istnieją w nim uczestnicy akcentujący swój brak zainteresowania technologiami w dzieciństwie (ale nie spotkałem się z wyrażaniem braku zainteresowania lub niechęci do matematyki), którzy nabrali tej biegłości w toku kolejnych doświadczeń naukowych lub zawodowych związanych z badaniami ilościowymi czy pracą z danymi w ogóle. Niemniej już **jako uczestnik świata DS „trzeba" być osobą biegłą technicznie i kolokwialnie – nie bać się technologii**. Brak strachu przed zepsuciem czy zniszczeniem systemów technologicznych, połączony z rozeznaniem w opisywanym wyżej repertuarze technologicznej wiedzy milczącej, w połączeniu ze znajomością języka angielskiego i umiejętnością korzystania z najbardziej aktualnych źródeł referencyjnych oraz z samodzielnością, poczuciem kontroli[85] nad systemami technologicznymi, a także przekonaniem, że technologie są dla człowieka zasadniczo proste, składa się na ową „konieczną" biegłość technologiczną uczestnika świata DS.

Podkreślam namacalność (*palpable*) (Strauss, 1978: 121) pracy na komputerze, którą wykonują uczestnicy świata DS – wciskanie klawiszy palcami. Skoro działanie podstawowe jest wykonywane za pomocą pisania kodu, a nierzadko uczestnicy to osoby piszące z dużą prędkością czy nawet bezwzrokowo, to w ogóle **odrywanie dłoni od klawiatury uznawać mogą one za niewygodne i nieefektywne** – ruch zbędny i nieergonomiczny. Jest to argument za posługiwaniem się terminalem, ale także **skrótami klawiaturowymi**. Nie należy nazywać tego wiedzą milczącą, ale do elementarza umiejętności uczestników świata DS należy co najmniej zestaw skrótów klawiaturowych dla systemu operacyjnego, którego się używa, oraz zestaw skrótów dla najczęściej używanego IDE (np. Jupyter Notebook, PyCharm, RStudio). Klikanie myszką, by zmienić okno aktywnej aplikacji lub uruchomić linię kodu[86], będzie dla uczestników badanego świata indykatorem komputerowego laika, nawet do granicy kwestionowania możliwości uznania takiej osoby za uczestnika świata DS. Jako **element wiedzy milczącej traktuję także samo korzystanie z klawiatury**, tj. szybkie pisanie czy znajomość znaków niebiurowych, np. ~ _ ^ | `.

---

niemal wszyscy uczestnicy badanego świata DS mogą być nazwani cyfrowymi tubylcami przynajmniej ze względu na swój wiek, to dla nich poziom wiedzy i umiejętności komputerowych tzw. cyfrowych tubylców jest poziomem laickim.

85 Kontrola nad maszyną była w różnych badaniach nad hakerami określana jako motywacja indywidualna hakerów (Zaród, 2018: 29–30, 36–37), a więc uznaję to poczucie za immanentny składnik opisywanej przeze mnie biegłości technicznej, nie za coś charakterystycznego dla DS.

86 Zmiana aktywnego okna to skrót *Alt*/*Cmd* + *Tab*, uruchomienie linii kodu (a także np. wysłanie e-maila) to *Ctrl*/*Cmd* + *Enter*.

4.3.6. Stos technologiczny data science

Podsumowując, warto przyjrzeć się wypowiedzi jednej z osób osadzonych w badanym świecie, która opowiada o własnych narzędziach pracy:

> *B: To chciałbym, w inną stronę teraz pójść i zapytać cię, jakie masz narzędzia pracy?*
> R: Moje narzędzia pracy? Yyy, ja pracuję na MacBooku Pro, większość, większość mojego zespołu pracuje na MacBookach. Ee, mój język programowania wyboru to jest Python, mimo że zaczynałam w eR, przesunęłam się w stronę Pythona. Yyy, pracuję w Jupyter Notebooku, zazwyczaj, bo moim zdaniem jest, jest łatwo, jest szybko, nie potrzeba dodatkowego oprogramowania. Eee, co jeszcze? I zazwyczaj rozwiązania w chmurze Amazon Web Services i tak dalej. Ostatnio też zaczęłam nowe rozwiązania testować, takie na przykład jak Databricks […] Do komunikacji Slack i Google Hangouts. Eee [pauza], no i oczywiście najważniejsza chyba jest konsola, eee, w której wszystko, wszystko robię. […] używam programu od Cisco, do VPNów, kiedy jestem, do używania VPNa, kiedy jestem poza biurem. Myślę, że to są wszystkie moje codzienne narzędzia. Czasami muszę poszukać czegoś nowego, ale to jest taki codzienny standard, co otwieram każdego dnia i sprawdzam. […] [Databricks – przyp. R.Ż.] To jest [krótka pauza] rozwiązanie, które, to jest środowisko, które łączy Sparka z Amazon Web Services, iii można bezpośrednio w nim uruchamiać kod, zarządzać swoim klastrem. Jest po prostu, powiedzmy, przyjaznym miksem wielu narzędzi. Jest takim centrum dowodzenia.
> *B: Mhm, mhm, ok. A, mówiłaś o konsoli, to chodzi o wiersz polecenia, tak, w komputerze?*
> R: Tak. Tak, tak, chodzi o terminal.
> *B: O terminal? A dlaczego, bo, bo, dobrze, dobrze zapamiętałem, że powiedziałaś, że to jest najważniejsze?*
> R: Dla mnie tak. Dla mnie to jest ważne, ponieważ, yyy, kiedy się łączy na przykład ze zdalną maszyną, z jakimś, jakimś komputerem w chmurze, tooo mogę to robić wyłącznie przez terminal i na przykład jak chcę odpalić na nim kod, to wszystko muszę, robię przez terminal, czy ściąganie danych, czy na przykład wrzucanie danych do chmury, też wszystko robię przez terminal, czy jak uruchamiam jakieś skrypty, to też staram się to robić przez terminal, co jest sporym ułatwieniem, eee, i dlatego jest dla mnie najważniejszy. Zaraz po Pythonie chyba, ale to [krótka pauza] jest oczywiste. […]
> *B: Mhm, mhm, ok. Yyy, aaa, a takie rzeczy jak kontrola wersji, też ci się przydaje?*
> R: Tak, używamy w firmie Gita, mmm, myślę, że Git jest standardem w środowisku, e, iii staramy się trzymać **dobrych** praktyk, używając Gita, takich typowych dla software engineering. Yee, to mi się przydaje, ale zazwyczaj [krótka pauza] albo dotychczas pracowałam w małych projektach, więc łatwo było, łatwo było wszystko ogarnąć. {wywiad 20}

Warto zwrócić uwagę na Databricks, które jedna z rozmówczyń określiła jako środowisko będące „przyjaznym miksem wielu narzędzi. Jest takim centrum dowodzenia" {wywiad 20}. Databricks to komercyjne narzędzie oferowane przez firmę Microsoft, „szybka i łatwa w obsłudze usługa analityczna do pracy zespołowej oparta na platformie Apache Spark":

> Uzyskuj szczegółowe informacje ze wszystkich swoich danych i twórz rozwiązania sztucznej inteligencji w usłudze Azure Databricks. Skonfiguruj środowisko Apache Spark™ w kilka minut, skaluj je automatycznie i pracuj nad wspólnymi projektami w interaktywnym obszarze roboczym. Usługa Azure Databricks obsługuje języki Python, Scala, R, Java i SQL, a także struktury i biblioteki nauki o danych takie jak TensorFlow, PyTorch i scikit-learn (Microsoft Azure, 2019).

W świecie DS praktyka stosowania tego rodzaju „przyjaznych miksów wielu narzędzi" jest rzadka. Należą do nich Databricks i DVC (narzędzie do kontroli wersji modeli i danych). Rozmówczynie mówiły o **testowaniu** tych rozwiązań, nie są to zatem rozwiązania wdrożone do systematycznej pracy polskich data scientistów. Można odnaleźć materiały porównujące zalety tego rodzaju platform, wymieniające np. Azure Databricks, Alteryx, IBM SPSS, KNIME, H2O.ai, RapidMiner, IBM Watson Studio, Dataiku Data Science Studio oraz Anaconda (Gualtieri i in., 2018; IT Central Station, 2019; Muenchen, 2019), jednak w świecie DS w Polsce poza opisywaną już Anacondą – która stanowi inną kategorię narzędzi niż pozostałe z wymienionych – żadne z tych narzędzi nie jest popularne, a część raczej nie byłaby uznana za narzędzia do data science[87].

Zestaw technologii używanych przez uczestników badanego świata, zarówno pracujących komercyjnie, jak i naukowo oraz hobbystycznie, zdecydowanie bardziej niż np. uporządkowane instrumentarium chirurga przypomina **warsztat majsterkowicza**. W tym warsztacie znajdują się narzędzia różnych firm, narzędzia stare i najnowsze, niestabilne wersje testowe, nierzadko narzędzia chałupniczo przerabiane, wykorzystywane niezgodnie z pierwotnym przeznaczeniem, narzędzia jakby posklejane w całość srebrną taśmą.

Zestaw technologii używany na różnych etapach i różnych poziomach projektu nazywany jest **stosem technologicznym** (*technological stack* lub *full-stack*). **Stos technologiczny jest podstawowym obszarem odniesienia w praktykach zaświadczania o autentyczności** w świecie społecznym DS. W odniesieniu do używanego przy realizacji projektów stosu technologicznego oceniana jest przynależność osoby do badanego świata, bycie prawdziwym uczestnikiem świata DS, oraz oferty pracy/współpracy w różnego rodzaju firmach/zespołach/projektach data science. Technologie są także podstawą w wyznaczaniu granic społecznego świata DS oraz jego subświatów, a więc rozeznanie w technologiach jest niezbędne do zrozumienia procesów zachodzących w społecznym świecie – głównie segmentacji i profesjonalizacji.

Na rysunku 4.15 przedstawiam szeroki stos technologiczny data science. Omawiane narzędzia pokazuję w relacji do działania podstawowego świata społecznego DS, nie zamieszczając na rysunku wszystkich omówionych technologii, a wskazując na przypadki typowe.

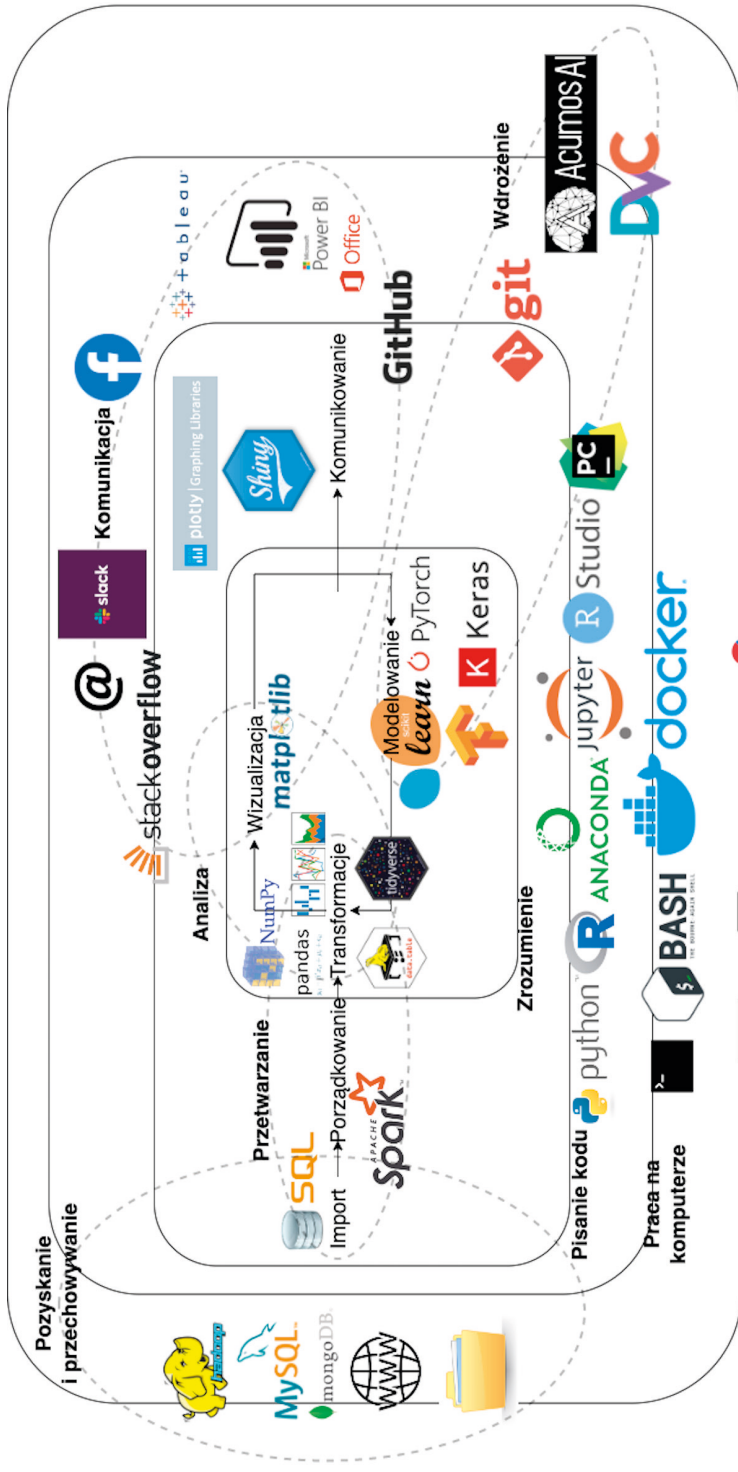
Uczestnicy mogą korzystać z różnych wycinków tego stosu technologicznego, szczególnie kiedy dotyczy to zakresu poza działaniem podstawowym badanego

87 Zbliżone rozwiązania to np. Orange, Caffe, WEKA, Rattle, które pozwalają wykonywać część analiz za pomocą GUI. Można wskazań także na narzędzia SAP i SAS Institute. W polskim świecie DS część z nich jest postrzegana jako narzędzia analityki biznesowej, tzw. BI, lub naukowej (te narzędzia to np. IBM SPSS, SAS, SAP, aplikacje Excel/Power BI firmy Microsoft), inne są używane marginalnie i nie traktuje się ich stosowania za właściwy sposób wykonania działania podstawowego, gdyż całość analiz nie jest lub nie musi być zapisywana w kodzie. Narzędzia takie jak Alteryx są nawet reklamowane jako narzędzia dla osób nieumiejących programować.

świata, czyli przechowywania i pozyskania danych, wdrożeń produkcyjnych oraz komunikacji. Przykładowo:

1) uczestnik pracujący naukowo i hobbystycznie może, działając wyłącznie na własnym laptopie, importować dane z plików (dane z własnych badań lub publicznych repozytoriów) oraz ściągać je z internetu metodami webscrapingowymi, używać języka R i IDE RStudio, przetwarzać, analizować i modelować dane za pomocą wybranych pakietów R (poza modelowaniem korzystając głównie z ekosystemu tidyverse), komunikować wyniki w artykułach naukowych i na meetupach, kontrolować wersje kodu przez Git i publikować je na GitHubie, korzystać ze źródeł wiedzy na Stack Overflow i komunikować się z innymi uczestnikami na Facebooku Data Science PL;
2) data scientista w zespole DS w dużej firmie zajmującej się ubezpieczeniami może pracować głównie na laptopie, a czasami, przy bardziej zasobochłonnych obliczeniach, łączyć się z firmowym serwerem analitycznym, używać głównie języka Python w dystrybucji Anaconda, pisać kod w notatniku Jupyter lub za pomocą IDE PyCharm, importować dane z firmowych serwerów – baz danych OLAP za pomocą języka SQL i odpowiednich pakietów pozwalających wykonać to w środowisku Pythona, przetwarzać, analizować i modelować dane głównie za pomocą pakietów „NumPy", „pandas", „matplotlib" i „scikit-learn", konteneryzować określone części projektów za pomocą Dockera, kontrolę wersji kodu prowadzić w SVN, komunikować wyniki za pomocą narzędzi pakietu MS Office (prezentacje PowerPoint, pliki Excela) oraz budować interaktywne raporty w MS Power BI, komunikować się w zespole na Slacku, z klientami wewnątrz firmy mailowo, nie brać udziału we wdrożeniach, ponieważ w tej firmie zespół DS ogranicza się do przekazywania plików .h5 z zapisem wytrenowanego modelu do działu IT, a po pracy zaliczać kolejne kursy MOOC i wykonywać hobbystycznie projekty z DL za pomocą biblioteki TF przy użyciu chmury Google'a i za pomocą Gita umieszczać kod na prywatnym GitHubie;
3) data scientista z kilkuletnim doświadczeniem w małej firmie/kilku firmach data science, które pracują dla różnych klientów w różnych branżach i krajach (tzw. start-upy lub firmy konsultingowe), może umieć posługiwać się każdym z narzędzi w prezentowanym na rysunku 4.15 stosie technologicznym i innymi technologiami, używając różnych narzędzi w różnych projektach, w zależności od specyfiki konkretnego zlecenia.

W tabeli 4.2 prezentuję opis dla każdego z wyróżnionych elementów stosu w podziale na czterdzieści trzy technologie i ich role.



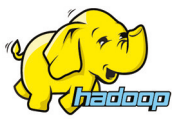
Rysunek 4.15. Stos technologiczny a działanie podstawowe DS

Źródło: opracowanie własne za pomocą draw.io, logotypy pobrane z oficjalnych stron WWW dostawców, inne rysunki na licencjach zezwalających na użycie.

Tabela 4.2. Stos technologiczny DS w podziale na technologie i ich role

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
	Komputer personalny	Laptop lub komputer stacjonarny pracujący pod kontrolą dowolnego systemu operacyjnego; własność firmy lub osoby fizycznej	Infrastruktura komputerowa
	Serwer wewnętrzny	Komputer w sieci lokalnej, bez połączenia przez internet; przeznaczony do wykonywania usług lub udostępniania zasobów dla innych komputerów w sieci, tzw. klientów; o większej niż komputer personalny mocy obliczeniowej i zasobach sprzętowych, przystosowany do nieprzerwanej pracy; raczej własność firmy	Infrastruktura komputerowa
	Amazon Web Services	Chmura obliczeniowa: duże serwery zewnętrzne z dostępem przez internet; wiele klastrów komputerowych, których zasoby i moc obliczeniową lub usługi na tych klastrach wypożycza się na żądanie od właściciela – firmy Amazon	Infrastruktura komputerowa
	Google Cloud Platform	Chmura obliczeniowa firmy Google	Infrastruktura komputerowa
	Microsoft Azure	Chmura obliczeniowa firmy Microsoft	Infrastruktura komputerowa
	Terminal	Program komputerowy służący do korzystania z innych programów i systemu operacyjnego za pomocą interfejsu tekstowego (a nie graficznego – GUI – jak np. w pracy biurowej); stosowane zamiennie są nazwy „powłoka” (*shell*), „linia poleceń” (*command line*), „konsola” (*console*), „bash”	Praca na komputerze
	Bash	Domyślna powłoka (*The Bourne-Again Shell*) systemów Linux/Unix, a także język programowania skryptowego, którego się w niej używa; *open source*	Praca na komputerze
	Docker	Technologia konteneryzacji umożliwiająca „zamrożenie” całego środowiska pracy wraz z używanymi w danej chwili wersjami narzędzi do ponownego wykorzystania; bezpłatna wersja *open source* i płatne wersje rozszerzone rozwija firma Docker Inc.	Praca na komputerze

Tabela 4.2 (cd.)

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
	System plików	Ogólnie metoda przechowywania plików na nośniku (np. dysku twardym, płytach CD/DVD) bez użycia systemu bazy danych; pliki takie mogą być źródłem danych dla zastosowań DS.	Pozyskiwanie i przechowywanie danych
	Internet/dane publiczne	Różne zasoby internetowe mogą być pozyskiwane do zastosowań DS za pomocą metod webscrapingowych, lub przez API; również dostęp do danych publicznych czy *open source* z użyciem internetu	Pozyskiwanie i przechowywanie danych
	MongoDB	System zarządzania nierelacyjną bazą danych w architekturze dokumentów; *open source* rozwijany przez firmę MongoDB, Inc. oferującą także usługi komercyjne, m.in. chmurę obliczeniową	Pozyskiwanie i przechowywanie danych
	MySQL	System zarządzania relacyjną bazą danych do przechowywania danych tabelarycznych, także do dokumentów; bezpłatna wersja *open source* i płatne wersje rozszerzone rozwija firma Oracle	Pozyskiwanie i przechowywanie danych
	Hadoop	System rozpraszania składowania danych i operacji na danych na klaster komputerowy, umożliwia wykonywanie wielu operacji równolegle; także cały ekosystem technologii (np. Hadoop MapReduce + Apache Pig + Apache Hive + Apache Spark); *open source* rozwijane przez Apache Software Foundation	Pozyskiwanie i przechowywanie danych
	Python	Język programowania skryptowego ogólnego zastosowania; najpopularniejszy język do zapisywania działania podstawowego świata społecznego DS; *open source* rozwijany przez organizację non profit Python Software Foundation	Pisanie kodu; **działanie podstawowe**
	R	Język programowania skryptowego przeznaczony do pracy z danymi (domenowy); drugi najpopularniejszy język do zapisywania działania podstawowego świata DS; *open source* rozwijany przez organizację non profit The R Foundation	Pisanie kodu; **działanie podstawowe**

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
	Anaconda	Dystrybucja języków Python i R, menadżer pakietów i środowisk Conda, kolekcja ponad 1500 pakietów dla DS, interfejs tekstowy i graficzny umożliwia m.in. łatwe korzystanie z notatnika Jupyter (dystrybucja głównie używana dla języka Python); bezpłatna wersja *open source* i płatne wersje rozszerzone rozwija firma Anaconda Inc. sponsorująca organizację non profit NumFOCUS	Pisanie kodu; **działanie podstawowe**
	Jupyter	Interaktywne środowisko obliczeniowe w formie notatnika w przeglądarce internetowej; pozwala na edytowanie i uruchamianie dokumentów analitycznych, które są przystosowane do czytania przez ludzi; może być uruchamiany lokalnie lub w chmurze; wspiera około 40 języków programowania (notatnik używany głównie dla języka Python, najpopularniejszy sposób pisania kodu w tym języku w DS); *open source*	Pisanie kodu; **działanie podstawowe**
	Rstudio	IDE (*integrated development environment*) mające wiele funkcji wspomagających powtarzalne czynności; IDE używane prawie wyłącznie do języka R (użycie Pythona jest możliwe), zdecydowanie najpopularniejszy sposób pisania kodu w R (nie tylko w DS); bezpłatną wersję *open source* i płatne wersje rozszerzone rozwija firma RStudio sponsorująca organizacje non profit NumFOCUS i Linux Foundation	Pisanie kodu; **działanie podstawowe**
	PyCharm	IDE dla języka Python używane powszechnie także poza DS; bezpłatną wersję *open source* i płatną wersję rozszerzoną rozwija firma JetBrains	Pisanie kodu; **działanie podstawowe**
	SQL	Język programowania (*Structure Query Language*, strukturalny język zapytań), w DS używany jest komponent DML (*Data Manipulation Language*, język operacji na danych), pozwala na import danych z bazy do środowiska pracy; możliwe jest także wprowadzanie, aktualizowanie i usuwanie danych; jest przeznaczony wyłącznie	Pisanie kodu/ pozyskiwanie i przechowywanie/przetwarzanie danych; **działanie podstawowe**

Tabela 4.2 (cd.)

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
		do pracy z bazami danych i na zbiorach danych, nie ma żadnego innego zastosowania; różne dystrybucje *open source* i komercyjne	
	Apache Spark	Zunifikowana technologia do rozproszonego przetwarzania danych; może wykonywać wiele rodzajów zadań, np. obsługa zapytań zbliżonych do SQL-owych, ML i przetwarzania danych w reprezentacji grafowej; *open source* rozwija Apache Software Foundation	Pozyskiwanie i przechowywanie/przetwarzanie danych; **działanie podstawowe**
	NumPy	Pakiet Pythona do pracy z danymi w formie tabelek oraz do obliczeń naukowych i inżynieryjnych; *open source* wspierane przez NumFOCUS	Przetwarzanie/analiza danych; **działanie podstawowe**
	pandas	Pakiet Pythona dostarczający struktur danych i narzędzi do ich analizy; *open source* wspierane m.in. przez NumFOCUS, RStudio, Anaconda	Przetwarzanie/analiza danych/komunikacja; **działanie podstawowe**
	data.table	Pakiet R o unikalnej konwencji zapisu, pozwala na wykonywanie szybkich manipulacji danymi do około 100 GB; *open source*	Przetwarzanie/analiza danych; **działanie podstawowe**
	tidyverse	Ekosystem pakietów R o spójnej konwencji zapisu; *open source* wspierane przez RStudio	Przetwarzanie/analiza danych/komunikacja; **działanie podstawowe**
	matplotlib	Pakiet do wizualizacji danych w języku Python; *open source* wspierane przez NumFOCUS	Analiza danych/komunikacja; **działanie podstawowe**
	scikit-learn	Pakiet metod tzw. tradycyjnego ML w języku Python; *open source* wspierane przez liczne uczelnie i firmy	Modelowanie; **działanie podstawowe**
	TensorFlow	Pakiet metod ML, w tym DL, głównie w języku Python; *open source* wspierane przez Google'a	Modelowanie; **działanie podstawowe**

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
	Keras	Nakładka wyższego poziomu, dostarczająca zestaw narzędzi do budowania modeli ML i DL w prostszy sposób; głównie jako zaplecze Kerasa (*backend*) używany jest TensorFlow; *open source* wspierane przez Google'a	Modelowanie; **działanie podstawowe**
	PyTorch	Pakiet metod ML, w tym DL, głównie w języku Python (zapewnia korzystanie z pakietu Torch dla języka Lua); *open source* wspierane przez Facebook	Modelowanie; **działanie podstawowe**
	plotly	Pakiet dla Pythona i R do interaktywnych wizualizacji danych; *open source* i komercyjne rozwiązania firmy Plotly	Komunikacja; **działanie podstawowe**
	Shiny	Pakiet R do interaktywnych wizualizacji danych; *open source* i komercyjne rozwiązania firmy RStudio	Komunikacja; **działanie podstawowe**
	Tableau	Oprogramowanie zamknięte (*proprietary software*), aplikacja do interaktywnych wizualizacji danych w interfejsie graficznym; wersje darmowe i płatne firmy Tableau Software	Komunikacja
	MS Power BI	Oprogramowanie zamknięte, aplikacja do interaktywnych wizualizacji danych w interfejsie graficznym; możliwe używanie języków programowania, m.in. SQL, Python, R; wersje darmowe i płatne firmy Microsoft	Komunikacja
	MS Office/ MS 365	Oprogramowanie zamknięte, pakiet aplikacji do prac biurowych; do komunikacji świata DS z klientami głównie Excel (arkusze kalkulacyjne) i PowerPoint (prezentacje multimedialne); wersje darmowe i płatne firmy Microsoft	Komunikacja
	e-mail	Poczta elektroniczna w dowolnej domenie, stosowana głównie do komunikacji świata DS z klientami lub współpracownikami	Komunikacja
	Facebook	Popularny serwis społecznościowy ogólnego przeznaczenia; w DS używany m.in. do komunikacji w grupach uczestników świata DS, także przez osoby chcące wejść do świata	Komunikacja

Tabela 4.2 (cd.)

Symbol	Nazwa	Opis	Rola w stosie technologicznym DS
	Slack	Aplikacja do pracy grupowej, platforma do wysyłania wiadomości, plików i połączenia z innymi narzędziami; oprogramowanie zamknięte; wersje darmowe i płatne firmy Slack Technologies	Komunikacja
	Stack Overflow	Publiczne forum internetowe przeznaczone do zadawania pytań i udzielania odpowiedzi związanych z pisaniem kodu w różnych językach programowania (używane szeroko także poza DS)	Komunikacja
	GitHub	Internetowe repozytorium kodu połączone z systemem kontroli wersji Git; funkcje uznawania autorstwa i pracy grupowej, funkcje społecznościowe (np. obserwowanie, wyrażanie uznania); funkcje publikacyjne (np. licencje, opisy, strony WWW, prezentacje multimedialne); publiczne repozytoria darmowe, prywatne płatne; należy do firmy Microsoft	Komunikacja/ pisanie kodu
	Git	Zdecentralizowany system kontroli wersji kodu; służy do zarządzania i śledzenia ewolucji całego zbioru plików cyfrowych, które składają się na projekt; używany szeroko w tworzeniu oprogramowania; obsługa przez interfejs tekstowy, w innych narzędziach (np. RStudio) także interfejs graficzny; *open source* wspierane przez firmę non profit Software Freedom Conservancy	Komunikacja/ pisanie kodu/ wdrożenie
	Acumos	Platforma do konteneryzacji (w połączeniu z Docker) określonych wersji modeli wraz ze środowiskiem obliczeniowym; *open source* wspierane przez Linux Foundation	Wdrożenie
	DVC	Platforma do kontroli wersji zbiorów danych i modeli oraz śledzenia projektu (*data science version control*); *open source* wspierane przez firmę Iterative.ai	Komunikacja/ wdrożenie

Źródło: opracowanie własne.

4.4. Wartości społecznego świata data science

Wartości w koncepcji społecznych światów to:

> […] orientacyjne punkty pozwalające na legitymizację danego działania, a jednocześnie na dokonanie oszacowania jego autentyczności i poprawności z punktu widzenia „interesów" danego społecznego świata lub jego segmentu. […]. Wartości ujawniają się w działaniu, gdy ludzie podejmują decyzje co do swego postępowania (Kacperczyk, 2016: 44, 47).

Opisuję siedem **wartości społecznego świata DS: efektywność, samodzielność, racjonalność, sprawczość, swobodę, samorozwój, ciekawość**.

Działania w badanym świecie są oceniane, legitymizowane, uznawane za autentyczne i poprawne zawsze w odniesieniu do **centralnej kategorii efektywności**.

Wyróżnione wartości łączą się ze sobą, są wzajemnie powiązane i wymieszane. Poniższe części książki mają organizować wywód, ale nie oznaczają rozłączności tych siedmiu wartości. Kategorie i kody będą powtarzały się, ujęte z różnej perspektywy. Pojawiające się rzadko odniesienia do literatury są wynikiem przeglądu dokonanego po opracowaniu opisywanych wartości świata DS na podstawie własnych badań terenowych.

4.4.1. Efektywność

Peter Norvig, nawiązując do określenia DS jako najbardziej seksownego zawodu świata, wskazał, że „ostatecznie seksowność sprowadza się do efektywności" (Gutierrez, 2014: viii–ix).

Efektywność jest centralną wartością badanego świata. Cenione jest takie działanie, które przynosi jak największy czy najlepszy wynik na jednostkę wkładu, traktowanego jako koszt (efektywność = wynik / wkład) oraz takie, którego wynik lub koszt jest uznany za odpowiedni w relacji do założonego rezultatu działania.

Efektywność związana jest nierozerwalnie z racjonalnym podejściem do działania i nawet jeśli nie jest precyzyjnie skwantyfikowana, to uczestnicy badanego świata chętnie posługują się w jej określaniu pojęciami ilościowymi, takimi jak „około trzech razy szybciej", „zrealizowaliśmy dzięki temu ponad 80% projektu". Używane są ilościowe wskaźniki, np. w wizualizacji danych za Edwardem Tufte mówi się o relacji ilości zaprezentowanych danych do ilości użytej grafiki – im więcej, tym lepiej (*data-ink ratio*, dosł. 'stosunek danych do tuszu') (por. Tufte, 1983; Inbar, Tractinsky, Meyer, 2007) lub o współczynniku kłamstwa (*lie factor*) – dla poprawnej wizualizacji wynosi on 1 i jest stosunkiem informacji prezentowanej graficznie do liczb, które grafika ma przedstawiać (Biecek, 2018d). Istnieje także wiele wskaźników służących do oceny modeli ML.

Efektywność może być mierzona i oceniana wtedy, kiedy uzyskano jakiś wyniki i zakładano jakiś rezultat, **w badanym świecie cenione są zatem przede**

wszystkim działanie zrealizowane oraz działanie celowe. Przykładem tego nastawienia jest traktowane przeze mnie jako kod *in vivo* powiedzenie „na koniec dnia" (będące kalką z *at the end of the day*). Tym samym efektywność jest nierozerwalnie związana ze sprawczością oraz racjonalnością. Przykładowo: działanie, który postronny obserwator mógłby określić mianem „bawienia się" nowymi narzędziami (np. pakietami dla Pythona), w świecie DS będzie legitymizowane jako „testowanie" nowych narzędzi, czyli jednocześnie bycie na bieżąco z pojawiającymi się nowinkami i ocenianie tych nowinek w kategoriach przydatności do poprawy swojego działania.

Iteracyjne podejście do wykonywania działania podstawowego sprzyja szybkiemu uzyskiwaniu wyników w realizacji jego elementów, co **uważane jest za sprzyjające efektywności**, wyniki służą bowiem do podejmowania bieżących decyzji o kierunku dalszych działań. Tym samym jakoby nie traci się czasu na długotrwałe dążenie w niewłaściwym kierunku, bo już na wczesnym etapie po uzyskaniu nieobiecującego wyniku zmienia się kierunek działania. Przypomina to nastawienie na szybkie i częste porażki (*fail fast, fail often*), które jest nazywane mantrą firm technologicznych z Doliny Krzemowej. To nastawienie ma być elementem mitologii amerykańskiego ducha przedsiębiorczości, gdzie porażki poprzedzające ostateczny sukces są traktowane jako powód do dumy, jako wskazujące na ryzykowność przedsięwzięcia (czyli odwagę podmiotu podejmującego to przedsięwzięcie), nierozerwalnie związaną z wysoko cenioną innowacją. Ikonami tak pojętego sukcesu są np. Thomas Edison, Henry Ford czy Steve Jobs (Draper, 2017). Co do zasady to na iteracyjnym wykonywaniu operacji i korygowaniu błędów w celu optymalizacji wskazanego parametru bazują metody nadzorowanego ML. Początkowo wydawało mi się zatem, że ML jest implementacją tego amerykańskiego ducha przedsiębiorczości, a świat DS jest, zgodnie z owym duchem, nastawiony na osiąganie sukcesów w procesie odnoszenia szybkich i częstych porażek. To przesadne uproszczenie.

Iteracyjna praca i poprawianie błędów to sposób działania uznawany w świecie DS za właściwy, ale ceniona jest efektywność i racjonalność, a nie ryzykowanie. **W świecie DS nie tylko nie ceni się ryzyka, ale też uczestnicy rzadko ryzykują utratę czegokolwiek innego niż czas.** Tak więc choć sprawczość, szczególnie rozumiana jako wpływanie na coś realnego, praktycznego, np. koszty określonego procesu w organizacji, jest w badanym świecie wysoko ceniona, to znakomita część pracy osób zajmujących się DS polega na operowaniu na danych w cyfrowym laboratorium, gdzie ryzyka inne niż strata czasu są bardzo małe. Takie eksperymenty mogą być kończone na różnych etapach zaawansowania, nie uzyskując wyniku praktycznego, tj. nie dochodząc do etapu wdrożenia. W laboratorium, w dużej mierze dzięki zapisywaniu operacji w kodzie, prawie wszystko jest powtarzalne, odwracalne, możliwe do zmiany. Z punktu widzenia osób zajmujących się DS i niebędących jednocześnie odpowiedzialnymi za budżet projektów (jak np. właściciele firm, kierownicy zespołów) nawet czas przeznaczony na „porażki"

projektów (czyli brak wdrożenia) może być uważany za cenny z uwagi na indywidualny samorozwój, zdobyte doświadczenie itp., a nie za bezsensowny koszt.

Pozostając przy równaniu „efektywność = wynik / wkład", w świecie DS **najczęściej w mianowniku jako wkład występuje czas** przeznaczony na osiągnięcie wyniku. Nie spotkałem się z praktyką negatywnej oceny żadnego działania jako wykonanego za szybko. Jeśli tylko rezultat jest oceniony jako odpowiedni, to mniej włożonego w realizację czasu może wyłącznie tę ocenę poprawić. Z uwagi na przekonanie o szybkim rozwoju DS i technologii cyfrowych w ogóle **istnieje duża presja do działania jak najefektywniejsze w kontekście poświęconego czasu**. Bardzo trudno byłoby legitymizować uzyskanie dokładnie tego samego rezultatu wolniej, niż to możliwe. Aby taka legitymizacja była skuteczna, mogłaby zostać sformułowana poprzez odniesienie do efektywności w konkretnej sytuacji, np. wykonałem coś wolniej za pomocą funkcji x z pakietu A, wiem o istnieniu pakietu B i że w nim inaczej zaimplementowana funkcja x działa o rząd wielkości szybciej, ale nie znam pakietu B tak dobrze jak A, czas poświęcony na użycie B przeze mnie przewyższyłby zatem czas zaoszczędzony dzięki szybszemu działaniu funkcji x w B niż A. Podsumowując – wybrałem rozwiązanie bardziej efektywne w danym kontekście.

Do **efektywności lub bezpośrednio do szybkości odwołuje się wartościowanie właściwie wszystkich opisywanych technologii świata DS**. Dotyczy to też programowania jako takiego:

> Współcześnie programowanie umożliwia wykonywanie trylionów (10^{18}) operacji arytmetycznych na sekundę. Pozwala to na znaczne zwiększenie wydajności dostępnych rozwiązań, otwiera możliwość praktycznego wykorzystania istniejących idei, lub też tworzenia nowych pomysłów (Nowosad, 2019: 12).

Dotyczy to również systemów bazujących na analizie danych czy ML. Powstałe w latach pięćdziesiątych pierwsze komercyjne systemy oceny zdolności kredytowej klienta (*credit scoring*) – Karol Przanowski (2014: 13–15) uważa, że są one „prekursorami" współczesnych modeli w DS – były budowane pod hasłem efektywności, optymalizacji wybranych aspektów procesu oceny „szybciej, taniej i obiektywniej" (Przanowski, 2014: 18).

Areny dotyczące sporów pomiędzy narzędziami konkurencyjnymi, takimi jak języki Python i R, ale także dowolnymi technologiami software'owymi czy hardware'owymi, są zawsze – choć nie wyłącznie – sporami o efektywność. Uczestnicy świata DS, nie tylko w Polsce, opracowują niezliczoną ilość tzw. benchmarków nastawionych na pomiar szybkości działania narzędzi (np. White, 2012; Fierro, Salvaris, Wu, 2017; BrodieG, 2018; Carneiro i in., 2018; Gorecki, Yuan, 2019), istnieją także pakiety do mierzenia czasu działania kodu (Wickham, 2018b; Chang, Luraschi, Mastny, 2019). W sporach o to, które narzędzie jest lepsze, uczestnicy mogą różnie definiować efektywność. Mogą podawać argumenty dotyczące nie tylko porównywania szybkości działania kodu dla tego samego zadania,

mierzonej w sekundach, ale także np. liczby linii kodu (czas pisania), dostępności pakietów/funkcji (także czas pisania, bo szybciej korzysta się z gotowej funkcji, niż pisze ją samodzielnie), czytelności kodu (łatwość i czas poprawiania błędów w kodzie, wdrożenia nowej osoby do projektu, łatwość ponownego wykorzystania kodu) czy nierzadko poczucia wygody lub łatwości pracy (pracuje mi się wygodniej w tym języku/pakiecie/IDE/systemie operacyjnym, a więc pracuję efektywniej). Pokazałem już przykłady narzędzi, które poprzez ułatwienia mają czynić pracę wygodniejszą, a w konsekwencji bardziej efektywną, np. dystrybucja Anaconda, IDE RStudio, pakiet-nakładka na Tensor Flow, czyli Keras. Korzystanie z terminala to przykład pozbycia się nakładki – czyli interfejsu graficznego – w celu poprawy efektywności oraz zwiększenia sprawczości i samodzielności człowieka w pracy na komputerze.

Samo uznanie, że jedynym prawidłowym sposobem wykonywania działania podstawowego jest zapisywanie operacji na danych w kodzie, jest wyrazem traktowania efektywności jako wartości w świecie DS. **Data science po prostu nie da się robić, klikając** (*you can't do data science in a GUI*) (Wickham, 2018d). Znany z ilościowych badań naukowych postulat powtarzalności wyniku (*reproducibility*) realizowany jest właśnie za pomocą kodu – można go uruchomić dowolną liczbę razy i uzyskać ponownie wynik zapisanych czynności. Kod jest możliwy do odczytania (*readable*), przeglądając zatem czyjś kod bez uruchamiania, można zrozumieć jego tok myślenia i operacji. Dzięki zapisowi w kodzie można śledzić zmiany i porównywać kolejne wersje zapisu (*diffable*), zobaczyć proces ewolucji projektu. Wickham wskazuje, że klikanie w ogóle zwiększa ryzyko pomyłek w procesie operacji na danych, szczególnie pomyłek niezauważonych. Sam, używając Excela, zwyczajnie boi się kliknąć w złą opcję, szczególnie że trudno potem odtworzyć to, co dokładnie się stało, nie wiadomo, co naprawiać (Wickham, 2018d). Przy pisaniu kodu taki problem nie występuje. Proces jest zapisany, człowiek widzi historię operacji, a jeśli dochodzi do pomyłki, skasowania części kodu, nadpisania zmiennej czy zagubienia się w skrypcie, wystarczy uruchomić odpowiednie linie kodu do powtórzenia ostatniego etapu wykonanego zgodnie z zamierzeniem. Pisanie kodu w porównaniu do klikania sprzyja zatem efektywności na wiele sposobów.

Klikanie się nie skaluje (Mason, 2010), a pisanie kodu tak. Co to oznacza? Jeżeli potrzeba np. pobrać jeden plik ze strony WWW, prawdopodobnie wykonanie kilku kliknięć będzie szybsze niż napisanie skryptu do ściągania pliku. Jeżeli jednak chce się pobrać sto plików i będzie się robić to codziennie przez miesiąc, klikanie będzie powtarzalnym, czasochłonnym wysiłkiem. Gdy sytuacja zmieni się i będzie trzeba pobierać tysiąc plików pięć razy dziennie, w skrypcie wystarczy zmiana odpowiednich parametrów. Tak więc dla rosnącej liczby zadań czy ilości danych kod będzie „skalować się", tzn. działać równie efektywnie (przy ograniczeniach technicznych) i nie wymagać dodatkowej pracy ludzkiej, a w przypadku klikania więcej zadań będzie zawsze oznaczało więcej pracy, a także zwiększy ryzyko ludzkich pomyłek, frustrację klikającego, jego zmęczenie itd.

Postulat powtarzalności wyniku (*reproducibility*) został przez świat DS zaczerpnięty z nauk przyrodniczych i przedefiniowany. Powtarzalność nie ma zapewniać – jak w nauce – gromadzenia dowodów na wsparcie albo odrzucenie hipotezy, by wyeliminować fałszywe roszczenia do nowych odkryć i strzec dyscypliny działalności naukowej (Peng, 2011b: 1126). Powtarzalność w DS sprzyja efektywności (Henderson i in., 2017; Wickham, Grolemund, 2017, Bryan i in., 2019). To przedefiniowanie jest przykładem „odgradzania się" od innych światów (Kacperczyk, 2016: 48), niemniej DS odgradza się od świata nauki tylko częściowo, częściowo powołuje się zaś na naukowe korzenie, odgradzając się od zwykłej działalności inżynierskiej, jak przez uczestników świata DS przedstawiane bywa np. tworzenie oprogramowania. Stąd mowa o DS jako działalności paranaukowej, czyli takiej, gdzie stosowane są metody naukowe lub zbliżone do naukowych, ale z nastawieniem na użyteczność i efektywność oraz sprawczość, a nie aspekty poznawcze i prawdziwość. Zdaniem badającego rosyjskich data scientistów Lowriego w inżynierii dominuje kryterium efektywności, natomiast w nauce kryterium prawdy. Twierdzi on, że choć DS posługuje się tylko kryterium efektywności, szczególnie w kontekście oceny działania modeli ML, to zmieniło ono technologiczne kryterium efektywności w standard epistemologiczny (Lowrie, 2017: 3–9). Te rozważania są słuszne w odniesieniu do odróżnienia DS od nauk przyrodniczych, co widać chociażby w opisanym wyżej rozróżnieniu w rozumieniu celu osiągania powtarzalnych wyników. Łatwo byłoby przyporządkować DS do kultury modelowania algorytmicznego (w której chodzi o wynik), a nauki przyrodnicze do kultury modelowania generatywnego (w której chodzi o wyjaśnianie) (Breiman, 2001; Donoho, 2015). Niemniej granica jest rozmyta, a w samym DS mogą być preferowane metody wyjaśnialne (np. regresja logistyczna, drzewa decyzyjne, różne metody modelowania statystycznego, metody optymalizacyjne znane z badań operacyjnych) w przypadku określonych problemów lub specyfiki klienta (np. branża finansowa, choć niekoniecznie działanie w niej będzie uznawane za DS). Istnieje także wiele metod służących wyjaśnianiu działania złożonych modeli ML[88], także w celu przeciwdziałania możliwym zarzutom prawnym lub etycznym o nieprzejrzyste i dyskryminujące decyzje podejmowane z ich użyciem. Jednak dominującą pozycją w świecie DS jest wartościowanie efektywności (i sprawczości) działań w opozycji do nauki akademickiej – zarówno nauk przyrodniczych, jak i ścisłych lub technicznych, tzn. akademickiej matematyki i informatyki. Data science bywa w tym kontekście przedstawiane jako działania, które nie są sztuką dla sztuki, a rozwiązywaniem rzeczywistych problemów, dostarczaniem rzeczywistej wartości biznesowej, pracą na prawdziwych, brudnych zbiorach danych, robieniem czegoś, co naprawdę działa.

88 To pojęcie to tzw. XAI (*eXplainable AI*), aktywny na tym polu jest znany w Polsce data scientista i naukowiec (matematyk) Przemysław Biecek.

Efektywność jako centralną wartość widać wyraźnie w czterech używanych w badanym świecie pojęciach: „MVP", „POC", „działa/nie działa" i „dowożenie". Są to kody *in vivo* (Charmaz, 2009: 76–79).

I tak **MVP** (*minimum viable product*) to produkt minimalnie gotowy do wprowadzenia na rynek, a w przypadku DS gotowy do wdrożenia produkcyjnego. Jest to produkt zrealizowany minimalnym wkładem, ale działający, prototyp gotowy dla pierwszego klienta:

> Nowo powstały zespół DS za poleceniem managera zdecydował się użyć bazy NoSQL, pomimo że nikt w zespole nie miał doświadczenia z taką technologią. Zbudowali model, potem podjęli próbę skalowania swojej aplikacji. Jednak ponieważ próbowali skalować produkt za pomocą nieodpowiedniej do tego celu technologii [NoSQL – przyp. R.Ż.], nigdy nie dostarczyli (*deliver*) go klientom. Kierownictwo firmy nigdy nie ujrzało zwrotów z tytułu inwestycji w ten projekt i doszło do wniosku, że inwestowanie w produkty bazujące na danych było zbyt ryzykowne i nieprzewidywalne. Gdyby zespół DS zaczął od MVP, nie tylko mogliby zdiagnozować problemy z samym modelem, ale także przejść na tańszą, bardziej odpowiednią technologię [bazodanową – przyp. R.Ż.] i zaoszczędzić pieniądze (Prendki, 2018).

Takie MVP mogą być testowane już nie w laboratorium DS, czyli niemal zawsze na danych historycznych, ale w środowisku będącym wycinkiem środowiska produkcyjnego, tzw. piaskownicy (*sandbox*) {wywiad 15}. Testy i ich kolejne iteracje wskazują na potencjalne kierunki zmian w produkcie – może dotyczyć to jakości danych, ścieżki przepływu danych (*pipeline*), przechowywania danych, przygotowania danych, etykietowania danych, mocy obliczeniowej sprzętu komputerowego, modelowania czy wdrożenia – a więc testowanie MVP prowadzić ma do bardziej efektywnego wykorzystania zasobów (np. pieniędzy, ludzi, sprzętu, czasu) na poziomie zespołu DS czy całej firmy, pozwala bowiem przy relatywnie małym koszcie sprawdzić produkcyjną użyteczność wypracowywanego rozwiązania. Jeden z rozmówców {wywiad 15} akcentował to, że testy w „piaskownicy" to sposób na zmniejszanie konsekwencji ewentualnej porażki produktu, co jest zmniejszaniem mianownika w równaniu efektywności.

Minimum viable product to zaawansowane stadium projektu DS. Zespół DS w większej organizacji może odpowiadać za monitorowanie testów MVP w „piaskownicy" i w porozumieniu z klientem wewnętrznym oraz ewentualnie zespołem IT ulepszać produkt w celu wykonania wdrożenia produkcyjnego. Mała firma DS, tzw. start-up lub firma konsultingowa, może – w zależności od konkretnego projektu – w ogóle nie brać udziału w takich działaniach, a osoby zajmujące się DS głównie naukowo czy hobbystycznie nie mają z nimi styczności. Warto przypomnieć, że „w data science wypróbowujesz 10 pomysłów, jeden działa bardzo dobrze, a reszta – nie działa wcale" (Vazquez, 2017). Data science to przede wszystkim eksperymenty, eksploracja – uczestniczenie w projekcie dochodzącym do etapu MVP czy do wdrożenia nie jest warunkiem bycia uznanym za uczestnika badanego świata, choć udane wdrożenia w portfolio będą podnosiły prestiż uczestnika świata i jego wartość na rynku pracy.

Pojęcie **„POC"** (*Proof of Concept*) jest używane zamiennie z prototypem lub MVP, jednak prototyp i MVP są rozumiane raczej jako rozwiązania nastawione na testowanie produktu przez klienta, a POC to rozwiązanie przeznaczone do testów technicznych, wewnętrznych, raczej bez udziału klienta. *Proof of Concept* może być wcześniejszym niż MVP etapem projektu DS, to właściwie komunikowanie wyników w celu uzyskania akceptacji osób decyzyjnych odnośnie do dalszych kroków w kierunku gotowego produktu, wdrożenia produkcyjnego.

> *B: Okey, i to jest koniec projektu? W tym, na ten moment?*
> R: Nie, nie, to, to miała być taka pierwsza, pierwsza faza, pierwszy POC, żeby tam też chodzi o to, generalnie chodzi o to, żeby te wytyczne sprzedać, zespół, że jesteśmy świeżym zespołem, dopiero co zbudowanym, i też chodzi o to, żeby się pokazać.
> *B: Jasne.*
> R: I to chyba się udało. Z drugiej strony to był POC, który robiliśmy na niewielkim obszarze w [nazwa kraju], gdyż kolejnym krokiem ma być już rozszerzenie całej analizy do całego obszaru [nazwa kraju].
> *B: Mhm, OK. A mm, a czy jest jakiś plan, co ma być takim, mm, ostatecznym zakończeniem tego projektu? Takim ostatecznym, ostatecznym produktem?*
> R: Mhm, takim ostatecznym produktem, który jest oczywiście poza nami, no to jest tak naprawdę plan rozwoju sieci, czyli w jakim miejscu i jakie te maszty, jakie anteny poustawiać. {wywiad 18}

> R: POC – poof of concept, czyli nie zakładasz, że rozwiązanie będzie produkcyjne, tylko robisz taką próbę, aby sprawdzić, czy się da, czy wyniki są obiecujące {konsultacja na LinkedIn z R po wywiadzie 18}

Pojęcia „MVP", „POC" i „prototyp" są zaczerpnięte z branży programistycznej i pomimo różnic pomiędzy nimi uważam je za wskazujące na praktyki, dzięki którym projekty stają się bardziej elastyczne, co prowadzić ma do unikania sytuacji brnięcia w rozwiązanie nieprzekładające się na zwrot inwestycji, czyli są to praktyki nastawione na efektywność – zmniejszanie kosztów i zwiększanie zysków.

Pojęcie **„działa/nie działa"** odnoszone jest zawsze do kodu – kod musi działać. Może nie działać dokładnie w sposób zamierzony, jednak musi działać w tym sensie, że zwraca jakiś wynik. Poza efektywnością, która w tym wypadku oznacza nastawienie na działanie celowe (na efekty), omawiana kategoria jest związana z wartościami sprawczości i samodzielności. Kod, który nie działa, tzn. nie zwraca wyniku, a tylko komunikat o błędzie, musi zostać naprawiony. **Nie ma znaczenia perfekcja** – znaczenie ma efektywność. Także w tym kontekście uczestnicy świata DS powołują się np. na zasadę 80/20. Jedną z jej odmian jest przekonanie, że 20% czasu poświęconego na najważniejsze elementy projektu przekłada się na 80% wartości uzyskanej w projekcie, a pozostałe 80% czasu daje tylko 20% wartości.

Prowadzi to do akceptowania niedoskonałości, legitymizowanych jako optymalizacja wkładu do rezultatu działania. Jeżeli zespół DS przygotował model, który ma dopasowanie (*accuracy*) np. 90%, to decyzja o uznaniu go za wystarczająco dobry i przejście do budowania prototypu może być uznana za optymalną w tym

sensie, że bardzo dużo czasu należałoby poświęcić na uzyskanie dokładności wyższej o kilka punktów procentowych, w tym samym czasie można by zaś przedstawić klientowi pierwszą wersję prototypu, uzyskać informację zwrotną i wykonać poprawki (które mogą dotyczyć czegoś zupełnie innego niż dokładność działania modelu, a ważnego z punktu widzenia klienta).

> Nie trzeba być super data cruncher, żeby komuś pokazać jakieś rekomendacje, które zmienią jego myślenie. Świat czeka na szybkie odpowiedzi, nie perfekcyjne, ale pozwalające działać i zbliżyć się do celu […]. Czasem konsultujemy firmy, które mają dział data science całkowicie odcięty od reszty firmy i nie daje żadnej wartości. {notatka terenowa, obserwacja 26}

W tym kontekście część uczestników badanego świata uważa za bezwartościowe branie udziału w konkursach modelowania na platformie Kaggle. Optymalizowana jest w nich wyłącznie techniczna miara dokładności modelu, a o wygranej decydują minimalne różnice, uznawane za nieznaczące w komercyjnym DS. Osoby biorące udział w konkursach cyzelują model, poprawiając jego wyniki o ułamki punktów procentowych, czego nie praktykuje się w komercyjnym DS. Konkursy te polegają niemal wyłącznie na modelowaniu, ponieważ – po pierwsze – dane do konkursu są przygotowane[89] (uznawana za żmudną i nudą, acz ważną, faza pozyskania i wyczyszczenia danych jest ograniczona do minimum), po drugie – osoby budujące model, nawet jeżeli wygrają i uzyskają nagrodę pieniężną, to nie uzyskują sprawczości, oddają bowiem organizatorowi zbudowany model i nie mają związku z jego ewentualnym wdrożeniem.

Konkursy Kaggle mogą być uznawane także za coś nieciekawego, bo znacznie łatwiejszego niż rozwiązywanie realnego problemu biznesowego. „Granie w Kaggle" bywa uznawane za stratę czasu, szczególnie przez silnie osadzonych uczestników świata DS (Brzeziński, 2018).

To, że nie ma znaczenia perfekcja, widać także na mikropoziomie działania, czyli w sposobie pisania kodu. Uczestnicy świata DS nie oceniają negatywnie praktyki interaktywnego uruchamiania kodu, poprawiania, sprawdzania dokumentacji do konkretnego pakietu czy sprawdzania komunikatu o błędzie na Stack Overflow, o ile odbywa się to szybko i skutkuje naprawieniem kodu. Mając działający kawałek kodu, ale nie do końca działający zgodnie z intencją uczestnika[90], wypada zadać pytanie na Stack Overflow, ewentualnie skonsultować się z koleżanką/kolegą. O pomoc z niedziałającym kodem wypada zwrócić się tylko w trakcie warsztatów z kodowaniem na żywo, gdyż zgłoszenie problemów osobom prowadzącym uważane jest wówczas za bardziej efektywne z perspektywy interesu uczestnika

89 Etap w projekcie DS, kiedy dane są dobrze przygotowane i można zacząć „bawić się" modelowaniem, nazywany jest czasem „Kaggle moment" {obserwacja 46}.

90 Może to także wskazywać na wykrycie błędu w cudzym kodzie, najczęściej będzie to autor pakietu. Wówczas, gdy osoba upewni się co do powtarzalności błędu i przygotuje działający przykład, wypada zgłosić błąd, np. jako issue na GitHubie. Takie działanie może być elementem budowania swojej reputacji jako osoby umiejącej programować.

warsztatu niż samodzielne szukanie odpowiedzi. Niemniej jeżeli będzie to pytanie o czynność uznawaną za elementarną, a nie za będącą treścią warsztatów, może być ono ocenione przez innych jako brak samodzielności, a także niepotrzebnie zajmujące czas prowadzących.

W kategoriach działania/niedziałania mówi się także o modelach ML i innych metodach analitycznych i statystycznych. I znów muszą one w ogóle zadziałać – jest to tożsame z działaniem kodu, za pomocą którego je zapisano – ale też powinny „działać dobrze". Dobre działanie jest w tym przypadku równoważne z uzyskiwaniem wyniku bliskiego zaplanowanemu rezultatowi. Istnieją różne podejścia do definiowania tego, co jest rezultatem. W kontekście modeli ML istnieje wiele ilościowych metryk ewaluacji modeli – w nadzorowanym ML najprostsze będzie dopasowanie odpowiedzi modelu do wartości zmiennej zależnej (*accuracy*), wyrażone jako stosunek wszystkich odpowiedzi do odpowiedzi poprawnych (czyli zgodnych z wartościami zmiennej zależnej), np. 0,9 lub 90% – i część uczestników badanego świata będzie koncentrować się właśnie na nich, uznając za pożądany rezultat np. *accuracy* wyższe od ustalonej wartości. W zadaniach klasyfikacji stosowane są bardziej szczegółowe miary, pozwalające ocenić różnego rodzaju niezgodności odpowiedzi modelu z wartościami zmiennej zależnej[91]. Cześć uczestników będzie nazywała takie podejście najprostszym, a jednocześnie nieużytecznym i będzie oceniała poprawność działania modelu także jako użyteczność odpowiedzi modelu dla rezultatu w kategoriach osiągnięcia celu o charakterze biznesowym (np. zmniejszenia czasu odpowiedzi na reklamację klienta, zwiększenia liczby płatnych subskrypcji wśród użytkowników w serwisie streamingowym) oraz w kontekście całego projektu. Te dwie skrajne strategie definiowania rezultatów działania modeli to przeciwstawne pozycje na arenie dotyczącej tego, czy uczestnicy realizują działanie podstawowe, żeby rozwiązać problem bądź zrobić model.

Dobrze działająca wizualizacja lub inna metoda analizy danych to taka, która pozwala na wgląd w dane, pokazuje zależności pomiędzy zmiennymi, ewentualnie komunikuje coś w sposób zrozumiały, pozwala na skonsumowanie wyniku przez odbiorcę. Część uczestników świata DS zwraca dużą uwagę na fizjologiczne i psychologiczne mechanizmy percepcji informacji podanych w formie graficznej, część stosuje różne, także zaawansowane metody statystycznego testowania hipotez, wielu jest na bieżąco z najnowszymi publikacjami (nierzadko preprintami) w dziedzinie ML. Dobrze działające są zatem metody nie tylko szybkie, ale i niekoniecznie szybkie, za to odpowiednio dopasowane do problemu – w sensie biznesowym i metodologicznym.

---

91 Typowe miary dla klasyfikacji binarnej to specyficzność (*specificity*), precyzja (*precission*) oraz czułość (*sensitivity* lub *recall*). Dla klasyfikacji binarnej i wieloklasowej stosuje się także metrykę F-1 (w podstawowej postaci jest ona średnią harmoniczną precyzji i czułości). Dla różnych modeli ML stosowana jest również krzywa ROC (*Receiver Operating Characteristics*) i obliczana na jej podstawie powierzchnia pod krzywą (*area under the curve* – AUC) (por. Powers, 2007).

Popełnianie i naprawianie błędów, szczególnie technicznych[92], ale także metodologicznych, jest uznawane za nieusuwalny element wykonywania działania podstawowego. Iteracyjna praca metodą prób i błędów będzie uznawana za normalną dopiero od poziomu umiejętności uznawanych za elementarne. Mylenie się przy napisaniu jednej linii kodu przy podstawowych funkcjach z pakietów, np. NumPy, pandas czy dplyr, będzie jasnym sygnałem niskich kwalifikacji – podobnie kłopoty z wykonaniem modelu regresji logistycznej czy drzewa decyzyjnego.

W badaniach psychologicznych i pedagogicznych popełnianie błędów, doświadczanie tzw. konstruktywnych porażek (*productive failure*) wskazywano jako to, co sprzyja nauce konceptów matematycznych i rozwiązywaniu problemów matematycznych (Kapur, Rummel, 2012). Moim zdaniem można to odnieść nie tylko do efektywności, ale także do innych wartości świata DS – samodzielności, sprawczości i swobody. W badaniach wskazywano, iż poczucie lęku (*anxiety*) przed matematyką u nastolatków jest związane z poczuciem bezsilności wobec rozwiązania problemu matematycznego (OECD, 2015), a to poczucie lęku może być powodowane m.in. przez nadmierny nacisk na prawidłową odpowiedź i zmuszanie do stosowania określonych procedur rozwiązywania zadań przez nauczycieli (Harper, Daane, 1998). Także w świecie *open source* przyzwolenie na popełnianie i poprawianie błędów jest uważane za to, co sprzyja poczuciu sprawczości, a w konsekwencji osiąganiu efektów. Twórca jądra (*kernel*) Linuxa – Linus Torvalds – wskazywał, że „ze złymi decyzjami technicznymi jest tak, że zawsze można je cofnąć" (Cass, 2016).

Uczestnik, którego działanie będzie pozytywnie oceniane przez innych uczestników świata DS w omawianych aspektach, podejmuje samodzielnie próby rozwiązywania napotykanych problemów, działa bez strachu przed popełnieniem błędu i z przekonaniem o własnym sprawstwie, czyli tym, że jest w stanie poradzić sobie z rozwiązaniem problemu. Samodzielnie i swobodnie wybiera sposoby podejścia do problemów oraz działa efektywnie, optymalizując różne aspekty wkładu pracy w porównaniu do jej efektu.

Nie jest więc tak, że data scientiści zwracają uwagę wyłącznie na szybkość: „wolny działający kod jest lepszy niż szybki niedziałający kod" (Nowosad, 2019: 15). Uczestnicy świata DS raczej **postrzegają swoje działanie w kontekście jak najefektywniejszej optymalizacji** jakości i szybkości pracy oraz **maksymalizowania wybranego aspektu**.

W świecie DS tak ważna jest efektywność, ale nikt nie uważa, że im więcej zrobi modeli dziennie czy im więcej linijek kodu napisze, tym lepszym jest data scientistą – nie jest to więc działanie „na ilość" w takim sensie. Szybkość jest bardzo ważna, ale

92 Przykładowo: pojawiają się żarty, w których według szablonów charakterystycznych okładek znanego w DS wydawnictwa O'Relly przygotowano fikcyjne tytuły technicznych podręczników, m.in.: „Kopiowanie i wklejanie ze Stack Overflow", „Googlowanie komunikatów o błędach", „Próbowanie rzeczy, aż zadziałają", „Pisanie kodu, którego nikt nie będzie mógł przeczytać", „Zmienianie rzeczy i patrzenie, co się stanie".

nikt nie liczy efektowności czasu pracy tak jak dla pracownika zbierającego truskawki. Data scientiści uważają swoją pracę za twórczą, techniczną i naukową, chodzi zatem o minimalizowanie czasu, ale nie o ilość, tylko o jakość rozwiązanych problemów. Sądzę, że uczestnicy badanego świata, spotykając się z zadaniem nastawionym na liczbę problemów, dążyliby do zautomatyzowania ich obsługi. Optymalizowanie postrzegam jako dążenie, by **rozwiązać problem najlepiej, jak to możliwe, w jak najkrótszym czasie**. Tak więc pomimo nastawienia świata DS na szybką, iteracyjną, niedoskonałą, efektywną pracę, jakość pracy – szczególnie w sensie metodologicznym – także jest dla uczestników kategorią odniesienia normatywnego. Raczej nie identyfikują się oni z tym, co nazywano kulturą firm technologicznych, wyrażającą się w hasłach: „wdrażaj i poprawiaj później" (*launch and iterate*) oraz „działaj szybko i psuj" (*move fast and break things*) (Crawford i in., 2018: 12) – bardziej może z akcentującym iteracyjne optymalizowanie hasłem „wydawaj wcześnie, wydawaj często" (*release early, release often*), pochodzącym z ruchu *open source* (Raymond, 2001). Świat DS ceni racjonalność, scjentystyczne podejście do realizowanych działań, a także ciekawość realizowanych projektów i raczej nie dąży do szybkich wdrożeń kosztem tych wartości. Tym samym wartości nie stanowi szybkość sama w sobie. Stanowi ją efektywność – rozumiana jako optymalizacja.

W kontekście optymalizacji uczestnicy badanego świata DS chętnie powołują się na *no free lunch theorem* (dosł. twierdzenie o tym, że 'darmowe obiady nie istnieją'). Nie ma zysku bez kosztu, maksymalizowanie jednego wskaźnika odbywa się zatem przez obniżanie innych i nie ma rozwiązań po prostu najlepszych (Wolpert, Macready, 1997). W odniesieniu do tego twierdzenia wskazuje się, nawet w materiałach do nauki podstaw ML, że:

> […] każdy algorytm klasyfikacji zawiera integralne założenia i żaden z modeli klasyfikacji nie przeważa nad innymi, jeżeli nie opracujemy założeń dotyczących danego zadania. W praktyce niezbędne okazuje się porównywanie przynajmniej kilku różnych algorytmów w celu wytrenowania i doboru najbardziej skutecznego modelu. Zanim jednak będziemy w stanie porównać różne modele, musimy najpierw ustalić metrykę służącą do pomiaru wydajności (Raschka, 2018: 35).

Tego rodzaju racjonalne, a najchętniej także kwantyfikowalne podejście do optymalizacji uczestnicy badanego świata DS stosują do wielu aspektów swojego działania – od oceny modeli ML, przez kwestie dotyczące używanych w projekcie narzędzi, kod i pisanie kodu, po np. skład zespołu projektowego do ogólnego rezultatu projektu. W rozmowach nieformalnych z uczestnikami świata DS spotykałem się ze stosowaniem kategorii optymalizacji, relacji wkładu do wyniku, ogólnie efektywności także w odniesieniu do sfer życia niezwiązanych z działaniem podstawowym ani w ogóle z DS, m.in. w kontekście wyboru środka transportu, zmiany pracy, przeprowadzki, wyboru studiów.

W stosunku do pisania kodu za przykład nastawienia na efektywność może służyć postulat przygotowywania kodu możliwego do powtórnego użycia (*reusable*)

– kod ma być modularny (*modular* – składać się z niewielkich, niezależnych części, co realizowane jest głównie za pomocą funkcji), poprawny (*correct* – robić to, co piszący ma na myśli), czytelny (*readable* – łatwo innej osobie czytającej zrozumieć, co robi kod), mieć styl (*stylish* – być pisany zgodnie z jednym stylem, np. PEP 8 dla Pythona), wszechstronny (*versatile* – rozwiązywać problem, który występuje więcej niż raz i przewidywać zmienność danych wejściowych) oraz kreatywny (*creative* – rozwiązywać problem dotychczas nierozwiązany lub znacząco poprawiać istniejące rozwiązanie) (Tatman, 2019).

Słowo **„dowozić"** jest traktowane jako nieformalne, stosowane w rozmowach pomiędzy uczestnikami badanego świata i ich współpracownikami, natomiast w rozmowach z klientami czy w wystąpieniach publicznych używane jest **„dostarczyć"**. Oba są kalką z angielskiego *deliver* (por. Ameisen, 2018), które oznacza zrealizowanie określonego zadania w umówionym terminie (*deadline*). Nastawienie na dostarczanie odróżniać ma DS od pracy naukowej, gdzie występują określone terminy realizacji, np. grantów, ale zdaniem jednego z rozmówców nie jest to presja czasowa, znana mu z pracy w komercyjnym DS:

> R: I, mmm, jasne jest, że to [data scientist – przyp. R.Ż.] jest bardzo dobra [pauza] praca, tak? Tylko dosyć niedużo osób się do niej nadaje. Czyli trzeba mieć dosyć specyficzny, jak to się mówi, skillset, czyli tak jak mówię – trzeba umieć całkiem nieźle programować, trzeba całkiem nieźle znać się na statystyce i czuć matematykę, trzeba też jednocześnie rozumieć procesy biznesowe klienta w domenie, w której klient pracuje i czwarta rzecz trzeba **dowozić**. Jest taka, na przykład, mmmm, sztuczna inteligencja na uczelniach [pauza], ale ona zazwyczaj nie jest skoncentrowana na tym, żeby w miarę szybko dostarczyć klientowi, eeee, efekt. Ktoś bierze sobie grant, ma dwa lata na wykonanie jakiejś, jakichś badań, no i sobie wykonuje tak? Tutaj w części biznesowej, komercyjnej jednak dosyć duży nacisk jest na tempo. {wywiad 2}

Dowożenie jest czwartym, zaakcentowanym przez rozmówcę elementem składającym się na tę rzadko występującą hybrydę, jaką jest osoba nadająca się do pracy w charakterze data scientisty. Pierwsze trzy z wymienionych elementów bliskie są słynnej definicji Conwaya (por. rys. 1.2) – umiejętność programowania (u Conwaya *hacking skills*, czyli w moim rozumieniu programowanie i biegłość technologiczna), znajomość matematyki i statystyki (*math and statistics knowledge*) oraz rozumienie procesów biznesowych klienta (*substantive expertise*, co oznacza bycie ekspertem w konkretnej dziedzinie i różni się od umiejętności zrozumienia biznesu kolejnych klientów na potrzeby kolejnych projektów, a taka jest właśnie specyfika pracy rozmówcy). Słowo „dowozić" pojawia się w Polsce w opisywanym znaczeniu także w innych środowiskach, gdzie praca ma charakter projektowy – np. w żargonie firm IT i korporacji.

W świecie DS cenione jest **dostarczanie czegoś, co działa**. Wysoko ocenione będzie wdrożenie produkcyjne modelu, nieco niżej zakończenie projektu na etapie POC lub MVP. Kiedy rezultatem projektu DS jest raport składający się np. z podsumowań danych, wyników testów statystycznych, wyników eksperymentów

z modelami ML, jest to uznawane za projekt nieciekawy, mało ważny, mało użyteczny. Jeżeli celem pracy jest samo generowanie podsumowań danych, np. w celu sprawdzenia, w którym sklepie odnotowano największy wzrost rok do roku w sprzedaży określonego produktu, to w świecie DS przeważnie nie będzie to w ogóle uznane za data science, a za analitykę danych (na pewno przy użyciu „tradycyjnego zestawu analityka", czyli arkusza kalkulacyjnego i zapytań w SQL-u). Dla uczestników świata DS to modelowanie ML odróżnia data science od analizy danych. Jeżeli zatem w projekcie przeprowadzono eksperymenty z modelami ML, to raczej nie ma wątpliwości co do uznania tego za data science. Jeżeli jednak rezultat projektu sprowadza się do prezentacji takich eksperymentów w raporcie, to również jest to nieciekawy, mało użyteczny projekt.

Jeden z rozmówców {wywiad 6} wskazał, że raport jest typowym rezultatem projektów, ale jeżeli to jest jedyny rezultat, to projekt nie jest użyteczny (jak wizyta lekarska, która polega wyłącznie na postawieniu diagnozy). Nieco bardziej użytecznym rezultatem projektu – rezultatem, który w badanym świecie jest już uznawany za coś działającego – jest aplikacja do wizualizacji danych. Takie wizualizacje, tzw. dashboardy, przygotowywane np. jako strony WWW/intranetowe przy użyciu pakietów „Shiny" lub „plotly", będą prezentować użytkownikowi podsumowania kluczowych danych, informacje w postaci różnych wykresów i wskaźników, tzw. KPI (czasami za KPI stoi proste działanie matematyczne, czasami zaawansowany model statystyczny lub ML), a także pozwalać użytkownikowi na interakcje, takie jak samodzielne filtrowanie zakresu danych w różnych wymiarach. Produkt jest uznawany za tym lepiej działający, im mniej ingerencji ludzkiej potrzeba do jego funkcjonowania (np. odświeżania danych), ale także działanie produktu może być oceniane w kontekście realnego wpływu w dokonywaniu zmian. Najlepiej działający jest produkt, który bazuje na produkcyjnym wdrożeniu modelu, dającym automatyczną odpowiedź bardzo wiele razy dla bardzo wielu osób – „jeżeli każdy tam tylko dolara dziennie zaoszczędzi czy tylko pięć dolarów, no to przez skalę, przez codzienność ma dużo większy wpływ niż to, że prezes wie, że tu ma na czerwono, tu ma na zielono" {wywiad 6}.

Widać, jak ważna jest nie tylko efektywność, ale i jak duża jest wartość sprawczości w badanym świecie DS. Rozmówca negatywnie ocenił sytuację, w której model, w sensie biznesowym, byłby użyteczny tylko teoretycznie, w raporcie, „na papierze", a nie praktycznie „po wdrożeniu":

> R: Powiedzmy, ktoś celowo robi [pauza], ktoś **celowo** [pauza], jak to powiedzieć, chyba trzeba powiedzieć, że to był sfingowany wypadek na tej zasadzie, że ludzie się umawiają, walą tanim samochodem w drogi samochód i z ubezpieczenia taniego samochodu biorą pieniądze na naprawę drogiego samochodu […] i teraz chodzi o to, żeby w momencie zgłoszenia szkody stwierdzić, **tak**, mamy do czynienia z wyłudzeniem, trzeba się temu przyjrzeć bliżej. I były dwie trudności w tym projekcie. Jedna trudność to nie wiadomo było, czy **da się** to zrobić i trudno było w jakiś systematyczny sposób stwierdzić: tak, na pewno da się to zrobić, trzeba było po prostu przeanalizować projekt, przeanalizować **proces** i korzystając z doświadczenia stwierdzić [pauza] **tak, da się** albo **nie, nie da się**. Druga sprawa było bardzo dużo zmiennych,

> które potencjalnie były użyteczne, zmiennych [pauza], tych zmiennych były **setki**, iiii, trzeba było uniknąć problemów związanych z tym, że danych się nie rozumie. Aby upewnić się, że dane są przygotowane w dobry sposób i ta analiza **też** nas zaskoczyła mimo sporego doświadczenia, yyy, w takim sensie, że jeśli byśmy się nie przyjrzeli zmiennym praktycznie jedna po drugiej, to wtedy byśmy zbudowali model, który na papierze może i działa, ale po wdrożeniu działa dużo, dużo **gorzej**. […] Eee, no i **to** było drugie zaskoczenie, natomiast **efekt** był **też fajny**, znaczy, on był zaskakujący, w zasadzie był do przewidzenia, w ten sposób, żeee tam było **kilka** modeli, w zależności od modelu było plus 10, plus 20 plus 30% do przodu, licząc w **pieniądzach**.
> *B: Mhm, zaoszczędzonych.*
> R: Potem, finalnie w udaremnionych wyłudzeniach. {wywiad 5}

4.4.2. Sprawczość

Sprawczość to **sprawianie, że coś się dzieje**. Oddaje to znane w socjologii pojęcie *agency*. To „dzianie się" jest tożsame ze zmianą, przejściem z jednego stanu w drugi. W badanym świecie najbardziej ceniona jest sprawczość w tzw. realnym świecie, rozpatrywana także w kategoriach skali – im większa skala wpływu, tym większa sprawczość. W powyższym przykładzie {wywiad 5} widać, że zmiana dokonywana jest z użyciem ML. Model, wdrożony do pracy operacyjnej w firmie ubezpieczeniowej, przyczynił się do redukcji kosztów wyłudzeń ubezpieczeń o około 10–30% {wywiad 5}. Choć z punktu widzenia naukowców z dziedzin humanistycznych i społecznych mógłby być to rewelacyjny przyczynek do rozważań nad sprawczością aktorów pozaludzkich, to nie podejmuję ich tutaj, ponieważ taka perspektywa nie pojawia się w badanym świecie społecznym DS. W świecie DS **sprawczość postrzegana jest wyłącznie jako sprawczość ludzka**. Mówi się o narzędziach jako potężnych, wydajnych, ale to człowiek opanowuje ich używanie i dzięki nim dokonuje zmian. Narzędziami nazywamy zarówno technologie (hardware i software), jak i metody używane w DS.

To przekonanie o panowaniu uczestnika badanego świata nad narzędziami wydaje się mi szczególnie ważne właśnie w DS, a konkretnie w modelowaniu metodami ML. Uczenie maszynowe jest używane do tworzenia systemów, w których decyzje podejmowane są automatycznie, bez udziału człowieka, na podstawie odpowiedzi modelu na dane wejściowe. W ML nie ma zaprogramowanych reguł, są one wyestymowane z danych. W dodatku część metod przetwarza takie dane wejściowe, które są przystosowane do percepcji ludzkiej – teksty, obrazy dźwięki. Jest to połączone wysokim poziomem skomplikowania matematyczno-informatycznego ML. To wszystko, podsycane marketingiem, składa się na obecne w różnych dyskursach – popkulturze, mediach, naukach (przyrodniczych, humanistycznych, społecznych), biznesie – określanie modeli ML jako magii, inteligencji, władzy, statusu boskiego czy demonicznego algorytmów (por. Thomas, Nafus, Sherman, 2018).

Z punktu widzenia uczestników świata DS tego rodzaju kategoriami, podkreślającymi sprawczość tzw. algorytmów, posługują się osoby spoza świata DS, które nie mają pojęcia o szczegółach technicznych i metodologicznych, ewentualnie osoby kierujące komunikaty marketingowe (mogą być to uczestnicy świata DS lub ich rzecznicy) do osób spoza świata DS. Modelowanie metodami ML to nie magia, a majsterkowanie, a tzw. AI to raczej hasło na slajdach w PowerPoincie niż coś działającego, zapisanego w kodzie. **W świecie DS podkreślana jest w ten sposób silna podmiotowość człowieka, biegłego w posługiwaniu się narzędziami** (technologicznymi i metodologicznymi). W popkulturowych obrazach AI bogiem są albo „konstruktorzy myślących maszyn", albo same te maszyny (Knapik, 2018: 13). Pierwszy typ jest bliższy temu, w którym sprawczość lokują uczestnicy świata DS.

W komercyjnym DS akcentowana jest sprawczość w sensie **wpływu działania na wyniki finansowe firm-klientów**. Stosowane jest pojęcie dostarczania wartości (*deliver value*), ewentualnie dostarczania wartości biznesowej. Pojęcie wartości (*value*) stosowane było już w definicjach big data. Tzw. definicje 5Vs wskazywały na wartość finansową danych cyfrowych, zbieranych i przechowywanych przez różne podmioty (Gandomi, Haider, 2015). W toku badań zauważyłem, że w komunikacji ze sceny, tak na konferencjach, jak i meetupach, często pada określenie „wartość" bez ujętego wprost odniesienia do finansów. Mowa jest raczej o dostarczaniu wartości, korzyściach, zmienianiu czegoś, poprawianiu i ulepszaniu. Do akcentowania wyników finansowych działań data scientistów dotarłem głównie dzięki wywiadom indywidualnym. Szczególnie rozmówczynie/rozmówcy z najdłuższym stażem zawodowym – rozmawiałem z pojedynczymi osobami pracującymi więcej niż 15 lub 20 lat, co niekoniecznie oznacza tyle lat w DS – podkreślały/podkreślali, że **data science musi się opłacać**.

Na wartość sprawczości w badanym świecie wskazuje uznawanie pisania kodu za prawidłowy sposób wykonywania działania podstawowego. **Sprawczość człowieka posługującego się do operacji na danych kodem jest znacznie większa niż posługującego się GUI.** Ma to zastosowanie do ogólnej pracy na komputerze. Dla osób będących doświadczonymi koderami[93] omawiana różnica jest tak oczywista, że nie werbalizowały jej w wywiadach. Wgląd w tę sprawczość osiąganą dzięki kodowaniu dała mi narracja jednej z rozmówczyń, która wywodzi się z akademii i kodować zaczęła po studiach magisterskich:

> R: […] w momencie, w którym puściłam ten skrypt i on mi podzielił te bodźce, eee, zobaczyłam, jaka jest w ogóle **potęga** w użyciu tego, eee, narzędzia, tak? Jakim jest znajomość języka programowania. Że ja sobie wymyślam problem, jestem w stanie znaleźć rozwiązanie, prawie że od tak, to po prostu do mnie dotarło [krótka pauza], miałam takie [śmiech], głupio mówić, objawienie [ze śmiechem], ale to było po prostu tak **niesamowite poczucie mocy**

93 Mogą to być np. ludzie z wykształceniem informatycznym lub matematycznym, programiści, którzy przeszli do DS, osoby interesujące się programowaniem od dzieciństwa, także osoby młode.

> **sprawczej**, tego, co można zrobić, że ja stwierdziłam, że jak ja bym się tego dobrze nauczyła, to ja bym w ogóle mogła rozwiązywać wiele problemów, które napotykam w swojej pracy badawczej i w ogóle [krótka pauza] **wznieść** to, co robię na wyższy poziom, że to nie musiałoby być zawsze takie sprowadzone do tego, co jest wykonalne manualne, manualnie, albo dostępne, zrobione już przez kogoś. {wywiad 21}

Podobne wrażenie „potęgi" osiąganej dzięki kodowaniu miałem, widząc działanie pierwszych własnoręcznie napisanych linii kodu i porównując je z wkładem powtarzalnego klikania w GUI, potrzebnego do uzyskania takiego samego wyniku. Rozmówczyni wskazała, iż umiejętności programistyczne uważa za dające większą swobodę pracy. W kodzie można zapisać niemal wszystko, co człowiek wymyśli (o ile pozwalają na to umiejętności), a klikając, człowiek pozostaje ograniczony przez polecenia zdefiniowane przez twórców oprogramowania. Pisząc kod, można **skoncentrować się na tym, jak coś zrobić, a nie czy da się to zrobić**. Dzięki tej sprawczości i swobodzie rozmówczyni odczuwała, że ma możliwość osiągnięcia wyższego poziomu pracy badawczej, niż gdyby nie potrafiła pisać kodu.

O ile osoby z małym doświadczeniem w kodowaniu mogą być zachwycone sprawczością, jaką daje korzystanie z kodu w porównaniu do korzystania z GUI, to bardziej doświadczeni koderzy mogą mieć podobne odczucia, porównując kodowanie (w sensie programowania czynności i reguł) z modelowaniem metodami ML.

Pewnych problemów nie dałoby się rozwiązać „normalnym" programowaniem (opisywaniem czynności i reguł), a da się je w sposób akceptowalny rozwiązywać dzięki ML (czyli estymowaniu reguł z danych). **Uczenie maszynowe jest postrzegane jako jeszcze wyższy niż pisanie kodu poziom sprawczości**, a także efektywności, osiąganej dzięki komputerom:

> R: […] A potęga komputerów polega na tym, że, załóżmy, ja napiszę jakiś program komputerowy, jestem w stanie go zmultiplikować, nie wiem, z miliard razy, dwa miliardy, nie wiem, ile mi tylko pozwoli, nie wiem, pamięci komputer, tak, to ja mogę sobie mnożyć, wysyłać, kopiować i tak dalej, i tak dalej. […] strasznie mi się spodobała ta koncepcja, że, że maszyny mają taką **siłę** w sobie, znaczy maszyny, no dokładnie komputery, ale to, co mi się najbardziej w tym wszystkim podobało, to to, że, yyy, no, komputer to jest tylko narzędzie, tak, too komputer, póki co jeszcze nie myśli iii właściwe wszystko to, co on zrobi, to to jest tutaj w głowie, tak? I strasznie mi się spodobała właśnie ta koncepcja, że wystarczy posiąść odpowiednią wiedzę i możesz wydawać, wiesz, polecenia komputerom, które będą to wszystko robiły i które zwielokrotnią, noo, w nieograniczonym stopniu twoje możliwości, tak. I to właśnie bardzo, bardzo mi się spodobało, wtedy właśnie jeszcze myślałem, że będę robił jakieś takie strony i tak dalej, eee, że w tę stronę to pójdzie i pojechałem po raz drugi, bo to był łączony taki wyjazd na studia do [nazwa kraju] i tam miałem pierwsze w swoim życiu zajęcia ze sztucznej inteligencji i wtedy, no, bardzo, bardzo mi się spodobało i stwierdziłem, że to jest to, że w to chcę pójść. {wywiad 11}

Osiąganie jeszcze większej niż dzięki tradycyjnym narzędziom IT sprawczości i efektywności jest ogólną legitymizacją stosowania ML, a nawet DS jako takiego. Wdrażanie modeli ML może zatem być przez uczestników badanego

świata uznawane za kolejny, po tzw. cyfryzacji, czyli w ogóle przejściu do pracy na komputerach, krok w rozwoju ludzkości[94]. W badanym świecie mogą pojawiać się odniesienia do tego, że jest to jakoby naturalny kierunek rozwoju ludzkości, łącznie ze stosowaniem analogii rozwoju technologii do ewolucji istot żywych – np. komputery uczą się widzieć (metody analizowania obrazów np. z użyciem DL).

Sprawczość osiągana dzięki technologii może być uwodzicielska dla ludzi zajmujących się DS, o czym mówił Christopher Wylie, pracujący niegdyś w Cambridge Analytica (Levy, 2019). Wspominał on o swoim zafascynowaniu możliwościami targetowania reklamy politycznej dzięki danym z Facebooka i choć – jako gej i obywatel Kanady – nie miał wspólnych poglądów z amerykańską prawicą, to zaangażował się w projekt tak bardzo, że nie myślał o realnych konsekwencjach swoich działań. „Kiedy piszesz skrypt w Pythonie, to nie masz wrażenia, że robisz cokolwiek komukolwiek. Nie myślisz: »jak to może faktycznie skrzywdzić ludzi?«" (Levy, 2019).

Wskazywano mi też, że oczekiwania klientów wobec usług związanych z DS bywają wygórowane, co świadczy o przecenianiu sprawczości, możliwej do osiągnięcia dzięki projektom data science. Także niektórzy uczestnicy badanego świata mogą przeceniać sprawczość własną lub dziedziny DS jako takiej:

> *B: Nie, chciałem spytać o twoją opinię. Co sądzisz o tym, że to jest jakaś najlepsza praca świata?*
> R: [śmiech]
> *B: Najseksowniejszy zawód świata.*
> R: Tak tak [ze śmiechem], słyszałem […] No wiesz, to jest z jednej strony fajne [pauza], bo to pomaga pracować. Tak uważam, bo buduje pewną percepcję. Znaczy, uważam, że to, co robimy, jest trudne, jest nowe, więc to jest tak naprawdę stwarzanie baaardzo wielu rzeczy od nowa, to bardzo dynamicznie się rozwija, więc ten cały szum medialny się w to wpisuje i pomaga, i dobrze, dobry, dobrze nas usytuawia na rynku, to na pewno. Ale z drugiej strony nie uważam, żeby moja praca była trudniejsza niż, nie wiem, wystrzelenie ludzi na Marsa, tak? Więc to nie jest tak, że tu będzie jesteśmy teraz **herosami** i w ogóle złotym palcem dotykamy i rozwiązujemy problemy świata i lekarstwo na raka się nagle objawia, nie? Więc też trzeba mieć w sobie pokorę i wiedzieć, że to też jest pewna praca, która, no, po prostu codziennie się ją wykonuje, raz lepiej się udaje, raz gorzej, duży poziom, procent projektów w ogóle bardzo szybko umiera, […] To nie jest proste, jak człowiekowi się wydaje, że jest supermenem, bo jest teraz data scientistą, to może wszystko, [cmoknięcie] no **nie** może, no. {wywiad 6}

Spotykałem się z historiami akcentującymi realny wpływ, zmianę rzeczywistości czy inaczej ujęte synonimy sprawczości osiąganej dzięki realizacji projektów DS. Zarówno indywidualnie, jak i w roli rzeczników swoich firm/instytucji uczestnicy badanego świata wypowiadają zdania w rodzaju: „my zmieniamy świat", „robimy rzeczy niemożliwe", „rozwiązujemy problemy, których nikt inny nie potrafi rozwiązać". Nie jest jednak tak, że uczestnikom badanego świata jest obojętne,

94 Nie zauważyłem, by pojawiało się to w badanym świecie, ale w literaturze istnieją pojęcia trzeciej i czwartej rewolucji przemysłowej – trzecia to cyfryzacja, a czwarta polega na stosowaniu tzw. AI (por. Paprocki, 2016).

w jakiej dziedzinie wpłyną na świat. Część z nich zwraca uwagę nie tylko na siłę wpływu, ale i na jego znaczenie.

Moim zdaniem marketing ma relatywnie najgorszą opinię w badanym świecie. Spotkałem się w moich badaniach z krytyką etyczną tej branży (Żulicki, 2019). Nie jest to specyficzne tylko dla polskiego świata DS – już w początkowym okresie kształtowania się branży DS w USA napisano, że „żaden młody hacker nie dorastał, marząc o sprzedawaniu reklam" (Conway, 2014). Jeff Hammerbacher, uznawany za jednego z twórców współczesnego znaczenia terminu „data science", były pracownik Facebooka i LinkedIn, powiedział publicznie, że „najlepsze umysły mojej generacji zastanawiają się, jak sprawić, by ludzie klikali na reklamy […] to jest do dupy" („[…] that sucks") (Savage, Halford, 2017: 1140).

Rozmówczynie/rozmówcy pracujący w obszarach, które określały/określali jako służące wyłącznie zarabianiu pieniędzy lub nieciekawe (m.in. marketing, usługi finansowe), raczej wskazywały/wskazywali na chęć zmiany branży na taką, która jest w ich opinii bardziej znacząca, wartościowa, ważna, służy ludzkości. W badanym świecie za taką uważana jest przede wszystkim medycyna. To w medycynie, czy ogólnie branży biomedycznej, wskazywano na możliwość wzięcia udziału w projekcie, który „zrewolucjonizuje coś w świecie" {wywiad 15} w takim sensie, że np. wydłuży ludzkie życie. Jednocześnie w tego rodzaju wypowiedziach wskazywano nastawienie na samorozwój i to, by praca była ciekawa. Takie nastawienie pojawia się też w formie porad dla początkujących data scientistów (Gutierrez, 2014: 317). Niemniej można spotkać się z opiniami, że charakterystyka pracy w komercyjnym DS jest zawsze taka, iż najważniejszy jest wynik finansowy i jest on właściwie jedynym kryterium oceny jakości projektów DS, gdyż taka jest „natura kapitalizmu" (O'Neil, 2017: 108).

Pojęcie „działa/nie działa" także wskazuje na wartość sprawczości w badanym świecie. **Istnieje praktyka kwestionowania autentyczności działania, tzn. nieuznawania działalności za data science wtedy, kiedy projekt nie jest nastawiony na rezultat w postaci czegoś działającego.** Nie chodzi o sytuację, w której pomimo starań nie udaje się uzyskać zadowalających rezultatów i projekt zostaje zakończony. Chodzi o intencjonalne nastawienie wyłącznie na rezultat w postaci deklaratywnej, czyli wykonanie analizy danych i sporządzenie raportu czy prezentacji z wynikami lub rekomendacjami. Część uczestników badanego świata może wyśmiewać takie rezultaty, mówiąc, że to nie jest DS, a np. konsulting, nie uznając takich deklaratywnych rezultatów za coś, co rozwiązuje rzeczywisty problem. Przykładem tej praktyki było zdarzenie na jednym z meetupów zorientowanych na język R {obserwacja 25}. Pierwszy prelegent prezentował model optymalizacji cen dla klienta z obszaru handlu detalicznego, akcentując, jak ważne w projekcie jest zrozumienie potrzeb firmy-klienta oraz teorie ekonomiczne. Gdy padło pytanie z publiczności: „Czyli co tak właściwie dostarczyliście? Slajdy?", sala gruchnęła śmiechem. Inny prelegent bronił się słowami „chcemy dawać coś działającego, chociaż prototyp".

W kodeksach etycznych tzw. AI zauważono przekonanie o potencjale sztucznej inteligencji. Zarówno pozytywny, jak i negatywny wpływ AI jest w takich kodeksach powszechnym, uniwersalnym zmartwieniem (Greene, Hoffman, Stark, 2019). W świecie DS panuje taki właśnie pogląd: narzędzia, jakimi dysponuje data science, mają ogromny wpływ na dalsze losy ludzkości, Ziemi i świata. Rozwój DS, a w szczególności ML, uważa się za nieunikniony, jest on imperatywem, dotknie każdej dziedziny życia. Wskazuje to na wysoką wartość sprawczości w świecie DS. Jednocześnie narzędzia, w tym ML, postrzegane są jako neutralne w ludzkich rękach. Według uczestników świata DS o pozytywnym/negatywnym wpływie wolno mówić dopiero na przykładzie konkretnych projektów, a najlepiej na zrealizowanych wdrożeniach. Może to akcentować ludzką odpowiedzialność za cel używania tych uznawanych za potężne narzędzi, co postrzegam jako kolejny argument za przywiązaniem badanego świata do mocnej, sprawczej podmiotowości ludzkiej. „Technologia jest **niewinna** [*innocent*, podkr. oryg.]. To data scientista ustawia dane wejściowe (*input*) i zadaje modelowi zmienną celu" (Casas, 2018). W badanym świecie DS oraz na przecięciu ze światem biznesu {obserwacja 22} pojawiają się porównania do noża/młotka, którymi można przygotować posiłek/wbić gwóźdź lub zabić człowieka[95]. Data science (czy konkretnie ML) uważane jest za narzędzie ogólnego zastosowania, które samo w sobie nie podlega ocenie moralnej. Prowadzi to do tego, że w badanym świecie nie kwestionuje się etyczności DS jako takiego. Nikt nie powie, że analiza danych, statystyka, programowanie i ML to z reguły złe i szkodliwe pomysły. Niemniej **artykułowana neutralność narzędzi jest równa temu, że są one domyślnie dobre** – oceniając tę dobroć w odniesieniu do wartości świata DS. Narzędzia te mają przecież zwiększać efektywność i sprawczość.

4.4.3. Samorozwój i samodzielność

W świecie DS szczególnie ceni się samodzielny wysiłek oraz panuje przekonanie, że rozwój człowieka polega głównie na samodzielnej pracy. Takie postrzeganie samorozwoju związane jest z charakterem wykonywania działania podstawowego DS – pisanie kodu odbywa się niemal wyłącznie samodzielne przed ekranem komputera. Fizyczna obecność innych osób pracujących na swoich komputerach nie ma dużego znaczenia, a praktyki programowania w parach (*pair programming*) (Williams i in., 2000) są niezwykle rzadkie. Zaród taki styl pracy w kolektywach hakerskich nazwał „wspólnotą indywidualnej koncentracji"

95 Przykłady można by mnożyć, np. tak w wywiadzie dla portalu sztucznainteligencja.org.pl wypowiedział się Przemysław Biecek: „przecież nie mówimy, że młotek jest etyczny albo nieetyczny, tylko że osoba, która go użyła, podjęła bardziej lub mniej etyczne decyzje. SI [sztuczna inteligencja – przyp. R.Ż.] jest narzędziem i w kwestiach etycznych nie zdejmowałbym odpowiedzialności z człowieka, który to narzędzie stworzył, lub który z niego korzysta" (Chojnowski, 2020).

(Zaród, 2018: 220) i ten styl obowiązuje w świecie DS w sytuacji fizycznego dzielenia przestrzeni przez uczestników badanego świata.

Samodzielność nie oznacza samotności. W świecie DS praktykowanych jest wiele form współpracy grupowej i wzajemnej pomocy. O ile działanie podstawowe jest realizowane samodzielnie i zazwyczaj za dany projekt DS jest odpowiedzialna jedna osoba, to występuje praktyka konsultowania się i dzielenia wiedzą.

Dzielenie się wiedzą, kodem i wynikami pracy własnej oraz udzielanie pomocy innym osobom są w świecie DS powszechne i cenione, tak w formie zapośredniczonej przez internet (np. na GitHub, LinkedIn, blogi, Stack Overflow, Slack), jak i w kontaktach twarzą w twarz (codzienna praca, meetupy, konferencje, warsztaty, hackatony). Każda z tych praktyk jest także nastawiona na samorozwój oraz autoprezentację – budowanie swojego wizerunku zarówno w badanym świecie, jak i wobec ludzi spoza świata, np. rekruterów, pracodawców.

Samorozwój jest w badanym świecie postrzegany jako działanie bezustanne. Polega on na:

1) byciu na bieżąco z nowymi technologiami i narzędziami dla DS;
2) poznawaniu domeny lub kolejnych domen, z których pochodzą dane, z którymi się pracuje;
3) poznawaniu pracy innych osób zajmujących się DS, także w innych domenach, stosujących inne metody i inne technologie;
4) zgłębianiu lub przynajmniej odświeżaniu wiedzy teoretycznej z zakresu matematyki, uznawanej za niezmienną podstawę DS (np. algebra liniowa, statystyka matematyczna);
5) podejmowaniu wyzwań – może pojawiać się określenie w rodzaju „wychodzenia ze swojej strefy komfortu", czyli np. zmiana pracy na inną domenę danych, udział w hackatonie, podjęcie się organizacji konferencji, podjęcie projektu niekomercyjnego itp.

Cześć uczestników świata DS podkreśla, że ważne jest także rozwijanie kompetencji biznesowych i miękkich, np. umiejętności doprecyzowania celów biznesowych projektów, rozumienia działania różnych rodzajów firm-klientów, zdolności komunikacji z osobami nietechnicznymi. W świecie DS nie uważa się, że raz osiągnięty wysoki poziom kompetencji zostaje na zawsze, a raczej, że nie używając określonych narzędzi i metod, zapomina się ich i trzeba uczyć się powtórnie. Ponadto można zostać w tyle.

Istnieje presja **bycia na bieżąco**. Dominuje przekonanie o tym, że DS zmienia się i rozwija bardzo szybko, oraz że podobne szybkie tempo przemian w różnych aspektach charakteryzuje współczesny świat jako taki. Nie dotyczy to tylko rozwoju narzędzi charakterystycznych dla DS, choć w praktyce duża część działań uczestników to śledzenie nowo pojawiających się narzędzi (konkretnie np. repozytoriów kodu, pakietów, artykułów naukowych prezentujących metody ML). Część uczestników może mówić o eksponencjalnym tempie zmian, przyspieszeniu technologicznym (co jest charakterystyczne np. dla Raya Kurzweila), bywa to także praktyką legitymizowania DS i praktyką

marketingową – upraszczając, trzeba wdrażać AI, bo firmę przegoni konkurencja – panuje w tej sferze „wyścig zbrojeń" {wywiad 19}.

W kontekście samorozwoju bycie na bieżąco oznacza, że należy nieustannie uczyć się i podejmować działania nierutynowe co najmniej w pięciu wyżej wyróżnionych aspektach. Jako kod *in vivo* traktuję używane w świecie DS pojęcie ***state of the art***, które oznacza najbardziej zaawansowaną technologię lub metodę w danym momencie. W badanym świecie DS należy zatem znać *state of the art* i wiedzieć, że za miesiąc czy dwa coś innego może być owym najbardziej zaawansowanym rozwiązaniem. Należy znać aktualne dokonania i rozumieć ich tymczasowość.

To przekonanie o konieczności bycia na bieżąco i znajomości *state of the art* występuje równocześnie z opisywanym nieco wcześniej przekonaniem o nieuchronności dalszego rozwoju DS. Jeden z rozmówców wskazał, że struktura ról w DS będzie się zmieniać (mniej stanowisk analitycznych, więcej związanych z różnymi aspektami ML oraz inżynierii danych), niemniej nie spodziewa się spadku wysokiego popytu na pracę osób zajmujących się DS, ponieważ „to nie jest tak, że jedna firma odpuści, żeby być w tyle" {wywiad 19}. Poproszony przeze mnie o spekulacje na temat tego, co mogłoby zatrzymać lub spowodować regres w branży DS, wskazał żartobliwie na wybuch na Słońcu niszczący całą ziemską elektronikę, ewentualnie silne regulacje prawne – choć te drugie raczej zmieniłyby charakter branży, niż doprowadziły do zapaści.

Bycie samodzielnymi i rozwijanie się jest przez część uczestników badanego świata DS uznawane za wynik ciekawości, a dokładniej bycia zainteresowanym data science. W poniższym fragmencie zwracam szczególną uwagę na to, że zdaniem rozmówczyni bycie zainteresowanym DS sprawia, że nie „trzeba" (w sensie przymusu), a „można" (w sensie możliwości) wciąż się rozwijać. To, czy będzie się mieć poczucie ciągłego samorozwoju, wskazywano jako kryterium decydujące o kontynuacji albo zmianie pracy.

Stykając się w wywiadach wyłącznie z wypowiedziami łączącymi ciekawość, samorozwój i samodzielność, sądziłem, że badany świat DS składa się tylko z uczestników „rzeczywiście zainteresowanych" {wywiad 4}, dla których praca jest pasją lub co najmniej hobby. Są jednak uczestnicy niebędący tak pojętymi „zainteresowanymi" DS. Wytłumaczono mi, że istnieje kilka poziomów bycia zainteresowanym, z których najwyższy to pasjonat, i że do takich właśnie osób mam w moich badaniach etnograficznych dostęp, bywając na meetupach i konferencjach. Nawet jednak w przypadku uczestników, dla których stabilna praca jest ważniejsza od samorozwoju, „nie jest tak, że oni w ogóle nie inwestują w rozwój, ale, no, jakby widać dysproporcje [w porównaniu z pasjonatami – przyp. R.Ż.]" {wywiad 15}. **Uczestnik świata DS jest zatem zobowiązany wobec tego świata do ciągłego samorozwoju.**

To przekonanie o konieczności ciągłego rozwijania się może być wzmacniane w badanym świecie przez to, że uczestnicy świata DS znają lub sami wykorzystują możliwości automatyzacji pracy ludzkiej, co może wywoływać poczucie bycia zagrożonym tzw. bezrobociem technologicznym.

> *B: Okey, aa czy teraz, ee, też musisz się jeszcze uczyć różnych rzeczy? Czy już wszystko umiesz?*
> R: Nie [śmiech], nie, nie. Jakbym uznała, że wszystko umiem, to by był koniec. {wywiad 17}

Biecek (2019b) podczas wystąpienia na warszawskim Data Science Summit 2019 stwierdził, że najważniejszymi obecnie trendami w DS jest właśnie automatyzacja modelowania (autoML) oraz wyjaśnialność modeli (XAI) {obserwacja 46}.

Osiągnięcie pewnego poziomu kompetencji technicznych i metodologicznych, połączone z przekonaniem o wartości samorozwoju i przyzwyczajeniem do podejmowania samodzielnego wysiłku, może prowadzić część uczestników badanego świata do **przekonania o możliwości skutecznego nauczenia się niemal wszystkiego**. Jest to odmiana poczucia sprawczości, która polega na przekonaniu podmiotu o byciu zdolnym do zrozumienia lub zrobienia, co zechce. Sama owocna nauka pisania kodu może sprzyjać kształtowaniu się takiego przekonania u osób uczących się, a jednocześnie nastawienie na samodzielność i poczucie własnej sprawności może sprzyjać osiąganiu sukcesów w pisaniu kodu. Kodowanie może być uznawane przez laików za umiejętność trudno dostępną, wymagającą elitarnych kompetencji umysłowych. Nowosad (2019: 15) jako antidotum na ten mit proponuje samodzielne wyszukiwanie informacji w internecie i ciągłe uczenie się.

W świecie DS ceniona jest samodzielność – także jako cecha opisywana w psychologii mianem **wewnątrzsterowności** lub wewnętrznego umiejscowienia kontroli (*internal locus of control* – LoC). Sądzę, że jest to związane – po pierwsze – z tym, że świat społeczny DS jest młodą i niezinstytucjonalizowaną całością społeczną, a po drugie, ze specyfiką działania uczestników świata DS. W pierwszym przypadku chodzi o to, że nie ma instytucjonalnej ścieżki zostania data scientistą, nie ma uznanego, formalnego sposobu zaświadczania o kwalifikacjach, a osoby, które obecnie określają się jako data scientiści czy zajmujące się data science, mogą mieć różne wykształcenie i doświadczenie zawodowe (w których niekoniecznie pada nazwa „data science"). Tym samym w badanym świecie zdecydowana większość uczestników to samoucy, a dopiero nieliczne osoby rozpoczynające karierę w 2018–2019 roku mogą być absolwentami studiów o profilu data science.

Samodzielne uczenie się narzędzi technologicznych jest szczególnie ważne w przypadku osób wywodzących się z matematyki lub nauk przyrodniczych, niemających wykształcenia informatycznego. Typowym scenariuszem, nie tylko dla takich osób, jest uczenie się nowych narzędzi na potrzeby konkretnego projektu czy problemu. Specyfika działania badanego świata DS sprawia, że wewnątrzsterowność i wewnętrzne LoC są w tym świecie pożądane – pozytywnie będzie oceniana w DS osoba, która samodzielnie osiąga rezultaty, samodzielnie definiuje cel lub cele szczegółowe, nie czekając na polecenia, z własnej inicjatywy zadaje pytania, szuka odpowiedzi, jest gotowa podejmować decyzje o nowym kierunku eksperymentów, a przede wszystkim ogólnie wychodzi z inicjatywą, nie czeka na instrukcje.

4.4.4. Ciekawość

Ciekawość oznacza **ciekawość jako cechę człowieka** – bycie ciekawym czegoś, zainteresowanym czymś, także bycie ciekawskim lub dociekliwym. Oznacza także **ciekawość jako cechę przedmiotu** – coś jest ciekawe lub interesujące. W świecie DS ciekawość ceniona jest we wszystkich tych znaczeniach.

Ciekawość, będąca cechą człowieka, jest tym, co wskazywano mi jako charakterystykę łączącą wszystkich uczestników świata DS. Jeden z rozmówców uważał, że DS to nie tylko zawód, ale typ myślenia (co było dla mnie markerem do szerszego niż kategoria zawodu czy pracy skonceptualizowania badanego wycinka rzeczywistości społecznej, a ostatecznie przyjęcia ramy pojęciowej społecznych światów):

> *B:* […] *Bo nie wiem, czy jest coś innego, co was łączy, czy to jest tak, że praca was łączy, nie? Że data scientist to jest zawód, tak bym to nazwał, tak?*
> R: Ale też typ myślenia na przykład. To znaczy, myślę, że taką istotną cechą, to znaczy każdy ma ją w różnym zakresie, jest ciekawość. U nas w zasadzie nie ma ludzi nieciekawych, każdego tam coś ciekawi [pauza], no, mamy swój nerdowski humor [pauza], mamy pewne specyfi [R wskazuje na swój T-shirt], no sam pan widzi, jak ja wyglądam, aaa, w garnitury się ubieramy na śluby i na pogrzeby, tak? No i tego typu, tak? Mmmmm [pauza] duża część, myślę, z nas jednak gdzieś tam błądzi myślami tak mocno i nawet podczas rozmów kwalifikacyjnych wychodziło. Rekrutowałem też dosyć sporą ilość informatyków [pauza], no to mam wrażenie, że jednak ta grupa data scientistów jest **jeszcze** bardziej specyficzna niż informatyków. {wywiad 2}

Bycie ciekawym jest rozumiane na dwa sposoby:

1) ciekawość czegoś, zainteresowanie czymś – tym czymś jest DS, ewentualnie dziedziny jej bliskie jak ML, programowanie, matematyka; tak pojęta ciekawość może być uważana w badanym świecie za coś, co sprzyja samorozwojowi w tym sensie, że osoba „rzeczywiście zainteresowana" DS rozwija się i uczy nowych rzeczy, wiedziona przede wszystkim ciekawością, czyli własną motywacją wewnętrzną; w konsekwencji to **bycie zainteresowanym jest w badanym świecie strategią potwierdzania autentyczności**, czyli legitymizacji siebie jako prawdziwego uczestnika świata DS; negatywnie są w świecie DS oceniane motywacje własnych działań w rodzaju „uczę się metody X i dzięki temu zatrudnię się w firmie A, gdzie zarobię dwa razy tyle, co obecnie"; raczej wypada wskazać na motywacje typu „słyszałem o metodzie X i tak mnie zaciekawiła, że wczoraj pół nocy nie spałem i czytałem o tym, a dzisiaj w pracy znalazłem nowy pakiet właśnie do tego i już nie mogę się doczekać, aż w domu to uruchomię";
2) ciekawskość lub dociekliwość – rozumiane jako niezadowalanie się łatwymi, powierzchownymi odpowiedziami i wyjaśnieniami, dążenie do poznania szczegółów, doprecyzowywanie i uściślanie, a – jak sądzę – także chęć zrozumienia, eksperymentowania, szukania, zgłębiania, rozpracowywania problemów; nie dotyczy to każdego i zawsze działania w badanym świecie,

> w odniesieniu do centralnej kategorii efektywności w pewnych sytuacjach dociekanie szczegółów będzie bowiem oceniane negatywnie jako strata czasu lub energii umysłowej; zdaniem niektórych uczestników świata DS tak rozumiana ciekawskość czy dociekliwość jest cechą indywidualną, bez której nie można być uczestnikiem badanego świata; **osoba zajmująca się DS musi zatem legitymować się zarówno chęcią, jak i zdolnością do zadawania pytań**; część uczestników badanego świata może uważać, że DS wymaga także pomysłowości, kreatywności, twórczego czy nieszablonowego podejścia do działania; początkowo wydawało mi się to niezwiązane z ciekawością, jednak wskazuje to zarówno na wartości ciekawości, jak i samodzielności oraz sprawczości w badanym świecie w tym sensie, że pozytywnie ocenia się tworzenie nowych rozwiązań (technicznych i metodologicznych), podejmowanie nierozwiązanych czy po prostu słabo zdefiniowanych problemów, a negatywnie rozwiązania szablonowe i wtórne, dobrze rozpoznane – takie traktowane są jako nieciekawe.

W podręczniku dla początkujących data scientistów używających języka R napisano:

> Kluczem do zadawania dobrych pytań doprecyzowujących (*follow-up questions*) będzie poleganie zarówno na swojej ciekawości (*curiosity*) – o czym chcesz dowiedzieć się więcej? – oraz na sceptycyzmie (*skepticism*) – jak może to być mylące? (Wickham, Grolemund, 2017).

Data scientista to nie osoba „tylko do klepania kodu" {wywiad 22}. Ciekawość, będąca cechą indywidualną uczestników oraz cechą przedmiotu (DS czy ML), jest w badanym świecie chętnie wskazywana jako to, co odróżnia DS od innych, raczej inżynieryjnych dziedzin IT – tam tylko „klepie się" kod. Uczestnicy świata DS uważają się za bliższych nauce niż zwykłej, raczej nieciekawej z ich perspektywy inżynierii, może pojawiać się określenie R&D (*research and development*, dosł. 'badania i rozwój'). Z tej perspektywy tworzenie oprogramowania może być uznawane w świecie DS właśnie za coś zwykłego, nudnego, nieciekawego, niezapewniającego samorozwoju, polegającego na szablonowym rozwiązywaniu dobrze zdefiniowanych problemów. Data science przeciwnie – jest ciekawe i wymaga bycia ciekawym, żeby je robić.

Ciekawe są sprawy niemające oczywistego rozwiązania: trudne do zdefiniowania, trudne w realizacji, niewyjaśnione. Wypowiedź jednego z rozmówców wskazuje także na wartość sprawczości – „ja najczęściej biorę projekty, które albo się nie udały, przynajmniej z dwa razy, albo nikt nie wie, jak się za nie zabrać" {wywiad 6}, co znaczy: inni nie dają rady realizować tak trudnych projektów, jak data scientista. Uczestnicy świata DS chętnie postrzegają „coś ciekawego" jako wymagającą ich wysiłku intelektualnego zagadkę. Ciekawy może być przede wszystkim problem, w drugiej kolejności dziedzina (wymieniane wcześniej finanse i marketing będą w badanym świecie raczej uznawane za mało ciekawe), ewentualnie metody (szczególnie DL lub metody niestandardowe), ale raczej nie technologie. „Ciekawy

projekt" oznacza „taki, którym warto się zająć" i uczestnikom uzasadniającym w ten sposób np. zmianę pracodawcy inni uczestnicy raczej nie zadają kolejnych pytań o motywy decyzji. Ciekawy projekt to także taki, w którym konkretny uczestnik zgodnie ze zobowiązaniem do samorozwoju będzie się dalej rozwijać, pewne przedmioty są zatem uznawane za ciekawe/nieciekawe na poziomie indywidualnym – zarówno w odniesieniu do bycia zainteresowanym jakimś przedmiotem, jak i relacji własnych kompetencji do poziomu trudności przedmiotu. W tym sensie dla określonego uczestnika ciekawa może być praca w agencji marketingowej czy w banku. W świecie DS istnieją także przedmioty powszechnie uznawane za ciekawe – medycyna, pojazdy autonomiczne i bezpieczeństwo (wykrywanie wyłudzeń/oszustw, bezpieczeństwo cyfrowe). Co do metod, to przynajmniej część uczestników za ciekawe uznaje DL, a raczej powszechnie za takie uznane będą metody niestandardowe, czyli np. połączenie typowych modeli ML w jeden bardziej złożony asamblaż (*model ensemble*), wykorzystanie typowego modelu ML lub statystycznego w sposób inny niż typowy, a także użycie mało znanych metod statystycznych, pochodzących z wąskiej dziedziny badań ilościowych (jak ekonometria, badania operacyjne, bioinformatyka, meteorologia) i zaadaptowanie ich do innej domeny. Za ciekawe mogą być uznawane dane nieustrukturyzowane (teksty, obrazy, dźwięk), które standardowo modelowane są za pomocą DL. Nie spotkałem się z określaniem technologii jako ciekawych, ewentualnie określano niektóre z nich jako fajne. Niemniej „fajne" jest mało znaczące poza tym, że wskazuje na pozytywną ocenę – mianem „fajne" określano nie tylko technologie (np. klastry, GPU), ale i konkretnych pracodawców, ciekawe problemy, rozwiązywanie zagadek, korzystanie z technologii *open source*, możliwość pracy zdalnej, a więc sprawy wskazujące na różne wartości badanego świata.

W badaniach twórców oprogramowania *open source* oraz hakerów wskazywano na ciekawość jako motywację do podejmowania i podtrzymywania ich działań. W tym wymiarze hakerzy podobni są do naukowców według tzw. klasycznego etosu naukowca Roberta K. Mertona (Zaród, 2018: 24–25, 288–289). Ciekawość, będąca motywacją do działania, w świecie DS jest **obowiązująca w tym sensie, że uczestnikom wypada powoływać się na nią jako własną motywację** do uczestnictwa w świecie DS.

Istnieje związek wartości świata DS – ciekawości, samodzielności, samorozwoju, efektywności – z postmodernistycznym rozumieniem pracy (Haratyk, Biały, Gońda, 2017: 154–156). Choć mowa o świecie społecznym, w którym działanie podstawowe może, ale nie musi być wykonywane komercyjnie i może, ale nie musi być źródłem dochodu, to rozumienie „pracy" (czy raczej działania podstawowego) jako rozwinięcia/spełnienia cech życia jednostki (Haratyk, Biały, Gońda, 2017: 154–156) oraz interpretowanie „pracy" jako kwestii wyboru i rozwoju osobistego lub rozwoju życiowego – zgodnie z postmodernistycznym imperatywem samorealizacji, opisanym m.in. przez Małgorzatę Jacyno (2007) – są zbieżne z poglądami uczestników świata DS i znajdują wyraz w wartościach.

4.4.5. Swoboda

Uczestnicy świata DS traktują swobodny wybór własnych narzędzi pracy raczej jako coś oczywistego. Negatywnie odnoszono się w moich badaniach do sytuacji, w której narzędzia są odgórnie narzucane, np. przez szefa zespołu, firmę/instytucję zatrudniającą czy klienta. Uczestnicy badanego świata negatywnie oceniają odgórne narzucanie narzędzi w postaci oprogramowania zamkniętego. Szczególnie dotyczy to podstawowego środowiska pracy DS, gdzie swoboda wyboru jest zawężona do dwóch dominujących języków programowania skryptowego – Pythona i R. Narzędzia typu SAS czy SPSS raczej nie są używane przez uczestników badanego świata z własnego wyboru. **Swoboda i elastyczność są w badanym świecie uznawane za zalety zarówno R, jak i Pythona jako takich.** Z jednej strony te zalety płyną z tego, że są to języki programowania, a nie oprogramowanie z GUI, użytkownik jest zatem znacznie mniej ograniczony w dostępnych opcjach. Z drugiej strony są to narzędzia *open source*, a więc istnieje swoboda w stosowaniu ich do rozmaitych celów, swoboda modyfikacji i budowania rozszerzeń (przede wszystkim w postaci pakietów), a te rozszerzenia, rzecz jasna, zwiększają elastyczność, a także możliwości (a więc i sprawczość).

Istnieje praktyka podkreślania, że używany język programowania jest językiem wyboru (kalka z *language of choice*) – osoba używa go dlatego, że chce, a nie musi. Jest to strategia nastawiona na wygodę i łatwość pracy w tym sensie, że człowiekowi pracuje się wygodnie za pomocą narzędzia, które mu odpowiada. Istnieje w badanym świecie postulat dobierania właściwego narzędzia do problemu, nie kłóci się to zupełnie ze swobodą wyboru narzędzia, biegłość np. w języku Python i sympatia do niego u konkretnej osoby mogą być bowiem traktowane jako jedno z kryteriów dopasowania narzędzia. Ostatecznie ta swoboda wyboru narzędzi ma służyć efektywności. Tak więc choć to, jakimi narzędziami posługują się inni uczestnicy, ma dla indywidualnego uczestnika znaczenie przy wyborze narzędzia, to pozytywnie oceniana jest w badanym świecie sytuacja, kiedy uczestnik może wejść do nowego projektu/zespołu/instytucji z wybranymi przez siebie narzędziami.

Jako coś oczywistego w badanym świecie traktowana jest także swobodna komunikacja i ubiór. Nie jest to swoboda w sensie dowolności, a w sensie nieformalności. Zwracanie się *per* „ty" przez dwudziestoletnich uczestników świata DS do dwukrotnie starszych uczestników z tytułami profesorskimi lub na stanowiskach kierowniczych jest uważane za normę, zgodnie ze zwyczajami angloamerykańskimi. Ta forma komunikacji obowiązuje zarówno na wydarzeniach świata DS, jak i pomiędzy uczestnikami tego świata.

Swobodna komunikacja dla uczestników świata DS oznacza, poza nieformalnością, także bezpośredniość i możliwość wyrażania własnego zdania, np. niezgody w kontakcie z przełożonym w pracy. Swoboda jest w tym wymiarze bliska samodzielności, dotyczy bowiem m.in. podejmowania decyzji o kolejnych

czynnościach we własnej pracy, organizacji czasu pracy, a także omawianego już doboru narzędzi. Nie każdy uczestnik badanego świata ma pełną decyzyjność w omawianych obszarach, ale tak pojęta swoboda jest traktowana jako coś cennego, szczególnie w kontekście zatrudnienia. Dalej powiem o dość powszechnych praktykach elastyczności czasu i miejsca pracy.

Wartość swobody w relacji z przełożonym widać w badanym świecie także w negatywnym ocenianiu przez uczestników praktyk tzw. mikromanagementu. Mikromanagement ogólnie oznacza sposób zarządzania ludźmi bazujący na szczegółowej kontroli nawet drobnych aspektów pracy. Uczestnicy cenią raczej przeciwny styl zarządzania i współpracy – taki, gdzie każdy ma dużą samodzielność i swobodę oraz bierze odpowiedzialności za własne działanie, a także gdzie panuje raczej płaska struktura:

> *B: OK, a co ci się najbardziej podoba w tym, co robisz teraz?*
> R: [pauza] Najbardziej podobają mi się ludzie. […] Natomiast, yyy, w przypadku takiego zespołu, gdzie jest nas kilkanaście osób, codziennie mamy spotkania, pracujemy, yyy, w pięciu miastach w czterech krajach, także spotykamy się **tylko i wyłącznie** na Skype [pauza] i pracujemy. Nikt nikogo nie potrzebuje, nikt nie ma potrzeby, żeby nikogo pilnować, stać na nim z **batem**, kontrolować godzin przyjść i wyjść z pracy, bo każdy wie, co ma robić. Każdy chce to robić, każdy chce to ulepszyć, to, co robimy, iiii jest ten taki **ferment** polegający na, yyy, ścieraniu się różnych idei pomysłów, yyyy, po prostu bycia **lepszym** w tym, co się robi. To jest, i to jest coś, co [pauza] co jest **najlepsze** w tym wszystkim. W tej pracy, tak. {wywiad 3}

Wiele treści zawiera w sobie końcówka powyższej wypowiedzi – widać w niej odniesienia nie tylko do omawianej wartości swobody, ale także do efektywności czy dokładniej do optymalizacji („każdy chce to ulepszyć to, co robimy"), ciekawości i samodzielności („każdy wie, co ma robić. Każdy chce to robić") oraz samorozwoju („jest ten taki ferment polegający na ścieraniu się różnych idei pomysłów, po prostu bycia lepszym w tym, co się robi" {wywiad 3}).

Jeśli chodzi o ubiór, to obowiązuje casual, osoba w marynarce/żakiecie wyróżnia się na wydarzeniach w badanym świecie, garnitur/kostium wskazuje raczej na osobę spoza świata DS, dodatki formalne, takie jak krawat, koszula ze spinkami, teczka czy buty ze skóry lakierowanej, widziałem wyłącznie na konferencjach biznesowych {obserwacje 22, 37}. W świecie DS należy ubierać się „tak, żeby ci było wygodnie" {wywiad 8}. Niemniej w toku badań nie widziałem osób noszących dresy. Jednocześnie w komercyjnym DS nie spotkałem się ze sformalizowanym dresscodem. Nie istnieje także nieformalny dresscode badanego świata, świat ten jest dość różnorodny, choć można wskazać pewne tendencje.

Po pierwsze, plecaki są znacznie bardziej popularne niż torby/teczki. Spotkałem się głównie z używaniem plecaków o pojemności około 20 litrów, nierzadko turystycznych (np. Deuter, Vaude, Thule). Wielu uczestników badanego świata nosi na co dzień laptop przy sobie, stąd używanie plecaków z odpowiednią przestrzenią do bezpiecznego i wygodnego przenoszenia komputera. Plecaki noszone

są zarówno do T-shirtów i jeansów, jak i do marynarek, sukienek. W rozmowach nieformalnych określano kobiecy zestaw składający się z casualowej sukienki i plecaka na laptop jako „stylówka z politechniki". Po drugie, obowiązują dwa nierozłączne style ubierania się, dla uproszczenia nazywam je „techniczny" i „schludny". Techniczny polega na noszeniu T-shirtów z logotypami technologii lub wydarzeń o charakterze technicznym[96]. Utrwalonym w popkulturze elementem tego stylu ubierania się jest bluza z kapturem, jednak jest ona w świecie DS mniej popularna niż T-shirty, ponieważ kojarzona jest z IT. Niemniej niektórzy uczestnicy badanego świata noszą takie bluzy, także z logotypami technologii, zdarza się też, że są to bluzy służbowe – z logotypem firmy, której uczestnik jest pracownikiem (szczególnie gdy występuje w roli oficjalnej reprezentacji firmy, np. prelegenta na wydarzeniu). Styl schludny składa się raczej ze stonowanych elementów ubioru casualowego – gładkie T-shirty, koszule, bluzki, swetry, jeansy, sukienki, buty sportowe, buty skórzane – w porównaniu ze stylem technicznym jest bardziej dopracowany i zadbany w sensie połączenia kolorów czy dopasowania rozmiaru do sylwetki i – jak mi się wydaje – jakości odzieży. Połączenia w rodzaju: casualowa marynarka i T-shirt z zeszłorocznej konferencji są także akceptowane. Latem w pełni akceptowane są osoby noszące krótkie spodnie, sandały lub klapki. Techniczny styl ubierania się może być w badanym świecie łączony z kategorią nerda i raczej z mężczyznami niż kobietami:

> *B:* […] *czy kogoś, kto się zajmuje data science, jesteś w stanie rozpoznać po wyglądzie?*
> R: Niee, no w ramach populacji, no może inaczej. Jako chodzi ogólnie o typ osoby technicznej, to jest często typ osoby, którą się rozpoznaje z daleka.
> *B: Na czym on polega?*
> R: No to by trzeba zobaczyć, to są bardzo często jakieś koszulki z konferencji, z technologii, ogólnie taki T-shirt i tak dalej. A takie coś też takie, że często to są takie z Aspergerem, z małym kontaktem z ciałem, tak chodzą, tak, tak właśnie [R przygarbia się i unosi barki] no tak dosyć sztywnie i tak widać **myśli** są, są w siebie, w myślach, no to może być programowanie dowolne czy tam nauka. Często to jest też styl wypowiedzi, ostatnio powiedziałem do jednej dziewczyny, czy jest matematyczką może. […] To nie jest, że data science ma zawsze takie coś czy że tak wygląda, ale cechy data scientisty to jest ciężko rozpoznać, bo to jest w wielu miejscach podobne. Bo to też są różne teamy, różne osoby, bardziej matematyczne, bardziej naukowe, bardziej projektowe, bardziej start-upy, bardziej coś tam [pauza] data science jest bardzo ogólne. Są osoby jak artyści, są jak programiści, jak biznesowi, więc to jest takie, wiesz, o co chodzi? Też są różni artyści, ale czasami widać, że ktoś coś kreuje, ten styl jest taki przemyślany, niektóre osoby są takie po prostu schludnie ubrane, są osoby, które przychodzą w ogóle, wiesz, w bluzie wyciągniętej, a to jest znany jakiś jeden z najlepszych specjalistów data science na świecie. {wywiad 26}

Po trzecie, z uwagi na ubiór nie byłem w stanie rozróżnić uczestników silniej i słabiej osadzonych w badanym świecie. Podobnie rzecz się miała ze stanowiskiem, branżą czy w ogóle obszarem działania uczestnika – wszyscy ubierają się

96 Inną praktyką jest oklejanie pokrywy laptopa naklejkami o zbliżonej tematyce – robią to prawie wszyscy uczestnicy badanego świata DS, o czym więcej w rozdziale piątym.

raczej w zbliżonym, nieformalnym stylu. Jest to kolejny wyraz traktowania swobody jako wartości w badanym świecie w tym sensie, że nie podkreśla się (przynajmniej w sferze wizualnej) hierarchii pomiędzy uczestnikami świata DS.

Osoby zajmujące się DS często pracują bez określonych godzin, mogą pracować zdalnie, a współpraca z osobami z innych miast czy międzynarodowa nie należy do rzadkości. Mowa także o osobach pracujących komercyjnie, w tym zatrudnionych na etatach, przed pandemią COVID-19. Nawet wśród takich osób nie spotkałem się ze sztywnym systemem godzin pracy, np. 8.00–16.00. Jeżeli takie osoby mają nałożony przez pracodawcę obowiązek ośmiogodzinnego dnia pracy, to raczej samodzielnie decydują o godzinie jej rozpoczęcia, np. między godziną 7.00 a 10.00. Zdecydowanie bardziej elastyczny lub nienormowany czas pracy, a właściwie podejmowania działania podstawowego i działań wspomagających, mają osoby zajmujące się DS akademicko oraz hobbyści. Uczestnicy badanego świata cenią szczególnie swobodę czasu pracy, natomiast swobodę miejsca mogą uważać za coś zwyczajnego, ponieważ i tak niemal całość zdań wykonują w środowisku cyfrowym, za pomocą laptopa z dostępem do internetu.

Wartości efektywności i swobody przekładają się na powszechną w badanym świecie niechęć do pracy nastawionej wyłącznie na jak najszybsze wykonywanie jak największej liczby zadań. Kiedy nie ma komponentu jakości, nie ma tak cenionego przez uczestników czasu na eksperymentowanie i popełnianie błędów, na testowanie nowych narzędzi, na konsultacje z innymi data scientistami, na samorozwój. Nie spotkałem się w komercyjnym DS z praktyką wyznaczania celów o charakterze sprzedażowym, co zdaniem jednego z rozmówców {wywiad 16} przekłada się na dobrą atmosferę w jego zespole DS. Stąd w badanym świecie raczej **mało cenione są firmy, w których istnieje duża odgórna presja na liczbę wykonywanych projektów w krótkim czasie, przekładająca się na praktyki przymusowego wyrabiania nadgodzin czy pracy w dni wolne, pracy na urlopie wypoczynkowym** itp. Data scientiści postrzegają swoją pracę jako w dużej mierze twórczą i polegającą na eksperymentach, od pracodawców i przełożonych spoza świata DS oczekują zatem swobody co do organizacji pracy w czasie. Model działania – może być on określany jako „orka" lub „praca na ciągłym ASAPie[97]" – jest dla uczestników świata DS po prostu nieoptymalny, nie tak pojmują oni efektywność.

Przy tym **niektórzy uczestnicy świata społecznego DS tłumaczą, że czasem lubią lub chcą pracować kilkanaście godzin dziennie, ale nie z przymusu**. Pojawiają się odniesienia do rozwiązywania ciekawego problemu, przeżywania przygody intelektualnej, doświadczania stanu przepływu[98] (*flow*). Jednocześnie nie

97 ASAP to akronim *as soon as possible* (dosł. 'tak szybko jak to możliwe').

98 Termin *flow* stworzył psycholog Mihaly Csikszentmihalyi. Oznacza on „przepływ" jako stan doświadczenia optymalnego, polegającego na skoncentrowanym wysiłku wykonania czegoś trudnego i wartościowego. Badania wskazują, że przepływu doświadczają zarówno wspinacze (Kacperczyk, 2016: 481–483), jak i hakerzy (Zaród, 2018: 220–221).

spotkałem się z praktykami nadużywania swojego zdrowia fizycznego dla pasjonackiego poświęcenia się pracy. Osoby mówiące o kilkunastogodzinnej czy całonocnej pracy zaznaczały, że kolejnego dnia pracują krótko lub biorą wolne.

4.4.6. Racjonalność

Racjonalność w sensie wartości społecznego świata DS pojmuję jako posługiwanie się rozumem, a nie emocjami czy intuicją. Są od tego nastawienia na rozum odstępstwa, lecz ogólnie ceniona jest racjonalność, na którą składają się ścisłość myślenia i kwantyfikacja.

Ścisłość myślenia polega na logice i precyzji. Uczestnicy badanego świata oceniają ją głównie na podstawie wypowiedzi – zarówno ustnych, jak i pisemnych. Najłatwiej było mi zauważyć to na meetupach. Początkowo byłem zaskoczony dociekliwością i krytycznością pytań zadawanych prelegentom przez uczestników. Typowe są pytania w rodzaju: „Co dokładnie oznacza…?", „Zdefiniuj…", „Jaki był cel tego…?", „Dlaczego użyliście metody/technologii…?", „Po co zrobiliście…?", „Jaka była dokłada wartość metryki…?". Takie interakcje są przykładem próby sił – testowania autentyczności jednych uczestników przez drugich poprzez wskaźniki, np. znajomości narzędzi, wiedzy matematycznej, bycia na bieżąco, efektywności, bycia prawdziwie zainteresowanym czy właśnie ścisłości myślenia. Występują one także w innych sytuacjach, np. w codziennych interakcjach członków zespołów DS:

> R: […] a jeśli chodzi o humor, mmm [pauza], oczywiście dużo jest docinek związanych z precyzją wypowiedzi i, eeee, i tym, co kto miał na myśli, co wynika jakby ze [pauza], z tego, że większość ludzi jest po matematyce, albo ja właśnie po fizyce matematycznej, gdzie, gdzie, em [pauza], to czym człowieka tłuką od samego początku, to jest definicje, precyzja i tak dalej. I to później siłą rzeczy przenosi się, eee, przenosi się na całe życie. [pauza] Amm, to myślę, że to taka wyróżniająca cecha, jeżeli chodzi o humor w moim zespole, bo poza tym, to już są bardziej kwestie personalne po prostu. Relacje między nami. Czyli, czyli się śmiejemy z tego, jacy jesteśmy, kto jaki jest, a nie, eeee, a nie opowiadamy żartów o, nie wiem, regresji logistycznej czy sieciach neuronowych. {wywiad 16}

Tego rodzaju zabiegi, jak sądzę w łagodnej formie, stosowali rozmówcy/rozmówczynie wobec mnie podczas wywiadów oraz w rozmowach przed i po nich. Pytano o metodologię, np. w jaki sposób są dobierane osoby do badania, czy wyniki będą generalizowane, o cel badań, hipotezy, a także delikatnie krytykowano i proponowano różne strategie. W trakcie wywiadów często pojawiały się prośby o doprecyzowanie, zdefiniowanie lub uściślenie wypowiedzi. Podobnie w odpowiedziach rozmówczynie/rozmówcy dążyły/dążyli do wypowiedzi logicznych i precyzyjnych, kierując się wytycznymi znanymi z ilościowych dziedzin nauki, np.:

1) nie używali tzw. wielkich kwantyfikatorów (jak: „zawsze", „wszyscy", „nigdy", „nikt"), relacjonując własne obserwacje, dodawali „moim zdaniem" lub „chyba"/„raczej";

2) ostrożnie wyrażali się o związkach przyczynowo-skutkowych, akcentując, że nie świadczy o nich ani korelacja, ani następstwo czasowe;
3) wskazywali na brak informacji o czymś, czasami z zaznaczeniem, że brak dowodu nie jest dowodem braku.

Co do logiki, to mam wrażenie, że uczestnicy świata DS przeważnie stosują w komunikacji werbalnej słowa „i", „lub", „albo", tak jak operatory logiczne koniunkcji, alternatywy i alternatywy rozłącznej. Ogólnie cenione są raczej wypowiedzi ścisłe, spójne, logiczne. Uczestnicy świata DS mogliby zaakceptować postulat dążenia do maksymalizowania współczynnika przekazanej treści do liczby wypowiedzianych słów.

Kwantyfikacje to według mnie tendencja do ilościowego postrzegania świata, posługiwania się ilościowymi metrykami i danymi ilościowymi. Uczestnicy badanego świata stosują w codziennej (i niekoniecznie dotyczącej data science) komunikacji takie kategorie, jak procenty (np. 20% czasu), ilorazy (np. 3 razy szybciej, stosunek A do B), pojęcia „dużo"/„mało", „często"/„rzadko" itp. W różnych kontekstach pojawiają się odniesienia do tzw. zasady Pareto czy inaczej 80/20. Istnieje praktyka kwantyfikacji działania podstawowego, np. w kategoriach czasu pisania kodu, liczby linii kodu potrzebnych do wykonania operacji, czasu działania kodu, a przede wszystkim używanych jest wiele ilościowych metryk w analizie i modelowaniu danych – od statystycznego testowania hipotez, przez metryki dobroci dopasowania modeli, do metryk wydajności czy kosztu produkcyjnego działania modelu. Truizmem będzie stwierdzenie, że same dane traktowane są zawsze jako ilościowe – zarówno ustrukturyzowane, jak i nieustrukturyzowane.

Choć w badanym świecie cenione są dane i informacje o charterze ilościowym, to nie zgadzam się na określenie, że uczestników cechuje dataizm – ideologia zasadzająca się na wierze w obiektywność kwantyfikacji i na zaufaniu do agentów zbierających dane (van Dijck, 2014: 198). W badanym świecie występuje wiara w kwantyfikację w tym sensie, że ceni się podejście ilościowe bardziej niż inne podejścia i porządki wyjaśniania świata, jako podejście racjonalne i umożliwiające optymalizację. Świat postrzega się raczej w kategoriach obliczeniowych; data scientista „myśli liczbami" {wywiad 16}. Na pewno nie jest to ślepa wiara w dane. W świecie DS powszechna jest troska o jakość danych i o rozpoznanie obciążeń (*bias*) oraz szumów występujących w zbiorach danych. Pojawia się także troska o rozpoznawanie własnych „biasów", rozumianych jako ukryte założenia, przekonania.

Nie jest to abstrakcyjna, epistemologiczna krytyczność wobec danych jako takich, która reprezentowana jest przez płynącą z nauk społecznych i humanistycznych krytykę tzw. big data czy pojęcia danych surowych, uczestnicy świata DS wierzą bowiem, że kwantyfikacja może być cennym – przede wszystkim w kategorii efektywności – sposobem wyjaśniania świata, a teoretycznie także obiektywnym. Przy tym wierzą (a jak moim zdaniem sami by to określili: wiedzą), już na poziomie metodologicznym czy wręcz operacyjnym, że kwantyfikacja, jako np. metoda zbierania czy przechowywania danych, jest w praktyce zawsze niedoskonała. Widać to wyraźnie w pojęciach sygnału i szumu, z którego zawsze składają się

tzw. realne, brudne zbiory danych. Uczestników świata DS cechuje zatem niskie zaufanie do agentów zbierających dane, a także pojawia się ograniczone zaufanie do zdolności intelektualnych ludzi oraz siebie samej/samego. Widać jednak silne przywiązanie do Weberowskiej „intelektualnej racjonalizacji", odczarowania świata za pomocą technologii (Weber, 1989: 121–122). Niemniej w świecie DS lub wśród biznesowych rzeczników świata DS pojawiają się odwołania do magii, próby zaczarowania DS, przynajmniej w oczach klientów.

Podsumowując, **w dataizm jako pułapkę ślepego entuzjazmu metodologicznego wobec kwantyfikacji nie wpadają osoby silnie osadzone w świecie DS**. **To one lub ich rzecznicy zastawiają owe pułapki na tych, którzy mogą być zainteresowani płaceniem za usługi DS.** W biznesie istnieje pojęcie *data driven* (dosł. 'napędzanie danymi'), które uważam za klarowną deklarację dataizmu i być może wyraz skuteczności działania wspomagającego świat społeczny DS, czyli marketingowego zastawiania pułapek myślowych związanych z kwantyfikacją (por. Żulicki, 2019).

Data science jako działalność paranaukowa jest pozornie bliska pozytywizmowi, czyli orientacji racjonalnej i modernistycznej. Andrzej Szahaj (2004: 16–17) takie podejście nazywa „scjentyzmem" i uznaje je za ideologię, która zakłada zdobywanie wartościowej wiedzy wyłącznie za pomocą metod naukowych zmatematyzowanego przyrodoznawstwa. Dalej pisze o scjentyzmie jako „religii nauki", która zasadza się na oczekiwaniu, że:

> […] poznanie naukowe zbliży nas ostatecznie do poznania jakiejś absolutnej Prawdy o świecie i naszym w nim miejscu. Kult prawdy ściśle łączy się z optymizmem poznawczym: poznanie nie zna żadnych niemożliwych do pokonania barier i przeszkód, prowadzi ono prostą drogą do ostatecznego celu – poznania świata takim, jakim on jest (Szahaj, 2004: 17).

Być może ów optymizm poznawczy jest charakterystyczny dla badanego świata. Jednak **zobowiązaniem świata DS nie jest dostarczenie prawdy absolutnej o świecie – jest nim zwiększanie efektywności, rozumianej jako optymalizacja**. W tym sensie zgadzam się z Lowriem (2017: 3–9), mówiącym o algorytmicznej racjonalności osób zajmujących się DS w Rosji.

Odstępstwa od racjonalności w omówionych wyżej aspektach pojawiają się w DS jako powoływanie się na intuicję i doświadczenie w wykonywaniu działania podstawowego. Data science jest uznawane w badanym świecie za dziedzinę wymagającą podejścia nieschematycznego, twórczego, a nie odtwórczego, rozwiązywania rozmytych i trudnych do zdefiniowania problemów, eksperymentowania. Doświadczenie i nabyta wraz z nim intuicja w wykonywaniu poszczególnych elementów działania podstawowego są traktowane jako cenne atrybuty osoby zajmującej się DS, szczególnie dlatego, że doświadczenia nabyć można wyłącznie przez czas i kolejne realizowane projekty – nie ma żadnego skrótu. Doświadczenie ma pomagać również w mikroaspektach wykonywania działania podstawowego, takich jak interpretacja wykresów na etapie eksploracyjnej analizy danych:

> Odpowiednio wyszkolony statystyk może odczytać kaprysy wykresu Q-Q, tak jak szaman może odczytać wnętrzności kurczaka, przy podobnym zastosowaniu zasad naukowych. Interpretowanie wykresów Q-Q jest bardziej trzewnym (*visceral*) niż intelektualnym ćwiczeniem. Niewtajemniczeni często są omamieni (*mystified*) przez ten proces. Kluczem jest doświadczenie. (Zeileis i in., 2016: cytat 105) {obserwacja 9}

W świecie DS pojawia się przekonanie, że wiedza i umiejętności rosną wykładniczo wraz z doświadczeniem w wykonywaniu działania podstawowego, ta intuicja doświadczonych uczestników jest zatem także racjonalnie wyjaśniana.

4.4.7. Wartości społecznego świata data science – podsumowanie

Opisywane wyżej wartości, ze szczególnym naciskiem na sprawczość i efektywność oraz nieujętą przeze mnie osobno, ale bez wątpienia obecną merytokrację, można uznać za typowe dla neoliberalizmu (Gershon, 2011; Lorenz, 2012; Wrenn, 2015). Węziej postrzegam je jako wywodzące się z jednej strony z tzw. światopoglądu technokratycznego, z drugiej zaś z etosu naukowca akademickiego według Mertona.

W odniesieniu do tzw. światopoglądu technokratycznego widać bliskość wartości świata DS do tego, co Kurczewska (1997: 18–35) nazwała konstytuującym ów światopogląd „terrorem przyszłości". Terror ten polega, po pierwsze, na tym, że technokratyczna wizja, odwołująca się do nauki i techniki, jest wizją jedyną, doskonałą, prawdziwą i pewną, daną bowiem przez przyrodę – nie tworem technokratycznych myślicieli, a dziełem samej przyrody zniewalającej całkowicie przebieg i kształt życia społeczeństw ludzkich. Po drugie, co Kurczewska określa mianem poglądu merytokratycznego, uznać tę wizję za konieczną i obowiązującą mogą tylko osoby mające odpowiednio wysokie kompetencje w scjentystycznie czy pozytywistycznie postrzeganej nauce. Po trzecie, osoby te to tylko technokraci, tak więc tylko technokraci są predestynowani do roli autorytetu w kwestii przyszłej rzeczywistości – autorytetu, od którego nie ma odwołania.

> Takie przeświadczenia musiały prowadzić technokratę do przekonania, że jego wizja przyszłości ma ze wszech miar charakter obligatoryjny, a on sam jest najlepszym, niezastąpionym i „wybranym" czy „zesłanym" przez Prawdę autorytetem intelektualnym, jedynym znającym prawdziwą przyszłość (Kurczewska, 1997: 19).

Uczestnicy świata DS nie są tak opisanymi modernistycznymi technokratami, niemniej cechuje ich podobnie jednoznaczne, choć niezupełnie bezalternatywne postrzeganie przyszłości. Nie wywodzą oni jednak swojej wizji przyszłości tylko z przekonania o jej zgodności z determinującą ludzkie życie przyrodą – choć, jak pokazałem wcześniej, pojawiają się takie legitymizacje, jak np. ML będący kolejnym stadium naturalnego rozwoju ludzkości – ale **z przekonania o efektywności jako ogólnoludzkiej wartości najwyższej**. Tak więc może i dałoby się wskazać

alternatywne wizje przyszłości, w których ludzie zrezygnują ze zbierania danych cyfrowych i ich analizy oraz modelowania, ale są to wizje kompletnie nieprawdopodobne, właściwie bezsensowne z punktu widzenia uczestników świata DS, ponieważ nieefektywne, nieoptymalne i nieopłacalne w sensie ekonomicznym. Zgadzam się z poglądem Lowriego (2017), że **świat DS zamienił scjentystyczne kryterium prawdy na kryterium efektywności, wizja przyszłości nie jest zatem wizją świata od początku do końca odczarowanego, wyjaśnionego dzięki kwantyfikacji** (choć w świecie DS część uczestników dąży np. do osiągnięcia wyjaśnialności modeli ML, opracowując metody otwierania czarnych skrzynek)**, a wizją świata efektywnego i optymalnego w najlepszy możliwy sposób**.

Wymieniane przez Mertona elementy etosu naukowca: bezinteresowność, swoboda poszukiwań naukowych, merytokracja, dostęp do wyników pracy naukowej i dochodzenie do prawdy w wyniku współpracy, wskazywał zarówno Manuel Castells – jako wartości przyświecające twórcom internetu (Juza, 2016: 200–201), jak i Zaród (2018: 33, 40, 90) – jako wartości podobne do etosu hakerów.

Z wymienionych wyżej elementów etosu naukowca jedynie bezinteresowność nie ma żadnego zastosowania do wartości świata DS. Interesowność w sensie dbania jednostki o siebie – własny interes finansowy, a szczególnie samorozwój – jest nie tylko całkowicie akceptowana, ale i ceniona tak dalece, że (w przypadku samorozwoju) staje się zobowiązaniem. Niemniej pamiętać należy, że uczestnicy zdecydowanie wyżej cenią uzasadnianie indywidualnych wyborów jako „bycie prawdziwie zainteresowanym" tą czy inną domeną, metodą, technologią niż uzasadnianie tylko korzyści finansowych. Pokazuje to, iż w badanym świecie silna jest kultura indywidualizmu samorealizacji, gdzie sukces osobisty uznawany jest za egorealizację (Jacyno, 2007: 242–243). Indywidualne wybory uczestnika świata DS, uzasadniane tylko względami finansowymi, będą zatem w badanym świecie uważane za prowadzące do „toksycznego sukcesu" (Jacyno, 2007: 242–243) – takiego, w którym życie jednostki jest niepełne, a ona sama zaniedbana, gdyż zrezygnowała z siebie. Wybory uzasadniane głównie ciekawością, a także pragnieniem samorozwoju będą postrzegane jako samorealizacja, czyli coś prowadzącego do sukcesu zdrowego, zgodnego ze sobą.

Kolejna różnica jest taka, że w świecie DS ceniona jest współpraca, szczególnie na poziomie indywidualnym, mniej zaś na poziomie organizacji, jednak jej celem nie jest dochodzenie do prawdy, a zwiększanie efektywności (a celami pośrednimi współpracy są ponownie: samorozwój, zwiększanie własnej efektywności, budowanie marki osobistej, budowanie sieci kontaktów i tym podobne indywidualne korzyści poboczne).

Badany świat DS jest bardzo merytokratyczny i jestem przekonany, że równanie zaproponowane w 1958 roku przez Michaela D. Younga: *IQ* + *effort* = *merit* (Jackson, 2001; Yair, 2007) zdobyłoby w tym świecie akceptację (nie tylko z uwagi na elegancką formę matematyczną). Widać w badanym świecie wielkie przywiązanie do wartości indywidualnego wysiłku (*effort*): indywidualnej nauki, samodzielnego

rozwiązywania napotykanych problemów technicznych i metodologicznych, doświadczenia, czasu poświęconego na wykonywanie kolejnych projektów. Pojawiają się jednak także odwołania do IQ, a właściwie szeroko pojętych dyspozycji indywidualnych, nawet wrodzonych – zdolności matematycznych czy intelektualnych w ogóle, „ciekawskości" jako posiadanego typu myślenia, bycia zainteresowanym matematyką lub programowaniem od dziecka, zdolności do szybkiego uczenia się. Może to prowadzić do merytokracji w sensie elitarnej klasy społecznej (Yair, 2007). Mam na myśli praktykowane przez część uczestników samopostrzeganie badanego świata jako wysoce obdarzonej różnego rodzaju kapitałem sprawczej elity, nagrodzonej ową elitarną pozycją przez społeczeństwo za swoje unikalne zdolności i wyjątkowy (ale i wyjątkowo dobrze zoptymalizowany) wysiłek, a także samopostrzeganie się indywidualnego, szczególnie dobrze osadzonego w świecie DS uczestnika jako elitarnego dobra, rzadkiego w kontekście rynku pracy i w odniesieniu do światów, na których przecięciu DS powstało (matematyka akademicka, ilościowe akademickie badania empiryczne, informatyka) oraz świata, z którym bezustannie się przecina (biznes).

Swoboda poszukiwań naukowych ceniona jest w zmienionej postaci: jako swoboda w zajmowaniu się takimi projektami, jakie ciekawią indywidualnych uczestników, swoboda w eksperymentowaniu i popełnianiu błędów w trakcie realizacji projektów i swoboda w korzystaniu z takich narzędzi, jakie uczestnik uznaje za najbardziej mu odpowiadające i dopasowane do rozwiązywanego problemu, a więc w konkretnej sytuacji najbardziej efektywne. Istnieje napięcie pomiędzy światem DS a nauką akademicką w tym sensie, że niektórzy uczestnicy świata DS (np. doktoranci lub byli akademicy) mogą podawać w wątpliwość to, czy DS, czy akademia dają jednostce większe możliwości tak pojętej swobody. Więcej o tym powiem, pokazując procesy stawania się uczestnikiem świata DS.

Dostęp do wyników pracy naukowej i nie tylko naukowej jest powszechnie akceptowany w świecie DS. Otwartość kodu (*open source*!) i metod, powszechne praktyki dzielenia się wiedzą, konsultowania się uczestników między sobą są uważane za wartościowe, bo sprzyjają efektywności, swobodzie, samodzielności, samorozwojowi oraz zwiększają sprawczość. Udostępnianie wyników własnych prac, np. w postaci pakietów, należy do typowych działań wspomagających w badanym świecie i uznawane jest za element strategii budowania własnego wizerunku i prestiżu, marki osobistej, upubliczniania swojej pracy w celu zaświadczenia o swoim doświadczeniu, wiedzy i umiejętnościach oraz zaświadczenie o własnej autentyczności jako uczestnika świata DS. Są to także elementy strategii budowania wizerunku organizacji – firm, zespołów DS czy jednostek akademickich.

Dobrze pasuje tutaj koncepcja późnej nowoczesności Anthony'ego Giddensa, Scotta Lasha i Ulricha Becka, którzy, jak zwięźle ujął to Piotr Sztompka, uznają, iż „nowoczesność bynajmniej nie przeminęła, lecz przeciwnie, występuje obecnie w najbardziej wyrazistych formach" (Sztompka, 2007: 576). Opisywane

przeze mnie wartości świata DS można by wywodzić nie tylko z etosu naukowca, ale i z cech nowoczesności Augusta Comte'a, m.in. pozytywizmu jako sposobu myślenia bazującego na metodologii nauk przyrodniczych i ścisłych, organizacji pracy nastawionej na zysk i efektywność, stosowania nauki i technologii w procesach produkcyjnych (choć nie dotyczy to bezpośrednio wartości, to nawet Comte'owska koncentracja siły roboczej w centrach miejskich jest zgodna z charakterystyką badanego świata DS). Zgadzam się ze Sztompką (2007: 559), że: „dostrzeżone przez Comte'a rysy nowoczesności znajdą się jak dotąd […] w każdym podobnym katalogu".

Patrząc na systemy budowane z udziałem zespołów DS jako na wdrożenia socjotechniczne, tzw. maszyny społeczne, zgadzam się także z Krzysztofkiem, iż:

> […] maszyny społeczne nie mogą funkcjonować w pustce aksjonormatywnej jako twory nastawione tylko na maksymalizację produktywności. Jeśli mają być osadzone tylko w logice społeczeństwa kierowanego, to będzie to konstruktywizm z określoną narracją ideologiczną. A może one same wymuszą *autopoiesis*, samoorganizację, samoregulację, agregowanie większych całości *bottom up*, czyli mielibyśmy wizję kontynuacji społeczeństwa liberalnego z indywiduum jako jego znakiem rozpoznawczym (Krzysztofek, 2011: 139–140).

Jako szeroką ramę aksjonormatywną, niebędącą tylko ramą dla samego świata społecznego DS (bo w nim najważniejszą wartością jest efektywność), ale ramą umożliwiającą i podtrzymującą istnienie świata DS, wybieram raczej drugą z wymienionych przez Krzysztofka – liberalizm/neoliberalizm. Przynajmniej w komercyjnej sferze działalności DS i wtedy, gdy dane dotyczą ludzi, projekty DS nastawione są głównie na personalizowanie. Jest to owoc liberalnego kultu wolnej, racjonalnej jednostki, w którym wyborca wie lepiej, a klient ma zawsze rację itd. (por. Harari, 2018: 281).

Rozdział 5
Dynamika społecznego świata

Pozostając w uniwersum podstawowych pojęć przyjętej ramy teoretycznej, opiszę dynamikę społecznego świata DS – zachodzące w nim procesy oraz areny tego świata.

Nie oznacza to, że elementy badanego świata są statyczne. Są one w koncepcji społecznych światów/aren traktowane nie tylko jako zmieniające się w czasie, ale także niestabilne, rozmyte i zamazane w sensie ontologicznym, tak jak i cały społeczny świat/arena (Clarke, 1991; Kacperczyk, 2016). Mowa o dynamice, czyli procesach i arenach społecznego świata, które także cechuje rozmycie. Są one jednak inne niż elementy społecznego świata, bo dynamiczne: proces polega na zmianie pomiędzy stanami lub etapami, arena polega zaś na sporze.

5.1. Procesy zachodzące w społecznym świecie data science

Technologie są ściśle związane z zachodzącymi w świecie społecznym procesami. W tym podrozdziale proponuję syntetyczne i abstrakcyjne ujęcie procesów: wyznaczania granic społecznego świata, legitymizacji i zaświadczania o autentyczności oraz segmentacji świata społecznego (w tym profesjonalizacji, będącej jedną z konsekwencji segmentacji). Realizuję zatem drugi z celów szczegółowych pracy.

5.1.1. Wyznaczanie granic społecznego świata

Wiele ze zidentyfikowanych procesów mógłbym opisywać jako wyznaczanie granic społecznego świata. Tak samo jest w przypadku aren. Należy jednak zacząć od legitymizacji i zaświadczania o autentyczności. Praktyki te są nastawione – choć niewyłącznie – na obronę granic społecznego świata przed „podobnymi innymi"

(Kacperczyk, 2016: 38–46) oraz na określanie innych uczestników i samookreślanie się jako uczestnicy świata społecznego.

Przechodząc do procesów segmentacji oraz aren, można je uznać za powiązane ze sobą dynamiczne składowe społecznego świata. Są one z jednej strony wyrazem tego, że uczestnicy świata pracują bezustannie nad definiowaniem jego granic (w sensie makro – tzn. co ten świat odróżnia od innych i w sensie mikro – czy konkretna osoba, także ja, jest uczestnikiem owego świata i w jakim stopniu). Z drugiej strony wskutek dokonywania się segmentacji oraz zachodzącego na arenie sporu redefiniowane są nowe granice „starego" świata i „nowych" subświatów.

Uczestnicy wciąż pracują nad określaniem granic, dlatego że granice te są płynne i niejednoznaczne.

Skupię się na tym, **jak odbywa się wchodzenie w granice społecznego świata DS** w Polsce – przekraczanie bariery efektywnej komunikacji (Kacperczyk, 2016: 56–57).

Opiszę pięć niezupełnie rozłącznych ścieżek wejścia do świata DS:

1) „młodzi absolwenci matematyki/statystyki";
2) „z programowania do DS";
3) „robiliśmy DS, zanim to się tak nazywało";
4) „eksperci domenowi przechodzą do DS";
5) „z doktoratu do DS".

Młodzi absolwenci matematyki/statystyki to osoby najczęściej rozpoczynające pracę na stanowisku data scientisty lub zbliżonym w trakcie studiów pierwszego stopnia (rzadziej drugiego) i kontynuujące ją po uzyskaniu dyplomu. Mogą planować karierę w DS już od pierwszych lat studiów i od tego czasu brać udział w kursach MOOC, stażach czy praktykach, uczestniczyć w wydarzeniach świata DS i organizować je, a w ramach studiów wybierać związane z DS przedmioty kierunkowe i specjalizacje. Osoby te uważają DS za wybór naturalny dla matematyków/statystyków i mogą sądzić, że posiadane wykształcenie stanowi o ich przewadze na rynku pracy DS. Przewaga ta jest kwestionowana przez inne pozycje w badanym świecie, które wyższą wartość przyznają np. umiejętnościom koderskim, doświadczeniu zawodowemu, rozumieniu problemów biznesowych.

Ścieżka ta dotyczy głównie osób kończących kierunek matematyka. Mogą one uważać się za elitę w elicie (jaką jest świat DS). Matematykę uznają za kierunek studiów trudny, wymagający pracy i wysiłku umysłowego. Uważają, że matematyka to najtrudniejszy, kluczowy i **niezmienny** fundament przetwarzania, analizy i modelowania danych dowolnymi metodami i z użyciem dowolnych narzędzi (które w przeciwieństwie do matematycznych fundamentów zmieniają się szybko, ale za to względnie łatwo się ich nauczyć).

> R: Zainteresowanie to było takie naturalne połączenie, z tym że zawsze się interesowałam matematyką, eee, i no już w liceum i wcześniej, poszłam właśnie potem na studia matematyczne, eee, więc myślę, że to jest jeden z najfajniejszych początków do data science, właśnie wiedza ta taka ustrukturyzowana, wiedza matematyczna, potem łatwo można wyjść, rozumiejąc te algorytmy i metody, nauczyć się samych narzędzi, już takich informatycznych, już nie jest trudno. {wywiad 4}

Zbliżona ścieżka może dotyczyć osób kończących kierunki z dziedzin obliczeniowych nauk empirycznych, jak ekonometria, bioinformatyka, geoinformacja, ale raczej nie występuje wśród nich przekonanie o charakterystycznej dla matematyków elitarności. Zarówno osoby kończące takie kierunki, jak i matematykę mają nierzadko epizody pracy jako programiści (lub w ogóle pracy w IT) lub epizody ze studiami doktoranckimi. Opisywana ścieżka splata się zatem z dwiema innymi – „z programowania do DS" oraz „z doktoratu do DS".

„Z programowania do DS" w sensie ilościowym może być ścieżką najczęściej występującą. W ankietach środowiskowych wykształcenie informatyczne deklarowano najczęściej – od nieco ponad 28% odpowiedzi respondentów w badaniu Kaggle 2017, do około 54% w ankiecie Stack Overflow 2018. Rosnąca w DS popularność języka Python (Nunns, 2017; Piatetsky, 2017b, 2018c; Strong, 2018; Borowiecki, Mieczkowski, 2019: 36–37) oznacza zapewne napływ osób z wykształceniem informatycznym/programistów do DS.

Osoby wchodzące do świata społecznego DS tą ścieżką uzasadniają własne decyzje koncentrujące się wokół kategorii ciekawością, samorozwojem, elitarnością oraz przeciwstawianiem sobie działalności badawczej i inżynieryjnej. W uproszczeniu bycie programistą jest **nudne**, a data scientistą **ciekawe**; programowanie jest łatwiejsze, dostępne, egalitarne, natomiast DS jest trudne, niedostępne, elitarne; programista jest „zwykłym" inżynierem wykonującym rutynowe prace, data scientista robi coś twórczego, odkrywczego, niestandardowego.

Osoby z dużym doświadczeniem programistycznym mogą być szczególnie cenione w DS – zarówno symbolicznie, jak i w pieniądzach – za wkład w dostarczanie (tzw. dowożenie) działających produktów (sprawczość jako wartość!). Wokół pracy, koncentrującej się na wdrożeniach produkcyjnych ML, wyrasta jeden z młodszych subświatów profesjonalnego świata DS: ML engineering (inżynieria ML, także DataOps). Spośród osób znanych publicznie w polskim świecie DS tę ścieżkę przeszedł m.in. Vladimir Alekseichenko.

„Robiliśmy DS, zanim to się tak nazywało" to ścieżka, z której wykorzystaniem nie tyle ludzie wchodzą do DS, co DS przychodzi do nich. Mowa o osobach zajmujących się od co najmniej kilkunastu lat przygotowaniem, analizą i modelowaniem danych, m.in. z użyciem języków programowania, w takich dziedzinach, jak np. bankowość, ubezpieczenia, telekomunikacja, energetyka. Mogą to być ludzie, którzy zdobyli stopień doktora i pracowali naukowo w dziedzinie matematyki stosowanej/statystyki lub w innych obliczeniowo zorientowanych dziedzinach. Takie osoby, określające się niegdyś np. jako statystycy, specjaliści od data miningu, specjaliści od modelowania ryzyka, mogą obecnie uważać się za data scientistów.

Cześć z nich współtworzyła świat społeczny DS, część dołączyła do świata, zauważając, że jest bliski ich działaniu i wartościom. Pojawia się tu tzw. **rebranding statystyki**, czyli zorientowane marketingowo nazwanie na nowo znanej działalności w celu wywołania wrażenia nowości, wyjątkowości, otwarcia dotąd niedostępnych możliwości, a w konsekwencji zdobywania nowych rynków, klientów

i zwiększanie zysków finansowych. Niemniej nie wszyscy robiący DS zanim się ono tak nazywało, czyli osoby o najdłuższym doświadczeniu zawodowym, uzyskują najwyższe zarobki.

Statystycy zajmujący pozycję w dyskursie – „DS to tylko rebranding tego, co robiliśmy od 30 lat" – będą odcinać się od bycia uczestnikami świata DS i określać DS mianem chwilowej mody: tak, jak minęła moda na data mining, tak minie moda na DS, statystyka zaś pozostanie. To obrona granic społecznych światów przed „podobnymi innymi" (Kacperczyk, 2016: 46). Taka pozycja jest reprezentowana przez część środowisk akademickich oraz branże o konserwatywnym podejściu do pracy z danymi, np. branżę farmaceutyczną. Dyskusje o relacji między DS a statystyką trwają od końca lat dziewięćdziesiątych. Postulowano przemianowanie statystyki na DS (Cao, 2017a: 7) w celu zorientowania tej dziedziny bardziej empirycznie (Wu, 1997), rozszerzenia statystyki o zainteresowanie problemami obliczeniowymi i współpracą z informatykami (Cleveland, 2001), zaadaptowania do statystyki odmiennego podejścia do modelowania – tzw. kultury modelowania algorytmicznego (Breiman, 2001), uznania statystyki i ML za odgrywające centralną rolę w DS (van Dyk i in., 2015). Mówi się zarówno, że DS to tylko przemianowanie (*rebranding*) statystyki, jak i przeciwnie – że statystyka jest najmniej ważną częścią DS, a DS bez statystyki jest nie tylko możliwe, ale i pożądane (Donoho, 2015: 4–7). Kształcenie na potrzeby DS jest odmienne od oferowanego obecnie nauczania statystyki (Bryan, Wickham, 2017). Dyskusje, rozważania i żarty o relacji statystyki i DS podejmowano na blogach i w mediach społecznościowych (Yau, 2009; Wills, 2012; Big Data Borat, 2013; Hyndman, 2014; Jarvis, 2014; Taylor, 2016). W branży farmaceutycznej centralną wartością jest, jak sądzę, dokładność analiz i z uwagi na szybko zmieniające się środowisko obliczeniowe (np. wersje pakietów) nie zaadaptowała ona szeroko języków R ani Python. Pozostaje przy oprogramowaniu SAS (z językiem SAS 4GL), zapewniającym powtarzalne działanie skryptów nawet z lat osiemdziesiątych. Nie zaadaptowała szeroko metod ML. Istnieją jednak firmy z branży medycznej identyfikowane z DS, np. mająca oddział w Poznaniu szwajcarska Roche.

Inni doświadczeni statystycy, głównie ci, którzy współtworzyli świat społeczny DS, mogą być w badanym świecie szczególnie cenieni właśnie ze względu na doświadczenie. Osoby te nierzadko zbudowały wizerunek i uzyskały status autorytetów świata społecznego poprzez swoje wystąpienia publiczne, książki, kursy, działalność naukową i dydaktyczną, ale **doświadczenie jako takie** w badanym świecie może być cenione wysoko. Data science to młody świat i osoby doświadczone są w nim rzadkim dobrem. Doświadczone osoby to te mające za sobą rozwiązanie wielu realnych problemów i znające wiele narzędzi, nie tylko obecnie najmodniejsze.

Ścieżka ta może splatać się z inną – „z doktoratu do DS", ale jest jakościowo różna. Dotyczy ona wyłącznie osób wykonujących działanie zbliżone do działania podstawowego świata DS, wcześniej niż zaczął się ów świat kształtować (jako nazwane uniwersum dyskursu). Spośród osób znanych publicznie w polskim świecie DS tę ścieżkę przeszli m.in. Przemysław Biecek i Artur Suchwałko.

„Eksperci domenowi przechodzą do DS" to ścieżka, która pokazuje, że deklaracje typu „dane mówią same za siebie" są przestarzałe. Pojawiają się one w innych niż DS środowiskach, przez które przechodzi fala fascynacji podejmowaniem decyzji bazujących na danych, a nie na teorii/opiniach, i większość omawianej krytyki pozostaje aktualna (Żulicki, 2019). Dane mówiące jakoby same za siebie były strategią legitymizacji wczesnego etapu upowszechniania się tzw. big data (w żargonie nietechnicznym), ale obecnie w badanym świecie DS straciły znaczenie. Być może deklaracje te były nastawione na zewnątrz i należy je uznać nie tylko za legitymizację DS, ale węziej – za marketing nastawiony na wywoływanie wrażenia o pojawieniu się nowych, niesamowitych możliwości (jak przy rebrandingu statystyki). Obecnie do świata społecznego DS i do zatrudnienia jako data scientiści wchodzą osoby specjalizujące się w konkretnej domenie, z której pochodzą dane. Uczestnicy osadzeni w badanym świecie uznają, że współpraca z ekspertem domenowym jest konieczna do wykonania projektu DS (Alekseichenko, 2019a).

Data scientiści będący wcześniej znawcami domeny mogą być uznawani za wartościowych członków zespołów DS. Jest tak raczej w dużych firmach, gdzie zespół/dział DS pracuje dla klienta wewnętrznego, w jednej/niewielu zbliżonych domenach. Osoby przechodzące omawianą ścieżkę wykonują ogrom pracy nad swym wizerunkiem i tożsamością, używając praktyk legitymizacyjnych i będąc poddawane testom na autentyczność. Muszą udowadniać bycie ekspertem od stosowania danych z użyciem programowania (czyli data scientistą), a odciąć się od bycia ekspertem dziedzinowym biegłym w pracy z danymi. Jeżeli takie osoby pozostają przy pracy w jednej domenie, kwestionowany może być ich samorozwój.

Ścieżkę **„z doktoratu do DS"** przechodzą osoby mające stopień doktora, pracujące na uczelniach, będące w trakcie doktoratu lub po porzuceniu studiów doktoranckich, także bez obrony pracy doktorskiej. Należy odróżnić osoby, które odeszły z akademii, od tych, które równocześnie robią kariery akademickie i uczestniczą w świecie DS.

Te drugie to szerokie spektrum uczestników – od tych, którzy „robili DS, zanim to się tak nazywało", poprzez osoby jednocześnie piszące doktorat/zatrudnione na uczelniach i na stanowiskach data scientistów lub świadczące usługi DS dla różnych klientów w modelu *freelance*. Dotyczy to raczej naukowców z dziedziny matematyki/statystyki lub informatyki. Charakterystyczne będzie np. realizowanie grantu naukowego w partnerstwie uczelni z firmą, publikowanie artykułów i występowanie na konferencjach naukowych oraz na wydarzeniach i w mediach badanego świata z podwójną lub wymienną afiliacją. Granice są tu płynne i zmienne w czasie. Nawet maleńkie subświaty spierają się o konkretne praktyki lub konkretne osoby, negocjując nieustannie granice tego, gdzie zaczyna się i kończy bycie uczestnikiem świata DS, „prawdziwym" data scientistą czy naukowcem. Spektrum zamykają osoby, których uczestnictwo w świecie DS jest kwestionowane, np. naukowcy tytułujący się jako data scientiści, a niemający doświadczenia

w komercyjnych zastosowaniach ML, będący wyłącznie dydaktykami. Te praktyki mogą być przez uczestników osadzonych w świecie DS określane mianem podszywania się pod świat DS w celach marketingowych, jak rekrutacja na płatne studia zaoczne lub podyplomowe – to tzw. *fake data science*. Inną kategorię stanowią naukowcy używający typowych dla DS narzędzi i metod w dziedzinach obliczeniowych. Ci mogą odcinać się lub w ogóle nie interesować światem społecznym DS lub wręcz przeciwnie – to właśnie spośród tej kategorii naukowców lub niedoszłych adeptów akademii rekrutują się osoby porzucające ją na rzecz pełnego i oczywiście zawodowego uczestnictwa w świecie DS.

Wracając do tych, którzy odeszli z akademii do DS, w ich narracjach widać odwołania do kategorii **ciekawości, sprawczości, swobody i samorozwoju**. Wartości te przyciągają młodych, uznających się za sprawnych intelektualnie i obliczeniowo ludzi do rozpoczęcia kariery w świecie nauki akademickiej. Po czasie, nierzadko krótszym niż ukończenie rozprawy doktorskiej, przychodzi **rozczarowanie wartościami akademii**. Osoby identyfikujące się z ciekawością, sprawczością, swobodną i samorozwojem nabierają przekonania, że świat akademii – będący dla nich grupą odniesienia normatywnego – kieruje się wartościami zgoła innymi. Uczelnie jawią się im jako „**feudalne instytucje, w których** […] **dominują rozgrywki polityczne** [podkr. R.Ż.] i żee to zżera bardzo wiele zasobów" {wywiad 21}. Nową grupą odniesienia normatywnego może stać się świat DS. Mówię o osobach już posługujących się narzędziami i metodami świata DS i nierzadko będących już na obrzeżach świata DS, np. bywają one na wydarzeniach świata DS, przeszły kilka kursów MOOC i zaczęły wprowadzać poznane narzędzia i metody do swoich analiz naukowych, mogą uczestniczyć w hobbystycznych projektach DS lub brać udział w konkursach Kaggle. Nazywam ich (oraz im podobnych) wannabesami, szczególnie jeżeli są to osoby chcące wejść w DS zawodowo.

Do DS przyciągają ich także inne wartości nieakademickie – przede wszystkim efektywność. Jedna z rozmówczyń {wywiad 21} wskazała, że uczelniane rozgrywki polityczne i feudalna organizacja pracy są nieefektywne, bo kontrproduktywne wobec tego, co uważa ona za cel działania akademii. Nie jest to ani jednostkowe, ani charakterystyczne dla Polski. Powstawały artykuły poradnikowe pokazujące, jak przejść z akademii do DS, istnieją kursy DS przeznaczone dla osób z doktoratem – w Europie np. S^2DS (Science to Data Science), w USA i Kanadzie Insight Data Science Fellows Program (Migdał, 2016). Przykłady można by mnożyć. O'Neil przeszła z akademii do DS (by później krytykować konsekwencje społeczne stosowania modeli), ponieważ na uczelni dokuczało jej „ślimacze tempo prac", chciała być „częścią postępującego szybkim krokiem prawdziwego świata" oraz „pokazać ludziom, że mój intelekt może przekładać się na pieniądze" (O'Neil, 2017a: 64).

Wywiady z osobami przechodzącymi opisywaną ścieżkę zbaczały w stronę krytyki współczesnej akademii. W trakcie badań dowiedziałem się m.in. o publicznej zbiórce pieniężnej zorganizowanej przez doktor nauk humanistycznych Katarzynę Dawidowicz na wydanie jej książki *Koszmar doktoratu*. Książka zawiera wywiady

z doktorantami i doktorami relacjonującymi poniżanie i wykorzystywanie ich, np. przez promotorów[1].

W języku świata DS doktoranci/doktorzy **monetyzują swe naukowe umiejętności** metodologiczne, analityczne, obliczeniowe, aplikując je na gruncie DS. Osoby przechodzące omawianą ścieżkę mogą być cenione w badanym świecie za rygor metodologiczny, a także ciekawość i tendencje do eksploracji. Nie może to być zagłębianie się w eksplorację, gdyż centralną wartością świata DS jest efektywność. W DS mówi się, że naukowcy pracują nieefektywnie, powoli i bez presji, cyzelując szczegóły. „Prawdziwe" DS to praca efektywna i „wystarczająco dobra", nastawiona na osiągnięcie optimum między zużytymi zasobami a osiąganym efektem, stanowiącym rozwiązanie realnego problemu. Tym samym osoby rekrutujące się z nauki akademickiej mogą być poddawane próbom sił pod kątem efektywności i sprawczości działania.

Spośród osób znanych publicznie w polskim świecie DS ścieżkę „z doktoratu do DS" przeszedł m.in. Piotr Migdał.

Każdą z wyróżnionych ścieżek można opisać w odniesieniu do różnic pomiędzy wartościami świata społecznego DS a wartościami w szerokich światach społecznych (będę je nazywać obszarami)**, z którymi DS przecina się i przenika**, a w pewnym stopniu z nich się wywodzi. Chodzi o obszary: nauki akademickiej (matematyki i empirycznych nauk obliczeniowych), technologii informacyjnych (dalej IT; szczególnie tzw. programowania oraz baz danych) oraz biznesu (rozumianego jako działalność komercyjna w rozmaitych dziedzinach). Określam wagę, jaką z perspektywy uczestników świata DS wyróżnione obszary (DS, nauka, IT, biznes) nadają siedmiu wartościom badanego świata. Na potrzeby porównania przyjmuję, że świat DS jako całość nadaje wysoką wagę (na rysunku 5.1 oznaczoną „3") każdej z siedmiu wyróżnionych wartości, tj. efektywności, ciekawości, samorozwojowi, swobodzie, sprawczości, racjonalności i samodzielności.

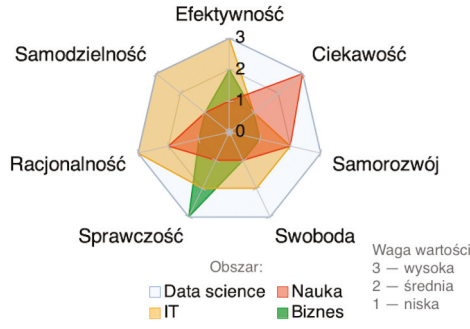
Obszar nauki akademickiej jest przez uczestników świata DS uważany za wysoko ceniący ciekawość, średnio (waga „2") samorozwój i racjonalność, a nisko (waga „1") pozostałe z wyróżnionych wartości. Dla uczestników świata DS perspektywa zrealizowania „ciekawego projektu" może zatem stanowić o atrakcyjności nauki akademickiej. Niska waga sprawczości spowodowana jest tym, iż nauka akademicka zajmuje się głównie wytwarzaniem tekstów mających niewielki wpływ na obszary poza akademią, a nie zajmuje się budowaniem „czegoś działającego". Niskie wagi efektywności, samodzielności i swobody spowodowane są hierarchiczną, „feudalną" organizacją pracy na uczelniach i głównie publicznym jej finansowaniem.

Obszar IT jest w perspektywie wartości świata DS najbardziej z nim zbieżny. Technologie informacyjne są przez uczestników świata DS postrzegane jako wysoko ceniące racjonalność, samodzielność i efektywność, średnio zaś samorozwój, swobodę i sprawczość. Niska jest waga ciekawości – jej powodem jest postrzeganie

1 Zbiórka na pomagam.pl (https://pomagam.pl/KoszmarDoktoratu, dostęp: 2.06.2020) zakończyła się zebraniem 5325 zł z planowanych 14 000 zł, a Dawidowicz wydała książkę samodzielnie.

informatyki czy konkretnie tworzenia oprogramowania tak jak zwykłej, nudnej pracy inżynieryjnej, w której rozwiązuje się problemy za pomocą dość schematycznych (w porównaniu do DS) metod i narzędzi. Uczestnicy świata DS, tak samo jak programiści czy ogólnie informatycy, postrzegają siebie jako osoby techniczne. Oznacza to nie tylko osoby biegłe techniczne, ale i wartościujące rzeczywistość w kategoriach efektywności, racjonalności i samodzielności. Może za wartość charakterystyczną dla światów technicznych należałoby uznać także samorozwój i sprawczość. Niemniej uczestnicy świata DS uważają swoją dziedzinę za bardziej sprzyjającą samorozwojowi niż informatyka. Wskazują, iż DS jest na znacznie wcześniejszym niż informatyka stadium rozwoju, rozwija się o wiele bardziej dynamicznie i w pewnych zastosowaniach wypiera rozwiązania informatyczne bazujące na zapisywanie reguł w kodzie, stanowiąc nowe i lepsze rozwiązanie wielu problemów. Innym wątkiem jest większe ryzyko niepowodzenia w eksperymentalnych projektach DS niż w szablonowym tworzeniu oprogramowania. Podobnie uznaje się DS za bardziej od informatyki sprawcze – głównie z uwagi na stosowanie ML, a nie „tylko" zapisywanie w kodzie reguł zaplanowanych przez ludzi.

Biznes jest przez uczestników świata DS postrzegany jako zbieżny z DS jedynie w wymiarze wysokiego wartościowania sprawczości. W świecie DS za kluczowy czynnik sprawczy mogą być uznawane pieniądze. Od budżetu biznesu-klienta zależy, jaki wpływ na rzeczywistość będzie miał realizowany projekt DS (np. jaki zespół ludzi i ich kompetencji można opłacić, jaka infrastruktura obliczeniowa będzie dostępna, jaka będzie skala i zakres wdrożenia produkcyjnego tworzonych modeli). Z punktu widzenia osób osadzonych w świecie DS biznes średnio ceni efektywność, a nisko pozostałe z siedmiu wartości badanego świata. Uczestnicy świata DS nisko oceniają także wartość racjonalności w biznesie, uznając, że biznes chętnie deklaruje bycie „napędzanym danymi" (*data-driven*), ale faktycznie decyzje zapadają z uwagi na przesłanki dalekie od racjonalności – gry polityczne lub personalne, przeczucia i intuicję, stereotypy, kaprysy decydentów.



Uwaga: oś liczbowa reprezentuje wagę wartości w obszarach z perspektywy uczestników świata DS.

Rysunek 5.1. Porównanie obszarów w perspektywie wartości społecznego świata – DS a nauka, DS a IT, DS a biznes

Źródło: opracowanie własne za pomocą Libre Office Calc.

Za ważny element w sporze o granice społecznego świata DS uznaję to, iż jego **uczestnicy postrzegają się w dużej mierze jako osoby techniczne**. Wskazuje na to fakt, że widzą oni dużą zbieżność w tym, jak ważne są wartości ich świata i obszaru IT. **Niemniej to nie tylko osoby techniczne** – w ocenie uczestników **świat DS jest jedyną w swoim rodzaju hybrydą**, która w niespotykany dotąd i nieporównywalnie do niczego efektywny sposób pozwala na rozwiązywanie realnych problemów. Robi to nie tylko za pomocą najnowocześniejszych technologii, ale także najlepszej aparatury matematycznej i metodologii empirycznych nauk obliczeniowych. Data science jako świat społeczny powstało i dynamicznie trwa na przecięciu trzech obszarów: nauki, informatyki i biznesu, a jego uczestnicy uważają siebie za elitę, która nie należy do żadnego z tych obszarów, a od każdego z nich jest „lepsza" (przyjmując za punkt odniesienia wartości świata DS).

Warto zobaczyć, jakie opowieści budują uczestnicy tego świata, by uzasadnić tę „lepszość" jako sens istnienia społecznego świata oraz jego normatywny ład (por. Kacperczyk, 2016: 56).

5.1.2. Legitymizacja i zaświadczanie o autentyczności

Światy społeczne uzasadniają swe istnienie. Czynią to zarówno światy „młode", jak i te bardziej ustabilizowane, konstruując opowieści dotyczące ich działalności oraz tożsamości. Praktyka ciągłego konstruowania uzasadnień wynika „z potrzeby uzyskania poparcia dla własnych działań i zawiera roszczenie zaakceptowania przez otoczenie" (Kacperczyk, 2016: 55).

Uzasadnienia mogą być wskazaniem na wyjątkowy sposób wykonywania działania podstawowego oraz na konkretną technologię, jaką dysponuje dany świat lub segment świata. W skali makro zachodzi proces „wyznaczania granic prawidłowej działalności" (gdzie negocjowane jest to, kogo wolno uznawać za uczestnika i „prawdziwego" uczestnika świata społecznego). W skali mikro tworzonych jest wiele objaśnień, uzasadnień, wymówek, pozwalających pojedynczym aktorom kontynuować działanie podstawowe (Kacperczyk, 2016: 36). Legitymizacja i zaświadczanie o autentyczności realizowane są głównie poprzez zabiegi komunikacyjne o charakterze objaśniania, neutralizowania lub teoretyzowania.

5.1.2.1. Dużo danych

Najpopularniejszym, relatywnie najstarszym i uznawanym za oczywistą prawdę o powstaniu świata społecznego DS (nie tylko polskiego, ale i globalnego) jest objaśnienie: **DS powstało, ponieważ z uwagi na gwałtownie rosnącą ilość dostępnych danych i rozwój mocy obliczeniowej komputerów przy jednoczesnym spadku ich kosztów** (zarówno danych, jak i mocy) **możliwe, a zarazem opłacalne stało się rozwiązywanie realnych problemów z wykorzystaniem metod ML**. Uczenie maszynowe znane jest od lat pięćdziesiątych XX wieku, ale w badanym

świecie objaśnia się, że dopiero w pierwszej dekadzie XXI wieku można było je realnie wykorzystać – jakoby przekroczony został wówczas punkt krytyczny, poniżej którego dostępnych danych było za mało, ich przechowywanie za drogie i zbyt pracochłonne, moc obliczeniowa zbyt kosztowna oraz powolna i mało kto wierzył nie tylko w opłacalność, ale w ogóle w możliwość efektywnego używania ML w rozwiązywaniu realnych problemów.

To wpływowe objaśnienie stosowano od czasów popularności terminu „data mining" i niemiłosiernie eksploatowano w okresie popularności nietechnicznego ujęcia terminu „big data". Akcentowano szybko rosnącą ilość dostępnych danych, które zaczęto przedstawiać jako **zasób do monetyzowania**. Już w 2012 roku opisywany zabieg sportretował Scott Adams (2012), odnosząc się z humorem także do Orwellowsko-panoptycznej narracji o wszechwiedzących danych, wyśmiewając metaforę „chmury" oraz trafnie pokazując wiarę w wielkie dane jako mesjasza, który za opłatą zbawi biznes od bankructwa.

Opisywane objaśnienie eksploatowano w entuzjastycznej *Big Data. Rewolucja...* (Cukier, Mayer-Schönberger, 2014). Nakreślono „epokę wielkich danych", poczynając od przykładu astronomii, gdzie dzięki rozpoczętemu w 2000 roku programowi badawczemu Sloan Digital Sky Survey zebrano w kilka tygodni więcej danych niż w historii tej dyscypliny. Przez kolejne dziesięć lat trwania badań zebrano 140 terabajtów informacji. Rozpoczęty w 2016 roku program Large Synoptic Survey Telescope miał gromadzić taką ilość danych co pięć dni. Ogromną ilość danych generują nie tylko badania naukowe. „Przy okazji" różnych czynności dane tworzą użytkownicy narzędzi technogigantów, np. Google'a i Facebooka. Pierwsza z firm w 2012 roku przetwarzała dziennie 24 petabajty informacji. Druga co godzinę otrzymywała do publikacji dziesięć milionów fotografii, a w ciągu doby trzy miliardy „polubień" oraz komentarzy (Cukier, Mayer-Schönberger, 2014: 21–22). Stosowano porównania do „zalewu" danymi czy „eksplozji informacyjnej", argumentując na rzecz korzystania z owych cennych zasobów (Larose, 2005: xi, 4; Cerf, 2007; Anderson, 2008; Cukier, Mayer-Schönberger, 2014: 22; Paharia, 2014: 60; Brynjolfsson, McAfee, 2015: 88–89).

W 2008 roku wielkość ogółu cyfrowych danych szacowano na pół zettabajta (ZB), czyli 500 milionów terabajtów. W 2011 roku było to około 2,5 ZB. Na rok 2020 prognozowano wielkość 35 ZB danych. Eksplozja towarzyszy dywersyfikacji rodzajów danych, szczególnie nieustrukturyzowanych, takich jak pliki tekstowe, muzyczne, wideo itp. (Sumbul, 2014). Ilość danych ustrukturyzowanych rośnie jakoby liniowo, a danych nieustrukturyzowanych wykładniczo (Sumbul, 2014).

Dane były postrzegane jako zasób naturalny, stały się „nową ropą" (Puschmann, Burgess, 2014; Watson, 2015; Gogołek, 2016; Sadowski, 2019), magiczną kopalnią nieustannie dostarczającą nowych minerałów, chociaż pierwotne złoża się wyczerpały (Cukier, Mayer-Schönberger, 2014: 140–141). Pochodną takiego postrzegania były tezy o tym, że „surowe" dane stanowią obiektywną prawdę o świecie lub iż „mówią same za siebie", więc wiedza teoretyczna i ekspercka stały się bezużyteczne (badany świat społeczny raczej odcina się od owych tez), oraz zjawisko zwane danetyzacją.

Opisywane wyjaśnienie, a szczególnie część mówiąca o gwałtownie rosnącej ilości wartościowych danych, pełni obecnie nie tylko funkcję legitymizacji świata społecznego DS (a niegdyś tego, co nazywano data mining i później big data). Opowieść tę za prawdziwą przyjęły nauka akademicka i biznes. Przenika ona do tekstów popularnonaukowych:

> W XXI wieku jednak zarówno ziemię, jak i maszyny przyćmią dane i to one staną się najważniejszym skarbem, a polityka będzie walką o władzę nad przepływem danych. Jeśli dane znajdą się w rękach zbyt niewielu ludzi – ludzkość podzieli się na różne gatunki (Harari, 2018: 111).

W badanym świecie DS opisywane wyjaśnienie uznawane jest za fakt – oczywistą prawdę, banał powtarzany do znudzenia:

> Jeśli pracujesz w analityce danych lub w data science, tak jak my, znany jest Ci fakt, iż dane są generowane przez cały czas w coraz szybszym tempie. Możesz być nawet trochę zmęczony ludźmi prawiącymi o tym fakcie kazania (Silge, Robinson, 2018).

Ta legitymizacja może być używana na poziomie mikro. Rozmówca {wywiad 2} z wieloletnim doświadczeniem w IT tak uzasadniał swój wybór ścieżki kariery – zajął się „sztuczną inteligencją" w czasie, kiedy po raz pierwszy zaczęła ona „naprawdę działać" dzięki dostępowi do dużej ilości danych i mocy obliczeniowej.

Opisywana legitymizacja stała się trywialna i straciła na znaczeniu na rzecz opowieści skoncentrowanej na terminie „AI", dotyczącym głównie „czegoś działającego" (szczególnie w zadaniach z zakresu pracy z tekstem, obrazem i dźwiękiem), a nie ilości danych czy mocy obliczeniowej. Pojęcie „AI" jest jednak raczej używane w komunikacji nietechnicznej, marketingowej, płynącej na „zewnątrz" ze świata DS. I choć w jego użyciu akcentuje się takie wartości badanego świata, jak sprawczość i efektywność osiąganą dzięki wdrażaniu produktów bazujących na AI, to uważam owo pojęcie za element areny, a nie strategię legitymizacji rozumianą przecież także jako uzasadnienie racji istnienia świata społecznego w komunikacji „wewnątrz" świata.

Można traktować opisywane wyjaśnienie wyłącznie jako fakt, i tak mógłbym odpowiedzieć na pytanie: „Co uczyniło możliwym przecięcie prowadzące do zaistnienia świata DS w konkretnym czasie?". Wydaje mi się, że koszty mocy obliczeniowej i przechowywania danych spadły poniżej punktu nieopłacalności. Podobnie ekonomicznie możliwy stał się dostęp do dużej ilości danych z nieistniejących wcześniej źródeł. Brakowało (i częściowo wciąż brakuje) regulacji prawnych dotyczących wykorzystania danych i ich analizy oraz modelowania, tak więc raczej nie istniały przeszkody inne niż techniczne. Usuwanie przeszkód technicznych pozwoliło na rozpoczęcie produkcyjnego stosowania osiągnięć akademickich, m.in. ML, na coraz większą skalę. Wysokie budżety firm i ich atrakcyjne, „nieakademickie" wartości pozwoliły pozyskać naukowców do pracy w DS. Sprzyjające były warunki kulturowe – z jednej strony względnie powszechna zgoda na danetyzację, przejawiającą się np. w globalnie akceptowanych modelach biznesowych

Facebooka i Google, gdzie użytkownik jest produktem generującym dane w zamian za bezpłatne usługi, z drugiej „późnonowoczesna" kultura firm technologicznych nastawionych na kwantyfikowalność, efektywność, sprawczość, akceptację niedoskonałości i iteracyjności pracy.

Faktem jest, że dane cyfrowe stały się czynnikiem produkcji (Śledziewska, Włoch, 2020: 69). Mają one wpływ na efektywność działalności gospodarczej i powodują rozwój nowych rozwiązań czy modeli gospodarczych, takich jak firmy technologiczne i platformy internetowe. Podmioty takie jak Uber, Alibaba czy Airbnb mają bardzo mało materialnych zasobów, bardzo dużo zaś danych cyfrowych, z których czerpią wartość ekonomiczną (Śledziewska, Włoch, 2020).

Faktem jest także to, że dane cyfrowe są przechowywane i przetwarzane za pomocą materialnej infrastruktury komputerowej. Infrastruktura ta, marketingowo nazywana chmurą obliczeniową, zużywa zasoby środowiska naturalnego Ziemi. Z uwagi na ciągły przyrost ilości danych stanowi to rosnące zagrożenie ekologiczne. Przy stałym wzroście ilości danych cyfrowych na poziomie 20% rocznie, za około 300 lat ilość energii niezbędnej do podtrzymania takiej produkcji danych przekroczy ilość energii, która jest obecnie konsumowana przez całą ludzkość (Vopson, 2020).

5.1.2.2. Prawdziwy data scientista

Data science w Polsce jest mało zinstytucjonalizowanym światem społecznym. **Nie istnieje formalna droga zaświadczania o autentyczności** uczestnika. Nie zauważyłem, aby jakiekolwiek formalne wykształcenie było w badanym świecie wskaźnikiem, że ktoś jest uznawany za jego uczestnika. Nie zmienia tego pojawianie się na rynku pracy absolwentów studiów z „data science" w nazwie. Nie widać w badanym świecie, by istniał uznawany certyfikat czy dyplom zaliczenia jakiegokolwiek kursu lub egzaminu. Nie istnieją uprawnienia do pracy w charakterze data scientisty.

Także w przypadku DS jako pracy zarobkowej formalna nazwa stanowiska „data scientist" (np. w umowie o pracę) nie jest wskaźnikiem bycia uczestnikiem badanego świata społecznego, a tym bardziej nie jest wskaźnikiem bycia prawdziwym data scientistą. Uczestnicy badanego świata wskazywali na praktyki podszywania się pod DS (tzw. *fake data science*), które stosują np. rekruterzy, promując ofertę pracy będącą faktycznie – tzn. z punktu widzenia uczestników badanego świata – pracą analityka danych z nazwą stanowiska „data scientist". To przyciągać ma wannabesów lub początkujących uczestników świata społecznego DS. Przykładem jest nazywanie stanowiska „data scientist" przy wymaganiach ograniczonych do znajomości Excela, aplikacji biurowych, elementów języka SQL lub narzędzi do wizualizacji danych (np. MS Power BI, Tableau). Z drugiej strony nazwy stanowisk pracy niezawierające słów „data science" nie są w ocenie konkretnych osób uznawane za wskaźnik bycia człowiekiem spoza badanego świata. Formalne nazewnictwo w miejscach pracy uczestników świata

DS jest uznawane za mało informatywne i zapóźnione wobec procesów zachodzących w tym świecie.

Aby być uznanym za uczestnika świata społecznego DS w Polsce, należy realizować działanie podstawowe, polegające na pisaniu kodu w ramach wszystkich trzech wyróżnionych obszarów, tj. przetwarzania, analizy i modelowania danych w jednym procesie, by osiągnąć postawiony cel (co określane jest w badanym świecie jako przeprowadzenie projektu od początku do końca). Poza tym należy mieć kontakt z innymi uczestnikami społecznego świata DS, uczestniczyć w wydarzeniach i w komunikacji badanego świata, czyli **brać udział w dyskursie o prawidłowym sposobie wykonywania działania podstawowego** (por. Kacperczyk, 2016: 18), a więc o granicach świata społecznego i jego wartościach.

Autentyczny uczestnik świata społecznego to taki, który poza spełnianiem tych podstawowych „kryteriów" uczestnictwa stanowi dla innych **uosobienie wartości** tego świata. Niemniej żaden świat społeczny nie jest jednorodny i niezmienny, a więc to, czyje działanie uosabiające wartości danego świata społecznego jest nieostre, płynne, zmienne w czasie, negocjowane, będzie przedmiotem sporów pomiędzy subświatami. Przedstawiam zidentyfikowane przeze mnie praktyki zaświadczania o autentyczności uczestnictwa w świecie DS w trzech kategoriach:

1) powołania na „właściwe" technologie;
   - narzędzia,
   - metody;
2) powołania na doświadczenie w rozwiązywaniu realnych problemów;
3) powołania na cechy osobiste.

Właściwe technologie – narzędzia w świecie DS – to te, które umożliwiają realizowanie działania podstawowego w sposób zgodny z wartościami. Typowym zestawem jest sześć technologii *open source* (Piatetsky, 2018d): **Python** – język programowania; **Anaconda** – dystrybucja Pythona do zastosowań DS (w tym notatnik Jupyter i repozytorium conda); **scikit-learn** – pakiet metod tzw. tradycyjnego ML; **TensorFlow** (TF) – pakiet stosowany głównie do metod DL; **Keras** – tzw. nakładka ułatwiająca korzystanie z TF; **Apache Spark** – zunifikowana technologia do rozproszonego przetwarzania danych. Ten zestaw pomija narzędzia oczywiste, i te nietraktowane jako narzędzia charakterystyczne wyłącznie dla DS, czyli pakiety Pythona do przetwarzania i analizy, w tym wizualizacji danych (np. **pandas**, **NumPy**, **matplotlib**). Pominięto też narzędzia, które opisuję mianem „przemilczane" – język **SQL**, znajomość tzw. **terminala i języka bash** oraz oczywistą znajomość **języka angielskiego**. Oferty pracy na stanowisko data scientisty zawierające „niewłaściwe" narzędzia oceniane są jako podszywanie się pod DS. Oferta wymagająca biegłości w Excelu i Tableau/Power BI będzie faktycznie ofertą dla analityka danych, a jeżeli wymagane są: język programowania Java czy inne niskopoziomowe oraz technologie konteneryzacji, będzie to oferta dla programisty lub związanego z DS stanowiska inżynierskiego (data engineer, ML engineer).

Łatwą do zaobserwowania praktyką autoprezentacji i zaświadczania o autentyczności uczestników świata DS jest ozdabianie pokrywy personalnego laptopa naklejkami. Laptop jest dla nich zazwyczaj urządzeniem nieodłącznym, noszonym w odpowiednim plecaku. Uczestnicy często widzą się z laptopami i rozpoznają je. Nieznajome osoby mogą rozpocząć interakcje pytaniem o naklejkę. Podczas jednego z obserwowanych wydarzeń byłem świadkiem następującej sceny: w chwili wychodzenia z pubu zauważono pozostawiony plecak. Znalazcy, nie rozpoznając plecaka, wyciągnęli laptop i oglądając naklejki oraz samo urządzenie zgadywali, do kogo należy zguba.

Fabryczny laptop etnografa jaskrawo wyróżnia się spośród oklejonych laptopów uczestników świata DS i pokrewnych środowisk technicznych (Lowrie, 2016: 26). Gromadziłem zatem naklejki i umieszczałem je na swojej pokrywie. Naklejki to często logo wykorzystywanych technologii oraz wydarzeń świata społecznego. Postrzegam oklejanie laptopów jako praktykę zaświadczania o autentyczności poprzez powołanie na „właściwe" narzędzia i prezentację swojego uczestnictwa w dyskursie badanego świata, a także odwołanie do zdobytego doświadczenia. Pojawiają się naklejki z logo firm, czasami uczelni, także np. grup meetupowych. Spotyka się naklejki z logo technologii spoza DS, związane np. z ruchem *open source*, oraz nalepki prezentujące światopogląd, gust czy zainteresowania (rys. 5.2).

Rysunek 5.2. Naklejki na pokrywie personalnego laptopa uczestnika świata DS

Źródło: fotografia wykonana przez Remigiusza Żulickiego za zgodą Piotra Migdała.

Zgadzam się z Lowiem (2016: 26–28), że laptopy uczestników świata DS są przyozdabiane raczej chaotycznie, naklejki mogą nachodzić na siebie, być brudne lub podniszczone. Traktowanie personalnego laptopa w sposób nieco niedbały, nonszalancki, swobodny, jako rzecz niewymagającą przesadnej troski – narzędzie do ciężkiej pracy i szybkiego zużycia – jest zgodne z podkreślaną w DS tymczasowością i dezaktualizowaniem się technologii, a więc ze zobowiązaniem uczestników do samorozwoju.

Posługiwanie się opisywanym wyżej zestawem sześciu narzędzi do wykonywania działania podstawowego jest w badanym świecie uznawane za wskaźnik bycia jego autentycznym uczestnikiem. Nie jest to wskaźnik jedyny ani zaświadczający o autentyczności. Jest to zestaw podstawowy, choć świadczący o wysokim poziomie znajomości technologii badanego świata przez osobę go użytkującą – wysokim, bo poziom „wysoki" jest najniższym z akceptowanych dla uczestników badanego świata. Dla większości uczestników, także silnie osadzonych, może być zestawem najczęściej używanym. Dostrzegam jeszcze dwa inne „wskaźniki" odwołujące się do narzędzi – strategie zaświadczania nie tylko o autentyczności, ale i o maestrii uczestnika świata DS.

Po pierwsze, o maestrii zaświadcza swobodne **posługiwanie się wieloma narzędziami konkurencyjnymi lub komplementarnymi do wykonywania działania podstawowego**. To tzw. dobieranie narzędzia do problemu, co jest oceniane jako dobre w kategoriach sprawczości i efektywności. Dotyczy to np. posługiwania się nie tylko najpopularniejszymi językami Python i R, ale też słabiej znanymi i posiadającymi inne zalety niż R i Python – np. Scala lub Julia. Podobnie jest z używaniem pakietów. O maestrii nie świadczy używanie PyTorcha zawsze i do wszystkiego. Wskazuje na nią używanie, w zależności od konkretnych wymagań danego projektu, optymalnego pakietu – np. PyTorch, TF, Theano czy CNTK. Nawet w obrębie jednego języka programowania o maestrii świadczy używanie pakietów Tidverse i data.table, nie tylko jednego z nich do każdego, kolejnego problemu. Kod *in vivo* to **„agnostyk technologiczny"**, co oznacza człowieka nieprzywiązanego do konkretnego narzędzia, a zwracającego uwagę głównie na rozwiązywany problem i korzystającego z szerokiej gamy narzędzi. Agnostykom „wypada" mieć na laptopie naklejki kilku konkurencyjnych technologii, a nie tylko tych komplementarnych, stanowiących jeden ekosystem.

Powoływanie się na doświadczenie w **posługiwaniu się różnymi rodzajami danych z różnych źródeł** także stanowi strategię zaświadczania o autentyczności. Uczestnikom dość łatwo ocenić rodzaje danych przez pryzmat technologii, za pomocą których były przechowywane i przetwarzane, a także przez pryzmat użytych metod. Jeżeli ktoś mówi, że w projekcie mają kłopoty z bazą w Mongo DB i planują jutro zacząć modelowanie od LDA, to mowa o dużym wolumenie dokumentów, czyli o danych tekstowych. Za autentycznego uczestnika raczej będzie uznana osoba, która pracowała zarówno z danymi tabelarycznymi z baz danych z dostępem przez SQL, z fotografiami zapisanymi w plikach, jak i z danymi tekstowymi pozyskanymi za pomocą webscrapingu, a nie osoba pracująca wyłącznie z jednym rodzajem danych.

Podobnie będzie z wolumenem danych, tak więc o autentyczności będzie zaświadczać raczej doświadczenie zarówno z małą, jak i dużą ilością danych. Wolumen i rodzaj danych (a nierzadko zastosowane metody) uczestnicy potrafią ocenić także przez pryzmat zastosowanej infrastruktury komputerowej. I znowu za bardziej autentyczne mogą być uznawane osoby, które realizowały projekty zarówno na laptopach, jak i na maszynach wirtualnych AWS oraz lokalnych serwerach firmowych.

Po drugie, o maestrii zaświadcza **samodzielne budowanie narzędzi**. Można być uznanym za autentycznego uczestnika świata DS, korzystając jedynie z gotowych rozwiązań, przeważnie przyjmujących formę pakietów. O maestrii świadczyć będzie pisanie własnych pakietów, początkowo raczej na użytek „chałupniczy". Może wymagać to użycia innych języków programowania niż Python/R, jak C++. Coraz bardziej dopracowane wersje budowanych narzędzi publikuje się w otwartym dostępie, np. na GitHubie. Wraz ze zdobywaniem uznania, obserwowalnego m.in. w liczbie pobrań pakietu czy tzw. gwiazdek na GitHubie, dalszym etapem może być publikacja autorskiego pakietu w repozytoriach uznawanych za oficjalne, takich jak CRAN dla R.

Właściwe technologie – metody w świecie DS – to metody różnorodne. Strategią zaświadczania o autentyczności jest powoływanie się na dobór właściwej metody do rozwiązywanego problemu praktycznego, z uwzględnieniem specyfiki projektu. Autentyczny uczestnik to zatem taki, który względnie swobodnie korzysta z metod tradycyjnego ML, DL, z podstawowych i zaawansowanych metod statystycznych, metod eksploracyjnej analizy danych, a także metod statystycznych lub analitycznych adaptowanych z węższych specjalizacji obliczeniowych dziedzin akademickich, np. biologii, badań operacyjnych. Nie spotkałem się z określeniem **„agnostyk metodologiczny"**, ale można je zastosować analogicznie, jak „agnostyk technologiczny".

O ile **uczestnik świata społecznego DS musi legitymować się znajomością i doświadczeniem w stosowaniu rozmaitych metod modelowania danych** (w tym tradycyjnego ML i DL), **to nie musi zawsze stosować ML**. Przeciwnie – za rozwiązanie rzeczywistego problemu za pomocą modelowania można uznać nawet automatyczne generowanie „tabelki" (częstości czy statystyk opisowych w podziale na grupy), o ile rozwiązuje to ów problem w optymalny sposób.

Stosowanie zawsze metod ML, a szczególnie DL, traktowane jest w świecie DS jako podejście początkujących. Co prawda muszą oni zaświadczać o swojej autentyczności znajomością tych metod. Jednak przy braku znajomości lub ignorowaniu metod tradycyjnych, w tym statystycznych i analitycznych, a także przy niewielkich umiejętnościach czy doświadczeniu w przetwarzaniu i analizie danych lub ignorowaniu tych etapów, początkujący mogą uzyskiwać „słabe" wyniki eksperymentów – słabe w sensie: ilościowych metryk dopasowania modelu do danych, zbliżania się do rozwiązania rzeczywistego problemu, kosztów mocy obliczeniowej niezbędnej do pracy modelu w systemie produkcyjnym. Możliwa jest sytuacja,

w której model bazujący na zaawansowanej sieci neuronowej, pracujący na 2 tys. zmiennych niezależnych, daje dopasowanie do walidacyjnego zbioru danych na poziomie 98%, jednak zespół DS przedstawia klientowi POC z użyciem prostszego obliczeniowo modelu pracującego na dwustu zmiennych, dającego dopasowanie 96%, ponieważ korzyść wynikająca z szybkości działania i kosztów ewentualnego wdrożenia i utrzymania prostszego modelu „na produkcji" jest uznawana za wyższą niż korzyść wynikająca z wyższej miary dopasowania modelu bardziej złożonego. Modelowanie, szczególnie z użyciem DL i ogromnej mocy obliczeniowej, jest w badanym świecie uznawane za fajne, za zabawę, coś przyjemnego – w przeciwieństwie do raczej nudnej i żmudnej pracy z przygotowaniem i analizą danych przed modelowaniem. Uczestnicy mogą czuć pokusę[2] używania DL jako tej najfajniejszej metody modelowania danych.

Niemniej o **autentyczności, a nawet maestrii uczestnika zaświadcza działanie z użyciem metod najefektywniejszych**, optymalnie przyczyniających się do rozwiązania problemu, a nie metod najbardziej skomplikowanych, zaawansowanych, najciekawszych, najfajniejszych czy najmodniejszych w danej chwili. ***Deep learning*, a nawet ML w ogóle, jest fetyszyzowane** zarówno przez wannabesów i początkujących uczestników świata DS, jak i przez jego klientów.

Silnie osadzeni uczestnicy świata DS także miewają ulubione języki programowania, pakiety i metody, z których mogą korzystać niemal zawsze na wstępnych etapach eksperymentów (wskazywano np. na metody z pakietu XGBoost). Dopiero później mogą korzystać z szerokiej gamy narzędzi i metod, nawet do etapu customizowania czy budowania narzędzia lub metody na potrzeby konkretnego projektu. Jest to zgodne z iteracyjnym podejściem do pracy w DS i wartościami tego świata.

Krytykowani przez uczestników osadzonych są początkujący lub wannabesi niemający podstaw matematycznych w zakresie np. będącej fundamentem ML algebry liniowej, a znający jedynie DL z kursów MOOC:

> *B: Mhm, czyli* [firmy-klienci – przyp. R.Ż.] *nie chcą tych danych, nie chcą, żeby dane wychodziły na zewnątrz?*
> R: Yyy, dwa czynniki. Paranoicznie nie chcą, żeby dane wychodziły na zewnątrz, w sumie nieważne, czy te dane rzeczywiście komuś mogą się przydać, czy nie, to jest jedna sprawa, a druga sprawa **nie widzą różnicy**, bo wszystkim się wydaje, że wystarczy **kręcić** tymi śrubkami i to wszystko, i to załatwia sprawę. Wtedy używam takiego przykładu [pauza], eee, kto sądzisz, że jest lepszym diagnostą? Ktoś, kto pracował w **szpitalu**? Przez, nie wiem [pauza], **20 lat**? I spotkał się z najróżniejszymi przypadkami czy ktoś, kto [pauza] **obejrzał sobie** wszystkie odcinki doktora House'a. I jak się poda taki przykład, to powoduje, że [pauza] klienci się **trochę** zastanowią, że nie do końca tak jest, że ktoś sobie potrzaskał sieci, eee, po kursie online {wywiad 5}

2 Powstają m.in. memy na ten temat – [tekst mema] Ja: „To nie wygląda na takie skomplikowane. Może nawet mógłbym to rozwiązać za pomocą regresji logistycznej." Ja do siebie: „Użyj uczenia głębokiego!" – https://www.facebook.com/artificialintelligencememes/photos/a.1653953787951182/1761188270561066/ (dostęp: 3.06.2020).

Uważam takie praktyki także za obronę granic autentycznego DS i własnej pozycji, również rynkowej, uczestników silnie osadzonych. W kontekście zaświadczania o autentyczności jest to przykład powołania na doświadczenie w rozwiązywaniu realnych problemów.

Jakie problemy są w społecznym świecie DS uznawane za realne? Działanie podstawowe uczestników badanego świata może być wykonywane komercyjnie, akademicko lub hobbystycznie, a sam fakt pracy w firmie lub na uczelni nie ma znaczenia w perspektywie uznawania się i bycia uznawanym za uczestnika świata DS. Określenie „data scientist" jest jednak raczej nazwą stanowiska pracy w podmiocie komercyjnym niż nazwą „stosowną" dla każdego uczestnika/każdej uczestniczki badanego świata. Biorąc powyższą konstrukcję za odzwierciedlającą perspektywę uczestników świata DS, **za realne problemy uznaję takie, których rozwiązanie ma wartość dającą się wyrazić w pieniądzach**. Jest to moje ujęcie. W dyskursie świata DS mówi się raczej o korzyściach utylitarnych: szybciej, taniej, w większej skali, łatwiej itp., odnosząc się do kategorii optymalizacji, efektywności i sprawczości. Sądzę jednak, że o autentyczności i maestrii uczestników zaświadcza pieniężna wartość rozwiązywanych problemów.

Zysk finansowy z dokonanej zmiany, ze zwiększonej efektywności jest traktowany jako mierzalna metryka sukcesu w projekcie DS. Tym samym **autentyczny uczestnik powinien legitymować się doświadczeniem w realizacji zyskownych projektów komercyjnych**. Nie chodzi o wysokość indywidualnych zarobków. Zarobki nie zaświadczają o autentyczności, ponadprzeciętnie wysoką pensję może bowiem uzyskiwać osoba oceniana negatywnie w kontekście sprawczości, w kontekście ciekawości lub samorozwoju albo wynagradzana za pracę poza działaniem podstawowym (np. prowadzenie szkoleń). Chodzi o efektywność i sprawczość osiąganą dzięki zrealizowanym projektom, a w szczególności dzięki wdrożonym modelom. Prestiż wdrożenia jest bardzo wysoki, ponieważ większość projektów DS kończy się niepowodzeniem, a wiele spośród udanych na poziomie POC/MVP nie zostaje wdrożonych. Celem części projektów w ogóle nie jest wdrożenie, a nierzadko osoby zajmujące się DS nie mają nic wspólnego z realizowanym – bądź nie – wdrożeniem.

Traktując ostrożnie tezy o formowaniu się jakichś „struktur" czy „hierarchii" wtajemniczenia w świat społeczny, bliskie perspektywie uczestników badanego świata, a przynajmniej jego dominującej pozycji, może być rozumienie doświadczenia w rozwiązywaniu realnych problemów jako elementu zaświadczania o autentyczności:

1) ktoś, kto potrafił wykonać „coś działającego", tzn. zrealizować projekt DS od początku do końca, niezależnie, czy w działalności komercyjnej, akademickiej, czy hobbystycznej, może być uznany za uczestnika świata DS (przy użyciu właściwych narzędzi, uczestnictwie w dyskursie świata i pozytywnej ocenie w odniesieniu do wartości świata);
2) ktoś, kto udowodnił, iż potrafi „dowozić", tzn. zrealizował kilka projektów DS w środowisku komercyjnym (przy co najmniej powyższych założeniach),

może być uznany za prawdziwego data scientistę – dotyczy to osób mających co najmniej kilka lat doświadczenia zawodowego;
3) za uczestnika osiągającego poziom maestrii może być uznana osoba, która (przy co najmniej powyższych założeniach) legitymuje się kilkoma produkcyjnymi wdrożeniami zrealizowanych projektów komercyjnych o dającej się oszacować wartości zysku dla klientów.

„Robię to od dziecka" jest strategią zaświadczania o autentyczności, odwołującą się zarówno do doświadczenia, jak i cech osobistych. Polega na deklarowaniu, że osoba interesowała się i zajmowała się matematyką lub informatyką już w wieku kilku/kilkunastu lat:

> [imię i nazwisko prelegenta]
> Człowiek, któremu tata zamiast bajek opowiadał zagadki matematyczne {autoprezentacja prelegenta na meetupie, obserwacja 4}

> R: […] I to się okazuje, że żeby to mieć, to dosyć dużą część życia trzeba się tymi rzeczami zajmować. Czyli to są ludzie, którzy nie kopali piłki na podwórku, nie grali w gry, tylko **siedzieli** i na przykład ich ciekawiło, dlaczego coś działa, a dlaczego coś nie działa w matematyce, tak? No i to są **dziwacy**, znaczy, jak się przekroi społeczeństwo, to mało jest takich osób, tak mi się wydaje. […] mamy dosyć dużo ludzi, którzy programowali już w gimnazjum, czyli to nie było tak, że się w szkole nauczyli, tylko mieli taki wewnętrzny push, żeby iść w tą informatykę, tak? {wywiad 2}

> R: […] padło pytanie, jakie miałaś marzenie w dzieciństwie. I w sumie czasami to tak jest, że jak usłyszysz reakcję innych, to dopiero rozumiesz, że to nie jest naturalne [z uśmiechem]. Ja zawsze chciałam być na styku, w takim sensie, mój tata bardzo szybko zaszczepił we mnie miłość do Exceli, ja **uwielbiałam** rozliczać się z wszystkiego w Excelu, więc ja już na bardzo wczesnym etapie podjęłam decyzję, że ja chcę się zajmować technologiami i wspomagać finanse. […] jestem jedną z nielicznych osób, które w gimnazjum wiedziały, na jaki kierunek studiów idą. Więc ja dość szybko byłam ukierunkowana, ja nie będę kryła, że to jest po prostu moja pasja. Ja męża mam również w tej samej branży, więc wiesz, nasze rozmowy wieczorne są głównie o tym [śmiech]. Wiesz, my w piątek sobie siadamy i odpalamy Andrew Ng [z uśmiechem] i my to lubimy. Ja naprawdę się tym strasznie pasjonuje i nie kryję […] {wywiad 22}

Podobne deklaracje pojawiają się w innych środowiskach związanych IT, np.:

> Po czym poznać, że dana osoba „czuje" postęp? Najłatwiej po tym, co robiła za młodu. Wybitni przedsiębiorcy często bawią się technologią i podejmują różne inicjatywy na długo, zanim wpadną na pomysł, że mogliby z tego zrobić sposób na życie (Schmidt, Rosenberg, Eagle, 2014: 212).

Data science to – szczególnie w Polsce – młody świat społeczny, składający się przeważnie z młodych ludzi. Rzadko można spotkać osoby z wieloletnim doświadczeniem w realizacji działania podstawowego. Powołanie się na zainteresowania i zajęcia kontynuowane od dzieciństwa ma wskazywać na długość doświadczenia w obcowaniu z technologiami, niekoniecznie tymi uznawanymi za właściwe w DS,

zaświadczać o swobodzie w ich użytkowaniu, o posiadaniu cennego, bo niemożliwego do nabycia inaczej niż przez doświadczenie, repertuaru wiedzy milczącej. Ta strategia odwołuje się do takich wartości świata DS jak samorozwój, samodzielność, ciekawość i racjonalność.

Jeżeli ktoś uczył się programowania i startował w konkursach matematycznych, mając 11 lat w roku 2002, kontynuował aktywność na tych polach, a w 2017 roku zaczął pracę w charakterze data scientisty, to dla innych uczestników badanego świata będzie to komunikat, że już od dziecka dużo pracował samodzielnie, uczył się i rozwijał. Pamiętać należy, jak w Polsce wyglądały technologie informacyjne i dostęp do materiałów edukacyjnych dotyczących programowania w 2002 roku (nawet nie mówiąc o programowaniu dla dzieci) – nasz bohater najpewniej korzystał ze zdobywanych z trudem źródeł anglojęzycznych. Jasne będzie również, że to osoba ciekawa, „prawdziwie zainteresowana", przy założeniu, że nikt nie przekupywał ani nie przymuszał jedenastolatka do matematyki i programowania. Komunikat dotyczy też tego, że bohater już jako dziecko myślał logicznie, z użyciem kwantyfikacji, racjonalnie. Potencjalnie wolał przeznaczyć wolny czas na matematykę i programowanie, lepiej czuł się bowiem w tej racjonalnej sferze, niż np. uprawiając sport czy uczestnicząc w życiu towarzyskim. Tu jednak opisywana strategia zaświadczania o autentyczności staje się niejednoznacznie akceptowana, łączy się bowiem z kategorią nerda lub tylko z jej stereotypowym i utrwalonym w popkulturze wizerunkiem:

> Tak więc jeśli należysz do tych, którzy uważają się za „niezbyt dobrych z matematyki", to być może chodzi o twój podświadomy strach, że jeśli w kursie statystyki pójdzie ci lepiej niż źle, to, powiedzmy, dostaniesz pryszczy (Babbie, 2006: 479–480).

Identyfikacja nerdów z introwertycznymi, racjonalnymi typami osobowości[3] **może być nie tylko stereotypem** – wskazywano to jako najważniejszą, obok wysokiego IQ, składową definicji prawdziwego nerda. Te charakterystyki są uznawane za podstawę do umiejętności myślenia racjonalnego, logicznego i abstrakcyjnego, typowego dla matematyki i informatyki. **Wśród osób uznających się za nerdów może panować przekonanie o tym, że należą one do elity intelektualnej, wąskiego grona ludzi myślących racjonalnie i logicznie w ogromnej rzeszy nieracjonalnych i nielogicznych przeciętniaków, których interesują towarzyskie gierki** (Tocci, 2009: 225–228). Nerdów wiąże się ze spektrum autyzmu lub z zespołem Aspergera – osobami niekomunikatywnymi czy nietowarzyskimi, ponadprzeciętnie inteligentnymi i wykazującymi zdolności matematyczne i techniczne z uwagi na jednostkę diagnostyczną[4] (Tocci, 2009: 228–232; Zaród, 2018: 229, 350). Przegląd literatury na ten temat opublikował jeden z najbardziej znanych w Polsce data scientistów (Krawczyk, Migdał, 2011).

---

3 INTJ (*Introversion*, *Intuition*, *Thinking*, *Judgment*) lub INTP (*Introversion*, *Intuition*, *Thinking*, *Perception*).

4 Od 1992 roku zespół Aspergera jest w Międzynarodowej Klasyfikacji Chorób ICD-10.

Nerd ma dla części uczestników świata DS pozytywne konotacje, może być kategorią pozytywnej autoidentyfikacji osoby technicznej, a także zaświadczać o byciu „prawdziwie zainteresowanym". Dla innych konotacje są neutralne lub negatywne, występuje czasem odcinanie się od bycia nerdem jako kimś, kto przesadnie skupia się na technologiach, a za mało uwagi poświęca rozwiązywaniu prawdziwych problemów.

Ta strategia zaświadczania o autentyczności pomaga zrozumieć przemilczane technologie świata DS. Przemilczane lub mało problematyzowane technologie to te znane od kilkudziesięciu lat (SQL, terminal i język bash, podstawy HTML/CSS/JavaScript).

Jeżeli bowiem jest tak, że obecnie liczne grono uczestników poznało pewne technologie, będąc dziećmi, i posługuje się nimi od kilkunastu czy dwudziestu kilku lat, to są one dla nich przezroczyste i w konsekwencji przemilczane. Tym samym raczej nie problematyzują tych technologii nawet ci, dla których nie są one przezroczyste i ci, którzy nie „robią tego od dziecka" – możliwe, że w ramach niepodkopywania własnej autentyczności.

Strategia „robię to od dziecka" może w badanym świecie zaświadczać o prawdziwym zainteresowaniu DS także dlatego, że odwołuje się do osobistych, nawet wrodzonych, jakoby uwarunkowanych biologicznie zdolności, predyspozycji czy zainteresowań obejmujących matematykę i informatykę:

> *B: OK, yyy, jak to się stało, że poszłaś na takie studia?*
> R: Ja w zasadzie miałam jedynie dylemat, czy pójść na informatykę, czy na matematykę, tak. Na informatykę na politechnikę, na matematykę na uniwersytet, iii, to były dwa miejsca, gdzie składałam papiery. Miałam **zawsze** zamiłowanie do matematyki i wręcz przeciwne uczucia do przedmiotów humanistycznych [z uśmiechem], więc tu wątpliwości nie było. […] Natomiast faktycznie, umysł ścisły odziedziczyłam po rodzicach i taką smykałkę. {wywiad 24}

Psychologowie wskazywali, że w tzw. zachodnim kręgu kulturowym matematyka jest uznawana za trudną i niekoniecznie ważną w życiu codziennym (w USA sprzedawano lalki Barbie z nagranym zdaniem „matematyka jest trudna"), oraz że osiągnięcia matematyczne przypisywane są w większym stopniu zdolnościom niż wysiłkowi włożonemu w naukę (Ashcraft, 2002). W Polsce jest podobnie:

> Przez ćwierć wieku polscy uczniowie nie zdawali matury z matematyki […]. Można było skończyć liceum, nie rozumiejąc podstawowych pojęć matematycznych. Dlatego matematyka nabrała cech wiedzy elitarnej (Skura, Lisicki, 2018: 52).

Tak więc strategią zaświadczania o autentyczności w badanym świecie może być powołanie się na zainteresowania, zdolności, wiedzę i umiejętności matematyczne. Jednak różne pozycje świata DS odmiennie akcentują wagę znajomości matematyki, choć nie jest obecna pozycja lekceważąca matematykę. **Za próg wejścia do świata DS może być przyjmowana znajomość matematyki zarówno od**

poziomu zaawansowanego egzaminu maturalnego, jak i od algebry liniowej/statystyki matematycznej. W książce przedstawiającej podstawy ML dla osób nietechnicznych, raczej zajmujących się biznesem niż chcących wejść do świata DS, stwierdzono, że temat „może zrozumieć każdy, kto podczas studiów miał semestr matematyki lub algebry liniowej" (Foreman, 2017: 17).

Mówi się o silniejszym nastawieniu na znajomość matematyki/statystyki w subświecie użytkowników języka R i silniejszym nastawieniu na umiejętności koderskie w subświecie używających Pythona. Taka wizja jest przesadnie uproszczona. Strategią zaświadczania o autentyczności może być powołanie się na zainteresowanie, zdolności, wiedzę i umiejętności informatyczne, a szczególnie koderskie. Jednak data scientista nie jest programistą i zaawansowane umiejętności programistyczne, np. płynne posługiwanie się niskopoziomowymi językami programowania, nie zaświadczają o autentyczności uczestnika. **Próg wejścia do świata DS stanowi znajomość R lub Pythona, pozwalająca realizować projekty od początku do końca.** Standardy pisania kodu są różne w różnych subświatach, w różnych firmach i zespołach DS, ale prawdziwy uczestnik musi samodzielnie i efektywnie radzić sobie z kodowaniem na każdym etapie iteracyjnej realizacji projektu. Podobnie data scientista nie jest matematykiem-teoretykiem, a o autentyczności nie zaświadcza znajomość ani abstrakcyjnych dziedzin matematyki, ani nawet zaawansowanych szczegółów matematycznych w ML. To drugie może być podstawą do osiągnięcia poziomu maestrii w posługiwaniu się metodami i narzędziami ML, ale samo w sobie nie będzie w badanym świecie wysoko cenione z uwagi na małą sprawczość.

Podsumowując, **uczestnicy badanego świata zaświadczają o autentyczności poprzez kategorię bycia osobą techniczną**.

> *B: A ci managerowie też są data scientistami?*
> R: **Tak**, wszyscy, którzy są w naszym zespole, są **technicznymi** osobami, nie ma takich osób, które na przykład zajmują się tylko, ymm, product managementem albo tylko są product ownerami. Nie, wszyscy muszą **usiąść** [ze śmiechem] i wszyscy, wszyscy programują, wszyscy zajmują się data science. {wywiad 20}

Może to oznaczać miks zainteresowań, zdolności, wiedzy, umiejętności matematycznych i informatycznych. Ten miks musi być jednak taki, by pozwalał na efektywne i samodzielne wykonywanie działania podstawowego – pracy na danych cyfrowych. W środowisku związanym z Politechniką Warszawską trwały dyskusje dotyczące tłumaczenia nazwy „data science" na polski – kierunek nazwano inżynieria i analiza danych. Argumentacja na rzecz określenia „danetyka" (jak informatyka czy robotyka), już nie jako kierunku studiów, ale dziedziny, jest przykładem na postrzeganie autentyczności uczestnictwa w DS w kategoriach „osoby technicznej" oraz w utylitarnej wartości realizowanych projektów:

> […] danologia rodzi we mnie odczucie, że jest to nauka humanistyczna o danych, w której prowadzi się rozważania filozoficzne o tym, czym są dane, jaką rolę odgrywają w życiu itd. Dlatego zupełnie nie pasuje to do technicznej natury tej dziedziny, w której na co dzień się programuje,

> pracuje z różnymi technologiami, stosuje lub opracowuje algorytmy matematyczne. […] Dlatego danetyka wpisuje się w tę prawidłowość – mi nasuwa ona skojarzenie z „inżynierami" danych, którzy działają w biznesie. Gdy usłyszymy od kogoś z biznesu, że pracuje w danetyce, to brzmi to rozsądnie – od razu czuć, że pracuje w obszarze technicznym w dziale/zespole związanym z danymi (Ryciak, 2019).

Mniej jednoznaczną strategią zaświadczania o autentyczności jest powołanie się na utylitarną wartość zrealizowanych projektów. Nazywano to rozwiązywaniem realnych problemów, ale i ta kategoria bywa uznawana za biznesową kliszę i część uczestników woli mówić o dostarczaniu wartości (bo np. w trakcie realizacji dopiero ustala się problem lub problem zostaje porzucony na rzecz innego, bardziej wartościowego). Legitymizacja nie następuje tylko poprzez odniesienie do tego, jaką wartość utylitarną miały zrealizowane projekty. To, że ktoś generował zyski dla firmy, będąc np. specjalistą w zakresie marketingu (tzw. ekspert dziedzinowy), nie zaświadcza przecież, że jest prawdziwym data scientistą. Taka osoba może przejść na stanowisko DS, a jej znajomość dziedziny może być uznawana za zaletę w tej roli zawodowej, niemniej, by być postrzegana jako data scientista, musi zaświadczyć o swoich kompetencjach technicznych w zakresie narzędzi i metod uznawanych przez uczestników świata DS za odpowiednie. Osoby nietechniczne, jak eksperci dziedzinowi czy różnego rodzaju osoby odpowiedzialne np. za zarządzanie, sprzedaż, marketing, HR, księgowość, produkcję, administrację, są przez uczestników badanego świata przyporządkowywane do kategorii osób biznesowych. Terminem „biznes" określa się klientów, wewnętrznych i zewnętrznych. Może być to stosowane także wobec przełożonych i osób wspomagających realizacje projektów od tzw. strony biznesowej (jak np. project manager, product owner, marketingowcy), o ile uważa się je za osoby nietechniczne.

Powrócę jeszcze do **nerdów w DS** i powiążę to z areną dotyczącą wykonywania działania podstawowego: „Czy uczestnicy realizują działanie podstawowe, żeby rozwiązać problem (dostarczyć wartość utylitarną), czy zrobić model z użyciem danych (dostarczyć układ technologiczny)?". Moim zdaniem w badanym świecie dominuje pozycja pozytywnego nastawienia do kategorii nerda (jako osoby technicznej i „prawdziwie zainteresowanej"), a także pozycja uznająca rozwiązywanie problemów za cel działania podstawowego. To w wybranych aspektach jest ze sobą sprzeczne.

Mające charakteryzować nerdów niskie umiejętności komunikacyjne (lub tylko brak zainteresowania tą sferą pracy) mogą być przeciwskuteczne w rozwiązywaniu problemów rzeczywistych. W sytuacjach innych niż stawianie problemów samemu sobie będzie to zawsze polegać na komunikacji z innymi ludźmi i zrozumieniu ich – na czym polega cel, jaki stan będzie oznaczał osiągnięcie celu, na czym będzie polegał/polegała sukces/porażka. Z innej strony przypisywane nerdom myślenie racjonalne, logiczne i abstrakcyjne będzie skuteczne (lub postrzegane jako skuteczne przez pozycję uznającą „nerdowatość" za pozytywną cechę) w rozwiązywaniu

problemów tak, że sprzyja koncentracji na samym problemie, a nie na ludziach, np. tym, że ktoś w firmie klienta boi się zmian, które jego zdaniem przyniesie skutecznie wdrożony model ML.

Nerd jest w świecie DS łączony z niewielką znajomością biznesu i niewielkim nim zainteresowaniem, z osobami biegłymi technicznie i zorientowanymi silnie na techniczne aspekty realizowanych projektów, a pomijającymi aspekty utylitarne. Zdaniem niektórych uczestników świata DS najbardziej pożądanym połączeniem jest jednocześnie bardzo dobra znajomość narzędzi, metod, biznesu i umiejętności komunikacyjnych:

> R: Osoby, które rozumieją biznes, to są **perełki**. Powiem wprost. Ja nie kryję też, że ja tak też miałam, dlatego miałam tyle ofert, bo jak ktoś ze mną rozmawiał i widział, że ja rozumiem biznes i jestem w stanie opowiedzieć o biznesie, to było takie [pauza] **OK**, mało takich osób, starajmy się wziąć. […] większość data scientistów mimo wszystko to są osoby, które wyrosły z programistów, a prawda jest taka, że i teraz jest coraz większy taki, znaczy, taki też powinien być trend, że te osoby z wiedzą biznesową powinny też się kształcić bardziej w data science i to wtedy są najlepsi data scientiści, którzy mają taką wiedzę biznesową. {wywiad 22}

Choć osoby legitymujące się jednocześnie bardzo wysoką znajomością technologii i wysoką znajomością biznesu mogą być uznawane za wyjątkowo wartościowych data scientistów, tzw. jednorożce, czyli cenne, rzadkie, bez mała mityczne hybrydy, to osoby biegłe technicznie, a nieposiadające wysokiej znajomości biznesu także definiuje się jako prawdziwych uczestników. Niemniej **samo bycie nerdem nie zaświadcza o byciu uczestnikiem świata DS ani o autentyczności uczestnictwa**. Pewne cechy nerdów korelują z byciem data scientistą, ale nie powodują go (Kuncewicz, 2019).

Znajomość technologii jest w świecie DS pierwotna w tym sensie, że bez niej nie można zostać uznanym za uczestnika. Znajomość biznesu czy abstrakcyjnie ujęte dostarczanie wartości utylitarnej jest uznawane za wtórne wobec technologii. Jest osiągane dzięki zastosowaniu „właściwych technologii" i wraz z doświadczeniem w stosowaniu tychże technologii. W tym sensie **osiąganie celów technicznych i celów utylitarnych może być dla data scientisów tym samym**.

Dla części uczestników, nawet silnie osadzonych, „data scientist" nie jest kategorią autoidentyfikacji. Częściowo ze względu na pojawianie się na rynku pracy stanowisk określanych terminem *fake data science*, uznawanych za faktyczne zadania analityka danych, częściowo chyba ogólnie, będąc wyrazem procesów segmentacji, a szczególnie profesjonalizacji badanego świata, **już od około 2018 roku w Polsce zauważalna jest utrata prestiżu określenia „data scientist"**. Niektórzy uczestnicy mogą unikać tego określenia jako np. nieostrego, coraz bardziej pospolitego czy powszechnie fałszowanego na rzecz określeń koncentrujących się wokół uczenia maszynowego – „ML engineer" oraz „ML researcher".

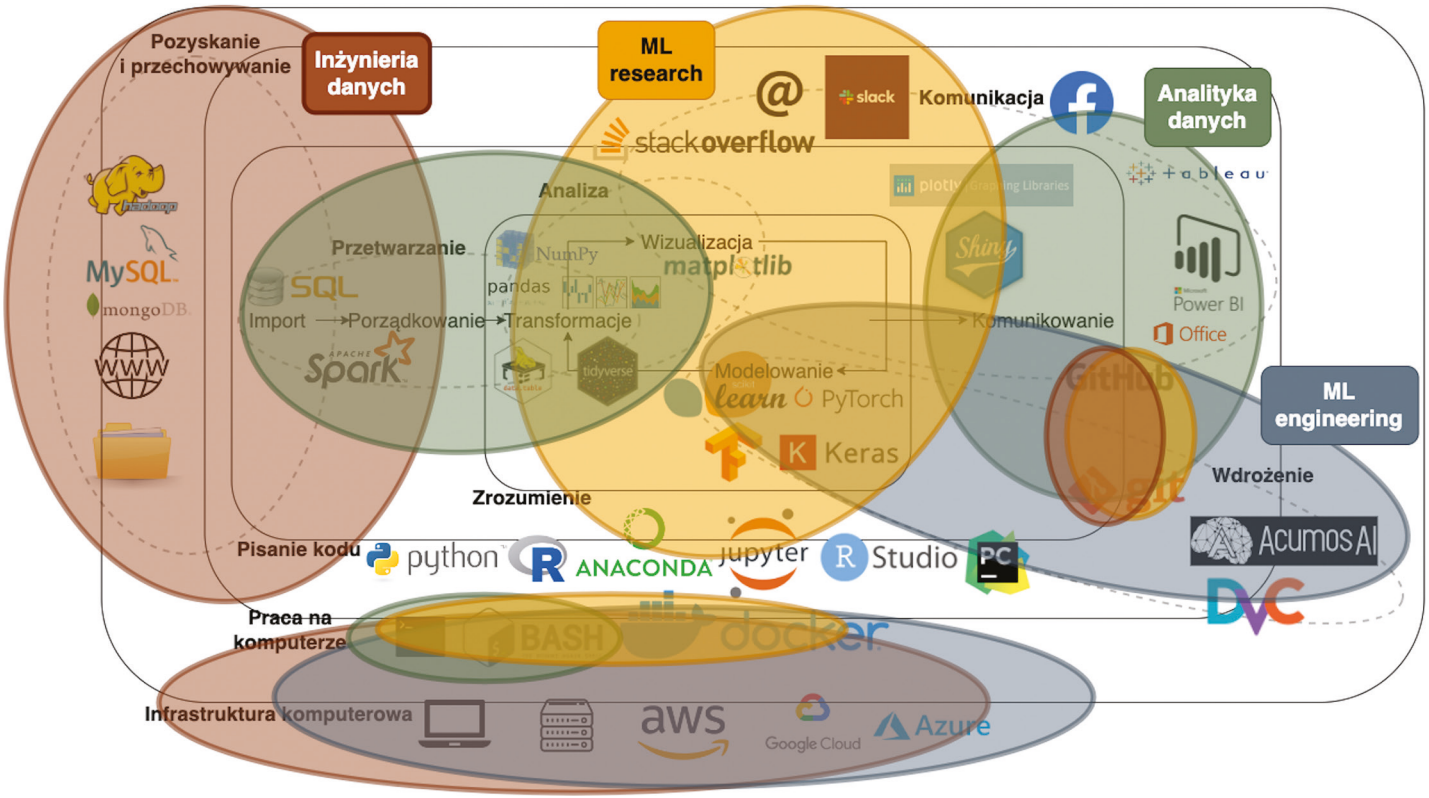
5.1.3. Segmentacja – profesjonalizacja

ML engineering, ML research to obok tzw. **inżynierii danych** subświaty profesjonalne badanego świata DS. Dwa pierwsze z wymienionych to wyodrębnianie się specjalizacji z DS. Inżynieria danych powstała zaś na przecięciu trzech obszarów – świata DS, big data w rozumieniu technicznym i baz danych w tradycyjnym rozumieniu.

Wyróżniam też inny segment/subświat – **analitykę danych** – starszą i bardziej ustabilizowaną niż DS. Ten segment został wchłonięty i przekształcony w części uznawanej za „podzbiór" świata DS. Inna część analityki danych jest przez uczestników świata DS postrzegana jako odrębna od DS. Granicę światów wyznacza właściwy sposób wykonywania działania podstawowego – pisanie kodu. Ta część analityki danych, która pracuje za pomocą interfejsów graficznych, jest poza DS, natomiast ta kodująca może być uważana za podzbiór. Subświaty ML engineerów/researcherów i inżynierów danych to subświaty domyślnie kodujące.

Skoro stos technologiczny jest obszarem odniesienia w praktykach zaświadczania o autentyczności w świecie społecznym DS, to jest także podstawą w wyznaczaniu granic społecznego świata DS oraz jego subświatów. Szkic granic omawianych subświatów w odniesieniu do stosu technologicznego świata DS prezentuję na rysunku 5.3.

W badanym świecie panuje względna zgodność co do tego, że **analityka danych różni się od DS brakiem stosowania metod ML**. Analityka, posługując się podobnymi narzędziami do przetwarzania i analizy danych, w bardzo małym stopniu nastawiona jest na budowanie „czegoś działającego", a raczej na raporty lub narzędzia do automatyzacji raportowania czy eksplorowania danych (np. dla zarządu firmy z użyciem Microsoft Power BI lub Tableau). Działanie podstawowe może być tu wykonywane zarówno w języku Python, jak i w R. Analityka jest w badanym świecie uznawana za „podzbiór" DS w tym sensie, że jakoby prawdziwy uczestnik świata DS potrafiłby sobie poradzić z zadaniami analityka – stanowią one wstęp do elementarza uczestnika badanego świata. Analityka postrzegana jest jako łatwiejsza, mniej sprawcza, mniej ciekawa, mniej rozwijająca niż DS. Spotkałem się z opiniami, że w najbliższych latach niemal całość pracy polegającej na opisowej analizie danych/raportowaniu może zostać zautomatyzowana. Zdobycie pracy kodującego analityka bywa postrzegane jako wstęp do pracy data scientisty. Informowano mnie także, że osoby zatrudniane na stanowiska data scientist bez doświadczenia zawodowego (tzw. junior) wykonują faktycznie zajęcia analityków danych. Analitycy danych raczej otrzymują wynagrodzenia niższe niż data scientiści, natomiast dla analityków publikowanych jest więcej ofert pracy – przy czym należy pamiętać, że analityka danych, posługująca się głównie arkuszami kalkulacyjnymi i językiem SQL, nie jest uznawana za część świata DS. Granica między kodującą analityką a data science jest bardzo rozmyta, bo choć mówi się o braku stosowania metod ML w analityce, to często zdarza się, że tzw. analitycy stosują pewne metody ML, a tzw. data scientiści stosują analizy opisowe/eksplorację/testowanie hipotez. Problem granic jest tu dyskutowany np. pod kątem tego, czy modele liniowe lub logitowe należy uznać za ML, czy znane od dziesięcioleci „proste" algorytmy ML (m.in. drzewa decyzyjne) należy uznać za charakterystyczne dla DS.



Rysunek 5.3. Subświaty/segmenty społecznego świata DS w Polsce – inżynieria danych, analityka danych, ML research, ML engineering w odniesieniu do stosu technologicznego i działania podstawowego DS

Źródło: opracowanie własne za pomocą draw.io, logotypy pobrane z oficjalnych stron WWW dostawców, inne rysunki na licencjach zezwalających na użycie.

W dyskursie nietechnicznym, np. w HR, w akademii i szeroko pojętym biznesie, trwa proces zbliżania się zakresów pojęciowych analityki danych i data science, co przez silnie osadzonych, „prawdziwych" uczestników świata DS może być postrzegane jako utrata prestiżu określenia „data scientist". Rysunek 5.3 jest w tym kontekście raczej przedstawieniem perspektywy takich silnie osadzonych, zaryzykuję stwierdzenie – konserwatywnie postrzegających DS uczestników. Dla innych uczestników, a także osób spoza świata DS (np. reprezentujących biznes czy HR) analityka danych i DS mogą być po prostu tożsame. Niemniej wskazywano mi, że to, co dawniej było nazywane analityką, od 2018 roku nazywa się DS.

> *B: Mmmm, dobra, to, to zacznijmy od czy ty jesteś data scientist?*
> R: […] Ee, jakby **formalnie** nie jestem, nie mam tego w tytule swojego stanowiska. Eee, ale myślę, że jeśli chodzi o zadania, które wykonuję, to w wielu firmach zostałoby to nazwane jako data scientist. Eeee, i podejrzewam, że gdyby teraz moja firma szukała kogoś na podobne stanowisko, też nazwałaby to, to, że szuka data scientist. {wywiad 16}

Ponieważ to modelowanie jest jakoby „najfajniejszą", najbardziej „seksowną", ciekawą, rozwojową i sprawczą częścią pracy data scientistów, to obecnie w badanym świecie rośnie prestiż subświatów ML engineer oraz ML researcher, a w komunikacji marketingowej i w ogóle biznesowej mówi się raczej o sztucznej inteligencji niż o analityce, big data czy nawet data science. Jedna z rozmówczyń {wywiad 8} wskazała, że aplikując do pracy w kraju Europy Zachodniej, zauważyła, że DS było tam postrzegane węziej niż w Polsce, gdzie prawdziwe DS to raczej tzw. *full-stack data science*. W tamtym kraju DS opisywano właściwie zgodnie z polskim rozumieniem kodującego analityka danych, a nie data scientisty.

ML engineer jest rolą profesjonalną odpowiedzialną za produkcyjne wdrażanie modeli i utrzymanie ich działania. Działanie podstawowe będzie tu wykonywane np. w językach Python, Scala, a także językach niskopoziomowych, jak C++. Nazwa wskazuje, że jest to stanowisko inżynierskie, zorientowane bardziej na stabilność rozwiązań niż eksplorację i eksperymenty. Rola ta może być uważana w badanym świecie za bardziej prestiżową niż DS z powodu większej sprawczości – koncentruje się przecież na wdrożeniach i utrzymaniu, a także za bardziej wymagającą i sprzyjającą samorozwojowi w obszarze znajomości technologii (również kojarzonych z tradycyjnym programowaniem, np. znajomością niskopoziomowych języków programowania). Ten subświat w znacznie większym stopniu niż DS zajmuje się problemami infrastruktury obliczeniowej, jej kosztów czy czasu działania. Może to wskazywać na przesunięcie w badanym świecie właśnie w kierunku wyższego wartościowania wdrożeń modeli niż modelowania (jako eksperymentowania z metodami i ich parametrami). Szczególnie że to drugie staje się coraz łatwiejsze, nawet do momentu, w którym nie wymaga samodzielnego pisania kodu (np. na platformach typu KNIME, RapidMiner, DataRobot, Dataiku czy platformie firmy Peltarion). Subświat ten zdecydowanie nie jest postrzegany jako „podzbiór" DS, a jako jego wyspecjalizowana część, w zakresie inżynierskim zaś także wykraczająca poza rozumienie DS (nawet szerokie, tzw. *full-stack*).

W prezentowanym niżej fragmencie wypowiedzi rozmówca {wywiad 19} wskazywał, że ML engineer to rola skoncentrowana na technologicznych aspektach projektów DS (ale nie przechowywania i dostępu do danych – to raczej inżynieria danych), natomiast data scientista na metodologicznych. Czasami praca zespołu/działu lub małej firmy zajmującej się DS jest taka, że klient wewnętrzny czy zewnętrzny otrzymuje zaakceptowaną wersję modelu, np. w pliku .h5, i wdraża go (lub nie) siłami własnego zespołu/działu IT. Trudno zatem określić, czy opisywany subświat raczej „wypączkował" ze świata DS, czy powstał na przecięciu DS i jakiegoś segmentu „tradycyjnej" informatyki (Strauss za Kacperczyk, 2016: 50).

ML researcher jest rolą profesjonalną odpowiedzialną za śledzenie nowości, tworzenie rozwiązań pośrednich/hybrydowych i rozwijanie autorskich metod ML. Działanie podstawowe będzie wykonywane zazwyczaj w Pythonie, lecz gama języków programowania może być szeroka, choć raczej nie obejmuje R. Być może za technologie do wykonywania działania podstawowego należałoby uznać zapis matematyczny i język ludzki (niemal zawsze angielski), a na niższym poziomie abstrakcji np. edytor LaTeX, czyli technologie do pisania artykułów naukowych w dziedzinie ML. Osoby takie nierzadko mogą być równolegle akademikami. Prestiż tego stanowiska postrzegany jest głównie w kontekście bardzo wysokiego poziomu znajomości szczegółów metod ML – zarówno matematycznych, jak i informatycznych – poziomu znacznie wyższego, niż potrzebuje data scientista do „efektywnego rozwiązywania prawdziwych problemów". Rola ML researchera może być postrzegana w badanym świecie jako elitarna, bliska nauce akademickiej, nadzwyczaj ciekawa, ale także nadzwyczaj sprawcza – tworząca nowe metody, potencjalnie dla milionów użytkowników. Subświat ten, podobnie jak ML engineering, jest uważany w świecie DS za specjalizację tak daleką, że wykraczającą poza jego granice. To proces pomiędzy „pączkowaniem" a przecinaniem się światów, tym razem świata DS i akademickich badań nad ML.

Inżynieria danych (*data engineering*) **tożsama jest z big data w sensie technicznym**. Data engineer cechuje się porównywalnym prestiżem i zarobkami co data scientista, ale nazwa ta określa węższy zakres obowiązków niż data scientisty. Działanie podstawowe data engineera koncentruje się wokół przechowywania i dostępu do danych, także przetwarzania ich oraz problemów dotyczących infrastruktury technicznej. W sensie działania podstawowego nie ma on nic wspólnego z analizą ani z modelowaniem danych. Używane są rozmaite języki programowania, technologie bazodanowe oraz rodzaje infrastruktury. Ten segment czy subświat inżynierii danych, a może po prostu świat oddzielny od DS, współpracuje z data science głównie w zakresie udostępniania zbiorów danych, a częściowo także przy wdrożeniach. Współpraca np. data engineera z ML engineerem może być niezbędna do przygotowania tzw. pipeline'u (dosł. 'rurociągu'), którym dane „płyną" w systemie produkcyjnym. Ten segment to profesjonalny subświat informatyki zajmującej się bazami danych, przecinający się ze światem DS. Inżynieria danych uznawana jest w badanym świecie za obszar względnie dobrze zdefiniowany w porównaniu do DS:

> R: Eee, czyli data engineer tak jak ja to rozumiem [śmiech], bo często to się może rozjeżdżać, chociaż to mi się wydaje jest **w miarę** już ustandaryzowane [krótka pauza], to jest ktoś, kto się zajmuje zbieraniem danych, yyy, czy do hurtowni danych, czy jak to się teraz nazywa data lake'u, coraz częściej jest to w chmurach, czyszczeniem ich oraz po prostu tworzeniem automatycznych procesów, które do tego służą. **Gdzieś** jakaś platforma na przykład internetowa generuje nowe dane, gdzieś już są zbierane i potem po kolei ileś tam kroków są automatyzowane, coś tam się dzieje, żeby te dane były w jakimś stopniu **przeczyszczone** i udostępnione ludziom, którzy potem się nimi zajmują, data scientistom, ludziom zajmującym się bardziej Bussines Intelligence itd. {wywiad 19}

W obszar inżynierii danych bywa jednak włączany zakres produkcyjnego wdrażania modeli ML, z kolei profesję ML engineera niektórzy uważają za wyłaniającą się właśnie z inżynierii danych (Gontarz, 2019: 9).

W badanym świecie buduje się zespoły realizujące to, co nazywam w niniejszej pracy projektami DS, wraz z wdrożeniami. Zespół złożony z data scientisty, ML engineera i ML researchera wskazywano jako wymarzony:

> R: […] złożenie dobrego zespołu, który będzie zawierał kogoś od data science, komu można szybko zlecić: zrób tą wizualizację, odpal ten algorytm, siamten algorytm, zobacz taką metodę [niezrozumiałe] i zrób to na próbce, zrób to na pliku CSV, zrób to na jakimś zrzucie sprzed miesiąca i niekoniecznie musisz kombinować, jak napisać dobrą końcówkę do jakiegoś mikroserwisu, który akurat prosto z produkcji ściął. No, no, no, data science nie będzie umiał tego kompletnie zrobić, będzie siedział i na to patrzył, co to w ogóle jest? Z drugiej strony jest ktoś, kto jest takim ML researcher, no to będzie ktoś, kto będzie zupełnie na bieżąco i będzie mówił [zmienia głos] dobra, no to może spróbujmy tu, a tutaj taki model opublikowano, a tutaj taki, a tutaj na GitHubie jest taka biblioteka, może tego spróbujmy, no i wreszcie ci ludzie na dole od ML, od inżynierii ML-a, którzy tak naprawdę to wszystko razem potem sobie spinają. To jest taki dream team, jeżeli chodzi o budowę takiego środowiska. {wywiad 23}

Organizacja pracy w zespołach DS (które mogą nazywać się np. zespołami AI czy analitycznymi) może być różna. Cassie Kozyrkow, Chief Decision Scientist w Google, pisze nawet o dziesięciu różnych rolach w perfekcyjnym zespole tego typu (Kozyrkov, 2018). Nawet nazwa jej stanowiska – *decision scientist* – bywa uważana za profesjonalny subświat DS lub za graniczący z DS, ale nietechniczny, raczej statystyczny lub naukowy i skoncentrowany na wspomaganiu decyzji. Jeszcze inaczej postrzegany jest nietechniczny (w Polsce przeze mnie niespotkany) segment o nazwie *analytics translator* – rola rozumiejąca ML, lecz niekodująca, skoncentrowana na biznesowych aspektach projektów i komunikacji z klientami (Henke, Levine, Mcinerney, 2018).

W czasie moich badań dyskusja o granicach i zakresach opisywanych subświatów dopiero się zaczynała, miałem okazję brać udział w meetupie poświęconym temu zagadnieniu {obserwacja 38}. Spór o kształtujące się granice tych profesjonalnych subświatów można by opisać jako szeroką arenę, w której stronami są np. świat DS, szeroki świat informatyki, świat marketingu, świat HR, świat decydentów.

Z punktu widzenia uczestników świata DS nierzadko przecież formalne nazwy stanowisk, sposoby wynagradzania pracowników, organizowania zespołów i ich pracy czy prowadzenia projektów DS nijak mają się do tego, jak oni postrzegają swój świat społeczny i wyróżnione subświaty wraz z ich specyfiką.

W badanym świecie panuje przekonanie, że DS znajduje się na wczesnym etapie rozwoju – technologicznym, specjalizacji ról profesjonalnych, świadomości potencjału biznesowego, adaptacji w rozmaitych branżach, regulacji prawnych itd. Można uznać to za kolejną legitymizację badanego świata. Ma ona głównie charakter neutralizacji. Nastawiona jest raczej na zewnątrz, do audytorium nietechnicznego. Szczególnie to „wczesne stadium rozwoju" dotyczyć ma Polski, postrzeganej w badanym świecie jako technologicznie i biznesowo peryferyjna, zapóźniona, z niskimi budżetami; przypomnę powszechną w polskim DS praktykę pracy dla klientów zagranicznych. Spotkałem się z porównaniem obecnego (w czasie realizacji moich badań) stadium rozwoju DS w Polsce do tego, jak wyglądał i działał polski internet na początku lat dziewięćdziesiątych. Pisano wówczas statyczne strony w języku znaczników HTML, uruchamiane przez modemowe łącza na stacjonarnych komputerach. E-maile były czymś egzotycznym, nie istniała skuteczna wyszukiwarka internetowa ani media społecznościowe, a tzw. internauci stanowili niemal subkulturę złożoną z informatyków, geeków i naukowców. Dziś strony internetowe może budować każdy, np. za pomocą szablonów Wordpressa, wiele osób jest połączonych z siecią 24 godziny na dobę za pomocą smartfona, a konta w mediach społecznościowych mają nie tylko noworodki, ale i zwierzęta domowe. Skoro tak rozwinęły się technologie i wzorce korzystania z internetu, to można sobie poprzez analogię wyobrazić, jak rozwiną się technologie i wzorce korzystania z produktów bazujących na ML.

5.2. Areny społecznego świata data science

Poprzez zaangażowanie w arenę społeczne światy i ich segmenty czy subświaty realizują swoje interesy i walczą o własne pozycje, warunkujące przetrwanie i rozwój. Arena jest dyskursywną, czasami także fizyczną przestrzenią spotkania i „pragmatycznej troski o daną sprawę" (Kacperczyk, 2016: 42). Społecznego świata nie sposób zrozumieć w oddzieleniu od innych światów, z którymi spotyka się na arenie – badacz powinien opisać stanowiska, sojusze i „wrogów" spierających się na arenie, a także to, co i jak chcą w owym sporze osiągnąć dla siebie. Badanie aren pomaga dotrzeć do wewnętrznej złożoności społecznych światów. Procesy np. segmentacji, legitymizacji społecznych światów mogą być najlepiej zrozumiane właśnie na arenach. Areny pojawiają się na każdym poziomie życia społecznego: od makro, przez mezzo i mikro, do pojedynczych aktorów – uczestników społecznych światów (Kacperczyk, 2016). Ja postrzegam areny w dużej mierze także jako spory o wartości.

W tym podrozdziale realizuję ostatni z postawionych sobie celów szczegółowych. Jest to określenie i opis głównych aren sporów oraz uchwycenie pełnej sytuacji badania. Odnoszę się często do wcześniejszych części książki, wprowadzając kilka nowych wątków. Dążę do nagłośnienia i objaśnienia wcześniej już sygnalizowanych kwestii, do syntezy moich ustaleń i podsumowania wykonanego opisu społecznego świata DS. Część domyka ujęcie pełnej sytuacji badania, na którą łącznie składają się wszystkie rozdziały tej książki.

5.2.1. Modelowanie

Modele ML postrzegam jako **obiekty graniczne**. W ujęciu Susan L. Star i Jamesa R. Griesemera (1989) obiekt graniczny jest czymś, co znajduje się na granicy społecznych światów, na ich przecięciach lub w pęknięciach. „Coś" może być zarówno materialne, jak i niematerialne, konkretne i abstrakcyjne, np. urządzenie, program komputerowy, lek, akt prawny, pojęcie. Różne światy czy subświaty zaangażowane w arenę mają różne interesy, wyrażające się w sposobach definiowania i interpretowania obiektu granicznego. Strony areny spotykają się „wokół" obiektu granicznego, każda ma wobec tego obiektu wymagania, będące wyrazem owych interesów. Problemem pojęcia obiektu granicznego może być to, czy wywołuje on podziały między światami/subświatami i krystalizowanie się areny, czy też to podziały wewnętrzne społecznego świata/światów (a więc potencjalna arena) „szukają" obiektów, które mogą stać się graniczne, tzn. obiektów pozwalających zarysować odmienność interesów i legitymizować swoją pozycję w dyskursie, forsując własną definicję obiektu (Kacperczyk, 2011: 192; 2016: 40–41). W opisywanej arenie skłaniam się ku pierwszemu sposobowi ujęcia ontologii obiektu granicznego, tzn. to komercyjne wdrażanie modeli ML jest faktem, który zapoczątkował krystalizowanie się owej areny.

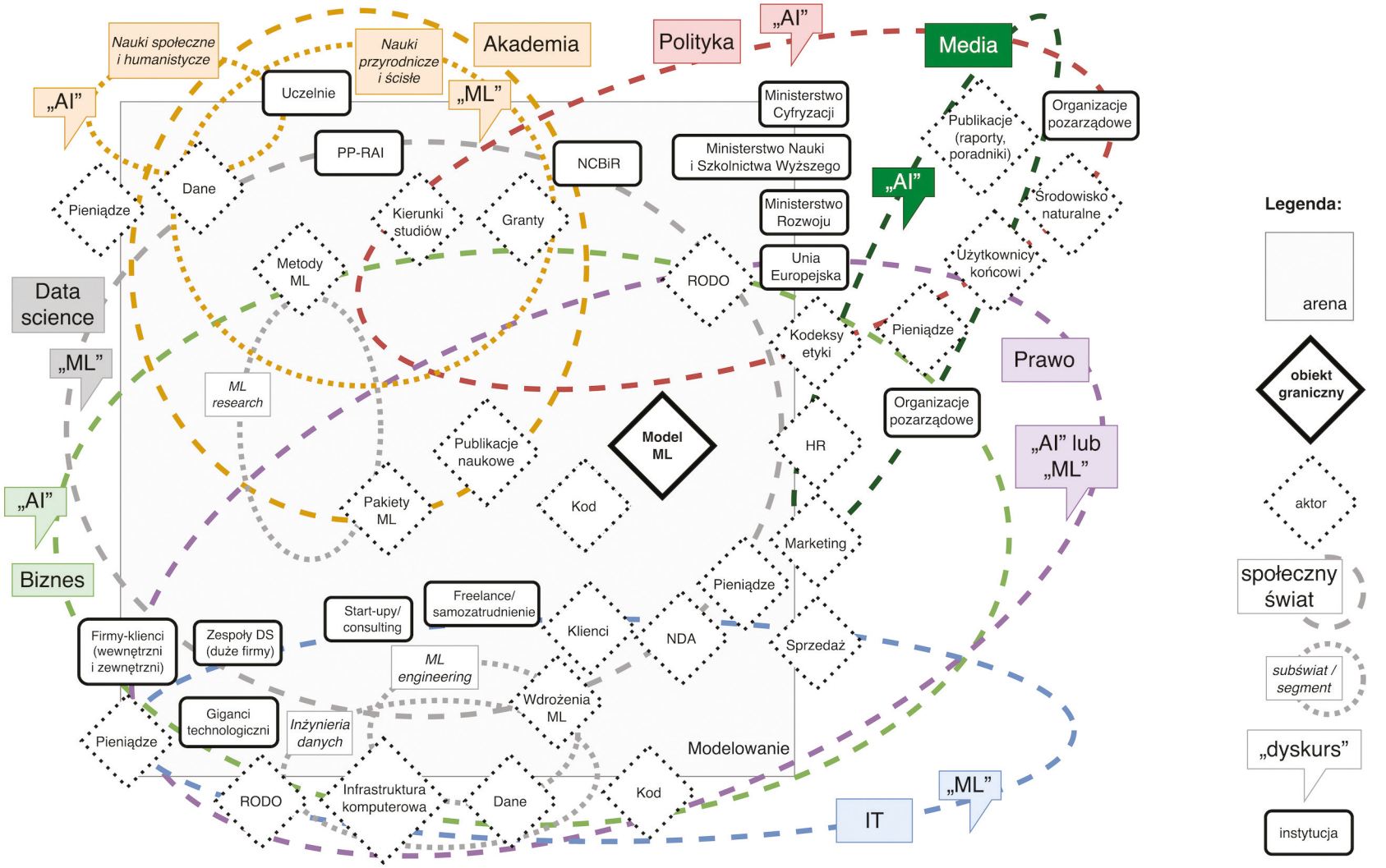
Modelowanie jest obiektem granicznym także „wewnątrz" badanego świata DS, o czym pisałem już wiele w odniesieniu do działania podstawowego, wartości, procesów, nie odwołując się bezpośrednio do pojęcia obiektu granicznego. Modelowanie jest nieodłącznym elementem składającym się na działanie podstawowe. Nie można być uczestnikiem badanego świata, nie wykonując modeli ML. Znajomość metod i narzędzi do modelowania, a także doświadczenie w komercyjnym tworzeniu modeli odróżnia uczestników słabiej od silniej osadzonych w badanym świecie. To modelowanie (a właściwie jego udział w obowiązkach służbowych oraz to, na jakich aspektach modelowania rola zawodowa jest skoncentrowana) jest tym, co odróżnia kodujących analityków danych od data scientistów, a tych drugich od ML engineerów i ML researcherów. To modelowanie jest „wisienką na torcie" pracy data scientistów, czymś najciekawszym, najfajniejszym, najbardziej seksownym. To modelowanie, a szczególnie wdrażanie modeli, daje sprawczość w świecie zewnętrznym, to ono zwiększa efektywność w wybranym wycinku rzeczywistości. Jednocześnie samo modelowanie nie wskazuje na uczestnictwo w badanym

świecie – osoba skoncentrowana tylko na budowaniu modeli ML może być zarówno akademikiem, jak i graczem w Kaggle lub wannabesem.

Na rysunku 5.4 przedstawiam modelowanie jako arenę, w której wokół modelu ML, bedącego obiektem granicznym, trwa spór społecznych światów – DS, biznesu, akademii, IT, mediów, polityki i prawa.

Względnie wspólne są interesy światów technicznych, czyli DS oraz IT (jako szeroko pojętej informatyki nieakademickiej). Światy te współpracują w miejscu znajdującym się „blisko" granic świata DS, czyli we wdrażaniu modeli ML. W zakresie uzyskania dostępu do danych i ich wstępnego przygotowania czy tzw. pipeline'u (przepływu danych w procesie – „rurociągu") oraz w zakresie różnych zadań dotyczących infrastruktury komputerowej zdarza się, że świat DS korzysta z usług świata IT. W innych przypadkach wszystkimi tymi czynnościami mogą zajmować się uczestnicy świata DS, a w większych działach/zespołach ludzie wyspecjalizowani w ML engineeringu lub inżynierii danych (szczególnie ci drudzy – raczej postrzegani jednak jako świat IT niż DS). Na rysunku 5.4 ująłem to jako subświat DS jedynie „stykający się" z granicą DS, a faktycznie będący „wewnątrz" świata IT. Charakterystycznymi aktorami w obu tych światach technicznych są dane, kod, klienci i pieniądze. Wspólne są zatem interesy finansowe, tj. światy DS i IT często zarabiają na tych samych projektach DS, dla tych samych klientów w ramach tych samych instytucji. Taka działalność komercyjna jest kluczowym czynnikiem przetrwania i rozwoju zarówno dla świata IT, jak i DS. Drugorzędnym, ale niezbędnym czynnikiem przetrwania i rozwoju dla świata DS jest współpraca ze światem akademickim. Klientami projektów DS są czasami również instytucje publiczne, jednak w Polsce skala takiej działalności jest na tyle mała, że nie uznałem jej za element zaangażowania świata polityki w opisywaną arenę. Kończąc wątek wspólnych interesów światów DS i IT, warto przypomnieć, że start-upy, zespoły DS czy pojedynczy freelancerzy mogą realizować projekty i zarabiać niemal bez wsparcia świata IT. Większość projektów DS kończy się na różnych etapach eksperymentów i nigdy nie dochodzi do produkcyjnych wdrożeń. Nie mam żadnych informacji na temat tego, aby uczestnicy świata DS byli wynagradzani tylko za efekty wdrożeń czy w ogóle za doprowadzenie do wdrożenia – wynagradzanie nie jest uzależnione od etapu zakończenia projektu.

W obu światach technicznych obiekt graniczny definiowany jest w sposób techniczny, czyli jako model ML. Wybór metody oraz przygotowanie modelu wytrenowanego, zwalidowanego i ocenionego jako „dostatecznie dobry" w odniesieniu do wymagań projektu jest kontrolowane niemal wyłącznie przez świat DS. Ze strony świata IT pojawiają się negocjacje związane z inżynieryjnymi aspektami, takimi jak czas działania do wygenerowania odpowiedzi modelu, jego stabilność, powtarzalność, ilość zużywanej mocy obliczeniowej, akceptowane przez model formaty danych itd. Raczej nie ma pomiędzy tymi światami negocjacji o aspektach metodologicznych. Dla świata IT modele ML są właściwie gotowymi czarnymi skrzynkami. Powodem tego nie jest brak rozeznania technicznego – jest nim ewentualnie brak rozeznania metodologicznego, ale nie w tym rzecz. Świat IT jest zobowiązany do innych, inżynieryjnych zadań, a sfera przygotowania modelu jest po stronie świata DS i zagłębianie się w nią nie leży w interesie ludzi z IT.



Rysunek 5.4. Modelowanie jako arena – mapa społecznego świata/areny w Polsce

Źródło: opracowanie własne z użyciem draw.io.

Skoro na arenie modelowania interesy finansowe światów technicznych realizowane są w instytucjach, czyli niemal wyłącznie w firmach komercyjnych, to **arena ta jest w dużej mierze kontrolowana przez światy biznesu oraz prawa**. Aktorem biorącym udział w każdym projekcie komercyjnym świata DS jest formalne zobowiązanie tajemnicy, często w formie tzw. NDA (*non-disclosure agreement*), które podpisują poszczególni uczestnicy projektu. Umowy typu NDA to ukonstytuowane zobowiązanie uczestników świata DS wobec organizacji/instytucji i osób, które płacą za usługę. Tajemnica raczej nie obejmuje kodu, szczególnie jeżeli część rozwiązania ujęta została przez data scientistów w pakiet, to raczej jest on publikowany jako *open source*. Nie są mi znane przypadki, by tajemnica obejmowała metody modelowania w sensie teoretycznym ani ujęte w kod/pakiet. Tym samym światy biznesu i prawa nie mają władzy nad kodem i metodami ML. Raczej mają władzę nad przygotowanymi, wytrenowanymi modelami ML – te mogą być objęte tajemnicą i w przypadku modeli komercyjnych nie ma praktyki publikowania gotowych modeli w *open source*. Nie byłoby to zgodne z interesem świata biznesu.

Świat biznesu ma dużą, może największą na opisywanej arenie władzę nad danymi, świat prawa walczy zaś o uzyskanie władzy. Prawo jest częściowo rzecznikiem polityki, a częściowo i pośrednio jednym z bardzo niewielu rzeczników użytkowników końcowych, będących przedmiotem działania wdrożonych produkcyjnie modeli ML. Ci „użytkownicy" nie zawsze będą ludźmi – wdrożenie może służyć np. klasyfikacji innych obiektów (książek, filmów, ogórków). Może i w tych wypadkach człowiek tylko pozornie nie jest przedmiotem – ostatecznie jest adresatem odpowiedzi systemu bazującego na modelu ML. Nie ma wątpliwości, że „użytkownikiem" jako przedmiotem działania owych systemów są ludzie, np. w przypadkach banków, firm windykacyjnych, zajmujących się edukacją, zdrowiem, marketingiem, ubezpieczeniami, telekomunikacją na podstawie danych dotyczących ludzi modelowane są ludzkie charakterystyki i działania.

Znaczący dla walki o władzę nad danymi aktor zaczął funkcjonować od 25 maja 2018 roku. To dzień, od którego w Polsce stosowane jest tzw. RODO (GDPR – *General Data Protection Regulation*), czyli Rozporządzenie Parlamentu Europejskiego i Rady (UE) 2016/679 z dnia 27 kwietnia 2016 r. w sprawie ochrony osób fizycznych w związku z przetwarzaniem danych osobowych i w sprawie swobodnego przepływu takich danych oraz uchylenia dyrektywy 95/46/WE (wcześniejszego rozporządzenia o ochronie danych). RODO ma na celu ujednolicenie i zwiększenie ochrony danych osobowych obywateli Unii Europejskiej. Wprowadza ono np. możliwość wystąpienia do administratora „swoich" danych o usunięcie ich (tzw. prawo do bycia zapomnianym) (Czapska, 2018; Ministerstwo Cyfryzacji RP, 2018a). W badanym świecie DS to prawo do bycia zapomnianym jest dyskutowane m.in. w kontekście sytuacji, w której już po przygotowaniu modelu ML osoba występuje o usunięcie „swoich" danych. Czy w takim przypadku legalne będzie korzystanie z modelu, w którym w procesie

przygotowania był użyty rekord z zapisem danych owej osoby, czy model należałoby uczyć i walidować ponownie, na danych bez rekordu, do którego usunięcia zobowiązuje RODO {obserwacja 46}? Model wyuczony i zwalidowany na określonych zbiorach danych byłby inny, gdyby wykorzystano inne zbiory danych. Ponowne przygotowanie modelu jest kosztowne, a żądania usunięcia danych mogą się powtarzać. To stawia pod znakiem zapytania opłacalność używania modeli ML w zastosowaniach dla danych osobowych, takie rozwiązanie prawne nie leży zatem w interesie światów IT, DS i biznesu. W chwili pisania tej części książki, w lipcu 2020 roku, nie istniała jednoznaczna interpretacja obowiązujących przepisów w opisywanej sprawie.

Z moich badań wynika, że problemy dotyczące legalności danych, z których korzystają uczestnicy świata DS, są postrzegane jako problemy świata DS tylko przez niektórych uczestników – osoby mające firmy lub zarządzające nimi, osoby zarządzające działami/zespołami DS. W akademii raczej takie problemy podnoszą osoby na stanowiskach profesorskich, poza tym problem jest zdecydowanie mniejszy. W środowisku komercyjnym osoby „szeregowe" zajmujące się DS uważają problemy prawne za nietechniczne oraz niebiznesowe, tak więc raczej nie leżą one w ich obszarze zainteresowania. Osoby „szeregowe" uważają, że odpowiedzialność za kwestie legalne spoczywa na działach prawnych (w dużych firmach wewnętrzny dział prawny bywa określany np. jako „bezpieka" {obserwacja 14}), na osobach zarządzających wyższego szczebla, a także klientach. Komercyjnie, akademicko oraz hobbystycznie używane są różne źródła danych, np. dane z wolnego dostępu, dane z systemów przemysłowych czy internetowych, niebędące danymi osobowymi (nawet gdy dotyczą ludzi, np. z Google Analytics). Czasami dane są ściągane z sieci za pomocą metod web scrapingu i akurat ta praktyka jest w badanym świecie powszechnie uznawana za mogącą prowadzić osoby/instytucje do negatywnych konsekwencji prawnych i stosowana jest albo w przypadkach niebudzących wątpliwości, albo potajemnie. W moich badaniach świata DS raczej nie miałem dostępu do informacji o nielegalnych praktykach dotyczących danych. Raz w rozmowie nieformalnej wskazano mi, że wcześniej stosowane przez tę osobę praktyki posługiwania się danymi „chyba były nielegalne" lub „chyba obecnie zostałyby uznane za nielegalne", więc lepiej o tym nie rozmawiać. W walce o władzę nad danymi świat prawa jest zawsze opóźniony w stosunku do możliwości technicznych i praktyki biznesowej, więc czasami nie wiadomo, co jest legalne, a co nie, ponieważ kwestie te są prawnie nieuregulowane.

Bardziej znaczącym dla świata DS elementem regulacji prawnych związanych z RODO jest tzw. prawo do wyjaśnienia. Nie jest ono precyzyjnie określone, ale kilka pozycji w opisywanym sporze (tzn. modelowaniu jako arenie) stoi na stanowisku, iż wynika ono z RODO. Chodzi o zapisy w sekcji 4 – „Prawo do sprzeciwu oraz zautomatyzowane podejmowanie decyzji w indywidualnych przypadkach", na które składają się artykuły 21 i 22 – „Prawo do sprzeciwu" oraz „Zautomatyzowane podejmowanie decyzji w indywidualnych przypadkach,

w tym profilowanie". W Polsce tzw. prawo do wyjaśnienia[5] obowiązuje od 4 maja 2019 roku tylko w odniesieniu do decyzji kredytowych. W społecznym świecie DS w Polsce raczej nie uznaje się oceny zdolności kredytowej za DS. To zastosowanie modelowania jest nie tylko starsze niż DS, ale bazuje na innych metodach i narzędziach niż te uznawane za „właściwe". Pewna część osób zajmujących się modelowaniem – nie tylko zdolności kredytowej – w instytucjach finansowych posługuje się np. językiem R i może być uznawana za uczestników świata DS, ale zdarza się to rzadko. Uczestnicy świata DS przeważnie uważają modelowanie w instytucjach finansowych za odmianę „tradycyjnej" analityki danych i statystyki, raczej nudną i konserwatywną, bo przez regulacje prawne ograniczoną do łatwo dających się wyjaśnić metod statystycznych, prawie nie ma w tej dziedzinie pracy z danymi nieustrukturyzowanymi ani używania dużych infrastruktur obliczeniowych w tzw. chmurze. Wejście w Polsce opisywanego prawa do wyjaśnienia decyzji kredytowej przedstawia jako swój sukces Fundacja Panoptykon, której działalność jest przykładem zaangażowania organizacji pozarządowych w arenę (Fundacja Panoptykon, 2019).

Tak zwane prawo do wyjaśnienia jest próbą prawnego uregulowania obiektu granicznego opisywanej areny, czyli modelu ML. W sensie technicznym całą władzę nad przygotowaniem modelu ML ma świat DS. To od konkretnych osób piszących kod zależą kluczowe szczegóły, to w gronie osób piszących kod zapadają decyzje techniczne. Kształt modelu jest negocjowany z klientami w obszarze

5 Ustawa z dnia 21 lutego 2019 r. o zmianie niektórych ustaw w związku z zapewnieniem stosowania rozporządzenia Parlamentu Europejskiego i Rady (UE) 2016/679 z dnia 27 kwietnia 2016 r. w sprawie ochrony osób fizycznych w związku z przetwarzaniem danych osobowych i w sprawie swobodnego przepływu takich danych oraz uchylenia dyrektywy 95/46/WE (ogólne rozporządzenie o ochronie danych) (Dz.U. z 2019 r., poz. 730): „Art. 70a. 1. Banki i inne instytucje ustawowo upoważnione do udzielania kredytów na wniosek osoby fizycznej, prawnej lub jednostki organizacyjnej niemającej osobowości prawnej, o ile posiada zdolność prawną, ubiegającej się o kredyt przekazują, w formie pisemnej, wyjaśnienie dotyczące dokonanej przez siebie oceny zdolności kredytowej wnioskującego. 2. Wyjaśnienie, o którym mowa w ust. 1, obejmuje informacje na temat czynników, w tym danych osobowych wnioskującego, które miały wpływ na dokonaną ocenę zdolności kredytowej. 3. W przypadku przedsiębiorców bank i inna instytucja ustawowo upoważniona do udzielania kredytów może pobierać opłatę za sporządzenie wyjaśnienia, o którym mowa w ust. 1. Wysokość opłaty powinna być odpowiednia do wysokości kredytu. 4. Przepisy ust. 1–3 stosuje się odpowiednio do przedsiębiorcy ubiegającego się o pożyczkę pieniężną". „1a. Banki, inne instytucje ustawowo upoważnione do udzielania kredytów, instytucje pożyczkowe oraz podmioty, o których mowa w art. 59d ustawy z dnia 12 maja 2011 r. o kredycie konsumenckim, a także instytucje utworzone na podstawie art. 105 ust. 4, mogą w celu oceny zdolności kredytowej i analizy ryzyka kredytowego podejmować decyzje, opierając się wyłącznie na zautomatyzowanym przetwarzaniu, w tym profilowaniu, danych osobowych – również stanowiących tajemnicę bankową – pod warunkiem zapewnienia osobie, której dotyczy decyzja podejmowana w sposób zautomatyzowany, prawa do otrzymania stosownych wyjaśnień co do podstaw podjętej decyzji, do uzyskania interwencji ludzkiej w celu podjęcia ponownej decyzji oraz do wyrażenia własnego stanowiska".

biznesowym, z IT w obszarze inżynieryjnym, z prawnikami w kwestiach legalnych, rzadziej z akademią w zakresie metod, ale nikt poza zespołem DS czy wręcz konkretnym człowiekiem piszącym kod nie ma władzy, by zadecydować o wyborze metody ML, użyciu konkretnego przekształcenia na danych, zastosowaniu konkretnej wartości hiperparametru. Chyba największy wpływ na kształt modelu (mowa o środowisku komercyjnym) mają klienci, których uwagi (w rodzaju np. „model działa za wolno", „jest za dużo fałszywych alarmów", „system jest za drogi, bo zużywa x zasobów infrastruktury obliczeniowej", „segmentacja użytkowników według modelu jest zupełnie inna niż nasza segmentacja z badań marketingowych") są w zespołach DS przy ewentualnej współpracy z IT tłumaczone na szczegóły techniczne i służą do optymalizowania przygotowywanego rozwiązania w kolejnych iteracjach. Z punktu widzenia „szeregowych" data scientistów sytuacja z tzw. prawem do wyjaśnienia wygląda podobnie jak z innymi przepisami wynikającymi z RODO czy dowolnych aktów prawnych[6] – nie zajmują się tym, ponieważ ich praca ma charakter techniczny, w razie potrzeby dostosowują się do zaleceń przełożonych i prawników. Dla DS jako działalności akademickiej lub hobbystycznej to prawo nie ma zastosowania.

Aktywnie w walce o „wyjaśnialne" modelowanie ML uczestniczą raczej osoby bardzo silnie osadzone w badanym świecie, jak Przemysław Biecek. Istnieje akademicki obszar badań i rozwoju ML, który jest odpowiedzią właśnie na powstające i przyszłe regulacje prawne, na kontrowersje etyczne stosowania modeli ML oraz na zapotrzebowanie klientów biznesowych na zrozumiałość modeli. Obszar ten nazywa się XAI (*eXplainable Artificial Intelligence*, dosł. 'wyjaśnialna sztuczna inteligencja'). Biecek publikował m.in. w tej dziedzinie artykuły naukowe[7] oraz pakiety (np. DALEX) i współtworzył wyposażone w chatbota narzędzie wyjaśniania modeli ML (Biecek, 2018c; Staniak, Biecek, 2018; Kuźba, Biecek, 2020), wypowiadał się o XAI na wydarzeniach świata DS (Biecek, 2018e; {obserwacja 46}), prowadził w tym zakresie zajęcia na Politechnice Warszawskiej i Uniwersytecie Warszawskim we współpracy z firmą McKinsey Digital (Biecek, 2020). Tego rodzaju działalność w dziedzinie XAI postrzegam jako wspólną walkę światów DS i akademickich badań nad ML (w subświecie nauk ścisłych) o wspólne interesy i pozycje, warunkujące ich przetrwanie i rozwój – szczególnie świata DS. Rozwiązania techniczno-metodologiczne, np. w formie pakietów takich jak DALEX, mają przecież pozwolić na wyjaśnianie nawet bardzo złożonych modeli ML, a więc zapobiec „wykastrowaniu" świata DS z najbardziej efektywnych, najbardziej sprawczych, być może najbardziej dochodowych metod zaawansowanego ML, DL i zestawów modeli. Żądanie wyjaśnienia może być:

6 Dowolnych aktów prawnych, ponieważ osoby zajmujące się DS w Polsce mogą pracować dla klientów na całym świecie.

7 „Praca została dofinansowana w ramach projektu RENOIR w ramach umowy grantowej Marie Skłodowskiej-Curie nr 691152 oraz grantu NCN Opus 2016/21/B/ST6/02176" (Biecek, 2018c: 4).

1) wymuszone przez legislację;
2) postulowane w mediach lub w świecie polityki (być może w formie rzecznictwa w imieniu marginalnie traktowanych użytkowników końcowych);
3) wymagane przez coraz bardziej wyedukowanych klientów niemających zaufania do systemów bazujących na ML (częściowo także z obawy przed konsekwencjami prawnymi czy wizerunkowymi), a więc i opornych wobec inwestowania w systemy będące tzw. czarnymi skrzynkami.

Metody XAI są zorientowane na zachowanie komercyjnych zastosowań niezbędnych, konstytutywnych dla świata DS technologii – czyli modeli ML – w obliczu żądań o wyjaśnialność. Rozwój XAI leży także w interesie akademii. Jako obszar nowy i modny w Europie może być obiecujący w perspektywie indywidualnych karier akademickich i budowania renomy instytucji poprzez publikacje naukowe, granty, udział w konferencjach, przedmioty i kierunki studiów, może mieć udział w debacie publicznej i wpływ na światy oraz możliwości współpracy ze światami polityki, prawa i biznesu. W ogóle przetrwanie i rozwój każdego rodzaju badań nad ML i innymi metodami czy narzędziami związanymi z DS leży w interesie akademii. Punktami przecięcia ze światem DS są przecież nie tylko metody i pakiety (same w sobie niekomercyjne), ale i granty (np. NCN lub NCBiR), doktoraty wdrożeniowe, publikacje naukowe. Są osoby określające się np. jako ML researcher, które łączą karierę akademicką i komercyjną, i to nie tylko w XAI. Wyjaśnialność, w sensie praktycznej możliwości wyjaśnienia działania (wdrożonego produkcyjnie modelu) konkretnemu operatorowi systemu bazującego na modelu ML lub konkretnemu użytkownikowi – przedmiotowi działania takiego systemu, leży także w interesie biznesu, o ile jest dostatecznie łatwa, efektywna i optymalna kosztowo w porównaniu z ryzykiem i kosztem konsekwencji prawnych, wizerunkowych lub etycznych.

Pojawiły się prace osób zajmujących się DS i ML, które nie są częścią XAI, a stanowią wołanie o etyczny ML lub też przykłady „czynienia dobra", działalności etycznie wzorcowej z zastosowaniem ML (Buolamwini, Gebru, 2018; Raji i in., 2020; Sefala i in., 2021). Są adresowane do środowiska technicznego, gdzie zyskują zainteresowanie i rozgłos, np. są przyjmowane na konferencję NeurIPS, niemniej w moich badaniach nie spotkałem się z ich obecnością na omawianej arenie.

Nauki społeczne i humanistyczne także włączają się w spór o rolę RODO w regulacji systemów bazujących na danych i modelach. Wskazywano między innymi, że:

> […] w RODO brakuje precyzyjnego języka i explicite wyrażonych praw (*rights*) oraz zabezpieczeń (*safeguards*), pojawia się zatem ryzyko, że to prawo będzie „bezzębne" (*toothless*) (Wachter, Mittelstadt, Floridi, 2017; tłum. R.Ż.).

Wskazano na pole do sprzecznych interpretacji RODO, a nawet na to, czy prawo do wyjaśnienia w ogóle rozwiązuje problem transparencji i odpowiedzialności/rozliczalności (Wachter, Mittelstadt, Floridi, 2017). W zasadzie humanistyka

i nauki społeczne uczestniczą w omawianej arenie modelowania tylko jako głos krytyczny. Jest to krytyka metodologiczno-epistemologiczna, gdzie czasem po tej samej stronie stają przedstawiciele nauk ścisłych i przyrodniczych, i przede wszystkim krytyka konsekwencji społecznych, gdzie znaczący jest wkład nauk prawnych. Raczej nie ma podstaw, by mówić o udziale w arenie tych przedstawicieli humanistyki i nauk społecznych, którzy używają ML w pracy naukowej – i tak rzadko są to rzeczywiście modele ML, są to raczej sposoby zbierania, przetwarzania i analizy danych, a jeżeli już, to według mnie model ML nie stanowi obiektu granicznego dla tej części subświata. Marginalny udział w opisywanej arenie mogą brać badacze społeczni, zajmujący się DS czy tzw. AI jako badanym wycinkiem rzeczywistości, jak ja. Niemniej rozwój, choć może niekoniecznie przetrwanie świata DS, leży w interesie nauk społecznych i humanistyki w ogóle. Świat DS dostarcza subświatowi akademii nowinek i modnych tematów: nowych narzędzi i strategii metodologicznych oraz pola badań i refleksji krytycznej w zakresie metodologii i epistemologii, konsekwencji społecznych, ekonomii, prawa, etyki, filozofii.

Wracając do własności danych, już pod koniec lat dziewięćdziesiątych w humanistyce wskazywano, że cyfryzacja prowadzi do coraz nowszych wyzwań prawnych; zasadnicze pytanie miało dotyczyć tego, do kogo będzie należała wiedza (Lyotard, 1997: 33). Zgadzam się z Yuvalem Hararim, że władza gigantów technologicznych staje się groźna w kontekście własności danych i to w kilku aspektach. Jego zdaniem sprzedawanie targetowanych reklam to obecna krótkoterminowa strategia Google'a czy Facebooka, a prawdziwym celem jest gromadzenie danych, aby „zhakować najskrytsze tajemnice życia" (Harari, 2018: 111–112). Harari przyjmuje za prawdę strategię legitymizacji badanego świata DS. Nie oznacza to, że jego tezy są bez sensu – pomijając futurystyczne wizje i skupiając się na etnograficznym „tu i teraz", słusznie wskazuje on na rzeczywisty problem olbrzymiej władzy gigantycznych firm technologicznych nad danymi. Nie będę powtarzać argumentacji i historii, takich jak afera Cambridge Analytica. Ta władza i wpływ na opisywaną arenę modelowania są ogromne.

Giganci technologiczni wspierają najpopularniejsze na świecie pakiety *open source* do modelowania ML. TensorFlow (TF), nakładka Keras (i mniej popularny TensorFlow.js) są wspierane przez Google, z kolei PyTorch (i mniej popularne Caffe) przez Facebooka. Giganci technologiczni oferują także popularne usługi udostępniania mocy obliczeniowej w tzw. chmurze – Microsoft Azure, Amazon Web Services (AWS) i Google Cloud Platform (GCP). Przykłady ich zaangażowania w arenę modelowania można by mnożyć: od kursów MOOC, oferowanych nie tylko przez wymienione firmy, ale także np. IBM, przez inne narzędzia, np. popularny do zastosowań dydaktycznych notatnik programistyczny Google Colab. Nawet jeżeli MOOC czy dowolny kurs (szkolenie/studia) oferuje jakaś firma bądź uczelnia, to uczy się na nich pracy na narzędziach technologicznych gigantów – inaczej nie jest to popularny kurs. Do Google'a należy także platforma Kaggle, za pomocą której ludzie uczestniczą w konkursach modelowania ML. W prasie

technologicznej wskazywano, że kilka lat temu dołączanie gigantów technologicznych do społeczności *open source* było nowością. Firmy te mogłyby przecież tworzyć zamknięte, komercyjne oprogramowanie do ML. Uznały jednak, że ten model nie sprawdzi się w przypadku DS.

Firmy takie jak Google i Facebook wspierają rozwiązania *open source* zapewne po to, by zachować możliwość szybkiego wprowadzania najnowszych rozwiązań metodologicznych, płynących do świata DS z akademii (ewentualnie przez subświat ML research). Inaczej środowisko DS szybko porzuciłoby „przestarzałe" oprogramowanie zamknięte, wypuszczone na rynek przez technologicznego giganta. Inną zaletą wspierania oprogramowania *open source* jest to, że przyciąga ono do gigantów technologicznych talenty z obszaru DS, dla których ważne są jednocześnie samorozwój, wyzwania, sprawczość i otwarty dostęp do wypracowanych rozwiązań. Takie talenty przyciągają z kolei innych data scientistów do pracy dla technologicznych gigantów, już nawet niekoniecznie w rozwijaniu narzędzi *open source*, ale w ogóle w zespołach/działach DS realizujących projekty dla tych gigantów. Autor pakietów Caffe i Caffe2, Yangqing Jia, pracował nad TF, będąc zatrudnionym w Google'u, a później przeszedł do Facebooka (Arakelyan, 2017). Wśród najsłynniejszych w świecie DS naukowców w Google'u pracowali m.in. Andrew Ng i Peter Norvig, a w Facebooku np. Yann LeCun. Giganci technologiczni są w świecie DS uznawani za wybitnie atrakcyjnych pracodawców, choć niektórzy uczestnicy wskazują, że nie podobają im się pewne ich działania. Podczas wywiadów, mówiąc o problemach etycznych związanych z modelami ML, wielokrotnie wspominano np. niesławny model używany do selekcji CV w firmie Amazon.

Rok 2018, szczególnie w związku z tzw. skandalem Cambridge Analytica, był przełomowy, ponieważ kwestie naruszeń prywatności i etyczność wykorzystywania danych weszły do głównego nurtu dyskursu publicznego. Skandal koncentruje się jednak na problemie dostępu do danych, własności danych i celu ich użycia, a nie na modelach. Problemy etyczne dotyczące używania modeli ML są poza głównym nurtem dyskursu publicznego, niemniej są znane i co najmniej od 2018 roku są na agendzie w opisywanej arenie modelowania. Zarzuty etyczne płyną ze strony światów: mediów (czasami w roli rzecznika użytkowników końcowych), polityki, prawa lub nauk społecznych i humanistycznych. Mogą one płynąć także od biznesu – klienta.

Odpowiedzi widzę dwojakie. Tą techniczną, ze strony świata DS i akademii (nauk ścisłych), jest XAI. Odpowiedzią nietechniczną, przede wszystkim ze świata biznesu, są tzw. kodeksy etyczne i rady/komisje ds. etyki. Kodeksy te to aktor usytuowany na peryferiach areny modelowania, mający styczność ze światem DS, ale ulokowany na przecięciu światów biznesu, mediów i polityki (można by opisać kodeksy etyczne jako obiekt graniczny dla „areny wewnętrznej" tych trzech światów). Kodeksy etyczne mają też styczność ze światem prawa, ale nie są aktami prawnymi, więc prawnicy przyjmują role pośrednie zarówno w ich tworzeniu (we współpracy z biznesem), jak i krytyce (wraz z mediami, naukami społecznymi

i humanistyką lub jako naukowcy). Kodeksy stanowią przede wszystkim działanie wizerunkowe firm, które w myśl liberalnej poprawności politycznej muszą pokazać światom mediów, polityki, prawa czy wreszcie klientom i użytkownikom, że dbają o prywatność, brak dyskryminacji itp. Jest to nastawione na budowanie albo odzyskiwanie zaufania do świata biznesu przez inne światy, od których zależy przetrwanie i rozwój firm. Firmy, a szczególnie technologiczni giganci, aby uniknąć twardych regulacji prawnych, chętnie tworzą miękkie rozwiązania deklaratywne, do których należą kodeksy (Wagner, 2018: 84). Brak regulacji prawnych może leżeć w interesie wielu firm zarabiających dzięki modelom ML i uważany być przez nie za stan pożądany, regulacje prawne zaś za przeszkodę w działaniu biznesu.

W polskim świecie DS miękkie rozwiązania deklaratywne odbierane są przez uczestników jako coś „spoza" świata DS, co ma znikomy lub żaden wpływ na arenę modelowania i na sam świat społeczny. Nie mam informacji, aby w polskim świecie DS istniał dokument w rodzaju np. kodeksu etycznego postępowania data scientistów. Istnieją dokumenty zwane kodeksami postępowania, tworzone przez organizacje uznawane za należące do świata DS (nie tylko polskiego). Nie dotyczą one modelowania, są tworzone głównie jako kodeksy dla konferencji (PyData, 2018; Why R? Foundation, 2018). Podobną treść mają kodeksy postępowania organizacji i inicjatyw nastawionych na włączanie do DS mniejszości genderowych (*gender minorities*) (por. WiMLDS, 2018; PyLadies, 2019; R-Ladies, 2019). Istnieje kodeks postępowania profesji DS, który mógłby się wydawać oddolnym, „wewnętrznym" dla świata DS dokumentem, ale w badanym przeze mnie świecie społecznym nie słyszałem o nim. Został on przygotowany około 2014 roku przez mało wpływowe i nieobecne w Polsce stowarzyszenie (Data Science Association, 2020). Propozycję profesjonalnej przysięgi Hipokratesa dla data scientistów przytoczyła też O'Neil (2017: 277). Jej treść[8] po załamaniu rynków w 2008 roku stworzyli specjaliści od inżynierii finansowej. Choć książka O'Neil jest znana w polskim świecie DS, to z przysięgą nie spotkałem się w trakcie badań. Z perspektywy uczestników świata DS nie ma ona znaczenia w badanym świecie ani w arenie modelowania. Podobnie raczej nie mają znaczenia tego rodzaju dokumenty[9] powstające lokalnie, poza światem DS. W Polsce poradniki/raporty poświęcone etyce

8 „Będę pamiętał, że to nie ja stworzyłem świat oraz że nie musi on dopasowywać się do moich wyliczeń. Będę śmiało używał modeli do określenia wartości, nie będę jednak ulegał nadmiernej fascynacji matematyką. Nigdy nie będę dążył do elegancji modeli kosztem zgodności ze stanem rzeczywistym bez wytłumaczenia, dlaczego tak postąpiłem. Nie będę stwarzał wobec klientów korzystających z moich modeli fałszywego poczucia pewności co do ich trafności. Zamiast tego będę wyraźnie informował o ich założeniach oraz brakach. Rozumiem, że moja praca może mieć ogromny wpływ na społeczeństwo oraz ekonomię, z którego w znacznej części mogę sobie nie zdawać sprawy" (O'Neil, 2017: 277).

9 Dla części uczestników mają znaczenie publikacje przygotowane przez podmioty spoza świata DS, np. zawierające wyniki badań związanych z branżą DS w Polsce (Borowiecki, Mieczkowski, 2019; Gontarz, 2019).

tzw. AI przygotowały np. organizacje pozarządowe: Klub Jagielloński z Fundacją Centrum Cyfrowe (Tarkowski, Paszcza, Mileszyk, 2020)[10] i Fundacja Panoptykon (Szymielewicz, Obem, 2020)[11]. W drugim dokumencie zacytowano Biecka i podano link do strony MI^2 DataLabu, zajmującego się m.in. XAI, ale wskazuje to jedynie na rozeznanie autorek w temacie, a nie na znaczenie dokumentu dla świata DS. Z nieco większym zainteresowaniem traktowane mogą być takie dokumenty jak tzw. Biała Księga AI (Komisja Europejska, 2020), szczególnie przez właścicieli firm zajmujących się DS i osoby na stanowiskach kierowniczych w DS. Tego rodzaju dokumenty wskazują na potencjalny kierunek rozwoju regulacji prawnych oraz przyszłe kierunki publicznego finansowania projektów badawczych. Dokumentów dotyczących potencjalnego lub realnego finansowania badań nad AI istnieje już wiele, jak wymieniono np. w raporcie z pierwszego Zjazdu Polskiego Porozumienia na Rzecz Rozwoju Sztucznej Inteligencji (PP-RAI):

> Panelistka zwróciła jednocześnie uwagę na znaczenie europejskiego partnerstwa publiczno-prywatnego (Big Data Value Association – BDVA), które odgrywa kluczową rolę przy definiowaniu tematów związanych z Big Data w ramach programu H2020. Następnie p. Szołucha zaprezentowała szacunkowe budżety przeznaczone przez KE dla poszczególnych obszarów tematycznych. Zgodnie z zapowiedziami KE inwestycje w projekty B+R w latach 2018–2020 mają wynieść ok. 1,5 mld euro. Według panelistki te inwestycje KE mają przygotować grunt do przyszłych inwestycji, planowanych na nową perspektywę budżetową, na lata 2021–2027. Zgodnie z zapowiedziami KE, obok Programu Horyzont Europa (następcy Programu Horyzont 2020), utworzony zostanie nowy program, Program Cyfrowa Europa (ang. Digital Europe), w ramach którego mają być realizowane wdrożenia i testy w środowiskach przemysłowych. Planowany budżet programu to ponad 9 miliardów euro. Jak podkreślała podczas swojego wystąpienia Pani Szołucha, jednym z pięciu strategicznych obszarów tego programu ma być sztuczna inteligencja, z budżetem 2,5 miliarda euro na działania związane z upowszechnianiem wdrożeń sztucznej inteligencji w całej gospodarce i społeczeństwie europejskim (Czarnowski i in., 2018: 11–12).

Tak zwana Biała Księga jest w komercyjnym DS traktowana jako niemająca znaczenia, szczególnie dla codziennej pracy „szeregowych" data scientistów. Może być uznawana za potencjalnie wpływową raczej przez przedstawicieli świata biznesu:

> Biznes aprobuje to, że Komisja [Europejska – przyp. R.Ż.] chce regulować wyłącznie zastosowania SI „wysokiego ryzyka", czyli np. w medycynie czy transporcie. „Biała księga" pomija jednak sprawę zasadniczą: odpowiedzialności za szkodę oraz krzywdę wyrządzoną przez rozwiązania oparte na sztucznej inteligencji – zauważa Aleksandra Musielak, ekspertka z Departamentu Prawa Gospodarczego Konfederacji Lewiatan w rozmowie z naszym portalem [sztucznainteligencja.org.pl – przyp. R.Ż.] (Zagórna, 2020).

---

10 Tekst pod tytułem zawierającym powszechne na opisywanej arenie odniesienie do popkultury – brytyjskiego, dystopijnego serialu sci-fi – *„AlgoPolska": 11 postulatów, których realizacja pozwoli uniknąć mrocznych wizji rodem z „Black Mirror"*. Działanie sfinansowane ze środków Programu Rozwoju Organizacji Obywatelskich na lata 2018–2030.

11 Tekst pt. *Sztuczna inteligencja non-fiction*. Materiał powstał dzięki wsparciu Samsung Electronics Polska.

W ramach europejskiej strategii cyfrowej na lata 2019–2024 trwają w UE dalsze prace nad „godną zaufania sztuczną inteligencją" (Komisja Europejska, 2022c) oraz nad dwoma powiązanymi obszarami – aktem o usługach (*Digital Services Act* – DSA) i o rynkach cyfrowych (*Digital Markets Act* – DMA) (Komisja Europejska, 2022a; 2022b). Powstał także projekt rozporządzenia w sprawie sztucznej inteligencji. Jest to zapowiedź wprowadzenia jednolitego uregulowania kwestii opracowywania, wprowadzania do obrotu i wykorzystywania systemów AI na obszarze UE (Komisja Europejska, 2021). Byłoby to na opisywanej arenie modelowania sprawczym aktorem, co najmniej na miarę GDPR. W tym projekcie rozporządzenia UE używany jest termin „systemy AI", podczas gdy w podobnym dokumencie w USA stosowany jest termin „automatyczne systemy decyzyjne" (ASD). To drugie określenie wydaje się lepsze w tym sensie, że nie wskazuje na konkretną technologię wykonania systemu, a na jego zastosowanie. Unika także sporów o to, czym jest AI (Mökander i in., 2022).

Swoje wytyczne dotyczące etyki AI, przygotowała także – poza tym epizodem nieobecna na arenie – strona Kościoła rzymskokatolickiego (Pontificia Accademia per la Vita, 2020b). Dokument sygnowali przewodniczący Papieskiej Akademii Życia Vincenzo Paglia, prezes firmy Microsoft Brad Smith, wiceprezes IBM John Kelly III, dyrektor generalny ONZ ds. Wyżywienia i Rolnictwa Dongyu Qu i włoska minister ds. innowacji technologicznych Paola Pisano. W ceremonii uczestniczył przewodniczący Parlamentu Europejskiego David Sassoli, odczytano list papieża Franciszka, wystąpił filozof Luciano Floridi, a sygnatariusze wezwania wyrazili chęć współpracy w celu promowania etycznego wykorzystania AI zgodnie z sześcioma[12] zasadami (Mróz, 2020; Pontificia Accademia per la Vita, 2020a). Kościół rzymskokatolicki oraz w ogóle kościoły, związki wyznaniowe i instytucje religijne to „światy nieobecne, jakich można byłoby oczekiwać, że się pojawią [na opisywanej arenie modelowania – przyp. R.Ż.]" (Kacperczyk, 2016: 573). Może współcześnie religie nie radzą sobie w odpowiadaniu na problemy techniczne ani polityczne, ale potrafią budować interpretacje kształtujące tożsamość (Harari, 2018: 120). Takie problemy jak np. rozwój AI wymagają rozwiązań na poziomie globalnym, w czym żadna z religii nie może być pomocna, gdyż każda uważa tylko swoją perspektywę za słuszną. Tym samym zdolność religii do kształtowania tożsamości Harari (2018: 115–122) postrzega jako część problemu, a nie coś, co przyczynia się do jego rozwiązania. Niemniej i tak poza wspomnianym dokumentem

12 1. Transparentność – rezultaty działania algorytmów SI [sztucznej inteligencji – przyp. R.Ż.] powinny być w pełni możliwe do wyjaśnienia. 2. Włączenie – powinny być uwzględnione potrzeby wszystkich ludzi, by wszyscy mogli na równi korzystać z możliwości rozwoju. 3. Odpowiedzialność – ci, którzy projektują i budują SI, powinni postępować odpowiedzialnie i transparentnie. 4. Bezstronność – algorytmy SI nie powinny działać według żadnych uprzedzeń, gwarantując przez to sprawiedliwość i ochronę ludzkiej godności. 5. Niezawodność – systemy SI powinny umożliwiać niezawodne działanie. 6. Bezpieczeństwo i prywatność – systemy SI muszą działać w sposób bezpieczny i z poszanowaniem prywatności użytkowników (Mróz, 2020).

nie widać dotychczas prób budowania takich kształtujących tożsamość interpretacji ze strony religii – w sensie instytucji, bo media związane z konkretnymi wyznaniami, także w Polsce, publikowały już teksty na ten temat (Merritt, 2018; Jóźwiak, 2019). Wydaje mi się to zaskakujące, ponieważ zajęcie stanowiska o charakterze tożsamościowym, moralnym, wobec zastosowań tak rozbudzającej dyskurs publiczny technologii AI leży w interesie każdej religii instytucjonalnej. Zdecydowane stanowisko mogłoby przecież być elementem wpływu na politykę i prawo, jak instytucjonalne religie czynią w rozmaitych sprawach. Wymieniony dokument Papieskiej Akademii Życia jest próbą wejścia Kościoła rzymskokatolickiego na omawianą arenę.

Swoje stanowisko w sprawach etyki AI publikowały przeróżne podmioty państwowe, prywatne, organizacje pozarządowe, międzynarodowe konsorcja, uczelnie i inne. Dokumenty te mają różny charakter – od zbioru zasad, jakie przygotowano np. z udziałem Elona Muska i Stephena Hawkinsa w Asilomar (Future of Life Institute, 2017), przez dokumenty bardziej nastawione na prezentowanie wartości i wizji, po tzw. strategie rozwoju AI w różnych krajach. Są nawet zestawienia i analizy dziesiątków takich dokumentów (Alekseichenko i in., 2018; Greene i in., 2019; Rothenberger, Fabian, Arunov, 2019; Fjeld i in., 2020; Taylor, Dencik, 2020). Takie publikacje można by uznać za obiekty graniczne dla miniaren usytuowanych na obrzeżach opisywanej areny modelowania. Dla świata DS mają one jednak nikłe znaczenie.

Zdecydowanie największy wpływ – choć i tak bardzo mały – mają dokumenty dotyczące uczestników świata DS nie jako uczestników świata, a jako pracowników. Wówczas są oni zobowiązani formalnie, by co najmniej otwarcie nie występować przeciwko stanowisku pracodawcy. Niemniej i tak wielokrotnie spotykałem się z interpretowaniem firmowych kodeksów czy wytycznych jako wyłącznie działań wizerunkowych, nierzadko robionych „na wyrost", czasami nieadekwatnych do pracy data scientistów.

W przypadku osób pracujących w dużych firmach z jednej strony istnieje w organizacji ogólny kodeks etyki, z drugiej pracują prawnicy troszczący się o zgodność działalności z przepisami w odpowiednim kraju. Data scientiści, budując i walidując modele, zwracają uwagę na tzw. biasy jako problem techniczny wynikający ze specyfiki użytego zbioru danych, metod ML i sposobu dostrojenia modelu. Powtarzają się opinie o kodeksach, będących elementem strategii wizerunkowej, szczególnie w przypadku gigantów technologicznych pracujących nad odbudową zaufania i dopasowaniem się do głównego nurtu politycznej poprawności po wpadkach ujawnionych w mediach.

W polskim świecie DS nie tworzy się jak dotąd kodeksów DS czy modelowania, choć za problem uznaje się możliwość dyskryminowania czy jakkolwiek pojętego krzywdzenia użytkowników końcowych. Jednak kodeks to nie jest „coś działającego". **Praca nad nim byłaby w świecie DS raczej uznana za bezwartościową, ponieważ deklaratywnych rezultatów nie uważa się za coś, co rozwiązuje**

rzeczywisty problem. Za rzeczywisty problem uznawane są tzw. **biasy, traktowane jako problem techniczny**, a więc rozwiązywalny **za pomocą kodu**. Nietechniczne, deklaratywne rozwiązania problemu biasów w modelach mają więc dla uczestników świata DS podobny status jak przygotowywane przez biznes materiały marketingowe, sprzedażowe, wychwalające oferowane rozwiązania z użyciem pojęcia AI lub ewentualnie zachwalające pracę przy rozwijaniu tejże – to wszystko tylko opakowanie dla rozwiązań bazujących na ML, tylko slajdy w PowerPoincie.

W technicznej książce poświęconej XAI podano wymogi, które powinien spełniać każdy model predykcyjny. Są one wzorowane na trzech prawach robotyki Isaaca Asimova z wydanej w 1942 roku książki sci-fi *Zabawa w berka* (Biecek, Burzykowski, 2020):

1) walidacja – dla każdej predykcji modelu należy być w stanie zweryfikować, jak silne są dowody wspierające tę predykcję;
2) uzasadnienie – dla każdej predykcji należy być w stanie zrozumieć, które zmienne wpływają na nią i w jakim stopniu;
3) spekulacja – dla każdej predykcji należy być w stanie zrozumieć, jak ta zmieniłaby się, gdyby zmieniły się wartości zmiennych ujętych w modelu.

Zdaniem Biecka i Burzykowskiego (2020) obecnie w ML standardem jest rozpoczynanie od eksploracyjnej analizy danych, następnie ma miejsce przygotowanie wielu rodzajów modeli, na koniec zaś wykonywana jest ich walidacja, np. metodą podział zbioru danych na treningowe i testowe oraz walidacja krzyżowa. Ich wyniki służą iteracyjnemu poprawianiu modeli. Być może w nieodległej przyszłości zarówno eksploracja (autoEDA), jak i modelowanie (autoML) zostaną w wielu zadaniach technicznych zautomatyzowane, walidacja będzie zaś odbywać się z użyciem metod XAI oraz zasad etyki i sprawiedliwości. Autorzy wskazują na wiele przykładów negatywnych konsekwencji społecznych wdrażania modeli będących tzw. czarnymi skrzynkami. W pracach takich jak publikacja Biecka i Burzykowskiego (2020) wymogi wobec modeli predykcyjnych nie są uznawane za rozwiązanie deklaratywne, a za wyjaśnienie założeń stojących za proponowanymi rozwiązaniami technicznymi. Są to wymagania wobec modeli, więc nie mogą być traktowane jako kodeks postępowania dla osób zajmujących się DS i nie mówią nic ani o domenie, w której budowany jest model, ani o potencjalnych konsekwencjach jego wdrożenia.

Być może należy spodziewać się powstania w Polsce „środowiskowego" kodeksu postępowania w DS, który stanie się kolejnym obiektem granicznym miniareny w opisywanej arenie. Będzie to miniarena „wewnątrz" badanego świata DS. Deklarowano mi indywidualną wrażliwość na kwestie etycznych konsekwencji rezultatów pracy data scientistów, na prywatne lektury (jak *Broń matematycznej zagłady* O'Neil czy prace Harariego), na niechęć do pracy w obszarach, które postrzegano jako wątpliwe etycznie. Niemniej w moich badaniach nie znajduję sygnałów o kształtowaniu się etyki DS o charakterze zbiorowym. Etyka, w sensie oceny rezultatów działalności pracy, jest uważana za indywidualną sprawę

każdego z uczestników i nie podlegają oni w tym zakresie zbiorowym zobowiązaniom wobec społecznego świata – zupełnie inaczej niż w przypadku wartości tego świata czy zaświadczania o autentyczności. Wskazywano mi, że na razie kodeksy powstają w firmach na zasadzie reakcyjnej – jako wyraz nastawienia biznesu, tak jak świata DS, głównie na efektywność. Po co robić kodeks, o który nikt nie prosi? Po co naprawiać coś, co nie jest zepsute?

Postrzeganie firmowych kodeksów przez uczestników świata DS jako działań wizerunkowych nie musi oznaczać, że są negatywnie oceniane. Jedna z rozmówczyń zaznaczała własną wrażliwość etyczną – „czuję gdzieś te zasady, […] żeby ten produkt, który tworzymy, eee, nie odzwierciedlał, nie wiem, czyichś przekonań" {wywiad 20}. Z jednej strony uczestnicy mogą nie uznawać kodeksów za sprawcze, z drugiej jednak mogą być one uważane przez nich za względnie potrzebne, ale też niekoniecznie, wystarcza im bowiem indywidualne poczucie etyki.

Tworzenie krajowych czy obszarowych strategii rozwoju AI jest dość popularnym w świecie polityki sposobem uczestniczenia w opisywanej arenie modelowania. Polskie prace w tym zakresie (Koloch i in., 2017; Borowik i in., 2018; Ministerstwo Cyfryzacji RP, 2018b; Kroplewski i in., 2019; Marczuk i in., 2019) są entuzjastyczne. W ich przygotowywaniu brały udział osoby reprezentujące organizacje pozarządowe (jak Fundacja Digital Poland i Klub Jagielloński) i uczelnie wyższe.

Ministerstwo Cyfryzacji RP[13] twierdzi, iż w UE brakuje 400 tys. pracowników na „rynku danych" i luka ta będzie się powiększać (Borowik i in., 2018). Wskazano, że „niezbędne jest, abyśmy w Polsce rozwijali dziedziny związane ze zbieraniem, analizowaniem oraz przetwarzaniem danych", zaznaczając, że:

> […] spośród dostępnych dla nas możliwości rozwojowych najkorzystniejsze jest inwestowanie w gospodarkę opartą o dane. Polska gospodarka może odnieść ponadprzeciętne korzyści dzięki zwiększonej produktywności i nowym modelom biznesowym (Borowik i in., 2018).

Program gospodarki opartej na danych nazwano Przemysł+ (Borowik i in., 2018). W innym dokumencie Ministerstwo Cyfryzacji posługuje się terminem „AI", a całość strategii bazuje na twierdzeniu, że **gospodarka oparta na danych stanowi kluczowy element rozwoju Polski i będzie „główną siłą napędową wzrostu gospodarczego i tworzenia miejsc pracy, nie tylko w obszarze zaawansowanych technologii, lecz całej gospodarki"** (Ministerstwo Cyfryzacji RP, 2018b: 7). Odniesiono się do wielu międzynarodowych strategii rozwoju AI[14] – są to dokumenty Unii Europejskiej, Organizacji Narodów Zjednoczonych i grup G-7

13 W tekście i na rysunku 5.4 prezentuję nazwy ministerstw z roku 2020. W roku 2022 nie istnieje już Ministerstwo Cyfryzacji, a Cyfryzacja w Kancelarii Prezesa Rady Ministrów. Rolę Ministerstwa Nauki i Szkolnictwa Wyższego przejęło Ministerstwo Edukacji i Nauki, a Ministerstwa Rozwoju – Ministerstwo Rozwoju i Technologii.

14 Przegląd takich strategii na świecie przygotowała Fundacja Digital Poland (Alekseichenko i in., 2018).

oraz G-20 (Ministerstwo Cyfryzacji RP, 2018: 104–109). Współtwórczyni omawianego dokumentu wskazuje, że stosowanie w biznesie rozwiązań bazujących na danych jest jednoznacznym źródłem przewagi konkurencyjnej:

> Sukces osiągną bowiem te spośród nich [firm i marek – przyp. R.Ż.], które będą w stanie reagować, być może w czasie rzeczywistym, umiejętnie łącząc dane, treści, kanały i technologie (Kaczorowska-Spychalska, 2019).

Ministerstwo Cyfryzacji stwierdziło także, że AI zarówno zlikwiduje, jak i utworzy miejsca pracy. Aby Polska znalazła się po tej stronie równania, po której powstaną nowe miejsca pracy, Ministerstwo Cyfryzacji postuluje istnienie ponad siedmiuset firm budujących AI w Polsce do roku 2025. „AI przetransferuje bogactwo do tych krajów, które będą potrafiły ją budować i kontrolować" (Ministerstwo Cyfryzacji RP, 2018: 22–23). Polska musi budować własne AI, ponieważ:

> [1] Dane to zasób. Na razie potrafimy go zdobyć, ale jeszcze słabo potrafimy go analizować. Jeżeli nie będziemy mieli własnych rozwiązań AI, staniemy się krajem „surowcowym". Będziemy dostarczać dane, ale będziemy zależni od tych, którzy potrafią je zmienić w wysokoprzetworzony i dochodowy produkt. „Scenariusz Wenezueli" [2] Spójrzmy prawdzie w oczy: sztuczna inteligencja będzie eliminowała kolejne zawody. Które, jak szybko i w jakim stopniu, to oddzielna dyskusja. To, co ważne, to na każde zlikwidowane 1000 miejsc pracy sztuczna inteligencja wygeneruje około 1280 nowych (Gartner – „AI and the future of work", grudzień 2017). Jeżeli nie będziemy budować naszego AI, to praca zniknie w Polsce, ale pojawi się w innym kraju. Tak jak rewolucja IT zabrała pracę w wielu krajach, ale wykreowała ją w Indiach czy… w Polsce (Ministerstwo Cyfryzacji RP, 2018b: 22).

Owe 720 firm ma odzwierciedlać potencjał intelektualny i biznesowy Polski w porównaniu do innych krajów budujących AI, przyjęto bowiem, że w 2025 nasz kraj ma znajdować się między 20. a 25. percentylem najlepszych w tej dziedzinie krajów na świecie. Chodzi o 720 firm „budujących AI, a nie tylko wdrażających", wyjaśnienia tego nie znajdujemy w dokumencie. Oceniono liczbę owych budujących AI firm na trzydzieści podmiotów w roku 2019[15] i mają one generować łączne przychody w kwocie około 0,35 mld PLN. W roku 2025 przychód tej branży ma wynosić 8,3 mld PLN, a więc 24 razy więcej (Ministerstwo Cyfryzacji RP, 2018b: 22–23). Zachowano zatem zbliżony przychód na firmę[16].

> W pierwszej kolejności Pan Robert Kroplewski [Pełnomocnik Ministra Cyfryzacji do spraw społeczeństwa informacyjnego[17] – przyp. R.Ż.] rozszerzył informacje na temat nowych inicjatyw międzynarodowych oraz aktualnego stanu prac nad tzw. europejską strategią sztucznej inteligencji. Podsumował swoje doświadczenia z udziału w różnych międzynarodowych grupach

15 Digital Poland ocenia ich liczbę wyżej, pisząc o 264 ankietowanych firmach AI (Borowiecki, Mieczkowski, 2019).

16 Dla trzydziestu firm zysk 0,35 mld PLN to około 11,67 mln PLN na firmę. Dla 720 firm przy zysku 8,3 mld PLN to odpowiednio 11,53 mln PLN.

17 Cyfryzacja KPRM, b.d.

> eksperckich – wskazał także na skuteczność działań ze strony ministerstw, przykładowo powołanie do grona wysokiego szczebla ekspertów AI przy OECD (tzw. AIGO). […] W tej chwili tematyką sztucznej inteligencji zajmują się głównie trzy ministerstwa, tj.: [1] Ministerstwo Przedsiębiorczości i Technologii, w kontekście innowacji gospodarki, [2] Ministerstwo Nauki i Szkolnictwa Wyższego, w zakresie głównie związanym z finansowaniem prac badawczo-rozwojowych (zwłaszcza NCBiR) oraz kształceniem kadr, [3] Ministerstwo Cyfryzacji, w zakresie możliwych zastosowań przez administrację publiczną oraz w kontaktach z Komisją Europejską (Czarnowski i in., 2018: 7).

W świecie DS na krajowe strategie rozwoju AI i podobne deklaracje zwracają uwagę raczej tylko osoby silnie osadzone w badanym świecie, nierzadko (czasem jednocześnie) pełniące funkcje kierowników zespołów DS lub właścicieli firm oraz piastujące stanowiska profesorskie na uczelniach państwowych. Nie jestem w stanie wypowiedzieć się o trendach co do oceny działania ministerstw RP w omawianym zakresie – raz tylko miałem okazję porozmawiać z osobą wyrażającą krytyczne opinie. Zdaniem rozmówcy panuje chaos i wzajemne zwalczanie się frakcji w partii rządzącej, przy jednoczesnej orientacji na samorozwiązanie się problemów gospodarczych dzięki niewidzialnej ręce rynku. Rozmówca upatruje pozytywnej roli państwa głównie w finansowaniu kształcenia ludzi w zakresie ML na uczelniach wyższych:

> R: […] budzi się świadomość w firmach, ona jeszcze cały czas chyba jest trochę sterowana takim hypem, ale generalnie jak tylko na jakiejś konferencji się pojawi hasło AI, to ludzie natychmiast biegną. Na pewno co przeszkadza, jakieś, jakieś takie ruchy ze strony rządowej i to jest bardziej szamotanina, żeby budować jakąś strategię cyfryzacji.
> *B: Właśnie, bo ministerstwo coś tam opublikowało, prawda?*
> R: Nie wiem, czy pan będzie chciał to wykorzystać, czy nie, ale prawda jest taka, że niestety są dwa ministerstwa, właściwie trzy, które próbują grać w tą grę, to jest Ministerstwo Cyfryzacji, Ministerstwo Gospodarki i Ministerstwo Nauki i Szkolnictwa Wyższego, i one są obsadzone przez szczerze nienawidzące się frakcje PiS-owskie. Oni, oni wewnątrz nienawidzą się bardziej niż wszystkich pozostałych dookoła, więc oni nawzajem sobie torpedują wszystkie pomysły, jakie mają. […] Pomijając ogólnie **burdel**, jaki ta ekipa organizuje, czy to w ramach tej nieszczęsnej reformy, czy w ogóle w swoich działaniach, to nie widać tutaj **żadnej** spójności, żadnej myśli przewodniej. To jest takie na zasadzie, wrzućmy trochę pieniędzy, może coś z tego powstanie. A niestety nie da się tego rozwiązać tylko siłami przedsiębiorców, bo niestety oni potrzebują tego początkowego motorka w postaci ludzi, którzy będą to robili. {wywiad 23}

Prawie nieobecni w dyskursie na arenie modelowania są użytkownicy końcowi i środowisko naturalne. Obaj wymienieni aktorzy są traktowani przedmiotowo, raczej jako zasób niż podmiot mający interesy i wpływy na arenie. Choć w dyskursie świata DS i wśród jego biznesowych rzeczników mówi się o danych jako zasobach, to dane nie biorą się znikąd. Często są zapisem działań ludzkich, działań osób płacących swoimi danymi w specyficznym barterze w zamian za usługi, np. Facebookowi czy Google'owi.

Słabo obecnymi w dyskursie aktorami, sprowadzonymi do roli zasobów, są także nisko opłacani „klikacze", np. zatrudniający się przez Mechanical Turk. Ich niewykwalifikowana praca polega na zbieraniu lub wzbogacaniu – tzw. etykietowaniu – danych, które mają zasilać modele nadzorowanego ML. Klikaczy zatrudniono m.in. do pozyskania danych w tzw. aferze Cambridge Analytica[18]. W świecie DS powstają rozwiązania techniczne i metodologiczne pomagające zmniejszyć ilość ręcznego etykietowania (Ratner i in., 2017), niemniej nie można tego uznać za wzmacnianie pozycji „klikaczy" na arenie, a raczej za wysiłek nastawiony na zwiększanie efektywności świata DS. Wskazywano także na postkolonialny charakter obszaru tzw. AI – w moim ujęciu właściwie tożsamej z areną modelowania – ponieważ niewidoczna praca klikaczy wykonywana jest często przez mieszkańców byłych kolonii anglojęzycznych, w warunkach ograniczonego prawa pracy, również przez osoby odbywające karę pozbawienia wolności lub osoby ubogie. Z jednej strony ma to wspierać rozwój ekonomiczny oraz elastyczność zatrudnienia w krajach postkolonialnych, z drugiej utrwala model dystrybucji wiedzy i pracy – pracy niskopłatnej, niewykwalifikowanej i przy ograniczonych prawach pracownika (Mohamed, Png, Isaac, 2020: 10).

Słabo obecni w dyskursie są także bardziej wykwalifikowani pracownicy zatrudniani obok, zamiast lub jako uzupełnianie tzw. AI. Są to np. moderatorzy treści na YouTubie lub Facebooku, którzy zapewniają ręczne filtrowanie materiałów przesyłanych do serwisów (Chemaly, Buni, 2016; Newton, 2019). To czasami „ludzie udający roboty, które udają ludzi" – firma sprzedająca jakoby bazującego na AI cyfrowego asystenta zatrudniała ludzi wykonujących skrycie zlecone zadania (Huet, 2016).

Wielu słabo obecnych w dyskursie aktorów składa się na pełen łańcuch dostaw systemów bazujących na modelach ML – od wykorzystujących duże ilości zasobów środowiska naturalnego chmur obliczeniowych do nisko płatnej pracy górników, w skrajnych warunkach wydobywających metale na komponenty infrastruktury komputerowej (Portmess, Tower, 2015; Joler, Crawford, 2018).

Czasami użytkownik końcowy jest równoznaczny z klientem (użytkownik jako ktoś, kto korzysta z systemu bazującego na ML, np. handlowiec, który przed spotkaniem uruchamia aplikację, gdzie widzi historię kontaktu z kontrahentem oraz maszynowo wygenerowaną klasyfikację tego kontrahenta i trzy potencjalne produkty do zaproponowania mu). W takim przypadku faktycznie jest klientem, czyli płaci firma zatrudniająca handlowców. W świecie DS za dobrą praktykę uznano by konsultacje z handlowcami już na etapie tworzenia systemu i zbieranie od nich informacji zwrotnej także po wdrożeniu systemu. Użytkownicy są tu zatem obecni i mają sprawczość. Z drugiej strony czasami za użytkownika należy uznać osobę na pozycji kontrahenta z powyższego przykładu, tzn. kogoś, kto nie tyle korzysta, a „z niego"

18 Czasem trudno odróżnić „klikacza" od użytkownika serwisu zbierającego dane, co zwane jest zamianą klienta w produkt. Pracę „klikacza" użytkownik może wykonywać nieświadomie, rozwiązując CAPTCHA – por. Kobielus, 2017.

korzysta i „na nim" działa system oparty na ML – z historycznych danych kontrahentów modele dają odpowiedzi dla konkretnego podmiotu. Występuje to w dużej skali u technologicznych gigantów, takich jak np. Facebook, Google, LinkedIn (należący do Microsoftu), Spotify, Netflix, gdzie modele odpowiadają za tzw. personalizowane treści rekomendowane konkretnym użytkownikom. To personalizowanie to owoc liberalnego kultu wolnej, racjonalnej jednostki – trzeba zatem komunikować produkty, usługi. Ogólnie rzeczy ujmując, marketing zwraca się do jednostki, bo: wyborca wie lepiej, klient ma zawsze rację, ludzie sami o sobie decydują (por. Harari, 2018: 281). Takie założenia o jednostce nie oznaczają jednak, aby głos tego rodzaju użytkowników końcowych był znaczący w opisywanej arenie.

Niemniej być może w konsekwencji obecnych w głównym nurcie dyskursu publicznego głosów podnoszących kwestie np. prywatności użytkowników końcowych w dużej mierze od 2018 roku – po aferze Cambridge Analytica oraz w związku z wprowadzeniem GDPR w Unii Europejskiej – użytkownicy końcowi, poprzez rzeczników takich jak organizacje pozarządowe, media, polityka, a w konsekwencji prawo, będą zdobywać nieco większą niż obecnie sprawczość na dalece nierównej w tym zakresie arenie modelowania. Ideologia danetyzacji przestaje być przyjmowana bezkrytycznie przez wszystkie zainteresowane strony (por. van Dijck, 2014: 200–201). Wołanie użytkowników o prywatność dostrzegły już firmy i włączyły zapewnienia dotyczące dbałości o ich prywatność do swoich komunikatów marketingowych. Od 2019 roku czyni tak nie tylko Apple (od dawna znane z podobnych zapewnień co do oferowanych przez firmę urządzeń), nie tylko firmy, dla których prywatność jest podstawą oferowanych usług (np. DuckDuckGo, Protonmail, Mozilla, Brave) – robią to także Facebook i Google (Lerman, O'Brien, 2019; Robertson, 2019). Problematyka pojawiła się już w mainstreamie popkultury – platforma Netflix emitowała serial w reżyserii Jeffa Orlowskiego, w którym pojawiają się spostrzeżenia w rodzaju „jeśli nie płacisz za produkt, to ty jesteś produktem" (Obem, 2020).

Użytkownicy końcowi mogą domagać się czegoś dalece innego niż swojej prywatności, a prywatność nie jest jedynym obszarem, w którym mamy do czynienia z konsekwencjami społecznymi wdrożeń modeli ML (prywatność związana jest raczej z pozyskiwaniem i użyciem danych, a nie z modelowaniem) czy w ogóle działalności świata DS. Poza tym użytkownicy nie są jednorodną, nietechniczną grupą. Użytkownikami końcowymi systemów bazujących na modelach ML nie są jakieś szare masy, o których – mam nadzieję, że nie – wypowiadam się tu z wyższością i troską, ale to tak samo socjologowie, politycy, data scientiści, prawnicy i przedstawiciele światów społecznych, organizacji i innych społecznych całości.

W moim ujęciu, być może niesłusznie, nieobecna jest także popkultura. Chyba wolno byłoby traktować ją **jako tło dla areny modelowania**. Właśnie z popkultury, z science fiction, jak np. współczesnego serialu brytyjskiego pt. *Czarne lustro* czy starszych filmów, np. *Terminator*, *Raport mniejszości*, wyobrażenia o tzw. AI nierzadko czerpie nie tylko użytkownik nietechniczny, ale i uczestnicy każdego z wyróżnionych na arenie modelowania światów, łącznie z DS i IT.

Pozycją nieobecną w dyskursie opisywanej areny modelowania jest to, że **co do zasady nie należy lub nie warto w ogóle używać modeli ML**. Taka pozycja nie reprezentowałaby interesów żadnego ze światów/subświatów zaangażowanych w arenę.

Dalej przyglądam się przecięciu światów DS i biznesu, szczególnie w kontekście działań podejmowanych pomiędzy data scientistami i ich rzecznikami, przedstawicielami firm-usługodawców (tzw. klientami wewnętrznymi) i firm-klientów oraz osobami chcącymi wejść do świata.

5.2.1.1. Sztuczna inteligencja to slajd w PowerPoincie

Z punktu widzenia uczestników świata DS do dyskursu o modelowaniu, właściwego dla ich społecznego świata, należy pojęcie uczenia maszynowego (ML) – obok innych terminów technicznych dotyczących danych, kodu i różnego rodzaju narzędzi oraz metod. Podobnie postrzegają oni dyskurs światów IT oraz nauk przyrodniczych i ścisłych w akademii. Tam także używane jest pojęcie ML. Uczestnicy świata DS zauważają posługiwanie się zarówno pojęciem „ML", jak i „sztuczna inteligencja (AI)" w świecie prawa. Inne światy nietechniczne, czyli nauki społeczne i humanistyczne w akademii, biznes, media i polityka raczej postrzegane są jako posługujące się pojęciem „AI". **Uważam pojęcie „AI" za tzw. opakowanie** (Kacperczyk, 2016: 45) **dla systemów bazujących na modelach ML oraz w ogóle dla działalności świata DS.**

Uczestnicy świata DS raczej nie posługują się pojęciem „AI" w swoim gronie. Moim zdaniem, gdy mówią o AI, wskazuje to na żart lub ironię. Pojęcie to stosują w komunikacji z innymi światami nietechnicznymi zaangażowanymi w arenę. Komercyjnie pojęcie AI jest stosowane głównie przez biznesowych rzeczników świata DS. To osoby na stanowiskach sprzedażowych i marketingowych, które formułują komunikaty do klientów projektów DS. W tej roli występują także osoby będące uczestnikami świata DS, np. kierujące zespołami/działami DS lub freelancerzy:

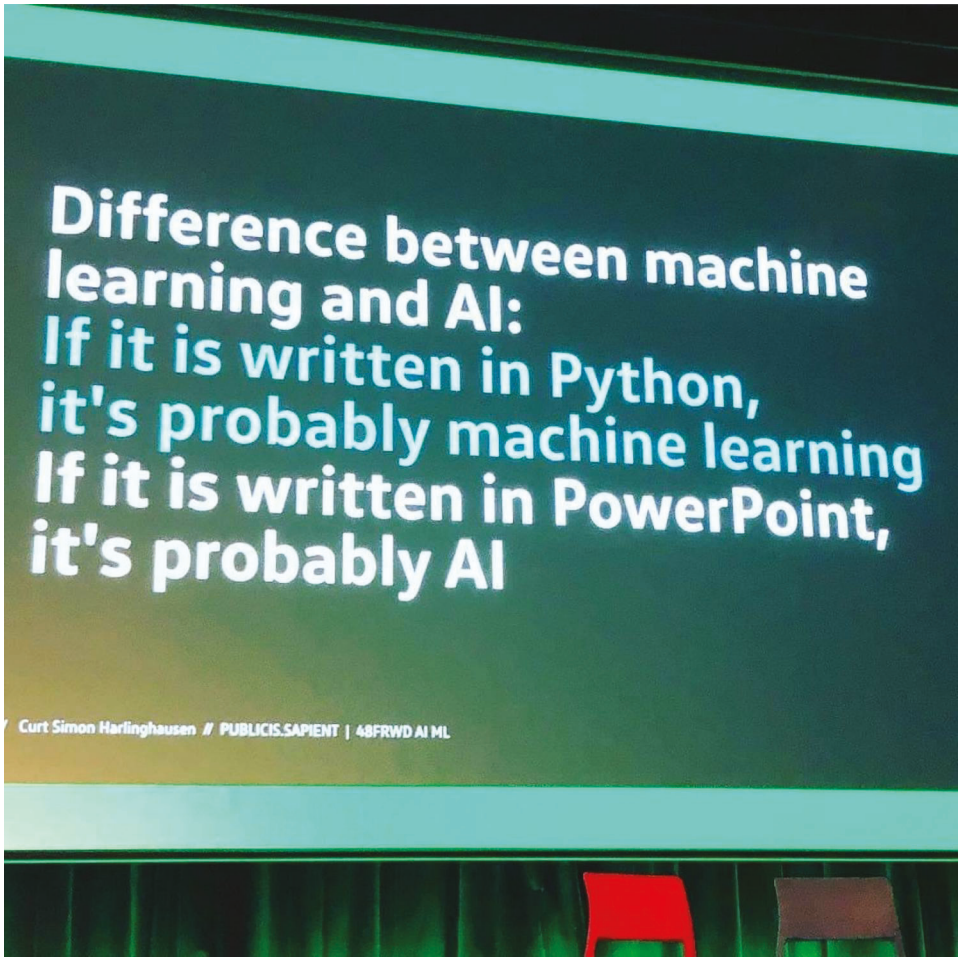
> Jako konsultant z łatwością zdobywałem klientów za pomocą kartki papieru i prezentacji PowerPoint, ponieważ nie odróżniali oni sztucznej inteligencji (AI) od analityki biznesowej (BI) lub analityki biznesowej (BI) od licencjackiego stopnia naukowego BS [*Bachelor of Science*] (Foreman, 2017: 16).

Rzecznikami mogą być także osoby zajmujące się zasobami ludzkimi (HR), formułujące komunikaty do ludzi zainteresowanych pracą w DS – do początkujących uczestników świata DS i wannabesów. Tak więc AI jako opakowanie dla działalności świata DS (a właściwie najbardziej atrakcyjnego, ciekawego, sprawczego, seksownego itp. jej wycinka, czyli modelowania ML) jest opakowaniem przygotowanym zarówno dla klientów – szeroko rozumianych jako osoby, które płacą za projekty i pracę data scientistów – jak i dla osób zainteresowanych pracą w DS.

Jaskrawym przykładem tego, że uczestnicy świata DS uznają AI za nietechniczne opakowanie dla swojej technicznej działalności, polegającej przede

wszystkim na pisaniu kodu do przetwarzania, analizy i modelowania danych cyfrowych jest żartobliwa sentencja, która zdobyła środowiskową popularność na początku 2019 roku – „Różnica pomiędzy ML a AI: Jeżeli to jest napisane w Pythonie, to prawdopodobnie ML. Jeżeli jest napisanie w PowerPoincie, prawdopodobnie to AI" (rys. 5.5).

Tę „sentencję", podczas wystąpienia na konferencji nietechnicznej, wygłosił Curt Simon Harlinghausen, który nie jest uczestnikiem świata DS, a przedsiębiorcą w branży internetowej (Fürg, 2017). Miało to być chyba komunikatem nastawionym na edukowanie potencjalnych klientów zamawiających usługi bazujące na modelach ML. Niemniej „sentencja" przyjęła się w badanym świecie DS.



Rysunek 5.5. AI jako opakowanie dla działalności świata DS – żartobliwa sentencja

Źródło: https://www.linkedin.com/feed/update/urn:li:activity:6501030890754314240/ (dostęp: 3.09.2020).

B: A ty, czym konkretnie się zajmujesz? W twojej pracy?
R: W mojej pracy jestem w zespole, który szumnie nazywa się AI Core Team[19] [śmiech]
B: AI Core Team? Tak?
R: Tak, tak dokładnie [ze śmiechem].
B: Ok, a dlaczego, a dlaczego szumnie przepraszam?
R: Nie, to kolega dosłownie w zeszłym tygodniu coś takiego powiedział, że, yyy, generalnie jak piszesz kod, to jest machine learning, ale na prezentacji to się nazywa AI. Nie no, śmieję się z tego AI, że to wszędzie teraz się pojawia takie supermodne hasło w każdym zespole, w każdej firmie, na każdej prezentacji, a de facto taka prawdziwa sztuczna inteligencja to jeszcze nie została stworzona. To jakby. {wywiad 18}

B: A słyszałem, yyy, coś takiego, w formie żartu, żeee jeżeli to jest machine learning, no to jest pewnie napisane w Pythonie.
R: [śmiech]
B: Wiesz, o czym mówię?
R: Tak, a AI to jest w PowerPoincie [śmiech]
B: No właśnie. I możesz mi wytłumaczyć ten żart?
R: Yyy, moim zdaniem to jest dlatego właśnie, że AI, lubią tego pojęcia używać osoby od marketingu, bo ono brzmi egzotycznie, ono, ludzie mają jakieś wyobrażenia o sztucznej inteligencji wykreowane przez filmy i media, więc dlatego PowerPoint to się kojarzy z takim dynamicznym managerem, który próbuje ci coś sprzedać, a machine learning brzmi już trochę nudniej i to są właśnie te osoby, które siedzą i klepią kod w Pythonie. Ja tak to widzę. {wywiad 20}

B: No dobra, a czy śmieszy Cię żart – sztuczna inteligencja jest napisana w Power Poincie, a ML w Pythonie?
R: Nie, bo ja robię sztuczną inteligencję w Power Poincie. Nie no, żartuję [z uśmiechem]. Zawsze jak robię slajdy i prezentację, to mówię no, zajmuję się AI [z uśmiechem]. [pauza] Trochę tak jeeeeest, bo my lubimy trochę rozdmuchać {wywiad 22}

R: […] bardzo często mylone są te dwa pojęcia sztuczna inteligencja i uczenie maszynowe, to ponieważ to jest mimo wszystko politechnika, tutaj mamy dosyć przyziemne na to spojrzenie. Jest taki złośliwy żart, który krąży po korytarzach, że jeśli sztuczna inteligencja, to prawdopodobnie to jest w Power Poincie, jeżeli uczenie maszynowe, to w Pythonie, i to trochę oddaje rzeczywistość. Ta sztuczna inteligencja to jest często też tak na etapie, bo byśmy chcieli, bo jakieś wizje, natomiast w praktyce, to co na co dzień działa, co da się zakodować w [niezrozumiałe], to jest uczenie maszynowe. […] {wywiad 23}

W badanym świecie AI jest traktowane jako termin nietechniczny, stosowany głównie przez biznesowych rzeczników świata DS w celu sprzedawania usług czy systemów, w których czasem używane jest ML. Terminem „AI" opakowywane są technologie niezwiązane z ML, bazujące na zaprogramowanych regułach, ale np. rozpoznające język naturalny, jak chatboty. W sensie technicznym nie jest to błąd, ML jest bowiem uznawany za subdyscyplinę AI (Russel, Norvig, 2009: 2; Goodfellow i in., 2016: 9; Raschka, 2018: 26). Niemniej z punktu widzenia uczestników świata DS termin „AI" jest tak wieloznaczny i budzący emocje, że w sensie

19 Nazwa zmieniona z zachowaniem słownictwa zbliżonego do oryginału, w celu zachowania poufności wywiadu.

technicznym nie da się zaakceptować jego użycia. Właściwie z punktu widzenia DS to, co nietechnicznie nazywane jest AI, po prostu nie istnieje. Sztuczna inteligencja to tylko kolorowe opakowanie dla różnych systemów, które mogą, ale nie muszą działać na bazie ML. Terminem „AI" określane są przecież np. chatboty działające na bazie zaprogramowanych reguł.

Nie ma w badanym świecie zgody co do tego, czy używanie rozmytego, obrośniętego popkulturowymi skojarzeniami terminu „AI" w komunikacji ze światami nietechnicznymi leży w interesie świata DS. Praktyka ta jest dość powszechna. Wśród 2800 europejskich start-upów twierdzących, że zajmują się AI, w około 40% nic na to nie wskazuje – „nie znaleziono żadnego potwierdzenia, by AI stanowiło ważną część oferowanych produktów" (Schulze, 2019). Opisana praktyka ma być stosowana w celu podniesienia atrakcyjności start-upów w oczach inwestorów (Schulze, 2019).

Spotykałem się z opiniami, iż bez AI jako opakowania dla DS nie da się funkcjonować na rynku. Zdaniem jednej z rozmówczyń {wywiad 25} osoby zajmujące się DS muszą deklarować, iż zajmują się AI, termin „DS" nie wzbudza bowiem zainteresowania klientów. Za „fasadą" kryją się zarówno firmy, których działania rozmówczyni oceniała pozytywnie, jak i takie, które poza opakowaniem nie mają wiele do zaoferowania. Te fasadowe firmy jej zdaniem nie osiągają dobrych wyników finansowych.

Uczestnicy świata DS raczej negatywnie oceniają używanie terminu „AI" przez swoich rzeczników (pracowników HR/rekruterów) w stosunku do osób zainteresowanych pracą w DS. Data scientiści nie chcą w zespołach osób przyciągniętych do pracy obietnicami o AI. Dość powszechne jest przekonanie, iż lepiej unikać firm rekrutujących pod hasłem „AI" z małą liczbą szczegółów technicznych w ofertach pracy. Jednak termin „AI" jest stosowany w nazwach grup meetupowych zajmujących się DS, a przyciąganie na spotkania osób zainteresowanych pracą w DS jest jednym z celów firm wspierających organizacje meetupów. W przypadku komunikacji z klientami stosowanie terminu „AI" jest w świecie DS oceniane ambiwalentnie. Trudno wskazać różne pozycje w dyskursie na ten temat. Jest raczej tak, że większość uczestników zajmujących się komercyjnym DS ocenia te praktyki jednocześnie jako leżące i nieleżące w interesie swojego świata społecznego, a nawet węziej – w interesie swoim i swoich firm, zespołów czy działów.

Wygórowane, bazujące na niejasnych wizjach i niewiedzy technicznej oczekiwania klientów być może przyciągają ich do rozpoczęcia współpracy z firmami oferującymi usługi związane z modelowaniem danych, co raczej leży w interesie badanego świata DS, bo traktowane jest jako popyt. Niemniej już na wczesnych etapach wiele projektów kończy się, a klienci mogą być rozczarowani brakiem rozwiązania problemu biznesowego, brakiem wdrożenia. Nierzadko mają trudności z akceptacją eksperymentalnego charakteru projektów DS i tego, że nawet najbardziej dopracowane systemy bazujące na ML nie mają stuprocentowej skuteczności. Problem biznesowy może być rozwiązany bez użycia ML, a za pomocą rozwiązań

z zakresu inżynierii oprogramowania (zapisywania reguł, tzw. if-else), jednak wówczas nie jest to w badanym świecie uznawane za projekt DS – nawet kiedy produkt ma elementy przetwarzania języka naturalnego, jak np. chatbot czy asystent głosowy (choć wówczas często będzie nazywany „AI"). Jedna z rozmówczyń {wywiad 22} uważała, że klienci stają się coraz bardziej wyedukowani – zaczynają postrzegać DS coraz rzadziej jako magię, a coraz częściej jako majsterkowanie.

Zdaniem uczestników świata DS będzie on się wciąż dynamicznie rozwijać, mieć coraz większy wpływ na rzeczywistość, choć opadnie entuzjazm wobec najmodniejszych terminów, jak np. „AI". Niemniej dominujące w badanym świecie jest przekonanie, że „prawdziwa" – równa czy wyższa ludzkiej – AI jest niezmiernie daleko, o ile w ogóle kiedykolwiek będzie możliwa. Kolejna rozmówczyni {wywiad 18} wskazała, że jak najbardziej DS rozwija się dynamicznie i należy być na bieżąco, by utrzymać się w branży. Jednak projekty DS są wykonalne raczej w dużych, bogatych i zaawansowanych technologicznie firmach o wysokiej „kulturze danych", głównie w największych miastach. Natomiast składające się na opakowanie dla ML (i innych technologii) obietnice związane z „AI" pozostaną w sferze deklaracji – w PowerPoincie – co najmniej do momentu powstania komputerów kwantowych:

> R: Bo tak jak ja obserwuję teraz trendy, przynajmniej w firmie [nazwa kraju z Europy Zachodniej], dużej, która ma na to budżet, to [pauza], no stety albo niestety, małe firmy się tym nie zajmują, bo nie mają na to pieniędzy. Więc wiedziałam, że gdybym chciała zmieniać pracę na przykład, no to opcja będzie tylko w Warszawie, Krakowie, Wrocławiu, ewentualnie Poznaniu. Nie ma co raczej marzyć o mniejszych miastach, no i tylko że to się w dużych firmach na razie rozwija i pewnie tak, tak to zostanie. [pauza] No i że raczej to będzie wymuszone, że będzie to istniało tylko w firmach, które, w których jest wysoka jakość, że tak powiem wysoka kultura danych. […] Eee, no więc te tematy będą się tylko kręciły pewnie w dużych firmach, które mają te dane jakoś sensownie ustrukturyzowane. Eee, a co do samego rozwoju [śmiech] AI, to myślę, że przez najbliższe lata jeszcze się nie musimy niczym martwić, raczej AI nie działa.
>
> *B: Czyli pozostanie w PowerPoincie, tak?*
>
> R: Tak, pozostanie w PowerPoincie i raczej nasza pozycja tu nie jest zagrożona. Możemy zacząć się martwić, jak powstaną komputery kwantowe, to trochę się sytuacja może zmienić [śmiech]. Yy, no ale właśnie fajne jest to, że ten, [pauza] bardzo dynamicznie się to rozwija, a że się zmieniają w bardzo szybkim tempie i narzędzia i możliwości [pauza]. Choćby teraz jakiś czas temu właśnie też na potrzeby projektu robiłam analizę bibliotek machine learningowych. No to też widać, że biblioteki, które powstały parę lat temu, już są u schyłku, a niektóre się bardzo intensywnie rozwijają. I to wszędzie, w całej tej branży są takie tendencje, że narzędzia niektóre będą się szybko dynamicznie rozwijały, a takie, które powstały i nie są rozwijane, to będą zanikać, to nie jest tak jak z Excelem, że będzie wieczny [śmiech]. Yyy, no ale myślę, że cały czas branża, działka, ta działka będzie się prężnie rozwijała i mam nadzieję, że uda mi się w niej utrzymać. {wywiad 18}

Ten fragment ilustruje oczywisty wniosek, że **na arenie modelowania największą władzę nad przetrwaniem i rozwojem zaangażowanych w nią światów mają aktorzy o najwyższym kapitale ekonomicznym i technologicznym**.

5.2.1.2. Magia i majsterkowanie

Klienci, wannabesi – osoby chcące wejść do świata DS, a nawet początkujący uczestnicy mogą postrzegać ML jako coś magicznego. Model ML staje się fetyszem, który sam w sobie ma rozwiązywać realne problemy biznesowe (z perspektywy klientów) lub stanowić o byciu prawdziwym data scientistą (z perspektywy wannabesów/początkujących). Dla uczestników silniej osadzonych w badanym świecie modelowanie z zastosowaniem ML (czy w ogóle projekty DS) to praca majsterkowicza, który sam wyposaża swój warsztat i chałupniczo przerabia narzędzia, używa ich niezgodnie z pierwotnym przeznaczeniem, pracuje metodą prób i błędów. Wykonanie modelu ML, który pomaga w rozwiązaniu realnego problemu, wymaga majsterkowania – czasami żmudnego i męczącego, czasami pomysłowego i nieszablonowego, zawsze zaś trochę eksperymentalnego, gdzie rezultaty są niepewne.

Modele ML trenuje się na zbiorach danych mających obciążenia, tzw. biasy. Przypadkowe współwystępowanie cech w zbiorze danych treningowych prowadzi do błędów w działaniu gotowego modelu. Takie błędy trudno zauważyć osobom przygotowującym modele. Słynny w świecie DS eksperyment (Ribeiro, Singh, Guestrin, 2016: 1142–1143) dotyczył właśnie tego. Przygotowano model nadzorowanego ML tak, aby dawał obciążoną odpowiedź, a następnie poproszono ludzi o ocenę jego działania. Model miał klasyfikować zdjęcia wilków i psów husky. W danych treningowych wilki zawsze prezentowano na śniegu, a husky zawsze bez. Walidacja wykazała, że zdjęcia zawierające śnieg lub inne, jasne tło przy dolnej krawędzi klasyfikowane były przez gotowy model jako „wilk", natomiast niezawierające takich cech jako „husky" – bez względu na kolor zwierzęcia czy pozę w kadrze. Dwudziestu siedmiu studentom studiów II stopnia, mającym ukończony co najmniej jeden przedmiot z zakresu ML, zadano następujące pytania:

1) „Czy ufają w prawidłowe działanie tego modelu w realnym świecie?";
2) „Dlaczego?";
3) „W jaki sposób model rozróżnia zdjęcia wilków i psów husky?".

a) Husky zaklasyfikowany jako wilk b) Wyjaśnienie

Rysunek 5.6. Obciążony model ML „wilk czy husky?" – błędna odpowiedź modelu i wyjaśnienie

Źródło: Ribeiro, Singh, Guestrin, 2016: 1143.

Badanym pokazano dziesięć odpowiedzi modelu, z których osiem było poprawnych, jedna pokazywała husky na śniegu zaklasyfikowanego jako „wilk", jedna zaś wilka bez śniegu zaklasyfikowanego jako „husky". Wypowiedzi 10 z 27 studentów oceniono jako wskazujące na zaufanie do modelu, a 12 z 27 badanych miało sądzić, iż śnieg na zdjęciu jest jedną z cech, których „wyuczył się" model. Następnie studentom pokazano graficzne wyjaśnienie odpowiedzi modelu (rys. 5.6).

Po wyjaśnieniu graficznym pytania powtórzono. Na zaufanie do działania modelu wskazały tylko trzy badane osoby, a śnieg jako cechę wyuczoną przez model wskazało aż 25 z 27 badanych. Metody wyjaśniania klasyfikacji konkretnych przypadków przez modele ML są przydatne dla uzyskania wglądu w modele i do oceny, czy ufać ich działaniu (Ribeiro, Singh, Guestrin, 2016: 1143).

W sensie technicznym nie ma wątpliwości, że trudne jest uzyskanie modelu ML dającego wysoki odsetek poprawnych odpowiedzi i mającego wysoką zdolność tzw. generalizacji, czyli „niezbiasowanego" ani przeuczonego, „nauczonego na pamięć" {wywiad 8}. W sensie realizacji celu całego projektu DS będzie to bez porównania trudniejsze. Realizacja projektu, którego małą częścią jest praca nad przygotowaniem modelu – a przy uwzględnieniu wielu czynników organizacyjnych, biznesowych, kulturowych, formalnych czy infrastruktury technicznej – produkcyjne wdrożenie i utrzymanie wypracowanego modelu ML tak, by ostatecznie przynosił klientowi wartość dodaną, jest bardzo, bardzo trudna. Niemniej spotykałem wśród uczestników badanego świata opinie, iż **zarówno klienci, jak i wannabesi oraz początkujący uczestnicy świata DS postrzegają modele ML jako coś, co w magiczny sposób rozwiązuje realne problemy**. Magiczny, czyli niedający się racjonalnie wyjaśnić. Magiczny, czyli sam w sobie cudownie działający, na zasadzie „weź problem, dodaj ML, problem rozwiązany". To alternatywny system ciągów przyczynowo-skutkowych, w którym imponujący efekt osiągany jest nie wiadomo jak.

Jest to fetyszyzacja modeli ML, a wśród klientów nawet fetyszyzacja ich opakowania, czyli AI (Thomas, Nafus, Sherman, 2018):

1) fetysz jest obiektem materialnym przesiąkniętym możliwościami (*capabilities*), które nie są z natury rzeczy właściwościami lub funkcjami samego przedmiotu;
2) te nadmiarowe możliwości są generowane w punkcie styku pomiędzy osobami o różnych pozycjach i tym samym rozszerzają one zakres ich wkładu (*outcomes*) społecznego, kulturowego i ekonomicznego;
3) ten wkład społeczny, kulturowy i ekonomiczny jest błędnie rozpoznany jako obietnica fetyszyzowanego obiektu lub zastąpiony tym obiektem;
4) to zastąpienie lub błędne rozpoznanie jest samo w sobie skuteczne: fetyszyzowany obiekt umożliwia coś, co bez niego by nie zaszło.

Fetyszyzowanie modeli ML, a szczególnie konkretnych metod, jak DL, jest w świecie DS uważane za cechę wannabesów lub początkujących. W przypadku uczestników silniej osadzonych w świecie DS dążenie do ciągłego stosowania DL

we wszystkich projektach przy pomijaniu innych metod ML oraz metod statystycznych i analitycznych jest oceniane negatywnie – jako brak profesjonalizmu.

Koncentracja wannabesów i początkujących uczestników świata DS na deep learningu czy ML jest przeciwskuteczną strategią zaświadczania o autentyczności. Postrzeganie tych osób w świecie DS wydaje mi się takie, że uznają one deep learning (lub ML) za magiczny artefakt czyniący zwykłego człowieka magikiem – data scientistą. Przy tym problemy czyszczenia i przygotowania danych, czasami analizy danych, i związane z tym narzędzia, takie jak np. SQL, uważają nie tylko za nudne, ale i nieważne w pracy data scientistów. Wśród silnie osadzonych uczestników rzeczywiście te problemy bywają uznawane za nudne, ale ważne dla realizacji projektu DS – jest to żmudna, brudna, ale niezbędna robota, w przeciwieństwie do fajnego, seksownego i modnego modelowania. Poza tym samo modelowanie nie ma w sobie niczego magicznego, a wymaga majsterkowania. Jest jeszcze „gorzej" dla uznania ich autentyczności w badanym świecie, gdy wannabesi lub początkujący uważają, że ML w magiczny sposób rozwiązuje rzeczywiste problemy. Jest to występek przeciwko wartościom badanego świata. Magiczne postrzeganie dowolnej metody czy narzędzia używanego w świecie DS jest sprzeczne z niezbędnym elementem świadczącym o byciu autentycznym uczestnikiem – z byciem tzw. osobą techniczną.

Elementem bycia autentycznym uczestnikiem świata DS jest używanie metod ML, ale w autentycznym uczestnictwie chodzi o opanowanie szerokiego zakresu narzędzi i metod oraz rozwiązywanie rzeczywistych problemów. W badanym świecie powtarzane jest powiedzenie „kiedy masz tylko młotek, wszystko zaczynasz traktować jak gwoździe" (Maslow, 1966: 15):

> *B: OK, a w tych projektach data science czy zawsze używacie uczenia maszynowego, ee, we wszystkich projektach? Czy…*
> R: Nie. Nie używamy uczenia maszynowego we wszystkich projektach, bo wiemy, co to jest uczenie maszynowe. […] Stosujemy uczenie maszynowe, kiedy coś nam to daje. Prowadzimy projekty z jasnym nastawieniem biznesowym, czyli możemy używać nawet bardzo, **bardzo** prostych narzędzi. Nie musimy zawsze robić czegoś skomplikowanego, data science to nie jest tylko używanie rzeczy, których nie rozumieją przeciętni ludzie. {wywiad 7}

Deklarowanie, że technologia działa magicznie, ma budzić skojarzenie imponującej i bezproblemowej funkcjonalności. Efekt jest zdumiewający, a środki, za pomocą których został osiągnięty, są nieistotne, a nawet tajemne. Już na początku lat siedemdziesiątych pisarz Arthur C. Clarke, stwierdził, że „każda wystarczająco zaawansowana technologia jest nie do odróżnienia od magii". Działanie „magiczne" to konotacja pozytywna. Oznacza, że technologia jest zaawansowana, wygodna, pozwala korzystającym z niej ludziom nie przejmować się szczegółami technicznymi, a wyłącznie cieszyć się z efektów jej używania. Cechą definiującą magiczne technologie jest także bezkosztowość. Działanie jest pomniejszone o niekorzystne elementy, np. walkę i wysiłek. Odwołanie do magii polega zatem nie tylko na zapewnieniu alternatywnego systemu związków przyczynowo-skutkowych, ale także

na odwróceniu uwagi od metod i zasobów, w sensie technicznym niezbędnych do uzyskania określonego efektu (Elish, Boyd, 2018: 67–68).

Wskazywano mi, że wiedząc o fetyszyzowaniu magicznych, zaawansowanych technologii po stronie klientów, data scientiści mogą celowo używać bardziej skomplikowanych narzędzi czy metod, niż wymaga tego projekt. Zdarza się także praktyka opakowywania relatywnie prostych rozwiązań w deklarowane „zaawansowanie", także w komunikacji zespołu DS z klientem wewnętrznym.

Zaczarowywanie DS poprzez odwołania do magii nie jest jednoznacznie uznawane za sprzyjające interesom badanego świata. Być może część rzeczników świata DS uważa, że marketing i sprzedaż projektów ML opakowywanych w AI i zaczarowywanych obietnicami magicznego działania leży w interesie biznesowym tego świata. Wydaje mi się jednak, że sytuacja zmieniła się w czasie i to, co prowadziło do pozyskiwania kolejnych klientów w roku 2017, było nieskuteczne w roku 2019. Klienci stają się coraz bardziej wyedukowani i coraz bardziej rozczarowani, ponieważ w świecie biznesu funkcjonuje wiele niespełnionych obietnic co do magicznych, inteligentnych systemów. Już od początku 2019 roku w świecie DS wzmacnia się pozycja, w której świadomi, racjonalni klienci są postrzegani jako sprzyjający interesom badanego świata. Klienci antropomorfizują ML, mogą pytać np. „czy tam jest mózg". Data scientistom rekomendowane bywa stosowanie metod XAI i przyjazne, uczciwe wyjaśnianie zawiłości oraz możliwości ML, by zapobiec wygórowanym oczekiwaniom klientów i późniejszemu rozczarowaniu {obserwacja 46}.

Majsterkowanie nie jest określeniem używanym *in vivo* w społecznym świecie DS. To kategoria analityczna wzorowana na ujęciu *bricolage*'u (majsterkowania) i *bricoleur* (majsterkowicza) Claude'a Lévi-Straussa. Konsultowałem zastosowanie tej kategorii do opisu pracy świata DS w rozmowach nieformalnych, zyskując akceptację jego uczestników. Pomocny w konsultacjach był komiks z serii „xkcd" autorstwa Randalla Munroe, pokazany przeze mnie na jednym z meetupów {obserwacja 33}, a znany nie tylko w DS, ale i innych światach technicznych (por. Zaród, 2018). Postaci komiksu rozmawiają o w pełni zautomatyzowanym procesie zbierania i analizy danych, komentując, że to domek z kart, zbudowany z przypadkowych skryptów, który zawali się przy jakimkolwiek dziwnym wsadzie. Moi rozmówcy reagowali na komiks lub jego opis albo rozbawieniem, albo konstatacją „niestety, to prawda", czasami łącząc majsterkowanie z wczesnym stadium rozwoju świata DS.

W socjologii stosowano kategorie majsterkowania/majsterkowicza w odniesieniu do technologii informacyjnych, nauki i inżynierii. Także w DS występuje działalność polegająca na:

1) używaniu gotowych materiałów i dostosowywaniu ich do swoich potrzeb, także niezgodnie z pierwotnym przeznaczeniem (Szpunar, 2012: 79–80);
2) pragmatycznej manipulacji narzędziami i aparaturą, w celu uzyskania działających i reprodukowalnych układów, przyjmującej postać wypróbowywania różnych konfiguracji materiałów i technik, czemu nie musi towarzyszyć refleksja teoretyczna (Afeltowicz, Pietrowicz, 2008: 45–46).

Podobnie jak Zaród potwierdzam (na gruncie DS) tezy Gabrieli Coleman i Johana Söderberga dotyczące tego, że tworzenie oraz modyfikacja narzędzi są niezbędne w stawaniu się i byciu uczestnikiem badanego świata[20], oraz że dla uczestników świata DS – tak jak dla hakerów – ważna jest nie tylko wolność modyfikowania narzędzia, ale i uznanie autorstwa (Zaród, 2018: 340).

Praca świata DS jest dla mnie majsterkowaniem szerzej pojętym: zarówno w sferze narzędzi, w sferze metod, jak i sferze praktycznego celu projektów DS, a nawet w sferze prawnej i etycznej. Majsterkowanie w DS nie jest czymś mitycznym, przeciwstawnym nauce i inżynierii, jak chciał Lévi-Strauss. Jest jakością, odróżniającą DS od działania w nauce akademickiej i w inżynierii. Majsterkowanie odnosi się do nauki i inżynierii (por. Afeltowicz, Pietrowicz, 2008), jednak z moich badań wynika, iż uczestnicy świata DS odróżniają swój świat od tych obszarów także poprzez odwołanie – nie wprost – do bycia światem „bardziej majsterkującym". Znajduje to wyraz w postrzeganiu tych obszarów jako różniących się od DS w sferze wartości: ciekawości, samorozwoju, swobody, samodzielności.

5.2.2. Data scientistka

Autorzy raportu AI Now Institute podkreślają, że specyficzna kultura firm technologicznych i zajmującego się tzw. AI środowiska naukowego nie sprzyja kompleksowemu podejściu do problemów etyki systemów AI. Twierdzą, że systemy AI utrwalają (*perpetuate*) uprzedzenia i dyskryminacje panujące także w samych firmach. Mówią o braku różnorodności, szczególnie płciowej, oraz o powtarzających się praktykach nękania (*harassment*), wykluczania i nierówności płacowych, które są nie tylko szkodliwe dla pracowników tych firm, ale wpływają na oparte na AI produkty, które te firmy tworzą (Crawford i in., 2018: 42).

Wskazuje się na ilościową dysproporcję płci w branży, nie tylko w Polsce. Na trzech najważniejszych konferencjach poświęconych ML w 2017 roku kobiety stanowiły 12% uczestników, a sytuacja nie zmieniła się od lat dziewięćdziesiątych, kiedy w badaniu osób publikujących w poświęconym AI czasopiśmie naukowym „IEEE Expert" kobiety stanowiły 13%. W zespołach zajmujących się tzw. AI w największych firmach sytuacja jest podobna. Kobiety stanowią odpowiednio 15% i 10% pracowników w Facebooku i Google'u. Wśród absolwentów studiów magisterskich z informatyki w USA było 18% kobiet (Crawford i in., 2018: 38). To nie pokazuje całości obrazu, ponieważ uczestnicy świata DS rekrutują się spośród absolwentów wielu kierunków. Podobnie wskazują badania rynku pracy w USA. Tylko 20% respondentów ukończyło studia informatyczne, około 25% data scientistów to matematycy lub statystycy, 20% jest absolwentami nauk przyrodniczych, a 35% innych dyscyplin (Burtch, 2018: 20). Frakcja kobiet ogółem jest jednak

20 Zaobserwowano to w różnych zbiorowościach profesjonalnych, takich jak inżynierowie, hakerzy, badacze i drwale.

zbliżona do podawanych przez AI Now i wynosi 15%, przy czym jest najwyższa dla grupy z najkrótszym doświadczeniem zawodowym i spada wraz z liczbą lat doświadczenia (Burtch, 2018: 28–29). Moja analiza ankiet związanych z polskim światem społecznym DS wskazuje na udział kobiet od 5% do 30% w zależności od źródła danych i podobną zależność udziału kobiet względem ich wieku. Niemniej tezy autorów raportu o odzwierciedlaniu kultury firm technologicznych w ich produktach mogą być słuszne. O tym, że technologia uosabia wartości, badacze z dziedziny STS wypowiadali się już od dawna (Winner, 1980; Friedman, Nissenbaum, 1996; Nissenbaum, 2001), a o technologii jako utrwalonym społeczeństwie Bruno Latour pisał w 1991 roku (Latour, 2013).

Podobne przekonanie występuje w świecie DS. Działania nastawione na włączanie kobiet i innych mniejszości do świata DS uzasadniane są m.in. w kategoriach ulepszania budowanych produktów/realizowanych projektów. Za uzasadnieniem stoi przekonanie, że skład zespołów DS ma wpływ na charakter budowanych przez zespoły rozwiązań, i że te rozwiązania będą bardziej różnorodne (*diverse*) i w konsekwencji „lepsze" wówczas, gdy w zespołach będzie więcej niż obecnie osób reprezentujących mniejszości (w Polsce są to kobiety, nie widzę innych mniejszości w dyskursie).

To, że kobiety stanowią 10–20% osób uczestniczących w badanym świecie, oznacza, że **kiedy w zespole DS pracuje kobieta, to niemal zawsze jest to jedna kobieta wśród kilku mężczyzn**. W Polsce zespoły złożone z większej liczby kobiet, w większości z kobiet lub tylko z kobiet powoływane są raczej przez uczestniczki organizacji i inicjatyw nastawionych na włączanie mniejszości genderowych do DS (jak R-Ladies, PyLadies, WiMLDS), np. w celu uczestniczenia w hackatonie lub jednorazowo na warsztatach. W komercyjnym DS takie zespoły zdarzają się bardzo rzadko. Z zespołem, na który składały się trzy kobiety pracujące jako data scientistki i trzech mężczyzn na stanowiskach biznesowych, spotkałem się raz {obserwacja 26}. Ten zespół z firmy McKinsey, jednej ze sponsorujących konferencję, w trakcie swojej prelekcji werbalnie podkreślał wyjątkowość swego składu i zalety pracy w takim różnorodnym zespole DS. Zazwyczaj jednak kobieta jest jedyną reprezentantką swej płci:

> *B: Mhm, okey, a jak ci się ogólnie pracuje w tej branży jako kobiecie?*
> R: Bardzo dobrze. [pauza] Dobrze pod tym względem, że znaczy, myślę, że rzeczywiście mi się zawsze dobrze z mężczyznami pracowało. Nie miałam nigdy jakichś takich problemów i nie czułam się nigdy specjalnie dyskryminowana. I wręcz widzę, że raczej w zespołach ludzie się cieszą, że, że tu mają jakąkolwiek kobietę, że to jest tak, że jak jest 5 czy 8 mężczyzn w zespole i rzeczywiście zawsze byłam jedyną kobietą. Niezależnie, czy to w tej pierwszej firmie, czy tej konsultingowej małej, czy teraz w tym trzecim zespole, zawsze byłam jedyną. I [śmiech] to było no dość mocno, że tak powiem, niekonwencjonalne, ale jak robiłam etat i przechodziłam rozmowę rekrutacyjną do teraz tej aktualnej pracy, no to rekruterzy, czyli z działu HR gość, powiedział mi, że [pauza], no, mam spore szanse i też ze względu na płeć mam spore szanse [śmiech], że jednak chciałby, żeby chociaż jedna kobieta w zespole była. Oczywiście **nigdy** nie chciałam, żeby moje, powiedzmy, mój udział w zespole był oceniany

> ze względu na płeć, tylko na umiejętności, ale mimo wszystko zawsze się dobrze w takim środowisku czułam. {wywiad 18}

> *B: OK. Ale to teraz w swoim zespole jest pani jedyną kobietą?*
> R: Tak i nic nie wskazuje na to, żeby było nas więcej. {wywiad 25}

Istnieją, także w Polsce, organizacje i inicjatywy nastawione na włączanie do DS mniejszości genderowych: kobiet tzw. cisseksualnych, czyli identyfikujących się z żeńską płcią przypisaną, kobiet i mężczyzn transseksualnych, osób niebinarnych, osób agenderowych (DataCamp, 2018; WiMLDS, 2018; PyLadies, 2019; R-Ladies, 2019). Aktywistka i data scientistka Shaikh Reshama wskazuje, że bardziej różnorodne i aktywne na rzecz różnorodności jest środowisko osób posługujących się językiem R niż językiem Python (Reshama, 2018). Konferencje użytkowników Pythona w DS/ML i użytkowników R (PyData, 2018; Why R? Foundation, 2018) mają jednak niemal identyczne kodeksy postępowania:

> PyData ma na celu zapewnienie udziału w konferencji wolnej od prześladowań każdemu bez względu na płeć, orientację seksualną, tożsamość płciową i ekspresję, niepełnosprawność, wygląd fizyczny, rozmiar ciała, rasę czy religię. Nie tolerujemy nękania uczestników konferencji w żadnej formie. […] Uczestnicy naruszający te zasady mogą zostać poproszeni o opuszczenie konferencji bez zwrotu kosztów według własnego uznania organizatorów konferencji (PyData, 2018).

Niemniej najważniejsza, odbywająca się od 1987 roku konferencja „Neural Information Processing Systems", znana jest jako NIPS. Słowo „nips" jest m.in. slangowym określeniem sutków (*nipples*) oraz obraźliwym określeniem osób pochodzenia japońskiego (od Nippon – Japonia w języku japońskim) (Hu, 2018). Konferencja jest popularna: około 2,5 tys. biletów na edycję 2018 wyprzedano w niecałe 12 minut (Koch, 2018). W 2018 roku jeszcze przed konferencją z inicjatywy John Hopkins University wystosowano petycję wzywającą do zmiany nazwy (Shead, 2018; Simonite, 2018). Wskazywano na seksistowskie incydenty w poprzednim roku: nieformalna impreza przed rozpoczęciem konferencji nazwana była akronimem TITS (slangowo cycki) (Hu, 2018). Elon Musk wzbudził aplauz, mówiąc ze sceny: „no NIPS without TITS" („nie ma sutków bez cycków") (Koch, 2018), natomiast nazywający się „zróżnicowanym zespołem" pracownicy firmy Dessa nosili i sprzedawali koszulki z napisem „My NIPS are NP-hard" („moje sutki są NP-twarde"[21]) (Koch, 2018; Simonite, 2018). Rada konferencji w odpowiedzi na petycję najpierw przeprowadziła ankietę wśród uczestników z ostatnich pięciu lat, w wyniku której postanowiła nie zmieniać nazwy (Koch, 2018; Simonite, 2018). Po kolejnej petycji z inicjatywy Animy Anandkumar, dyrektorki badań

21 Lub „NP-trudne", bo żart ma jednocześnie konotacje anatomiczne i matematyczne. Te drugie dotyczą tzw. NP-trudnych problemów obliczeniowych (*NP-hardness*; *non-deterministic polynomial-time hardness*). Tłumaczę jako „twarde", by podkreślić, że w przywołanych źródłach zaznaczano seksistowski wymiar żartu.

firmy NVIDIA[22], nazwę jednak zmieniono na NeurIPS (Koch, 2018; The Neural Information Processing Systems Foundation Board of Trustees, 2018). Konferencja wprowadziła też kodeks postępowania pierwszy raz w 2018 roku (Merity, 2017; NeurIPS, 2018).

Działania tych grup, przynajmniej przez najbardziej nagłośnioną pozycję w dyskursie opisywanej areny dotyczącej data scientistek, są oceniane pozytywnie. Rozmówcy argumentowali, iż kobiety mają trudniejszy niż mężczyźni dostęp do świata DS z powodu stereotypów i wieloletniej socjalizacji raczej do ról nietechnicznych. Grupy nastawione na włączanie mniejszości genderowych mają dawać kobietom (głównie wannabesom i początkującym) poczucie przynależności, bezpieczeństwa, dostarczać przykładów innych kobiet silnie osadzonych w badanym świecie.

Ponownie pojawia się nawiązanie do nerda, uważanego m.in. za osobę mało „towarzyską", zainteresowaną raczej abstrakcjami lub technologiami niż ludźmi, a będącą punktem odniesienia dla świata DS. „Nerdki" są, zdaniem jednej z rozmówczyń, jeszcze bardziej niż podobni mężczyźni poddawane presji otoczenia, muszą wykazać się jeszcze większą niż oni wytrwałością w rozwijaniu swoich jakoby „nienormatywnych" w ogóle, a szczególnie dla kobiet, zajęć i zainteresowań:

> *B: A jak pani sądzi, czy coś dają te inicjatywy grup właśnie przeznaczonych dla kobiet, typu R-Ladies czy tam PyLadies? Są też w Polsce.*
> R: Tak, myślę, że tak, ponieważ to ośmiela kobiety do tego, żeby się z kimś identyfikować. […] Nie czuje się człowiek taki, taki **wyautowany**. Wie pan, ja mam wrażenie, że naprawdę większość kobiet, które dzisiaj pracują zawodowo w tych technicznych zawodach tak w miarę wysoko, znakomita większość z nas, gdyby nas poddać diagnostyce psychiatrycznej, okazałoby się, że mamy spektrum autyzmu. Takie jest moje podejrzenie, to nie jest poparte niczym. Natomiast takie jest moje podejrzenie, bo kto normalny oparłby się tym wszystkim nagabywaniom, namowom, przekonywaniom? [pauza] Musiało być coś, co spowodowało, że to wszystko nam poszło gdzieś mimo uszu. No bo, bo nie ma siły. Ja jestem Aspergerem i spodziewam się po prostu, że większość z nas ma tego rodzaju jakieś tam, ee tam, jesteśmy nieneurotypowe. W przypadku mężczyzn matematyków czy informatyków takich, z którymi mam do czynienia, to też to obserwuję […] ale większość z nas jako dzieci, jak się tak porozmawia, czuła się kompletnie wyautowana, z takiego czy innego powodu nie była w tej takiej maistreamowej grupie, tylko gdzieś z boku. Takżeee tego się spodziewam, że większość kobiet, które się dzisiaj przebiły, gdzieś tam ma coś za uszami trochę inaczej. Nie jest to szkodliwe, pozwala nam to normalnie funkcjonować, ale pozwoliło nam też dojść tu, gdzie jesteśmy pomimo tego, że byłyśmy kierowane, no, może ty jednak byś została fryzjerką, kosmetyczką, może pójdziesz do szkoły uczyć matematyki. No nie, nie pójdę, to mnie nie interesuje.
> *B: Rozumiem. Ale to nie dotyczy tylko tej, no, relatywnie nowej branży data science, tylko w ogóle tych techniczno-matematycznych dziedzin, tych dziedzin STEM tak zwanych, tak?*
> R: Tak mi się wydaje. {wywiad 25}

22 Firma będąca producentem m.in. popularnych procesorów graficznych, tzw. GPU, które standardowo służą do wyświetlania wysokiej jakości grafiki (np. trójwymiarowej w grach komputerowych). W ML, szczególne deep learningu, obliczenia wykonuje się często na GPU zamiast na CPU, czyli podstawowym procesorze komputera, z uwagi na znacznie krótszy czas trenowania takiego samego modelu na GPU niż na CPU.

Na czym polega arena dotycząca data scientistek – kobiet w świecie społecznym DS? **Spór toczy się wokół tego, jak w badanym świecie traktować kobiety.** Obiektem granicznym areny jest płeć.

Najbardziej widoczna i pozornie jedyna pozycja (najbardziej nagłośniona, poprawna politycznie, a z uwagi na poprawność bez obaw wyrażana) w dyskursie świata DS to dążenie do budowania bardziej różnorodnych zespołów, które mają tworzyć lepsze produkty i realizować lepsze projekty, bo wypracowują rozwiązania jakoby właśnie bardziej różnorodne, „niezbiasowane" w kierunku perspektywy młodych, białych mężczyzn – grupy ilościowo dominującej. Jest to pozycja reprezentowana np. przez członków grup nastawionych na włączanie mniejszości genderowych do DS (lub tylko przez rzeczników tych grup) oraz przez biznesowych rzeczników świata DS (bez wątpienia można wyróżnić osoby zajmujące się HR, raczej także ludzi odpowiedzialnych za wizerunek firm i za marketing). Pozycja ta ma więc uzasadnienie ekonomiczne, utylitarne, nakierowane na efektywność, jest zatem zgodna z wartościami świata DS. Odwołuje się także do wartości humanistycznych, takich jak etyka, w kontekście odpowiedzialności etycznej za sprawczość, jaką mają ludzie realizujący projekty DS i budujący elementy systemów bazujących na ML. Różnorodne zespoły mają przyczyniać się do tego, że budowane rozwiązania, przede wszystkim modele ML, będą mniej dyskryminować czy krzywdzić osoby z mniejszości. Niemniej osią argumentacji jest uzasadnienie utylitarne. Warto przyjrzeć się wypowiedzi rozmówczyni, będącej organizatorką jednego z meetupów nastawionych na włączanie mniejszości genderowych do DS:

> *B: Dlaczego to jest w ogóle ważna sprawa, prawa kobiet w data science?*
> R: Prawa kobiet w data science? Znaczy generalnie.
> *B: Bo to jest ta sprawa, tak?*
> R: Wiesz cooo, ja [krótka pauza] tak jak mówiłam na początku, że jestem daleka od etykietek, jak mówiliśmy o data scientistach…
> *B: Ok, tak.*
> R: To myślę, że tutaj patrzę podobnie, czyli generalnie, eee, prawa kobiet, tak, ale spojrzałabym na o szerzej, generalnie prawa ludzi, komfort pracy, rozwój technologii w kierunku etycznym? Chodzi o to, że, mmm, w zasadzie tutaj są dwie kwestie, tak jakby spojrzeć nawet na to z biznesowego punktu widzenia, to firmy, które wdrażają tak zwane diversity, mają wysoki ten współczynnik różnorodności pod, pod różnymi aspektami, to są jednocześnie te firmy, które mają najwyższą, eee, skuteczność, yyy, gospodarczą. […] to są te firmy, które na przykład w poziomie innowacyjności, eee, przodują między innymi dlatego, że [westchnienie] algorytmy, które tworzą, eee, osoby tak, zajmujące się machine learningiem od, odzwierciedlają tak naprawdę te biasy, czyli te różne uproszczenia czy skrzywienia, eee, percepcyjne, czy przetwarzania informacji, które mają ludzie. […] I wydaje, ja staram się na to szerzej właśnie patrzeć, w taki sposób, tak, bo to nie chodzi tylko o, o, o prawa kobiet, tylko o, o, o genera, o generalnie [ze śmiechem] oo prawa ludzi, tak. Ja bym chciała, żeby nie było z tym takiego problemu jak problem praw kobiet, żeby to było zupełnie naturalne, żee funkcjonują pracownice [ze śmiechem] o wysokich kompetencjach czy pracownice uczące się, wdrażające po prostu na takich samych zasadach, nie, no niezależnie od, właśnie nie wiem, płci, pochodzenia, orientacji seksualnej, poziomu niepełnosprawności i cokolwiek sobie wybierzemy. {wywiad 21}

W dalszych rozważaniach tę pozycję nazywam pozycją różnorodności.

W polskim świecie DS nie spotkałem się z problematyzowaniem innych mniejszości niż kobiety. Istnieją prace dotyczące defaworyzowania w DS z uwagi na rasę i pochodzenie, co nazywane jest formą kolonializmu. Chodzi o defaworyzowanie nie tylko w ramach działania wobec użytkowników końcowych, czy tzw. klikaczy, ale także wobec osób zajmujących się DS (Crawford i in., 2019; Mohamed, Png, Isaac, 2020) oraz wobec krajów Globalnego Południa (Birhane, 2020). Stanowisko wyróżnionych autorów także odpowiada pozycji „różnorodności".

Przeciwstawna pozycja zakłada doskonałą merytokrację, w której nie liczą się takie cechy jak płeć, a tylko kompetencje techniczne i metodologiczne konkretnych osób. Jest to także zgodne z wartościami badanego świata DS. Pozycja „różnorodności" jest z tej perspektywy krytykowana za sprzeciwianie się merytokracji, za aktywne wspieranie np. kobiet – wypływające z samego faktu bycia kobietą. Z tej pozycji to, że w DS i w ogóle w technologiach jest mało kobiet, postrzega się jako różnorako uwarunkowany fakt, którego nie można zmienić decyzjami np. w dziale HR. Nie występują sygnały o postrzeganiu którejkolwiek płci jako lepszej/gorszej w DS – płeć ma być cechą nieistotną:

> *B: Ok. A jeszcze mnie to ciekawi, jak to jest z płciami, u ciebie w pracy?*
> R: [pauza] Mmmm, no z płciami jest, znaczy nie ma 50 na 50, yaaa, **ale** jest powiedzmy coś koło 20, 30% kobiet, co myyyyślę, że względem odsetka **kobiet**, które ogólnie są w branży, ee, myślę, jest nawet nadreprezentacją. Mmm, no teraz na przykład przy tej rekrutacji, ee, zgłosiła się chyba w sumie, na 40 CV jedna kobieta tylko, która jak dostała wstępne zadanie, mmm, przestała się odzywać. […] Więc, no, nawet, jakbyśmy chcieli podwyższać, yy, no to niezbyt, znaczy nawet jakbyśmy na upartego chcieli wziąć kobietę, na podstawie tej rekrutacji, to nawet by się nie dało.
> *B: Mhm, czyli co, jest, jest bardzo ciężko znaleźć kobiety, tak?*
> R: **Tak**. [pauza] Mmm, myślę, że w branży jest [pauza], tylko pytanie właśnie, ile może być w branży, bo na przykład w [nazwa miasta] działa, mmm, ten meetup [nazwa organizacji wspierającej mniejszości genderowe w DS]. I **tam** uczestników, w sensie mężczyzn, jest zwykle 70, koło 80%. […] I powiedzmy tam no, można by się spodziewać nadreprezentacji kobiet, nie? W sensie względem tego, co jest w branży. […]
> *B: Tak. Rozumiem. Hmm, ok, ale czy, czy chciałbyś, żeby było więcej kobiet?*
> R: [pauza] Znaczy, pfffff, moim zdaniem to jest **każdego** wybór ścieżki życiowej, co chce robić.
> *B: Mhm, ale czy, czy dla ciebie w pracy ma to jakieś znaczenie, czy, czy, czy zatrudnisz kobietę czy faceta?*
> R: Dla mnie nie. I mam nadzieję, że nigdy nie będę **musiał**, na przykład decydować się, że zatrudnię, eee, gorszą kobietę niż inny kandydat mężczyzna, tylko dlatego, że jest kobietą.
> *B: Mhm, ale to. Ok. Czy to znaczy, że jest jakaś presja na kobiety, że, nie wiem, twój szef chce kobiety?*
> R: Nie ma na szczęście, ale są, słyszałem, że inne firmy mają mocne preferencje, żeby zatrudniać, typu żeby wyrównywać. Znaczy nie tylko w data science, tylko ogólnie w technologii i tak dalej, nie?
> *B: Mhm, ale to ma być jakieś wizerunkowe, czy* [pauza] *nie wiem, o co to chodzi?*

> R: [pauza] Znaczy dla **części** firm mi się wydaje, że wizerunkowe. Na przykład w **nauce** to już jeden profesor mówił z 10 lat temu, że w grantach musi się tłumaczyć, dlaczego nie ma po równej liczbie mężczyzn i kobiet. Po prostu, skąd ma napłodzić tych kobiet.
> *B: Skoro ich nie ma, tak?*
> R: Znaczy w sensie na informatyce na przykład, u mnie na kierunku było [pauza] 5 kobiet na 130, 140 osób. {wywiad 19}

Osoby stojące na pozycji „doskonałej merytokracji" krytykują takie praktyki jak kierowanie dofinansowania konferencji/szkoleń DS tylko dla kobiet, podczas gdy mężczyźni płacą pełne stawki (DataCamp, 2018). Powszechnie krytykowane są praktyki nieformalnego lub formalnego nacisku na to, by w składzie zespołu DS znajdowały się kobiety (co czasami nazywane jest kwotą, od *gender quota* – np. „zespół HR zaproponował, żebyśmy dążyli do kwoty 30% kobiet w dziale DS"). Te i podobne praktyki uznawane są z punktu widzenia opisywanej pozycji za niesprawiedliwe, za obniżanie poprzeczki dla wybranej grupy, a w konsekwencji za działanie nieleżące w interesie świata DS, bo zmniejszające jego elitarność, jakość pracy, poziom biegłości technologicznej poprzez dopuszczanie do uczestnictwa osób, które na „normalnych" warunkach nie weszłyby do tego świata.

Z pozycji „różnorodności" owe „normalne" warunki są takie, że kobietom jest znacznie trudniej wejść do świata, obniżanie poprzeczki jest wyrównywaniem szans, rzekoma doskonała merytokracja jest tylko stwarzaniem pozorów egalitarności w imię zachowania elitarnego statusu grupy dominującej, a większa różnorodność leży jednoznacznie w interesie świata DS z uwagi na opisywaną już zależność między różnorodnym składem zespołów DS a lepszymi produktami i projektami.

> R: […] I teraz tak, **jest** mniej kobiet, co nie jest dziwne, bo po prostu historycznie, już abstrahując od branży, historycznie rola kobiety, jednak społeczna rola była trochę inna, teraz trochę ewoluuje ta rola, ale ja bardzo nie lubię czegoś takiego, że teraz się mówi, o boże, **nadal** mamy **tylko** 20% kobiet, bo powinniśmy spojrzeć na czas, na przestrzeń czasową, że kilka lat temu to było 2–3%, tak? To, że my mamy teraz 20%, to jest sukces. Jeśli my zaczniemy teraz tak naciskać, jak zaczynają naciskać niektóre kobiety czy niektóre nurty, to my to tylko zepsujemy i ja to już mocno widzę. […] Druga sprawa, są też kobiety, które to wykorzystują. Ale z drugiej strony spójrzmy, no jest więcej mężczyzn. Czy spotkały mnie nieprzyjemności duże z tego powodu, że jestem kobietą, na mojej przestrzeni kariery? Tak. Ze strony mężczyzn i ze strony kobiet, także z obu tych stron. Ja zarówno pracowałam dla firm, które bardzo promowały kobiety, jak i w których było bardzo mało kobiet i był większy nacisk na mężczyzn i uważam, że i tu, i tu są dobre i złe strony. Ja uważam, że my powinniśmy ewolucyjnie po prostu do tego podejść i tak rozsądnie, naprawdę bazując na danych. Czyli tak, jest mniej kobiet, ale trend jest taki, że jest ich coraz więcej i tak z ciekawostek właśnie ta firma, w której właśnie teraz pracuję, nie ma kwoty. Nie ma kwoty i jest 27% kobiet, co pokazuje, że po kompetencjach tylko i wyłącznie, zostało wybranych 27% kobiet i to pokazuje, że rzeczywiście jakby ten trend idzie w górę i ja to widzę po wszystkich statystykach, że to idzie. Ja nie lubię nacisku. A czy kobietom jest trudniej, czy łatwiej? To ciężko powiedzieć. To bardzo zależy też, nie? Ja mam też silne zdanie i raczej umiem je wypowiedzieć, więc mi jest łatwiej. Czy to jest kwestia płci, czy nie? Kwestia bardzo dyskusyjna. Mnie jest z tym

> dobrze. To, że jest więcej kobiet, fajnie, ja też promuję, współzałożyłam też w Polsce [nazwa organizacji wspierającej mniejszości genderowe w DS – przyp. R.Ż.]. […] I fajnie, że tak się wspieramy i że jest więcej kobiet, ale po prostu nie róbmy tego **na siłę**. To mi przeszkadza, mówiąc szczerze, bardzo nie lubię tego, że to jest robione na siłę, bo ja tylko odczuwam tego negatywne skutki, wynikające z frustracji właśnie kobiet, mężczyzn. {wywiad 22}

Obie pozycje – „różnorodność" i „doskonała merytokracja" – są zgodne z wartościami świata DS, ale wzajemnie sprzeczne. Sprzeczne jest, by jednocześnie aktywnie promować i wspierać różnorodności oraz nie zauważać np. płci i stawiać tylko na kompetencje. Pozycja „różnorodności" koncentruje się na kategoriach pozakompetencyjnych (jak płeć), a pozycja „doskonałej merytokracji" postuluje wyciszenie, wymazanie takich kategorii ze świata społecznego. Pozornie te pozycje są zgodne, ponieważ zgodnie z wartościami badanego świata jedna pozycja nie krytykuje drugiej w pełnej rozciągłości. Osoby z pozycji „różnorodności" także zgadzają się ze stanowiskiem, że merytokracja w świecie DS jest dobra, i że nie należy np. zatrudniać słabiej wykwalifikowanej kobiety kosztem lepiej wykwalifikowanego mężczyzny i mogą argumentować, że obecne wspieranie mniejszości służy temu, by w przyszłości można było pomijać kwestię płci – por. {wywiad 21}: „ja bym chciała, żeby nie było z tym takiego problemu jak problem praw kobiet, żeby to było zupełnie naturalne, żee funkcjonują pracownice o wysokich kompetencjach". Osoby z pozycji „doskonałej merytokracji" zgadzają się za to z tezą, iż bardziej różnorodne zespoły mogą realizować lepsze, bardziej różnorodne projekty i przyczyniać się do budowy lepszych, mniej „zbiasowanych" produktów. Tak więc przez różnorodny skład zespołów (mówi się czasem nie o płci, ale np. o różnorodnym wykształceniu, doświadczeniu zawodowym, używanym stosie technologicznym itd.) jak najbardziej może być osiągana wyższa efektywność, a także bardziej etycznie wrażliwa sprawczość – która i tak przekłada się przecież na korzyści utylitarne, takie jak zadowolony klient, brak kłopotów prawnych, zyski finansowe, ale również przyczynia się do zmieniania świata na lepsze.

Te dwie pozycje w dyskursie na opisywanej arenie pozostają bez związku z płcią uczestniczących w badanym świecie DS osób, które zajmują owe pozycje.

Opisany spór odbywa się na tle starych, jawnie w świecie DS potępianych stereotypów o kobietach, jakoby naturalnie gorzej predestynowanych do obszarów ścisłych, technicznych. Jeżeli w świecie DS istnieją osoby przyjmujące ten stereotyp jako swój pogląd, to są one ukryte, nie zabierają głosu na arenie, nie można uznawać ich za pozycję w dyskursie. Jednak uczestniczki badanego świata wciąż spotykają się z osobami obu płci reprodukującymi opisany stereotyp. Rozmówczynie opisywały sytuacje bycia uważanymi za osoby nietechniczne ze strony przedstawicieli świata biznesu (klientów wewnętrznych/zewnętrznych oraz koleżanek i kolegów z innych działów), a także akademii (jako studentki przez profesorów, jako współpracowniczki przez koleżanki i kolegów po fachu). Na całej arenie modelowania **występuje zjawisko podważania kompetencji technicznych kobiet, wywodzone z samego faktu bycia kobietą**. Warto przyjrzeć się dłuższej

wypowiedzi jednej z rozmówczyń, osoby silnie osadzonej w badanym świecie i mającej kilkanaście lat doświadczenia zawodowego w akademii i biznesie:

> *B:* […] *jak się pani pracuje, jak się pani czuje w takim w cudzysłowie męskim świecie?*
> R: Ciężko. **Zawsze** było ciężko. Tak, jak pan mówi. Na studiach zaczęło nas studia pięć, skończyłyśmy chyba trzy.
> *B: A pięć na ile na roku osób?*
> R: Przyjęto około setki, eee, o ile dobrze pamiętam. W każdym razie skończyło nas studia w ogóle 25 osób. Więc to były takie studia, które rzeczywiście mocno przesiały towarzystwo. No i prawda jest taka, że dziewczyny nigdy się nie ciągnęły w ogonie. Niestety, nasz świat jest taki, jaki jest i spotkałyśmy się z baardzo nieprzyjemnymi sytuacjami. Był na przykład prowadzący, który w momencie, gdy nie działały porty USB, takie stare komputery, e, z przodu miały porty USB, mieliśmy coś dawać, jakieś programy, które napisaliśmy w domu. Wetknął USB do portu, okazało się, że port nie działa i facet sobie pozwolił na uwagę [zmienionym głosem] no, to tak jak z babą, do jednej dziury daje dostęp, do drugiej już nie. No i człowiek się zastanawia, czy ma już iść na skargę, czy nie, prawda? No bo jeżeli pójdę na skargę, to i tak u tego idioty muszę zaliczyć. A jeżeli pójdę na skargę, to będę mieć kłopoty. Więc jest ciężko. Generalnie ja jeszcze jestem osobą niezbyt wysoką, co też nie dodaje mi mocy. Wszystko [pauza], wszystko trzeba tuszować pewnością siebie. Trzeba mówić z dużym przekonaniem, bo w przeciwnym wypadku nikt nam nie wierzy. Na dodatek nasze osądy czy wszystko to, co robimy, jest wielokrotnie sprawdzane. Nie to, żebym miała coś przeciwko, ponieważ to tylko i wyłącznie potwierdza, umacnia moją pozycję. Natomiast nawet mój współpracownik z firmy wielokrotnie, i to nie jest trzykrotnie, pięciokrotnie myślę, że co najmniej kilkanaście razy, przyznał się do kilku, ee, **sprawdzał**, czy to, co ja mówię, jest prawdą. I to jest chłopak, który jest świeżo po studiach. Naprawdę, nie to, żebym mu to zarzucała, to nie w tym rzecz, bo skoro za każdym razem się potwierdzało, to jest OK. Ale to jest robione **notorycznie**, regularnie. Tak samo właśnie w firmie moje jakieś tam prace były potwierdzane przez firmy zewnętrzne. I firmy zewnętrzne próbowały wprowadzać inne metody, że na pewno wyjdzie lepiej, że to, że tamto, że coś jeszcze, **nigdy** nie okazało się, żeby ktokolwiek był w stanie osiągnąć lepsze wyniki. Nie umiem powiedzieć, czy te wyniki osiągane przez inne firmy były gorsze, czy były porównywalne. Wiem, że nie były lepsze. Ale proszę sobie wyobrazić, jak się człowiek czuje, kiedy się dowiaduje, bo wiesz, zatrudniliśmy tutaj z zewnątrz firmę, żeby nam to sprawdziła innymi metodami. Ktoś zdecydował, że wywali **kupę kasy** na to, żeby coś potwierdzić, żeby potwierdzić, że moja robota nie jest do bani. Ja z jednej strony rozumiem to ze względów marketingowych, czy generalnie ze względów związanych z wydawaniem grubych pieniędzy. Że mając tego rodzaju potwierdzenie, wiadomo, że kierunek jest słuszny, ale z drugiej strony to zawsze działa na głowę, zawsze się w jakiś sposób obrywa, bo to jest tak jak powiedzenie: wiesz co, ufamy ci, ale tak nie do końca. Ta branża tak wygląda. […] No i tak jak pan mówi, na dodatek, jak wchodzi kobieta, no to w ogóle jest spora wątpliwość, czy ona może mieć jakiekolwiek pojęcie o tym, co mówi. […]
> *B: No i, jak rozumiem, z drugiej strony w środowisku analityków na przykład pani praca jest, jak rozumiem, bardziej sprawdzana niż praca kolegi?*
> R: Tak, zawsze jest weryfikowana co najmniej dwukrotnie. I co ciekawe, również przez kobiety. Współpracujemy też z firmą, która też w ramach [nazwa projektu], która pisze cześć oprogramowania, tą taką bezpośrednią dla użytkownika. No i wiadomo, integrujemy różne fragmenty oprogramowania naszego z ich oprogramowaniem, no i potrzebowali wykonać pewne testy, żeby mieć szacowanie czasu. No i jak dziewczyna zaczęła puszczać testy, aa, akurat tutaj też współpracuję z dziewczyną, o ile się nie mylę to też jedyną kobietą u nich w zespole, to też dzwoniła do mnie, słuchaj, ale czy to możliwe, bo ja na tej próbce danych od

> ciebie to mi działa w takim i takim czasie. Czy to jest możliwe? Ja mówię, wydaje mi się, że jest możliwe, ale przygotuję ci drugą próbkę danych, odpowiednio większą, żebyś wiedziała też, jak to się skaluje w czasie, żebyś sobie mogła przeliczyć. No i okazało się, że rzeczywiście to tak działa po prostu. Ale puszczała testy. Bo wiesz, bo ja już to sprawdzałam kilka razy. No właśnie **wiem**. Także niestety, jesteśmy przyzwyczajone do tego, że patrzy się nam na ręce dużo bardziej niż komukolwiek innemu. {wywiad 25}

W powyższej wypowiedzi, poza kolejną w moich badaniach narracją dotyczącą dyskryminacji ze względu na płeć na polskiej, publicznej uczelni wyższej, uzyskałem jedyny tak wyraźny obraz praktyki podważania kompetencji kobiet.

Kompetencje kobiet traktowane są jako wyjątek, zarówno przez mężczyzn, jak i inne kobiety, i to nie tylko w polskim świecie DS ani nawet nie tylko na arenie modelowania, ale w różnych obszarach technicznych – tak w działaniach hobbystycznych, jak i komercyjnych, także w nauce i edukacji (Zaród, 2018: 352–353). Kobietom trudniej niż mężczyznom uzyskać status autentycznej uczestniczki świata DS właśnie z uwagi na domyślne traktowanie kobiet jako osób nietechnicznych. Nawet przez osoby niezajmujące się DS ani żadną sferą techniczną kobieta na stanowisku technicznym może być uważana za kogoś nie na swoim miejscu, wyłącznie z uwagi na płeć.

Rozwijam dalej związek bycia poddawanym kwestionowaniu kompetencji z byciem osobą młodą i niedoświadczoną w DS. Warto przyjrzeć się wypowiedzi dotyczącej tej zależności z byciem kobietą:

> *B: A czy chciałabyś, yy, opowiedzieć o jakiejś sytuacji nieprzyjemnej, która ciebie spotkała?*
> R: Nie chciałabym chyba opowiedzieć, nie chcę ich sobie tak przypominać, ale raczej podtekst seksualny. Niestety, spotykałam się z podtekstami seksualnymi, ale tak na co dzień z czym się spotykam, ze względu zarówno na mój wiek, jak i to, że jestem kobietą, no ja nie mam tak, że wchodzę do sali i wszyscy myślą [zmienionym głosem] ta to ma wiedzę, to jej posłuchajmy.
> *B: Ale ze względu na wiek?*
> R: Wiesz co, bardzo często jednak różnica wieku między moimi odbiorcami a mną jest znaczna.
> *B: Czy mówimy o tym, że jesteś młoda i młodo wyglądasz?*
> R: Tak. Tak, tak, tak, tak, tak, tak, dokładnie o to chodzi. [pauza] No to jednak tak, trzeba na dzień dobry często udowodnić, jak rozmawiam z innymi kobietami, też z takimi, które bardzo wysoko zaszły, no to też jednak widzą to, że często jest tak, że na dzień dobry trzeba wejść i trochę udowodnić. Ale ja uważam, to znaczy to nie jest złe, to wynika właśnie jakby z tej ewolucji, ze względów czasowych i ja muszę przyznać, że tak jak na początku, jak byłam jeszcze młodsza i byłam taka bardziej [głośno] „jak to nie chcecie mnie słuchać!?". I próbowałam to siłą zrobić, to przynosiło odwrotne efekty. Teraz, kiedy przychodzę i ktoś podważy moją merytorykę na dzień dobry, ja po prostu staram się dalej robić swoje i powiem ci, że **zawsze**, zawsze spotykam się z superodbiorem. To, ile razy, nawet nie to, że obraził mnie ktoś jakoś, ale wielokrotnie słyszałam takie „przepraszam cię", myślałem, że nie masz takiej wiedzy. Albo przepraszam cię, wziąłem po prostu, nawet dosłownie miałam takie sytuacje, jak z mężem rozmawialiśmy często z osobami, że zwracano się do mojego męża, który jest mniej techniczny niż ja i ja zaczynałam odpowiadać na pytanie i mieliśmy taką sytuację, że właśnie jeden chłopak powiedział, mężczyzna, jedna osoba powiedziała „przepraszam cię, spojrzałem

> na ciebie, mówię kobieta, na pewno nie wiesz". „Przepraszam cię, bo sam się złapałem na tym, że to zrobiłem". Więc my też musimy wziąć pod uwagę, że to nie jest tak, że ci wszyscy ludzie to robią celowo. To czasami jest po prostu taki kontekst, który przez ileś lat trwał.
> *B: Ale to rozumiem, że tak zachowują się też osoby z różnych biznesów, z którymi ty się spotykasz, że to nie jest nawet kwestia IT?*
> R: Niee, nie. Przeróżnie. Bo czy to jest tylko kwestia IT? Nie oszukujmy się, osób na wysokich poziomach jest w ogóle dużo więcej mężczyzn. Tak jest no, po prostu. Więc tak, ale nie jest to tak dotkliwe i tak straszne, jak to jest promowane. Częściej to jest kwestia tego, że po prostu jest na początku taka konsternacja, ale co też mi słusznie kiedyś kumpel powiedział, to nie zawsze jest kwestia tego, że jestem kobietą. To nie zawsze jest kwestią tego, że jestem młodsza. Po prostu kontekst, wejście, dzień, cokolwiek, tak? Oni też czasami muszą się bronić. {wywiad 22}

Zaród (2018: 348–349) wskazuje, że w hakerspejsach kobiety nie mogą w pełni uczestniczyć w tzw. laboratorium, czyli miejscu charakteryzującym się minimalizowaniem kosztów porażki osób eksperymentujących, „karnawałem poznawczym" i wspólnotą indywidualnej koncentracji. Te „podwójne standardy" oceny kompetencji dla mężczyzn i kobiet polegają, zdaniem Zaroda, na tym, że **mężczyznom łatwiej, a kobietom trudniej oddzielić błąd od osoby**. W związku z tym „mężczyźni mają szerszy margines błędu, mogą stawiać ryzykowniejsze hipotezy bez groźby kompromitacji" (Zaród, 2018: 348–349). Na gruncie badanego przeze mnie świata kwestionowanie kompetencji także można uznać za praktykę co najmniej utrudniającą pełne uczestnictwo w DS. Projekty DS polegają przecież w dużej mierze na eksperymentowaniu – poprawianiu własnych błędów, testowaniu pomysłów, szukaniu wyjścia ze ślepych uliczek, chałupniczych przeróbkach narzędzi pracy. Właściwie jeżeli tylko praca uczestnika jest samodzielna i efektywna, to błądzenie i pomyłki co do zasady (być może „zasada" ta dotyczy mężczyzn) pozostają w zgodzie z wartościami badanego świata, a nawet mogą być pozytywnie oceniane w perspektywie ciekawości i samorozwoju.

Wśród osób z obszarów technicznych, zaangażowanych w szeroką arenę modelowania pojawia się wyrażana w dyskursie identyfikacja praktyki kwestionowania kompetencji osób reprezentujących mniejszości, zarówno w DS, jak i w ogóle w IT. Nie mam informacji, by działo się to w Polsce, lecz przykładowo podczas obywającej się w Portland konferencji programistycznej The Monktoberfest w 2019 roku na ten temat wypowiedział się Wickham. W swojej prelekcji, skierowanej do osób rozwijających oprogramowanie *open source*, mówił, iż bardzo dużo zawdzięcza on uczeniu się na własnych, publicznych pomyłkach. Zauważył jednak, że kiedy biały mężczyzna popełnia błąd publicznie, ludzie mówią „ale on jest odważny" i działa to na jego korzyść; kiedy zaś robi to samo kobieta lub osoba kolorowa, ludzie mówią „ale głupia/głupi" i uznają za osobę niekompetentną (Wickham, 2019). Na tej samej konferencji Abby Fuller, z wykształcenia politolożka, pracująca na stanowisku inżynierskim Principal Technologist w Amazon AWS, wskazywała na ślepą koncentrację branży IT na byciu osobą techniczną, podczas gdy dla niej w IT najważniejsze jest budowanie czegoś działającego, odnoszenie się w swojej pracy do

danych ilościowych, ciekawość, chęć uczenia się i popełniania błędów (kategorie te są tożsame z wartościami świata DS w moim ujęciu), a nie jakaś niedookreślona techniczność sama w sobie. Wskazywała, że ona sama była wyznawczynią techniczności – uważała, że jeżeli mówi na konferencjach cokolwiek nietechnicznego, to szkodzi swojej karierze, bo przecież ciągle musi udowadniać swoje techniczne kwalifikacje jako osoba z mniejszości. Jej zdaniem dla większości ludzi w branży IT domyślny stan jest taki, że nie wierzą w kompetencje kobiety i ona musi je udowadniać. Według niej nie mówią prawdy ludzie twierdzący, że zwracają uwagę tylko na umiejętności koderskie (Fuller, 2019), co uważam za krytykę pozycji „doskonałej merytokracji".

Z pozycji „doskonałej merytokracji" wskazywano mi, że kwestionowanie kompetencji dotyczy niekoniecznie kobiet, a osób młodych i niedoświadczonych, oraz jest strategią klientów w radzeniu sobie z negocjacjami biznesowymi w trudnej merytorycznie materii. Tak więc, szczególnie jeżeli chodzi o osoby młode i niedoświadczone, to podważanie, a raczej testowanie kompetencji ma jakoby racjonalne uzasadnienie, a po prostu w sensie ilościowym bardzo mało jest w DS kobiet starszych i o dużym doświadczeniu. Relatywnie częściej można spotkać kobiety początkujące i młode, więc zależność między byciem kobietą a kwestionowaniem kompetencji tej osoby w dużym stopniu wyjaśniać mają zmienne wieku i liczby lat doświadczenia. Może wydawać się, że kwestionowane są kompetencje kobiet jako takich, a chodzi o testowanie osób młodych i niedoświadczonych, przy czym najczęściej, kiedy spotyka się w DS kobietę, to jest ona właśnie młoda i niedoświadczona.

Kwestionowanie kompetencji spotyka także kobiety silnie osadzone w DS, z wieloletnim doświadczeniem, powoływanie się głównie na wiek i doświadczenie w wyjaśnianiu opisywanej praktyki odczytuję zatem jako charakterystyczne dla pozycji „doskonałej merytokracji". Niemniej bez wątpienia testowanie kompetencji jest praktyką stosowaną też wobec młodych i początkujących uczestników mężczyzn. Tego typu doświadczenia relacjonowali mi np. byli doktoranci nietechniczni z dziedzin obliczeniowych nauk przyrodniczych.

Kwestionowanie lub testowanie kompetencji może spotykać także osoby bez wykształcenia technicznego lub matematycznego, a szczególnie te, które nie studiowały na politechnikach. Takim osobom, podobnie jak kobietom, trudno uzyskać status autentycznych uczestników świata DS właśnie z powodu domyślnego braku wiary w ich biegłość techniczną ze strony co najmniej części uczestników legitymujących się dyplomem politechniki:

> *B:* [włącza dyktafon – dogrywamy drugą, nieplanowaną część wywiadu po chwili rozmowy towarzyskiej] *Czyli absolwenci, coś jest na rzeczy z absolwentami uniwersytetu?*
> R: Tak, ewidentnie. Większość moich, yyy, współpracowników jest po polibudzie, a jak nie, no to chociaż po jakimś doktoracie w Nowym Jorku, z fizyki [pauza], więc generalnie są, są takie śmieszki iii są takie żarciki, to jest, wydaje mi się, bardzo takie uderzające, takie niemiłe pod tym względem, zwłaszcza że to [pauza], ale to nie tylko data science, pójdziesz na jakikolwiek meetup programistyczny i to samo. Wyjdzie gdzieś, że jesteś po uniwerku, to do razu jest takie, wiesz [wykonuje teatralny gest strzepywania czegoś z ramienia].

B: Ale że w ogóle po uniwerku to jest w ogóle jakaś lipa, czy?
R: Tak. Tak. Jak jesteś po polibudzie, niezależnie jaki kierunek, spoko, jesteś po uniwerku, niezależnie jaki kierunek, chujówka.
B: Aha. Ok.
R: Ale to jest takie trochę z przymrużeniem oka, nie? Nikt ci jakby nie [pauza]. No dobra, nie wiem, zależy, w którym towarzystwie, w moim czasami jest bardzo z przymrużeniem oka, czasami troszeczkę mniej nawet, więc tak dziwnie trochę, nie?
B: No dobra. Ale no jakiś przykład, czyli jak to wygląda?
R: No nie, no właśnie, rozmawiamy sobie o technologiach, ja mówię o data science, że robię w tym i w tym, i w tym, a on mówi: „aaa no pamiętam ze studiów, nie?", a ja mówię: „a no to fajne, fajne miałeś studia, nie? zazdroszczę, ja nie miałam tego na swoich studiach", a on mówi: „ha, a to co ty nie po polibudzie", to ja mówię: „nie po uniwerku, po informatyce z ekonometrią", no i wiesz, taki taki gest, że tam, że się nie umywam, nie? Chwilę później mówi: „nie no żarcik, nie?". Ale w sumie wiesz, o co chodzi, to to jest takie bardzo między wierszami, ale jednak zostaje gdzieś. Co mam powiedzieć, no miałam taką samą matematykę jak on, taką samą algebrę liniową, rachunek prawdopodobieństwa, wszystko to samo, bo wiem, co mieli ludzie, moi znajomi na polibudzie, nie? Więc jak dla mnie jest to kompletnie nieuprawnione, a sami uczą się z dupy różnych rzeczy, które wiem, że na informatyce na uniwerku są super poprowadzone, nie są tak przestarzałe jak na polibudzie, gdzie jakiegoś [niezrozumiałe] się uczą, podczas gdy na uniwerku jest to już jakby dawno wykreślone z programu, więc jak dla mnie jest to strasznie niesprawiedliwe i ten stereotyp takiego niedojdy jest nadawany niesłusznie, znaczy trochę słusznie, trochę nie. Ale wiadomo, są wyjątki i tu, i tu, więc ja się czuję jako wyjątek, który został niesłusznie, jakby wrzucony do worka, ja nie wiem komu.
B: Ale to dlaczego słusznie?
R: Dlatego słusznie, że no więcej ścisłych przedmiotów tam i ludzie generalnie, tam bardziej idą tacy ściśli, więc no jakby to się rozumie, że tam, generalnie jakby więcej jest osób, które mogą być programistami, generalnie, nie? No ale to nadal uogólnienie. [pauza] Ja rozumiem to, no sama znam ludzi z uniwerka i wiem, że **generalnie to nie są osoby, którzy idą później w programowanie**, że raczej to są osoby, które generalnie gorzej statystykę rozumieją i lubią na przykład, i ogólnie jakiekolwiek rzeczy z matmy, więc rozumiem to, z tego powodu, ale to nie zmienia faktu, że [krótka pauza] jak już ktoś widzi, że ja robię to, co robię, no to już w sumie mógłby nie wątpić w to, że mam potrzebne do tego wykształcenie i wiedzę, i powiedzmy doświadczenie, a nadal to gdzieś tam wychodzi, nie? […] tutaj nawet nikt nie mówi, że po polibudzie ktoś jest programistą, tylko że bardziej po polibudzie to jest ktoś, kto jest ścisłym umysłem po prostu, więc to bardziej jakby pasuje do charakteru, do data science i do statystyki, do programowania i do wszystkiego w sumie. […] Tak jak mówiłam, w ogóle się nie liczy, jaki masz papierek, no ale po prostu później to gdzieś wychodzi na meetupach, nie wiem głupich rozmowach w kuchni i tak dalej, nie? Jak ktoś jest, powiedzmy, burakiem, to akurat to jest pierwsza rzecz, z której sobie zażartuje, można tak powiedzieć. [pauza] Więc w tym kontekście dziewczyna po polibudzie ma prawo wyśmiać chłopaka po uniwerku, a nie, że chłopak wyśmieje dziewczynę, bo jest dziewczyną, nie? {wywiad 10}

B: No dobra, a z politechniką i [nazwa uczelni ekonomicznej A – rozmówczyni jest jej absolwentką]*?*
R: Yyy, z politechniką i [nazwa uczelni ekonomicznej A] to z nas się zawsze śmiali, że my matematyki nie umiemy i programować też nie umiemy w sumie [śmiech], ja to się z nich zawsze śmiałam, że oni to w ogóle rozmawiać nie potrafią i z nimi się nie da porozmawiać tak normalnie. {wywiad 18}

Dogłębne opisanie sporu dotyczącego traktowania kobiet w DS wymagałoby oddzielnej pracy badawczej, dążąc zatem przede wszystkim do ujęcia pełnej sytuacji mojego badania polskiego świata DS, poprzestaję na powyższym szkicu. Ta arena była dla mnie szczególnie trudna do zbadania.

Istnieje wiele publikacji (Beede i in., 2011; Drury, Siy, Cheryan, 2011; Zawistowska, 2013; Williams, Ceci, 2015; Breda, Napp, 2019; Crawford i in., 2019), w tym także publicystycznych, jak bestsellerowa *Brotopia* (Chang, 2018), poświęconych sytuacji kobiet w branży IT, w nauce i edukacji technicznej, z reguły w obszarze STEM. Przez osoby zajmujące się DS, w porównaniu do branży IT, badany świat postrzegany jest jako różnorodny, otwarty i sprzyjający kobietom.

5.2.3. Python, R i Excel

Zastosowania Pythona i R w DS różnią się od siebie. W badanym świecie uznaje się, że **te różnice wynikają z historii rozwoju każdego z tych języków**, a w konsekwencji z cech technicznych tych języków, dostępności technologii ułatwiających korzystanie z nich (przede wszystkim pakietów), a także różnic w środowiskach, które preferują posługiwanie się jednym czy drugim językiem programowania. Język R zbudowali statystycy dla statystyków i zastosowań innych niż operacje na danych jest dla R niewiele. Język R jest też kojarzony z akademią – zarówno z matematykami/statystykami rozwijającymi metody analityczne, jak i naukowcami stosującymi analitykę do swojego obszaru. Python kojarzony jest raczej z zastosowaniami komercyjnymi, co, jak wskażę dalej, jest łączone z łatwiejszą niż dla R integracją z systemami IT. Warte odnotowania jest porównywanie R z pakietami statystycznymi, takimi jak SAS czy SPSS, w liczbie publikowanych artykułów naukowych (choć od 2017 roku rośnie użycie Pythona) (por. Muenchen, 2019).

Python jest językiem programowania ogólnego zastosowania, którego część rozwinęła się w kierunku pracy z danymi. Uczestnicy badanego świata nazywają Pythona np. prawdziwym lub pełnoprawnym językiem programowania, co bywa elementem legitymizowania działań (czy sposobu ich wykonywania). Python jest historycznie krócej niż R wykorzystywany do operacji na danych w sensie zbliżonym do DS. Choć oba języki powstały w latach dziewięćdziesiątych, to R od początku służyć miał tym celom, w Pythonie takie „odgałęzienie" powstało później {wywiad 13}. Historię rozwoju języków jeden z rozmówców przedstawiał następująco:

> R: Znaczy, generalnie jest tak, że eR, yy, powstało jako narzędzie typowe do analizy danych. Wiele lat temu, tak? Było **z góry** myślane i pisane pod analizę danych. Natomiast Python jest językiem, szerokim językiem programowania, który generalnie zajmuje się programowaniem aplikacji. No i tam oczywiście są głó, głównym takim, kością niezgody jest fakt, że eR jest [pauza] **jednowątkowy**, nie wiem, na ile jesteś technologiczną osobą, na ile nie, no w każdym razie w Pythonie można szybciej i więcej danych zanalizować. No ale czy trzeba? Ja uważam, że niekoniecznie. Faktycznie, Python będzie szybszy […]. {wywiad 6}

Wskazywano, że R lepiej sprawdza się w sytuacjach, gdzie rezultatem pracy ma być wgląd w dane, a Python przy tzw. wdrożeniach produkcyjnych modeli i wtedy, gdy mają być zastosowane metody ML, a szczególnie metody DL:

> *B:* […] *Dlaczego niektórzy robią w eRze, a inni w Pythonie? Nie można robić w jednym?*
> R: Ooo! Ooo [śmiech] Wiesz co, no eR jest świetny do takich szybkich analiz, jest, podobno jest dużo łatwiejszy do nauczenia się dla osób, które wcześniej nie miały styczności z programowaniem w ogóle.
> *B: Mhm.*
> R: Ale nie wiem, czy to mi się tylko tak wydaje, że właśnie trudniej jest zrobić w eRze, eee, duży taki projekt software'owy, który ma chodzić gdzieś na serwerze. Który ma chodzić cały czas na przykład, nie? Wydaje mi się, że to jest trudniejsze, może się mylę, ale wydaje mi się, że tak. Więc to też trzeba po prostu dostosowywać do tego, co chcesz, co chcesz osiągnąć, nie? Co chcesz robić. Jeżeli to mają być po prostu **analizy** danych, to myślę, że eR super. Czy ze zdjęciami by sobie poradził, nie wiem. […] u nas w ogóle w [nazwa firmy] nie używamy eRa, w ogóle. Wydaje mi się, że eR jest w takich bardziej analitycznych sprawach, właśnie jakiś marketing na przykład, alboo banki, no coś takiego. Bo wtedy faktycznie jest dużo danych, ale no po prostu robisz tylko raport dla klienta. I eR jest do tego świetny, bo jest szybki. Ma duże, dobre biblioteki do radzenia sobie z danymi taki, ale żeby właśnie zrobić coś, co jest dedykowane na serwer, to nie wiem, czy takie najlepsze, więc my nie piszemy w eRze w ogóle.
> *B: Mhm, ok. No dobra.*
> R: Znaczy, nawet nie wiem, czy takie deep learningowe rzeczy jak sieci neuronowe są w eRze. Wiesz może? {wywiad 8}

Warto zwrócić uwagę na wykrzyknik w reakcji na pytanie „nie można robić w jednym?" – dla uczestnika badanego świata oczywiste jest, że nie można. Język R uznawany jest za narzędzie łatwiejsze dla osób, które nie miały nigdy styczności z programowaniem, a Python dla osób znających już inny język programowania. Osoby z wykształceniem lub doświadczeniem informatycznym mogą także znać Pythona w związku z innymi zastosowaniami. Jest to jedna z obiegowych opinii, podkreślić jednak należy, że początkujących zainteresowanych ML bez względu na wykształcenie zachęca się obecnie do rozpoczęcia nauki od Pythona. W toku badań spotkałem osoby, dla których Python dla operacji na danych był pierwszą stycznością z programowaniem w ogóle. Jedna z rozmówczyń podkreślała, że należy **dobierać narzędzia do problemu**: „trzeba po prostu dostosowywać do tego, co chcesz osiągnąć" {wywiad 8}.

Przydatność języka do określonych zastosowań oceniana jest przez pryzmat dostępnych pakietów. Przy tym osoba, która korzysta już w Pythonie z narzędzi spełniających jej potrzeby, nie podejmuje wysiłku szukania innych rozwiązań – takie działanie mogłoby zostać uznane za bezcelowe, bo nieefektywne.

Konkurencyjność omawianych języków dobrze ilustruje wypowiedź: „na potrzeby takiego ogólnego data science nie ma dużej różnicy między eR i Pythonem poza tym, że według mnie Python jest mniej wygodny", natomiast **komplementarność lub rozłączność**: „deep learning w Pythonie jest dużo lepszy niż

w eR […] natomiast jak się spojrzy […] na jakieś zastosowania […] naukowe, nie ogólne data science, no to Python jest cieniutki, bo bibliotek ma dużo, dużo mniej" {wywiad 5}.

Tak więc czy w badanym świecie panuje zgoda, że „do prototypów eR, do wdrożeń Python" {wywiad 13}? Język R ma być łatwiejszy w pisaniu kodu, szybszy do celów uzyskania pierwszego, obiecującego wyniku – czasami nazywanego POC (*proof of concept*) – natomiast w kolejnym etapie projektu, tzn. wdrażaniu wyników na produkcję, preferowany ma być Python – z uwagi na łatwiejszą integrację z systemami informatycznymi, wyższą niż R szybkość i wydajność działania:

> R: […] Poza tym, no, eR jest właśnie bardzo fajny do takich szybkich analiz, nie? Dostajesz ramkę danych, szybko piszesz, od razu widzisz plus minus, co tam się dzieje, tak? W momencie kiedy to trzeba zacząć jakby wrzucać na produkcję, zarządzać wersjami, no w sensie, żeby to szybko działało, optymalnie z czymś tam łączyć, no to wtedy się pojawia już taki problem, że jednak Python. Bo Python jakby całą tą otoczkę programistyczną, ma znacznie lepszą. Python jest po prostu językiem programowania i on, jakby, może tak, eR został stworzony do analizy danych, a Python jest językiem programowania, który ma jakby odgałęzienie, w analizie danych, tak? Więc jeśli ktoś się nie zna na programowaniu, a nie potrzebuje tak naprawdę programowania, tylko chce sobie poanalizować dane, to raczej będzie bardziej skłonny do eRa, ale w momencie, kiedy ktoś potrzebuje właśnie budować jakieś systemy, które potrafią analizować dane, tak, to to jest Python, bo on ma wszystko, co potrzebne, żeby budować serwisy, jakieś tam, no masę różnych rzeczy. To jest taki pełnoprawny język programowania, więc no nie ma wojny, raczej są plusy i minusy. **Do prototypów eR, do wdrożeń Python**, tak, to wszystko. {wywiad 13}

W badanym świecie rzecz jasna nie ma w tej kwestii konsensusu. Za lepsze niż Python do wdrożeń mogą być uznawane inne języki, jak Scala czy kompilowane, np. Java lub C++. Przy tym zarówno Python, jak i R są wykorzystywane i do prototypów, i do wdrożeń (Borowiecki, Mieczkowski, 2019: 37–38).

Postulowane w świecie DS czytelność i prostota kodu zostały w przypadku Pythona ujęte w spisane standardy, znane jako „Zen of Python", PEP20 i PEP8. Użytkownicy Pythona, nie tylko w DS, spierają się, czy jakieś rozwiązanie w kodzie spełnia te standardy, co nazywa się kodem pytonicznym (*Pythonic*) (Reitz, 2018: 53–54). Dla języka R także podjęto próby standaryzacji, określenia zasad stylu pisania kodu (Baath, 2012; Google, 2012), ale w badanym świecie dominuje opinia, że w R niemalże każdy pakiet ma swoją osobną konwencję, oraz że użytkownicy nie wypracowali konsensusu (Muenchen, 2014; Wickham, 2014a). Wycinkami języka R z bardziej standaryzowanym sposobem pisania kodu są ekosystemy pakietów, jak tidyverse[23]. Moim zdaniem pokazuje to, że w DS społeczność użytkowników R

23 Przykładowo: w całym ekosystemie działa operator zwany „pipe", zapisywany jako %<% (dosł. 'rurka', por. wcześniejszy fragment „no i są te pipe'y, to już w ogóle, jest bosko" {wywiad 13}). Pipe nie działa w wielu pakietach poza ekosystemem, w tym w domyślnych pakietach GNU R, choć niektórzy twórcy spoza ekosystemu zapewniają działanie operatora w swoich pakietach. Konwencja pisania kodu w tidyverse obejmuje też wiele innych

jest mniej niż społeczność Pythona zainteresowana tzw. dobrymi praktykami programistycznymi, znanymi ze świata IT. Ma być to także argumentem za tym, że język Python jest „lepszy" niż R do wdrożeń – kod w Pythonie jest bardziej standaryzowany, co np. ułatwia komunikację zespołu DS z zespołem programistów podczas wmontowywania modeli w infrastrukturę informatyczną (Borowiecki, Mieczkowski, 2019: 37–38, 41). Relatywnie łatwiejsza niż dla R komunikacja z, ogólnie rzecz biorąc, światem IT to także przykład powodu, dla którego Pythona uważa się w badanym świecie za prawdziwy czy pełnoprawny język programowania. Może to wskazywać na traktowanie go jako języka dającego uczestnikowi świata DS większą sprawczość, a w konsekwencji użytkownikom Pythona umożliwiać łatwiejsze niż użytkownikom R osiąganie legitymizacji jako autentycznych uczestników (choć już nie poziomu maestrii, który charakteryzuje znajomość kilku języków programowania).

Uznaję **użytkowników Pythona i R za dwa subświaty świata społecznego DS, zaznaczając, że są to subświaty przenikające się i rozmyte**. Część uczestników – mniej więcej między 10% a 25% – używa przecież obu języków. Mogą oni preferować użycie jednego lub drugiego i kierować się zasadą doboru „najlepszego" narzędzia do danego problemu. Widać jednak specjalizację tych rozmytych subświatów, tak w sensie używanych metod, zastosowań DS, jak i specyficznych narzędzi ułatwiających korzystanie z języka preferowanego w każdym z subświatów. Nie jest to wniosek nowy:

> Podział według rodzaju rozwiązywanych problemów odzwierciedla do pewnego stopnia różnice w używanym sprzęcie i technologiach używanych do tego celu. Zatem różne subświaty tradycyjnie używają różnych maszyn, różnych języków [programowania – przyp. R.Ż.], i często różnych podejść, rozwijając swoje procedury i aktywności (Kling, Gerson, 1978: 27).

Istnieje tutaj arena dotycząca tego, jakie elementy języków są konkurencyjne, komplementarne i rozłączne. Inne niż Python i R języki programowania mogą mieć silnie specjalistyczny charakter w sensie zastosowania (jak Scala, Julia, Lua, Lisp, Prolog), mogą być charakterystyczne dla wybranych obszarów nauki (MATLAB, Octave) lub biznesu (SAS), mogą być językami świata IT użytecznymi w działaniach wspomagających (we wdrożeniach produkcyjnych – np. Java, C++; w tworzeniu interfejsów dla użytkowników wdrożeń – Ruby, JavaScript, nawet HTML i CSS).

Spór między R i Pythonem na początkowym etapie moich badań wydawał mi się niezwykle istotny dla badanego świata, obiektem granicznym jest przecież narzędzie do wykonywania działania podstawowego. Spodziewałem się zatem głębokiego podziału na dwa subświaty użytkowników dwóch języków. Szczególnie

---

wytycznych, jak np. używanie spacji po obu stronach znaku „=" (por. https://style.tidyverse.org/, dostęp: 2.02.2020); istnieje pakiet służący formatowaniu kodu zgodnie z tą konwencją (Müller, Walthert, 2020).

w formie online wyjątkowo łatwo było mi dotrzeć do materiałów porównujących wady i zalety obu języków, spór wydawał się żywy i intensywny. Podczas obserwacji często spotykałem się z żartami na temat omawianego sporu. W toku wywiadów swobodnych moja wstępna wizja areny szybko okazała się mylna, a robocze opisy – powierzchowne, ponieważ:

1) spory o konkurencyjność, komplementarność i rozłączność cech języków Python i R oraz tego, który w jakim zakresie jest lepszy, uczestnicy badanego świata interpretują głównie w kategoriach prorozwojowych, przyjacielskich sprzeczek, które przyczyniają się do budowania społeczności DS, ewentualnie wskazując na to, iż są one próbami sił nastawionymi na legitymizacje biegłości technicznej spierających się osób;
2) te spory widziane online wydają się znacznie bardziej intensywne, niż są one w opinii osób osadzonych w badanym świecie – jest to łatwe, a nawet rozrywkowe czy wręcz zabawne pole do spierania się między uczestnikami i subświatami świata DS;
3) spory od 2018 roku straciły na znaczeniu, gdyż Python zaczął zdecydowanie „wygrywać" w sensie ilościowym – bazy użytkowników i dominacji w uznawanym za najbardziej atrakcyjny elemencie działania podstawowego, czyli modelowaniu metodami ML i deep learning (głównie pakiety scikit-learn, PyTorch, TensorFlow), R stał się zaś bardziej niszowym językiem, cenionym jako dobre narzędzie przede wszystkim do statystyki, analizy i wizualizacji oraz raportowania danych (np. pakiety tidyverse, Markdown, Shiny), niemniej w R także buduje się modele, w tym deep learningowe, ma on również bazę użytkowników nieidentyfikujących się z DS, np. w bioinformatyce lub akademii (Olszewski, 2017; 2018).

Odnośnie do pierwszego wniosku, warto przyjrzeć się wypowiedzi silnie osadzonego w badanym świecie rozmówcy, który wskazuje właśnie na prorozwojowy, właściwie budujący społeczność DS charakter sporu o wybór jednego z dwóch języków do wykonania działania podstawowego. Należy podkreślić koncentrację silnie osadzonego uczestnika na doborze właściwego narzędzia do problemu, na tzw. agnostycyzmie technologicznym:

> R: […] jeżeli ktoś przychodzi z własną technologią, ja zawsze to mówię na rozmowie kwalifikacyjnej, jeśli to jesteś w stanie robić w swojej komórce i na kartce papieru, dla mnie to nie jest problem, póki to, co robisz, jest [pauza] wysokiej jakości i projekt zmierza w tę stronę, w którą chcielibyśmy, żeby zmierzał, nie ma sprawy. To jest wy, to jest, wiesz, to są takie święte wojny, nie? Każdy to w różnych dziedzinach ma. […] Nie widzę sensu w ogóle dzielenia tego, natomiast oczywiście wszyscy w tym zawodzie, trzeba być, są ambitni, każdy uważa się za specjalistę w czymś, bardzo często jest, więc mamy te swoje małe wewnętrzne zabawy w kłócenie się o wyższości Pythona nad R lub przeciwnie, co moim zdaniem jest fajne, bo też powoduje, że obie strony się uczą, tak? […] Więc summa summarum, no, czymś musimy się zajmować oprócz tej analizy danych. O czymś musimy przy piwie tam dyskutować. {wywiad 6}

Sam udział osoby w sporze o wady/zalety Pythona i R w DS jest nakierowany na legitymizowanie tej osoby jako uczestnika świata DS. Jeżeli osoba bierze udział w sporze, to zaświadcza o przynajmniej początkującym poziomie rozeznania w najważniejszych narzędziach do wykonywania działania podstawowego i jakimkolwiek doświadczeniu w wykonywaniu działania podstawowego, a także o byciu osobą rozeznaną w technologii w ogóle. Wydaje mi się, że wraz z rozwojem różnych pakietów i różnych specjalizacji Pythona i R w DS ten spór stał się sporem zwyczajowym, charakterystycznym raczej dla wannabesów i początkujących uczestników niż dla „prawdziwych" uczestników badanego świata. Ze względu na jasną opozycję R vs. Python i powtarzanie tych samych argumentów w sieci, uczestnictwo w sporze stało się łatwo dostępnym elementem enkulturacji do badanego świata. Odnosząc się do drugiego wniosku, powtórzę, że jest to spór znacznie bardziej wyraźny online niż z perspektywy uczestników świata DS, spór „internetowych napinaczy" podkolorowany przez mechanizmy działania mediów społecznościowych, a także rzeczników świata DS:

> *B: Czy jest jakaś wojna pomiędzy eRem a Pythonem, czy pomiędzy użytkownikami, czy to jest w ogóle jakieś złe podejście? Czy ja nie powinienem tak o tym myśleć?*
> R: Nieee. Sądzę, że [pauza] yyy, kiedyś, czasy, kiedy się ludzie jeszcze lubili o coś kłócić, to już chyba minęły, ja pamiętam czasy jeszcze, jak się kłócono o to, która wersja Linuxa jest najlepsza, yyy, jakieś jeszcze dawnymi czasy to oczywiście najwięksi internetowi fighterzy kłócili się, które sporty walki są najlepsze, czy by wygrał karateka z judoką [z uśmiechem] i tak dalej. […] wszyscy to traktują z przymrużeniem oka, przynajmniej poważni, mówię o poważnych ludziach, bo wiesz, zawsze są internetowi napinacze, co tam właściwie nie używają ani jednego, ani drugiego, tylko generalnie, yyy, eee, lubią sobie tak pona, byyy się opowiedzieć za którymś obozem, ale ludzie, którzy po prostu normalnie tym pracują, z tego, żyją z tego, pracują na co dzień, no to po prostu zdają sobie sprawę, że to jest, że zazwyczaj jest to mixem wielu różnych przyczyn [krótka pauza], ten wybór danej technologii, yyy, i to nie jest takie proste, w ogóle nie można, yyy, żeby powiedzieć, że ta jest lepsza, czy ta jest gorsza, często po prostu jest to po prostu nawet przypadek, iii [krótka pauza], ok, jak trzeba to będą pracować w eRze, jak trzeba, to będą pracować w Pythonie. {wywiad 14}

Osoby opowiadające się – nie w sposób żartobliwy ani ironiczny – zdecydowanie „po stronie" Pythona lub R, ostro i niemerytorycznie krytykujące język, którego nie używają, są w badanym świecie oceniane negatywnie. Takie działanie świadczy o byciu wannabesem, ewentualne początkującym, który serio traktuje internetowe spory i nie ma doświadczenia w wykonywaniu działania podstawowego, nie rozumie projektów DS, właściwie nie działa zgodnie z wartościami badanego świata. Nawet najbardziej doświadczeni uczestnicy świata DS, uznawani za osoby o najwyższym poziomie maestrii, mają swoje jawnie deklarowane preferencje wobec używanego języka programowania do wykonywania działania podstawowego, jednak nie spotkałem się w toku badań z ostrą krytyką „tego drugiego" języka ze strony takich osób. Taka krytyka podważałaby status maestra w oczach innych uczestników.

W ramach Pythona (do działań zbliżonych do DS) i R istnieją także wewnętrzne miniareny narzędziowe, które mogą wykraczać poza świat społeczny DS. Wśród użytkowników R występują np. spory użytkowników ekosystemu tidyverse z osobami korzystającymi z R bez jego użycia (Matloff, 2020), a także spory użytkowników tidyverse (lub węziej – pakietu „dplyr") z użytkownikami pakietu „data.table", dotyczące głównie zastosowań w przetwarzaniu danych (BrodieG, 2018; Wickham i in., 2019). W ramach Pythona użytkownicy spierają się m.in. o wady i zalety pakietów do ML i DL: TF i PyTorcha (por. Sopyła, 2019) lub Keras + TF a PyTorch (Misal, 2018; Agarwal, 2019).

Mniej więcej **od 2018 roku Python uznawany jest w komercyjnej sferze działalności świata DS za domyślny język do wykonywania działania podstawowego**. Język R należałoby poznać, czy chociażby potrafić posłużyć się nim jako drugim, w konkretnych zastosowaniach, np. analitycznych czy statystycznych, posługując się już swobodnie Pythonem i stając się silnie osadzonym uczestnikiem świata DS. Istnieją rozwiązania pozwalające na używanie obu języków w jednym środowisku programistycznym (Allaire, Ushey, Tang, 2018; Jolly, 2018). Pod koniec moich badań, w pierwszej połowie roku 2019 spotykałem się z opiniami użytkowników R, że uczą się czy nawet „przesiadają" na Pythona, bo jest to korzystne dla rozwoju ich kariery zawodowej.

Python jest postrzegany w badanym świecie raczej jako język bardziej sprawczy niż R. Jest to spowodowane znacznie większą bazą pakietów do ML i DL, obecnie wysoko komercyjnie cenionych technologii, a także wieloma cechami technicznymi samego języka, silnie powiązanymi z jego rodowodem – „pełnoprawnego" języka programowania do ogólnego zastosowania. Język Python „wygrał" z R na opisywanej arenie, ponieważ bardziej odpowiada komercyjnemu kierunkowi rozwoju społecznego świata DS i ma duży napływ początkujących data scientistów rekrutujących się z obszaru IT. Te osoby najczęściej znają Pythona z innych zastosowań. Obszar IT jest ilościowo większy niż obszar matematyki/statystyki akademickiej i ma większy kapitał finansowy oraz więcej władzy (Matloff, 2017). Ponadto pakiety używane w najbardziej atrakcyjnym i komercyjnie cenionym elemencie wykonywania działania podstawowego wspierane są przez gigantów technologicznych (jest to zbieżne z wyższym kapitałem IT):

> *B:* […] *jaką wróżysz przyszłość językowi eR i językowi Python w data science?*
> R: Znaczy, tak jak teraz na to patrzę, to obawiam się, że to jednak trochę duże korporacjeeee **wymuszą** przyszłość czegoś. No właśnie tak jak teraz jest nadal na machine learningu na deep learning, to właśnie rozwijają się różne frameworki typu Tensorflow, PyTorch czy Keras, a to wszystko było pierwotnie rozwijane przez Google albo przez Facebook. Mmmm, no więc to, co te duże korporacje sobie, że tak powiem, wymyślą, narzucą, będą finansować, to to się będzie rozwijało, ale no mimo wszystko kiedy jedne i drugie jest open source'owe, ale mimo wszystko ludzie z dobrej woli nie piszą tych pakietów i tak zawsze stoi ktoś, jeśli chodzi o takie poważniejsze pakiety, no to zawsze stoi za tym ktoś, kto jednak ma budżet. Tak więc po czyjej stronie będzie ten **budżet**, ten wygra [ze śmiechem]. {wywiad 18}

Być może wolno opisać spór pomiędzy językami Python i R także w perspektywie makro – jako spór o wpływy w świecie DS dwóch obszarów, na których przecięciu powstało DS. Chodzi o informatykę/IT oraz matematykę/statystykę, a z mojej rekonstrukcji sporu na opisywanej arenie wynikałoby, że zwycięża wpływ informatyki/IT dlatego, że ten bardziej sprawczy, mniej teoretyczny obszar niż matematyka/statystyka uzyskał silniejsze wsparcie trzeciego gracza – obszaru szeroko pojętego biznesu. W amerykańskim środowisku DS tego rodzaju wnioski pojawiły się już kilka lat temu (Matloff, 2017). Ta walka jest w moim przekonaniu kontynuacją walki opisanej już prawie dwadzieścia lat temu jako konkurencja dwóch różnych kultur modelowania (Breiman, 2001). Być może pozostaję pod wpływem praktyk legitymizacyjnych badanego świata, polegających na twierdzeniach, iż wciąż mamy do czynienia z wczesnym stadium rozwoju DS, ale **kierunek rozwoju badanego świata DS może być taki, że w sferze komercyjnej zostanie on niemal całkowicie wchłonięty przez IT**, stanowisko data scientisty zniknie na rzecz inżynierów danych, ML engineerów i ML reasercherów, **a systemy bazujące na modelach ML staną się w szerokim dyskursie publicznym tak samo mało magiczne jak strony internetowe czy aplikacje mobilne**, a nawet jak samochody czy lodówki. Obszar analizy danych, ich wizualizacji i raportowania być może stanie się na powrót przede wszystkim sferą nietechniczną, bardziej biznesową, być może dalej bazując na takich narzędziach jak arkusze kalkulacyjne, ale prawdopodobnie na elementach języków SQL, R, a przede wszystkim na przyjaznych dla użytkowników narzędziach z GUI, jak np. MS Power BI, Google Data Studio lub Tableau.

W ramach podsumowania przytaczam wypowiedź rozmówcy określającego się mianem ML researchera, pracującego w akademickiej informatyce. Rekonstruuje on tworzenie się obszaru DS jako walkę o wpływy pomiędzy informatyką (w której trzeba odróżnić co najmniej obszary baz danych, ML i hardware) a matematyką/statystyką, częściowo innymi dziedzinami obliczeniowymi, a także jako walkę na narzędzia i metody do operacji na danych:

> *B:* […] *Jak pan profesor sądzi, czy jakaś ewolucja nastała, czy to są rzeczywiście nowe rzeczy, czy to jest taki rebranding, co roku jakiś nowy fajny termin?*
> R: Tak, dokładnie. To jest w bardzo dużej mierze, to jest face lifting, to jest niestety tak, że ten kawałek tortu jest ograniczony, a każdy próbuje się na niego wepchnąć. I, historycznie rzecz biorąc, to było tak, że u podłoża wszystkiego leży statystyka i statystycy patrzą na nas, na informatyków z prawdziwą pogardą, mówią: [zmienionym głosem] „przecież my to wszystko od 100 lat wiemy, a wy wyważacie, nie to, że te drzwi są lekko uchylone, to są drzwi od **stodoły**, które są otwarte na oścież, a wy tam biegniecie i krzyczycie, że coś odkryliście". A my im próbujemy udowodnić, że my jednak robimy rzeczy nowe, że nigdy żaden statystyk nie wziął 100 tys. transakcji sklepowych i nie policzył fizycznie, zakupy jakich produktów z czym się korelują i w jaki sposób. I faktycznie data mining, czyli to, co jest tłumaczone jako eksploracja danych, to był świat ludzi od baz danych, którzy dysponowali ogromnymi wolumenami danych i próbowali zobaczyć, jakie metody statystyczne dadzą się zastosować, biorąc pod uwagę ówczesne ograniczenia technologiczne, obliczeniowe, no itd., itd. Ale bardzo wyraźnie to byli ludzie pochodzący ze świata baz danych, hurtowni baz danych. Zawsze była część ludzi, która się zajmowała terminem uczenia maszynowego, chociaż to było

traktowane trochę po macoszemu, bo chętniej używali terminu decision support system, czyli systemy eksperckie czy systemy wspomagania decyzji. I oni nagle stwierdzili, że ten termin jest mało sexy, tym bardziej że część ich tortu zjadali ludzie, którzy się zajmowali ekonomią, bo to też trochę taka teoria decyzji, podejmowania decyzji, troszeczkę w tym jest ekonomii, trochę psychologii, różne rzeczy. No więc oni też się zaczęli rozglądać, bo tutaj ci nam wchodzą na nasze podwórko z jakimś data mining, a ci z kolei zabierają, no więc oni wpadli na pomysł, może odgrzebać ten termin machine learning i my teraz będziemy ML-owcy. Na to oburzyli się ludzie, którzy się zajmowali bazami danych, ale ponieważ popularność terminu data mining trochę zaczęła się wypłaszczać, trzeba było znaleźć coś nowego, no to big data. No to big data, teraz to są już w ogóle tryliony i to się nie mieści w pamięci i większość waszych algorytmów o kant tyłka roztłuc, bo jak algorytm wymaga obliczenia globalnego i byśmy potrzebowali terabajta RAM-u, no to po prostu się nie da. Więc znowu teraz będziemy pracowali nad algorytmami, które się skaluje. Jak się to zaczęło skalować, to nagle wrócili ludzie od sprzętu i mówili: halo, halo, ale my mamy tutaj procesor GPU i my możemy wam wszystko zrównoleglić, i teraz to wszystko będzie równolegle. I się zaczęło obliczanie równoległe i teraz jest big data, ale ona jest na GPU. Więc tu z kolei obudzili się statystycy i matematycy z ręką w nocniku i mówią: „ale co z nami?", więc **oni wymyślili data science**. Tyle tylko, że oni niestety wszyscy umieli programować tylko w MATLAB-ie, informatycy im powiedzieli, że ten MATLAB to jest kompletny bull shit, nikt w tym nie będzie programował. Oni najpierw się obrazili, a potem zrobili sobie język eR. Przebrandowili tego MATLABA, pododawali trochę, zrobili z tego więcej statystyki, i powiedzieli, to już się doskonale programuje, to jest prawdziwy język programowania. Czyli data science się narodziło tak naprawdę w tym środowisku R. Jako statystyka plus troszkę programowania plus wizualizacja. Rzeczywiście po raz pierwszy wizualizacja poszła tak mocno do przodu, pojawiła się ta cała gramatyka dla wizualizacji, potem cały ten ggplot itd. No to znowu wrócili informatycy, którzy zobaczyli statystyków i matematyków, którzy piszą w jakimś języku, który jest nieczytelny, w którym nie ma żadnych standardów, żadnego programowania obiektowego, to wszystko w ogóle kompletnie jest [pauza], jak informatyk patrzy na eRa, to mówi: „to jest jakiś koszmar zupełny, my weźmiemy normalny język programowania, w którym są pewne standardy programowania, biblioteki są itd., i my z tego zrobimy swojego eRa". I zaczęli pracować nad Pythonem i zaczęli go rozwiązywać, no i teraz zaczęła się ta bitwa, co jest ważniejsze. Na to się nałożyło to, że paru ludzi w Toronto odgrzebało sprzed 20 lat sieci neuronowe i nagle się okazało, że te sieci neuronowe biją wszystkich kompletnie na głowę w przetwarzaniu języka naturalnego do obrazu i dźwięku. No to trzeba było ukuć jakiś nowy termin, no to deep learning. Cały ten deep learning na dobrą sprawę to jest połączenie dwóch rzeczy, to jest trochę algebry liniowej i trochę optymalizacji kombinatorycznej i te dwie rzeczy, jak się do siebie doda, to wychodzi deep learning. Ale faktem jest, że to, co zmienia całkowicie obraz, to jest skala. Co innego to jest zastosowanie tego do niewielkiego zbioru danych, pobawienie się, to jest 1000 przykładów i coś tam z tego wychodzi. A czym innym jest przepracowanie 10 mln zdjęć i nagle pojawienie się modelu, który dostaje zdjęcie i mówi „proszę bardzo, to jest kot, a to jest płot, a to jest zielone słońce". Ale myślę, że tę ewolucję terminologii nomenklatury trzeba rozpatrywać z jednej strony jako trochę wyszarpywanie sobie rynku, bo z punktu widzenia ludzie, którzy się tym zajmują, po drugiej stronie, tam, gdzie są firmy i gdzie są pieniądze, nie ma tak naprawdę ani wiedzy, ani potrzeby rozumienia tych subtelnych różnic, więc trzeba przepchnąć swoje. Cokolwiek tam zadziała, jakikolwiek hejt będzie aktualny, to na tej fali po prostu popłyniemy. Ludzie, którzy w tym siedzą głęboko, pewnie dostrzegają jakieś subtelne różnice, z mojego punktu widzenia chyba nie ma to aż tak dużego znaczenia, bo wszystko tak koniec końców, jak się na to spojrzy z całkowicie praktycznego punktu, to ma znaczenie dla ludzi, którzy siedzą w akademii, piszą artykuły

> i próbują siebie wpakować w jakiś taki wąski fragment i zbudować sobie jakieś swoje audytorium i to jest zbiór czasopism i konferencji dla big data, to jest konferencja dla ludzie data science. Natomiast jak się popatrzy na rzeczywisty problem, gdzie przychodzi jakaś firma i mówi taki, siaki, owaki problem do rozwiązania, to tam i tak wszystko wraca do jednego wielkiego worka, bo się okazuje, że jest big data, bo się okazuje, że są ogromne wolumeny danych, jest potrzebny taki niskopoziomowy feature engineering i tam trzeba po prostu usiąść i zakasać rękawy i programować. I jest potrzebny ktoś, kto ładnie to zwizualizuje, i jest potrzebny ktoś, kto odpali eksperymenty i jest potrzebny ktoś, kto zna się na przetwarzaniu równoległym, no bo inaczej to się nie skończy itd. Potem to i tak wszystko wraca do jednego dużego kubka. Myślę, że nie ma co specjalnie się tym ekscytować, zawsze będzie kolejne słowo, teraz jest reinforcement learning[24] jako pomysł, że teraz **wszystko** będziemy w ten sposób uczyli, niezależnie od tego, czy to ma sens, czy nie ma sensu. Za chwilę znowu pewnie wybuchną algorytmy ewolucyjne, bo ludzie już kombinują nad tym. Myślę, że to jest bardziej szum informacyjny, niż jakąś prawdziwa wartość jest w tej ewolucji terminów. {wywiad 23}

Technologie, które mają ułatwiać korzystanie z języków programowania, oferują wiele udogodnień w zakresie wykonywania działania podstawowego (pisania kodu do operacji na danych) i technicznych działań wspomagających (instalacji i zarządzania środowiskiem, publikacji wyników, kontroli wersji). O ile jednak **udogodnienia do działań wspomagających mogą przyjmować formę interfejsu graficznego do wyklikania operacji, to udogodnienia do działania podstawowego takiej formy nie przyjmują**. Stanowi to dla mnie wyraźny wskaźnik tego, że pisanie kodu jest immanentną częścią działania podstawowego w badanym świecie, moje ujęcie działania podstawowego jako „pisania kodu do przetwarzania, analizy i modelowania danych" oddaje zatem perspektywę uczestników badanego świata.

Stąd, że działanie podstawowe jest wykonywane za pomocą kodu, a nie w interfejsach graficznych, biorą się w świecie DS (nie tylko w Polsce) żarty dotyczące używania niezwykle popularnego Excela do pracy z danymi, przykłady używania tej aplikacji w sposób niezgodny z przeznaczeniem – np. jak bazy danych czy interfejsu użytkownika, powtarzające się listy problemów z tego typu narzędziami (Smart, 2013; Gutierrez, 2014: 110; Hamideh, 2015; Wickham, 2018d). Chodzi nie tylko o Excela, ale stał on się w badanym świecie synonimem arkusza kalkulacyjnego w ogóle, więc będę używać tego określenia – tak samo do działania podstawowego nie służą np. programy Google Sheets, LibreOffice Calc ani Apple Numbers. Jeden z rozmówców {wywiad 9} pokazał w trakcie wywiadu mem, w którym Python i R walczą, ale zgadzają się co do tego, że „nienawidzą" Excela (rys. 5.7).

24 Oznacza ML ze wzmocnieniem (*reinforcement learning*), stosowane np. w AlphaGo z 2015 r. W uproszczeniu polega na tym, że model gra w grę przeciwko sobie (por. Silver i in., 2016; 2017; 2018).

Rysunek 5.7. Co do tego jesteśmy zgodni: Python i R walczą, ale obaj nienawidzą Excela – mem internetowy

Źródło: https://www.facebook.com/Rmemes0/photos/a.1230204967031792/2078838445501769/ (dostęp: 9.06.2018).

Uczestnicy badanego świata podają wiele argumentów technicznych dotyczących tego, że nie da się wykonywać działania podstawowego w arkuszach kalkulacyjnych. **Używanie np. Excela jako podstawowego narzędzia pracy odróżnia uczestników świata DS od innych osób pracujących z danymi.**

Excel jest popularnym, ograniczonym i jednym z najprostszych narzędzi do pracy z danymi. Data scientista nie używa Excela do wykonywania działania podstawowego, tak jak kucharz w restauracji klasy premium nie przyrządza posiłków z użyciem plastikowego noża, może jednak bez uszczerbku na profesjonalnym wizerunku użyć go na biwaku z rodziną. Fanatyczne unikanie lub krytykowanie arkuszy kalkulacyjnych będzie w badanym świecie raczej postrzegane negatywnie, a także jako cechujące wannabesów lub początkujących, chcących w niewłaściwy sposób legitymizować autentyczność swojego uczestnictwa w świecie DS. Definiowanie np. stanowiska pracy w ofercie rekrutacyjnej jako „data scientist" przy wymaganiu przede wszystkim zaawansowanej znajomości Excela jest uważane za tzw. *fake data science*, czyli podszywanie się pod świat DS – „[Excel] to nie jest poważne narzędzie" {wywiad 12}.

Excel do wykonywania działania podstawowego w świecie DS jest uznawany za narzędzie nieodpowiednie, nieprzydatne, niewygodne, nieefektywne, mało sprawcze, nieciekawe (ponieważ od wielu lat względnie niezmienne) i bardzo ograniczone. **Excel to nie jest narzędzie techniczne, to narzędzie biznesowe.** W projektach DS jedynym akceptowanym w badanym świecie użyciem Excela jest komunikowanie się z biznesem czy ogólnie z osobami nietechnicznymi. Istnieje książka do nauki podstaw ML, adresowana dla osób nietechnicznych, nieumiejących programować, gdzie przykłady i ćwiczenia z modelowania wykonane są w arkuszach kalkulacyjnych (Foreman, 2017: 17). Status Excela w świecie DS jest zatem bliski PowerPointowi, czyli oprogramowaniu z tzw. pakietów biurowych typu MS Office, a nie językom Python lub R.

Zakończenie

W niniejszej książce dokonałem obszernego opisu etnograficznego polskiego społecznego świata data science, co stanowiło jej cel główny.

Na podstawie rekonstrukcji tworzonych przez uczestników świata DS definicji działania podstawowego zaproponowałem własne ujęcie, realizując pierwszy z celów szczegółowych. Działaniem podstawowym uczestników badanego świata jest, według mnie, pisanie kodu do przetwarzania, analizy i modelowania danych.

Przedstawiłem własne rozumienie roli technologii umożliwiających wykonanie działania podstawowego w odniesieniu do zachodzących w społecznym świecie procesów, m.in. wyznaczania granic społecznego świata, legitymizacji i zaświadczania o autentyczności oraz segmentacji – profesjonalizacji. W ten sposób zrealizowany został drugi cel szczegółowy. W odniesieniu do technologii granice społecznego świata DS ogólnie wyznacza wykonywanie działania podstawowego w językach programowania: przede wszystkim w Pythonie i w mniejszym stopniu w R. Bardziej szczegółowo, ale pozostając w sferze języka Python, można wskazać wiele narzędzi i metod uznawanych w badanym świecie za właściwe. Jest to np. Anaconda – dystrybucja Pythona do zastosowań DS, a wraz z nią notatnik Jupyter i repozytorium conda. Do przetwarzania, analizy danych służą m.in. pakiety pandas, NumPy, matplotlib, a także narzędzie Apache Spark, czyli zunifikowana technologia do rozproszonego przetwarzania danych. Do modelowania metodami tzw. tradycyjnego ML używa się najczęściej pakietu scikit-learn. W przypadku metod DL za właściwe uznawane są pakiety Tensor Flow (TF), PyTorch oraz narzędzie Keras, ułatwiające korzystanie z TF. Zidentyfikowałem także dyskursywne odgraniczanie się od obszarów bliskich DS poprzez wskazywanie na odmienne wartości. W ten sposób np. obszar IT określany jest jako nudny (bardziej inżynieryjny), a DS jako ciekawy (bardziej eksperymentalny); nauka akademicka uznawana jest za „sztukę dla sztuki", natomiast data science za sprawcze. Zaświadczanie o autentyczności odbywa się w badanym świecie, rzecz jasna, poprzez odwołanie do używania właściwych narzędzi i metod. Zidentyfikowałem technologie przemilczane, a niezbędne do pełnego uczestnictwa w świecie DS – np. podstawy języków SQL i bash. Inne praktyki zaświadczania o autentyczności to powołanie się na doświadczenie w rozwiązywaniu realnych problemów i cechy osobiste, np. bycie zainteresowanym matematyką lub informatyką od dziecka. Ogólnie zaświadczanie

o autentyczności uczestnictwa dokonywane jest poprzez kategorię bycia osobą techniczną.

Według mnie ktoś, kto uosabia wartości świata społecznego, powinien być uznawany za autentycznego uczestnika tego świata. Opisałem także praktyki prowadzące do bycia uznanym za uczestnika silnie osadzonego, osiągającego poziom maestrii – np. samodzielne budowanie narzędzi do DS oraz agnostycyzm technologiczny i metodologiczny.

Wskazałem ogólną legitymizację świata DS, jaką jest narracja o jego powstaniu – DS powstało, ponieważ z uwagi na gwałtownie rosnącą ilość dostępnych danych cyfrowych i rozwój mocy obliczeniowej komputerów, przy jednoczesnym spadku ich kosztów (zarówno danych, jak i mocy), możliwe, a zarazem opłacalne stało się rozwiązywanie realnych problemów z wykorzystaniem metod ML. Opisałem tworzące się subświaty/segmenty świata DS, głównie w kategoriach profesjonalizacji i zmiany zakresu wybranych aspektów działania podstawowego. Wyróżniłem inżynierię danych (*data engineering*), badania nad uczeniem maszynowym (*ML research*), inżynierię uczenia maszynowego (*ML engineering*) i analitykę danych. Zarówno działania podstawowe, jak i właściwe technologie są dla nich inne niż dla świata DS.

Rozpoznałem i opisałem wartości uczestników świata DS – efektywność, sprawczość, samorozwój, samodzielność, ciekawość, swobodę i racjonalność, realizując trzeci z celów szczegółowych. Za centralną wartość badanego świata uznałem efektywność. Są to wartości merytokratyczne, charakterystyczne dla późnonowoczesnego neoliberalizmu i wywodzące się z jednej strony z tzw. światopoglądu technokratycznego, z drugiej zaś z etosu naukowca akademickiego według Mertona. Pokazałem, w jaki sposób w tych siedmiu wyróżnionych kategoriach aksjonormatywnych oceniane są działania uczestników świata DS. W kategoriach wartości oceniane i budowane są także technologie używane w tym świecie. Za wyraz wartości uznaję preferencje i wybory technologiczne. W dużej mierze w kategoriach wartości opisałem także zachodzące w świecie DS procesy – szczególnie areny proponuję postrzegać jako spory o wartości.

Realizując czwarty cel szczegółowy, określiłem i opisałem główne areny sporów dla społecznego świata DS. Za szeroką arenę uważam modelowanie, z kolei model ML potraktowałem jako obiekt graniczny, wokół którego toczy się spór o władzę, zasoby, wartości, o przetrwanie i rozwój każdego z zaangażowanych światów. W tę arenę zaangażowane są społeczne światy DS, biznesu, akademii, IT, mediów, polityki i prawa. Dostęp do obiektu granicznego (szczególnie jeżeli zawęzimy jego ujęcie do produkcyjnie wdrażanych modeli ML) mają światy techniczne, czyli DS i IT, oraz te światy nietechniczne, które dysponują władzą nad wpływowymi aktorami na arenie. Jest to świat biznesu (aktor – pieniądze) i świat prawa (regulacje prawne). Podkreślam rolę gigantów technologicznych, takich jak Google, Facebook, Amazon, we wspieraniu rozwoju ważnych na arenie technologii, także tych „wolnych" (*open source*). Dość znaczące na arenie są też: świat polityki – jako

najbardziej sprawczy w zakresie kształtowania regulacji prawnych, oraz obszar nauk ścisłych i przyrodniczych – jako subświat nauki akademickiej, kształtujący metody ML, publikujący teksty i pakiety dotyczące modelowania. Jednak te światy nie mają bezpośredniego dostępu do obiektu granicznego. Jeszcze mniej wpływowe na arenie są światy mediów oraz obszar nauk społecznych i humanistycznych, będące subświatami nauki akademickiej, do którego sam przynależę.

Podkreśliłem różnice w dyskursach zaangażowanych w arenę światów. Światy DS, IT oraz subświat nauk ścisłych i przyrodniczych traktują obiekt graniczny w kategorii „uczenia maszynowego". Świat prawa posługuje się zarówno tą kategorią, jak i „sztuczną inteligencją". Pozostałe światy posługują się głównie nietechnicznym pojęciem „sztucznej inteligencji", stanowiącym opakowanie dla modeli ML (i wielu innych technologii). Te światy pozostają pod wpływem praktyk celowo stosowanych przez biznesowych rzeczników świata DS i IT (głównie marketing i HR), częściowo także pod wpływem popkulturowych ujęć AI, które można uznać za tło dla areny. Uczestnicy świata DS postrzegają realizowane przez siebie projekty, których małą częścią jest budowanie modeli ML, jako majsterkowanie, podczas gdy z perspektywy nietechnicznej zarówno ich praca, jak i samo działanie modeli ML jest przedstawiane i widziane w kategorii magii. Stąd popularny w świecie DS żart: „jeżeli to jest napisanie w Pythonie, to prawdopodobnie uczenie maszynowe; jeżeli jest napisanie w PowerPoincie, prawdopodobnie to AI".

Zarówno użytkownicy końcowi systemów bazujących na modelach ML, osoby wykonujące nisko płatne i niewykwalifikowane zadania (tzw. klikacze mogą np. etykietować dane źródłowe na potrzeby trenowania modeli nadzorowanego ML), jak i środowisko naturalne, obciążane energochłonną infrastrukturą obliczeniową (a w wybranych przypadkach marketingowo zwaną chmurą) to aktorzy słabo obecni na arenie, traktowani nierzadko przedmiotowo, niczym zasoby do zużywania.

Jako nieco węższą postrzegam arenę dotyczącą sytuacji kobiet w badanym świecie społecznym. W Polsce nie zauważyłem, aby problematyzowano sytuacje innych mniejszości. Kobiety stanowią 10–20% osób uczestniczących w badanym świecie, co oznacza, że kiedy w zespole DS pracuje kobieta, to niemal zawsze jest to jedyna kobieta wśród kilku mężczyzn. Opisałem spór o wartości pomiędzy dwiema pozycjami w świecie DS – pozycją „doskonałej merytokracji" a pozycją „różnorodności". Pierwsza z nich zakłada doskonałą merytokrację, w której nie liczą się takie cechy jak płeć, a wyłącznie kompetencje techniczne i metodologiczne konkretnych osób. Druga – bardziej poprawna politycznie i wyraźniej widoczna pozycja „różnorodności" – afirmuje włączanie do komercyjnego DS osoby reprezentujące mniejszości. Takie zespoły mają budować lepsze produkty i realizować lepsze projekty, bo wypracowują rozwiązania jakoby właśnie bardziej różnorodne, nieobciążone (*in vivo* „zbiasowane") w kierunku perspektywy młodych, białych mężczyzn – grupy ilościowo bezwzględnie dominującej.

Zaprezentowałem także arenę dotyczącą używanych w świecie DS narzędzi, szczególnie spór o konkurencyjność, komplementarność i rozłączność zastosowań

języków Python i R. Przedstawiłem różnice historyczne w rozwoju tych języków i dzisiejsze różnice w ich zastosowaniach. Ogólnie język Python jest obecnie w badanym świecie uważany za pierwszy wybór, jest językiem ilościowo znacznie bardziej popularnym – w Polsce może być on używany w około 87% firm zajmujących się projektami DS. W około 38% jest to język R, a w co najmniej 25% wykorzystywane są oba języki. Ze względu na cechy samego języka i dostępne pakiety Python uznawany jest za lepszy do modelowania metodami ML (szczególnie do DL), natomiast R do statystyki i analizy danych. Mniej więcej od 2018 roku Python uznawany jest, przynajmniej w komercyjnej sferze działalności badanego świata, za domyślny język do wykonywania działania podstawowego. Język R należałoby poznać, czy chociażby potrafić się posłużyć nim jako drugim, w konkretnych zastosowaniach, np. analitycznych czy statystycznych, posługując się już swobodnie Pythonem i stając się silnie osadzonym uczestnikiem świata DS. Spór stracił zatem na znaczeniu, a w ogóle przesadna koncentracja na jednej tylko technologii jest w badanym świecie uznawana za coś, co cechuje początkujących uczestników. Ci silnie osadzeni charakteryzują się raczej agnostycyzmem technicznymi i metodologicznym, czyli dość swobodnie dobierają narzędzia i metody do rozwiązywanego problemu. Opisywana arena jednoznacznie definiuje uczestników świata DS jako osoby wykonujące działanie podstawowe za pomocą kodu. Narzędzia analityczne bazujące na interfejsach graficznych, jak np. Excel, są uznawane za nietechniczne, niewłaściwe do działania podstawowego DS.

Zrealizowałem również piąty cel szczegółowy, czyli ujęcie pełnej sytuacji badania. Łącznie składają się na nie rozdziały: drugi – poświęcony przeglądowi literatury akademickiej, gdyż naukowców uważam za zaangażowanych w arenę; trzeci – gdyż moje badania wraz z ich ramą teoretyczną i metodologią stanowią wyraz właśnie takiego zaangażowania, a ten rozdział pokazuje moje usytuowanie w odniesieniu do świata DS; i na końcu rozdziały czwarty i piąty – jako prezentację wyników moich badań i analiz.

Dwa zagadnienia zostały w tej książce przemilczane (*sites of silence*, por. Clarke 2005: 85): szczegóły dotyczące metod stosowanych w DS oraz procesy wychodzenia uczestników z badanego społecznego świata. Pierwsze zagadnienie pominąłem świadomie, ponieważ zagłębienie się w szczegóły matematyczne stosowanych metod byłoby dla mnie niewspółmiernie pracochłonne w stosunku do efektu w postaci głębszego zrozumienia badanego świata, a więc i założonych celów. Pozostałem przy doraźnych, raczej intuicyjnych wyjaśnieniach, np. czym różni się ML nadzorowane od nienadzorowanego, co to jest propagacja wsteczna itp. Zdecydowałem się zagłębić bardziej szczegółowo w technologie świata DS jako narzędzia, a nie metody, bo moim zdaniem narzędzia są znacznie silniej obecne w dyskursie tego świata. Z uwagi na uzyskany przeze mnie, właśnie głównie przez pryzmat narzędzi, wgląd w elementy i procesy w społecznym świecie DS ta decyzja wydaje mi się właściwa. Zagadnienie wychodzenia ze świata DS pominąłem nieświadomie. W toku własnych badań i analiz nie spotkałem się z takim procesem ani nie

wpadłem na pomysł, aby go poszukać. Mogę powiedzieć jedynie o wychodzeniu ze świata w kontekście procesów segmentacji i profesjonalizacji, np. przechodzenia z definiowania się jako data scientisty do inżyniera uczenia maszynowego (*ML engineer*). Nie znam jednak praktyk wychodzenia „na zewnątrz" badanego świata, a bez wątpienia takie istnieją.

Ufam, że moja praca może przyczynić się do zmian w szerokim dyskursie nietechnicznym dotyczącym tego, co związane jest z badanym przeze mnie społecznym światem DS, a nazywane np. chmurą, big data, sztuczną inteligencją. Proponuję cztery stwierdzenia, stanowiące końcowe podsumowanie moich wniosków:

1) data scientista to nie programista, a ktoś, kto pisze kod do przetwarzania, analizy i modelowania danych;
2) dane i obliczenia w chmurze to dane i obliczenia na cudzym komputerze, a rzędy takich komputerów – serwerów w betonowych halach nie mają w sobie nic zwiewnego, potrzebują stałego zasilania energią elektryczną, chłodzenia wodą i ludzkiej obsługi;
3) „big data" to nieco przestarzałe określenie technologii rozproszonego i równoległego składowania i przetwarzania danych, jak np. Apache Hadoop i Spark, a nie inwigilacja dużych grup ludzi ani nie coś, co odmieni oblicze nauki i przyniesie śmierć ekspertom dziedzinowym;
4) sztuczna inteligencja/AI nie istnieje – istnieją różnego rodzaju automaty, a sami twórcy komercyjnie stosowanych modeli ML, czyli jakoby najbardziej inteligentnych, bo wyszukujących prawidłowości w zbiorach danych cyfrowych części systemów zwanych AI, uznają „sztuczną inteligencję" przede wszystkim za termin marketingowy, mający oczarowywać klientów; pojęcie „AI" jest z punktu widzenia uczestników świata DS tak wieloznaczne i budzące emocje, że w sensie technicznym nie da się zaakceptować jego użycia.

Są to dyskursywne postulaty – moim zdaniem nie warto używać określeń: „chmura", „big data", „AI" – chyba że w kontekście krytycznym. Stanowią one opakowania dla wielu skomplikowanych technologii, o które jako naukowcy, konsumenci i obywatele powinnyśmy spierać się jak najbardziej merytorycznie. Patrząc tylko na opakowania, pozostaniemy zaczarowani na powierzchni problemu. Najbardziej sprawcze decyzje techniczne zapadną bez naszego udziału i zostaną wdrożone przez podmioty posiadające kompetencje, infrastrukturę i kapitał – Google, Facebook, Amazon, Tencent czy innego giganta technologicznego.

Socjologia od początku swego istnienia jest nauką o nowoczesności. August Comte uznał, że kapitalizm, będący reżimem ekonomicznym, jest tym, co ukształtowało nowoczesność. W końcu drugiej dekady XXI wieku nie zmieniło się wiele. Późną nowoczesność – w rozumieniu Giddensa, Lasha i Becka – w dużej mierze kształtuje neoliberalny kapitalizm. Stosowanie nawet najbardziej zaawansowanych technologicznie i metodologicznie automatyzacji, tzw. AI, nie jest zatem rewolucją. Nawet kiedy wzrośnie skala stosowania tego rodzaju technologii i zmieni się struktura zawodowa rynku pracy, to zapewne będzie nieomal taki sam

tryumfujący kapitalizm – wciąż napędzany pozytywizmem, choć obecnie mówi się o napędzaniu biznesu danymi (*data-driven*), a jeszcze niedawno o rewolucji big data. Stąd mogłoby się wydawać, że zasadniczo nic nie zagraża przetrwaniu i dalszemu rozwojowi opisanego w tej książce świata DS w Polsce – dopóki będzie on zatrudniony w służbie kapitalizmowi. Uczestnicy polskiego świata DS widzą swój świat, szczególnie w kategoriach branży i rynku na usługi DS, jako obszar na wczesnym etapie rozwoju. Polscy klienci powoli i nieufnie wchodzą na rynek, a ulokowane w Polsce zespoły DS czy freelancerzy pracują w znakomitej większości dla klientów z Europy Zachodniej i USA. Uczestnicy spodziewają się zatem dalszego rozwoju i stabilizacji świata DS w wielu wymiarach (np. edukacji, technologii, prawa, rynku). Spodziewają się także spadku entuzjazmu wobec AI i przesunięcia postrzegania ich pracy w stronę IT, w stronę technologii tak mało magicznych jak strony internetowe.

Źródła potencjalnej „rewolucji", czyli zmiany społecznej, upatruję gdzie indziej. Pandemia COVID-19 pokazała, że sprawczość racjonalna i sprawczość pieniądza są zawodne. Wirus, czyli coś, co nie ma nawet DNA, nie poddaje się przecież temu porządkowi. Można rozumieć to szeroko, ja powiem jedynie, że wiele systemów opartych na modelach ML przestało działać zgodnie z oczekiwaniem, dane wejściowe przestały bowiem przypominać te, na jakich trenowano modele przed przełomem 2019/2020 roku. Być może zmieni to kierunek rozwoju DS, o ile neoliberalny, scjentystyczny, efektywnościowy paradygmaty działania tego świata będzie w wymiarze globalnym tracił na znaczeniu. Być może obserwujemy właśnie początek końca neoliberalizmu, bazującego na niekończącym się wzroście, egoizmie i konkurencji, na rzecz systemu społecznego opartego na zrównoważonym rozwoju i współpracy.

Aneks

Spis badań i analiz własnych

Spis wywiadów swobodnych – badania własne

Nr wywiadu	Termin realizacji	Długość nagrania [minuty]
1	21.10.2016	28
2	21.06.2018	47
3	4.07.2018	46
4	9.07.2018	45
5	10.07.2018	46
6	25.07.2018	62
7	25.07.2018	55
8	26.09.2018	52
9	26.09.2018	74
10	1.10.2018	57
11	5.10.2018	74
12	30.10.2018	51
13	30.10.2018	57
14	4.01.2019	40
15	5.01.2019	67
16	9.01.2019	55
17	13.01.2019	69
18	4.03.2019	68
19	9.03.2019	106
20	3.04.2019	68
21	24.04.2019	135
22	21.05.2019	97
23	27.05.2019	88
24	27.05.2019	82
25	28.05.2019	108
26	30.05.2019	167

Źródło: opracowanie własne.

Spis obserwacji uczestniczących – badania własne

Nr obserwacji	Termin realizacji	Długość obserwacji [godziny]	Rodzaj obserwacji	Czy online?
1	5.10.2016–5.05.2017	10	Kurs	–
2	7.11.2016	2	Wykład	–
3	3.12.2016–20.05.2017	40	Kurs	Tak
4	12.12.2016	1	Meetup	–
5	22.02.2017	1	Meetup	–
6	28.03.2017	1,5	Meetup	–
7	24.04.2017	2	Warsztaty	Tak
8	2.06.2017–11.03.2019	12	Kurs	Tak
9	27–29.09.2017	22	Konferencja/warsztaty	–
10	19–21.10.2017	2	Konferencja	–
11	16.11.2017	1	Wykład	-
12	12.12.2017	6	Konferencja	–
13	13.12.2017	2	Meetup	–
14	15.12.2017	2	Konferencja	–
15	15.01.2018	1,5	Meetup	–
16	5.02–15.09.2018	45	Kurs	Tak
17	2–29.03.2018	1	Kurs	Tak
18	2.03.2018	1,5	Konferencja	–
19	6.04.2018	3	Wykład	–
20	17.04.2018	1	Inne	Tak
21	17.04.2018	1	Inne	Tak
22	18.04.2018	8	Konferencja	–
23	25.04.2018	2	Konferencja	–
24	26.04.2018	6	Warsztaty	–
25	21.06.2018	2	Meetup	–
26	2–5.07.2018	28	Konferencja	–
27	27.08.2018	0,5	Inne	Tak
28	16.10.2018	1,5	Meetup	–
29	30.10.2018	1,5	Meetup	–
30	18.11.2018	6	Warsztaty	–
31	19–22.11.2018	4	Kurs	Tak
32	6.12.2018	1	Kurs	Tak
33	11.12.2018	1,5	Meetup	–
34	14.01.2019	2	Kurs	Tak
35	11–15.02.2019	10	Kurs	Tak
36	12.02.2019	2	Meetup	–
37	27.02.2019	7	Konferencja	–
38	4.04.2019	2	Meetup	–

Nr obserwacji	Termin realizacji	Długość obserwacji [godziny]	Rodzaj obserwacji	Czy online?
39	8–12.04.2019	10	Kurs	Tak
40	13.04.2019	7	Konferencja	–
41	2.-15.05.2019	5	Kurs	Tak
42	8.05.2019	1,5	Meetup	–
43	13.-17.05.2019	10	Kurs	Tak
44	16.05.2019	5	Warsztaty	–
45	18.-31.05.2019	5	Kurs	Tak
46	14.06.2019	8	Konferencja	–

Źródło: opracowanie własne.

W analizach użyto pakietów GNU R:

1) cowplot (Wilke, 2019);
2) ggrepel (Slowikowski, 2019);
3) jsonlite (Ooms, 2014);
4) lubridate (Grolemund, Wickham, 2011);
5) maptools (Bivand, Lewin-Koh, 2019);
6) PerformanceAnalytics (Peterson, Carl, 2020);
7) RColorBrewer (Neuwirth, 2014);
8) readODS (Schutten i in., 2020);
9) rgeos (Bivand, Rundel, 2019);
10) rmarkdown (Xie, Allaire, Grolemund, 2018; Allaire i in., 2019);
11) scales (Wickham, 2018c);
12) sf (Pebesma, 2018);
13) tidyverse (Wickham i in., 2019).

Kod w języku R, umożliwiający powtórzenie całości analiz i wizualizacji, dostępny jest publicznie na koncie GitHub autora:

1) dane o meetupach DS w Polsce – https://github.com/zremek/meetup-harvesting;
2) dane z sondaży „środowiskowych" DS – https://github.com/zremek/survey-polish-data-science;
3) dane z Google Trends – https://github.com/zremek/google-trends-ds-ml-ai;
4) wizualizacja wyników analizy jakościowej (mapa pozycyjna) – https://github.com/zremek/qualitative-maps.

Bibliografia

Abadi Martín, Barham Paul, Chen Jianmin, Chen Zhifeng, Davis Andy, Dean Jeffrey, Devin Matthieu, Ghemawat Sanjay, Irving Geoffrey, Isard Michael, Kudlur Manjunath, Levenberg Josh, Monga Rajat, Moore Sherry, Murray Derek G., Steiner Benoit, Tucker Paul, Vasudevan Vijay, Warden Pete, Wicke Martin, Yu Yuan, Zheng Xiaoqiang, Google Brain (2016), *TensorFlow: A system for large-scale machine learning*, 12th USENIX Symposium on Operating Systems Design and Implementation, USENIX, Savannah.

Abriszewski Krzysztof (2010), *Wszystko otwarte na nowo. Teoria Aktora-Sieci i filozofia kultury*, Wydawnictwo Naukowe Uniwersytetu Mikołaja Kopernika, Toruń.

ActiveState (2018), *Python: A Lingua Franca*, https://www.activestate.com/wp-content/uploads/2018/10/Python-Lingua-Franca-Whitepaper-2018_1.pdf (dostęp: 2.01.2019).

Adams Scott (2012), *Dilbert*, http://dilbert.com/strip/2012-07-29 (dostęp: 12.10.2016).

Afeltowicz Łukasz, Pietrowicz Krzysztof (2008), *Koniec socjologii, jaką znamy, czyli o maszynach społecznych i inżynierii socjologicznej*, „Studia Socjologiczne", nr 3(190), s. 43–73.

Agarwal Rahul (2019), *A Layman guide to moving from Keras to Pytorch*, https://mlwhiz.com/blog/2019/01/06/pytorch_keras_conversion/ (dostęp: 2.05.2019).

AI Now Institute (2018), *The AI Now Institute*, https://ainowinstitute.org/ (dostęp: 14.06.2018).

Alekseichenko Vladimir (2018), *Sztuczna inteligencja, cyberbezpieczeństwa i hackerzy*, http://biznesmysli.pl/sztuczna-inteligencja-cyberbezpieczenstwa-i-hackerzy/ (dostęp: 20.06.2018).

Alekseichenko Vladimir (2019a), *10 mitów o sztucznej inteligencji*, http://biznesmysli.pl/10-mitow-o-sztucznej-inteligencji/ (dostęp: 29.04.2019).

Alekseichenko Vladimir (2019b), *Drony zmieniają branże ubezpieczeń, budowlaną i inne*, https://biznesmysli.pl/drony-zmieniaja-branze-ubezpieczen-budowlana-i-inne/ (dostęp: 11.01.2019).

Alekseichenko Vladimir, Borowiecki Łukasz, Chojecki Przemysław, Czapska Martyna, Kowalczyk Witold, Mieczkowski Piotr, Pietrzak Piotr, Siudak Robert, Sztokfisz Barbara, Rachwalski Hubert (2018), *Przegląd strategii rozwoju*

sztucznej inteligencji na świecie, Fundacja Digital Poland, Warszawa, https://www.digitalpoland.org/assets/publications/przegląd-strategii-rozwoju-sztucznej-inteligencji-na-swiecie/przeglad-strategii-rozwoju-ai-digitalpoland-report.pdf (dostęp: 11.01.2019).

Allaire J.J. (2012), *RStudio: Integrated Development Environment for R*, [w:] *The R User Conference, useR! 2011*, University of Warwick, Warwick, s. 14, https://www.r-project.org/conferences/useR-2011/abstracts/180111-allairejj.pdf (dostęp: 11.01.2019).

Allaire J.J., Ushey Kevin, Tang Yuan (2018), *reticulate: Interface to "Python"*, https://cran.r-project.org/package=reticulate (dostęp: 11.01.2019).

Allaire J.J., Xie Yihui, McPherson Jonathan, Luraschi Javier, Ushey Kevin, Atkins Aron, Wickham Hadley, Cheng Joe, Chang Winston, Iannone Richard (2019), *rmarkdown: Dynamic Documents for R*, https://rmarkdown.rstudio.com (dostęp: 11.01.2019).

Amatriain Xavier, Basilico Justin (2012), *Netflix Recommendations: Beyond the 5 stars (Part 1)*, https://medium.com/netflix-techblog/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429 (dostęp: 9.07.2019).

Ameisen Emmanuel (2018), *How to deliver on Machine Learning projects: A guide to the ML Engineering Loop*, https://blog.insightdatascience.com/how-to-deliver-on-machine-learning-projects-c8d82ce642b0 (dostęp: 25.10.2018).

Anaconda (2018a), *Anaconda Distribution*, https://www.anaconda.com/distribution/ (dostęp: 3.12.2018).

Anaconda (2018b), *Travis Oliphant*, https://www.anaconda.com/people/travis-oliphant (dostęp: 3.12.2018).

Anaconda (2019a), *Anaconda Distribution Starter Guide*, https://docs.anaconda.com/_downloads/9ee215ff15fde24bf01791d719084950/Anaconda-Starter-Guide.pdf (dostęp: 4.12.2018).

Anaconda (2019b), *The end-to-end data science platform*, https://www.anaconda.com/enterprise/ (dostęp: 22.07.2019).

Anaconda (b.d.), *NumFOCUS – Anaconda*, https://www.anaconda.com/numfocus/ (dostęp: 3.12.2018).

Anderson Chris (2008), *The End of Theory: The Data Deluge Makes the Scientific Method Obsolete*, https://www.wired.com/2008/06/pb-theory/ (dostęp: 14.11.2017).

Anderson Ken, Nafus Dawn, Rattenbury Tye, Aipperspach Ryan (2009), *Numbers Have Qualities Too: Experiences with Ethno-Mining*, „Ethnographic Praxis in Industry Conference Proceedings", no. 1, s. 123–140, https://doi.org/10.1111/j.1559-8918.2009.tb00133.x

Andrus Calvin, Cook Jon, Sood Suresh (2017), *Data Science: An Introduction*, https://en.wikibooks.org/wiki/Data_Science:_An_Introduction (dostęp: 21.03.2018).

Angrosino Michael (2010), *Badania etnograficzne i obserwacje*, Wydawnictwo Naukowe PWN, Warszawa.

Angwin Julia, Larson Jeff, Mattu Surya, Kirchner Lauren (2016), *Machine Bias: There's software used across the country to predict future criminals. And it's biased against blacks*, https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing (dostęp: 30.04.2018).

Arakelyan Sophia (2017), *Tech giants are using open source frameworks to dominate the AI community*, https://venturebeat.com/2017/11/29/tech-giants-are-using-open-source-frameworks-to-dominate-the-ai-community/ (dostęp: 17.07.2019).

Arena Michael J., Pentland Alex, Price David (2010), *Honest Signals – Hard Measures for Social Behavior*, „Organization Development Journal", vol. 28(3), s. 11–20.

Ariely Dan (2013), *@danariely: Big data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everyone…*, https://twitter.com/danariely/status/287952257926971392 (dostęp: 14.02.2018).

Ashcraft Mark H. (2002), *Math Anxiety: Personal, Educational, and Cognitive Consequences*, „Current Directions in Psychological Science", vol. 11(5), s. 181–185, https://doi.org/10.1111/1467-8721.00196

Associated Press (2015), *Big Brother is watching: how China is compiling computer ratings on all its citizens*, https://www.scmp.com/news/china/policies-politics/article/1882533/big-brother-watching-how-china-compiling-computer (dostęp: 21.01.2019).

Avram Abel (2013), *Docker: Automated and Consistent Software Deployments*, https://www.infoq.com/news/2013/03/Docker/ (dostęp: 23.01.2019).

Awad Edmond, Dsouza Sohan, Kim Richard, Schulz Jonathan, Henrich Joseph, Shariff Azim, Bonnefon Jean-François, Rahwan Iyad (2018), *The Moral Machine experiment*, „Nature", vol. 563(7729), https://doi.org/10.1038/s41586-018-0637-6

Awad Edmond, Levine Sydney, Kleiman-Weiner Max, Dsouza Sohan, Tenenbaum Joshua B., Shariff Azim, Bonnefon Jean-François, Rahwan Iyad (2020), *Drivers are blamed more than their automated cars when both make mistakes*, „Nature Human Behaviour", vol. 4(2), s. 134–143, https://doi.org/10.1038/s41562-019-0762-8

Azam Anum (2014), *The First Rule of Data Science*, „Berkeley Science Review", http://berkeleysciencereview.com/article/first-rule-data-science/ (dostęp: 26.01.2018).

Azevedo Ana, Santos Manuel Filipe (2008), *KDD, SEMMA and CRISP-DM: a parallel overview*, „IADIS European Conference Data Mining", January, s. 182–185.

Baath Rasmus (2012), *The State of Naming Conventions in R*, „The R Journal", vol. 4(2), s. 74–75.

Babbie Earl (2006), *Badania społeczne w praktyce*, Wydawnictwo Naukowe PWN, Warszawa.

Baitu N.T. (2014), *What is a data scientist? 14 definitions of a data scientist!*, https://hub.biz/blog/what-is-data-scientist-14-definitions-data-scientist-8155414827284578184 (dostęp: 11.02.2018).

Barabási Albert-László (2002), *Linked: The New Science of Networks*, Perseus Publishing, Cambridge.
Barlow Mike (2013), *The Culture of Big Data*, O'Reilly, Tokio.
Barnet David (2017), *The robots are coming – but will they really take all our jobs?*, http://www.independent.co.uk/news/science/robots-are-coming-but-will-they-take-our-jobs-uk-artificial-intelligence-doctor-who-a8080501.html (dostęp: 14.02.2018).
Barry Dwight (2016), *Business Intelligence with R. From Acquiring Data to Pattern Exploration*, https://leanpub.com/businessintelligencewithr (dostęp: 17.02.2018).
Batorski Dominik (2004), *Sieci społeczne: Charakterystyka, uwarunkowania i konsekwencje struktur relacji społecznych na przykładzie komunikacji internetowej*, Uniwersytet Warszawski, Warszawa.
Batorski Dominik (2018), *Facebook. Wąż, który pożera własny ogon*, „Magazyn Opinii Pismo", 5 czerwca.
Bauman Zygmunt, Lyon David (2013), *Płynna inwigilacja. Rozmowy*, Wydawnictwo Literackie, Kraków.
Becker Howard (1974), *Art as Collective Action*, „American Sociological Review", vol. 39(6), s. 767–776.
Becker Howard (1986), *Doing Things Together*, Northwestern University Press, Evanston.
Beede David N., Julian Tiffany A., Langdon David, McKittrick George, Khan Beethika, Doms Mark E. (2011), *Women in STEM: A Gender Gap to Innovation*, „SSRN Electronic Journal", https://doi.org/10.2139/ssrn.1964782
Berman Jules J. (2013), *Principles of Big Data: Preparing, Sharing, and Analyzing Complex Information*, Elsevier, Waltham.
Bhatt Niraj (2013), *NoSQL, Big Data, and MapReduce*, https://nirajrules.wordpress.com/2013/05/ (dostęp: 13.02.2018).
Białko Michał (2005), *Sztuczna inteligencja i elementy hybrydowych systemów ekspertowych*, Wydawnictwo Uczelniane Politechniki Koszalińskiej, Koszalin.
Biecek Przemysław (2015a), *Pogromcy Danych. Przetwarzanie danych w programie R*, http://pogromcydanych.icm.edu.pl/ (dostęp: 28.12.2016).
Biecek Przemysław (2015b), *Pogromcy Danych. Wizualizacja oraz modelowanie danych*, Interdyscyplinarne Centrum Modelowania Matematycznego i Komputerowego Uniwersytetu Warszawskiego, Warszawa.
Biecek Przemysław (2015c), *PogromcyDanych: PogromcyDanych/DataCrunchers is the Masive Online Open Course that Brings R and Statistics to the People*, https://search.r-project.org/CRAN/refmans/PogromcyDanych/html/00Index.html (dostęp: 5.03.2018).
Biecek Przemysław (2016), *Odkrywać! Ujawniać! Objaśniać! Zbiór esejów o sztuce przedstawiania danych*, Fundacja Naukowa SmarterPoland.pl, Warszawa.
Biecek Przemysław (2017), *Przewodnik po pakiecie R*, Oficyna Wydawnicza GiS, Wrocław.

Biecek Przemysław (2018a), *Ceteris Paribus Plots – a new DALEX companion*, http://smarterpoland.pl/index.php/2018/06/ceteris-paribus-plots-a-new-dalex-companion/ (dostęp: 2.06.2018).

Biecek Przemysław (2018b), *CV – Przemysław Biecek*, http://biecek.pl/CV/ (dostęp: 11.08.2018).

Biecek Przemysław (2018c), *DALEX: Explainers for Complex Predictive Models in R*, „Journal of Machine Learning Research", vol. 19(84), s. 1–5.

Biecek Przemysław (2018d), *Który z nich zostanie najgorszym wykresem 2018?*, http://smarterpoland.pl/index.php/2018/12/najgorszy-wykres-2018/ (dostęp: 3.04.2019).

Biecek Przemysław (2018e), *RODO + DALEX, kilka słów o moim referacie na DSS*, http://smarterpoland.pl/index.php/2018/05/rodo-dalex-kilka-slow-o-moim-referacie-na-dss/ (dostęp: 2.06.2018).

Biecek Przemysław (2019a), *MDP: Model Development Process v. 0.1*, https://github.com/ModelOriented/DrWhy/blob/master/images/ModelDevelopmentProcess.pdf (dostęp: 3.07.2019).

Biecek Przemysław (2019b), *XAI or DIE*, https://www.slideshare.net/PrzemekBiecek/xai-or-die-at-data-science-summit-2019-149824427 (dostęp: 19.05.2020).

Biecek Przemysław (2020), *XAI Stories*, https://pbiecek.github.io/xai_stories/ (dostęp: 5.08.2020).

Biecek Przemysław, Burzykowski Tomasz (2020), *Explanatory Model Analysis. Explore, Explain, and Examine Predictive Models. With examples in R and Python*, CRC Press, https://pbiecek.github.io/ema/ (dostęp: 19.05.2020).

Biecek Przemysław, Kosiński Marcin (2017), *archivist: An R Package for Managing, Recording and Restoring Data Analysis Results*, „Journal of Statistical Software", vol. 82(11), https://doi.org/10.18637/jss.v082.i11

Big Data Borat (2013), *@BigDataBorat: Data science is statistics on Mac*, https://twitter.com/bigdataborat/status/372350993255518208 (dostęp: 19.02.2018).

Birhane Abeba (2020), *Algorithmic Colonization of Africa*, „SCRIPT-ed", vol. 17(2), s. 389–409, https://doi.org/10.2966/scrip.170220.389

Bivand Roger, Lewin-Koh Nicholas (2019), *maptools: Tools for Handling Spatial Objects*, https://cran.r-project.org/package=maptools (dostęp: 19.05.2020).

Bivand Roger, Rundel Colin (2019), *rgeos: Interface to Geometry Engine – Open Source ('GEOS')*, https://cran.r-project.org/package=rgeos (dostęp: 19.05.2020).

Blei David M., Ng Andrew, Jordan Michael I. (2003), *Latent dirichlet allocation*, „Journal of Machine Learning Research", no. 3, s. 993–1022, https://dl.acm.org/doi/10.5555/944919.944937

Blumenkrantz Deena (2018), *Slack Workspaces for Data Science*, https://medium.com/deena-does-data-science/all-the-slack-workspaces-for-data-science-323380abf8ba (dostęp: 5.09.2019).

Błaszczak Anita (2016), *Specjaliści big data będą wkrótce na wagę złota*, „Rzeczpospolita", 6 września.

Bobriakov Igor (2017), *Top 15 Python Libraries for Data Science in 2017*, https://medium.com/activewizards-machine-learning-company/top-15-python-libraries-for-data-science-in-in-2017-ab61b4f9b4a7 (dostęp: 3.02.2018).

Bomba Radosław (2015), *„Simowie" na wspak. Gra „This War of Mine" w perspektywie retoryki proceduralnej*, „Wielogłos", nr 3(25), s. 87–95.

Bomba Radosław (2017), *Bazodanowe interfejsy. Projektowanie interakcji z dużymi zasobami danych kulturowych*, „Kultura Popularna", nr 4(50), s. 64–73, https://www.ceeol.com/content-files/document-598587.pdf (dostęp: 5.09.2019).

Bornakke Tobias, Due Brian L. (2018), *Big-Thick Blending: A method for mixing analytical insights from big and thick data sources*, „Big Data & Society", vol. 5(1), s. 1–16, https://doi.org/10.1177/2053951718765026

Borowiecki Łukasz, Mieczkowski Piotr (2019), *Map of the Polish AI*, Fundacja Digital Poland, Warszawa, https://www.digitalpoland.org/assets/publications/mapa-polskiego-ai/map-of-the-polish-ai-2019-edition-i-report.pdf (dostęp: 22.04.2019).

Borowik Magdalena, Maśniak Leszek, Kroplewski Robert, Romaniec Hubert (2018), *Gospodarka oparta o dane – Przemysł+*, Ministerstwo Cyfryzacji Rzeczypospolitej Polskiej, Warszawa, https://www.gov.pl/cyfryzacja/gospodarka-oparta-o-dane-przemysl- (dostęp: 2.01.2019).

Botsman Rachel (2017), *Big data meets Big Brother as China moves to rate its citizens*, http://www.wired.co.uk/article/chinese-government-social-credit-score-privacy-invasion (dostęp: 18.06.2018).

Bowker Geoffrey C., Latour Bruno (1987), *A Booming Discipline Short of Discipline: (Social) Studies of Science in France*, „Social Studies of Science", no. 17, s. 715–748.

Boyd Danah (b.d.), *what's in a name? – danah michele boyd*, http://www.danah.org/name.html (dostęp: 11.08.2018).

Boyd Danah, Crawford Kate (2011), *Six Provocations for Big Data*, „SSRN Electronic Journal", https://doi.org/10.2139/ssrn.1926431

Boyd Danah, Crawford Kate (2012) *Critical questions for big data*, „Information, Communication & Society", vol. 15(5), s. 662–679, https://doi.org/10.1080/1369118X.2012.678878

Brackenbury Will, Liu Rui, Mondal Mainack, Elmore Aaron J., Ur Blase, Chard Kyle, Franklin Michael J. (2018), *Draining the Data Swamp*, [w:] *Proceedings of the Workshop on Human-In-the-Loop Data Analytics – HILDA'18. New York*, ACM Press, New York, s. 1–7.

Breda Thomas, Napp Clotilde (2019), *Girls' comparative advantage in reading can largely explain the gender gap in math-related fields*, „Proceedings of the National Academy of Sciences", vol. 116(31), s. 15435–15440, https://doi.org/10.1073/pnas.1905779116

Breiman Leo (2001), *Statistical Modeling: The Two Cultures*, „Statistical Science", vol. 16(3), s. 199–231, https://www.jstor.org/stable/2676681 (dostęp: 18.06.2018).

BrodieG (2018), *data.table vs dplyr: can one do something well the other can't or does poorly?*, https://stackoverflow.com/questions/21435339/data-table-vs-dplyr-can-one-do-something-well-the-other-cant-or-does-poorly/27840349#27840349 (dostęp: 5.12.2018).

Broek Elmira van den (2019), *Hiring Algorithms : An Ethnography of Fairness in Practice*, Fortieth International Conference on Information Systems, Munich.

Brooks Hannah (2014), *Interviews with Data Scientists*, „Data Science Weekly", no. 1 (April).

Brosz Maciej, Bryda Grzegorz, Siuda Piotr (2017), *Od redaktorów: Big Data i CAQDAS a procedury badawcze w polu socjologii jakościowej*, „Przegląd Socjologii Jakościowej", t. XIII, nr 2, s. 6–23.

Bryan Jennifer (2018), *Excuse Me, Do You Have a Moment to Talk About Version Control?*, „The American Statistician", vol. 72(1), s. 20–27, https://doi.org/10.1080/00031305.2017.1399928

Bryan Jennifer, Wickham Hadley (2017), *Data Science: A Three Ring Circus or a Big Tent?*, „Journal of Computational and Graphical Statistics", vol. 26(4), s. 784–785.

Bryan Jennifer, Hester Jim, Robinson David, Wickham Hadley (2019), *reprex: Prepare Reproducible Example Code via the Clipboard*, https://cran.r-project.org/package=reprex (dostęp: 18.06.2018).

Bryan Jenny (2018), *Happy Git and GitHub for the useR*, http://happygitwithr.com/ (dostęp: 6.02.2018).

Brynjolfsson Erik, McAfee Andrew (2015), *Wyścig z maszynami. Jak rewolucja cyfrowa napędza innowacje, zwiększa wydajność i w nieodwracalny sposób zmienia rynek pracy*, Kurhaus Publishing, Warszawa.

Brzezińska Ana, Przegalińska Aleksandra (2015), *Oko w oko z androidem. BINA48: Jestem jak gąbka. Pochłaniam każdą wiedzę, z którą się stykam*, http://weekend.gazeta.pl/weekend/1,152121,18816606,oko-w-oko-z-androidem-bina48-jestem-jak-gabka-pochlaniam.html?order=najstarsze&v=1&obxx=18816606#opinions (dostęp: 1.08.2018).

Brzeziński Tomasz (2018), *Lenistwo matką wynalazków, czyli dlaczego nie gram w Kaggle. Wystąpienie na konferencji Data Workshop Club Conf*, https://www.youtube.com/watch?v=Y5i05jBhQE8&feature= (dostęp: 14.09.2019).

Bugalski Piotr (2019), *Kurs Raspberry Pi – #8 – praca w konsoli, podstawy Linuksa*, https://forbot.pl/blog/kurs-raspberry-pi-praca-w-konsoli-podstawy-linuksa-id23911 (dostęp: 21.10.2019).

Bugnion Pascal (2016), *Scala for Data Science*, Packt Publishing, Birmingham–Mumbai.

Buolamwini Joy, Gebru Timnit (2018), *Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification*, [w:] S.A. Friedler, C. Wilson (red.), *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, vol. 81, PMLR, s. 77–91.

Burda Katarzyna (2018), *Uczłowieczanie komputera*, „Newsweek Polska", 3 kwietnia, s. 64–67.

Burrell Jenna (2016), *How the machine 'thinks': Understanding opacity in machine learning algorithms*, „Big Data & Society", vol. 3(1), https://doi.org/10.1177/2053951715622512

Burtch Linda (2014), *Tell Your Kids to Be Data Scientists, Not Doctors*, https://www.wired.com/insights/2014/06/tell-kids-data-scientists-doctors/ (dostęp: 14.02.2018).

Burtch Linda (2018), *The Burtch Works Study. Salaries of Data Scientists*, Burth Works Recruiting, Evanston, https://www.burtchworks.com/wp-content/uploads/2018/05/Burtch-Works-Study_DS-2018.pdf (dostęp: 3.02.2019).

Cadwalladr Carole (2018), *'I made Steve Bannon's psychological warfare tool': meet the data war whistleblower*, „The Guardian", 18 marca, https://www.theguardian.com/news/2018/mar/17/data-war-whistleblower-christopher-wylie-faceook-nix-bannon-trump (dostęp: 5.10.2018).

Cadwalladr Carole, Graham-Harrison Emma (2018), *Facebook and Cambridge Analytica face mounting pressure over data scandal*, „The Guardian", 19 marca, https://www.theguardian.com/news/2018/mar/18/cambridge-analytica-and-facebook-accused-of-misleading-mps-over-data-breach (dostęp: 22.03.2018).

Campbell Heidi A., Pastina Antonio C. (2010), *How the IPhone became divine: New media, religion and the intertextual circulation of meaning*, „New Media and Society", vol. 12(7), s. 1191–1207, https://doi.org/10.1177/1461444810362204

Canton James (2016), *From Big Data to Artificial Intelligence: The Next Digital Disruption*, https://www.huffingtonpost.com/james-canton/from-big-data-to--artifici_b_10817892.html (dostęp: 12.02.2018).

Cao Longbing (2017a), *Data Science: A Comperhensive Overview*, „ACM Computing Surveys", vol. 50(3), s. 1–42, https://doi.org/10.1145/3076253

Cao Longbing (2017b), *Data science: challenges and directions*, „Communications of the ACM", vol. 60(8), s. 59–68, https://dl.acm.org/doi/fullHtml/10.1145/3015456

Carneiro Tiago, Medeiros Da Nobrega Raul Victor, Nepomuceno Thiago, Bian Gui-Bin, De Albuquerque Victor Hugo C., Pedrosa Reboucas Filho Pedro (2018), *Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications*, „IEEE Access", no. 6, s. 61677–61685, https://doi.org/10.1109/ACCESS.2018.2874767

Caro Daniel H., Biecek Przemysław (2017), *intsvy: An R Package for Analyzing International Large-Scale Assessment Data*, „Journal of Statistical Software", vol. 81(7), s. 1–44, https://doi.org/10.18637/jss.v081.i07

Casas Pablo (2018), *Data Science Live Book: An intuitive and practical approach to data analysis, data preparation and machine learning, suitable for all ages!*, https://livebook.datascienceheroes.com/ (dostęp: 22.03.2018).

Cass Stephen (2016), *Linux at 25 Q&A with Linus Torvalds*, https://spectrum.ieee.org/computing/software/linux-at-25-qa-with-linus-torvalds (dostęp: 24.10.2019).

Centre for the New Economy and Society (2018), *The Future of Jobs Report 2018 Insight Report Centre for the New Economy and Society*, Geneva.

Cerf Vint (2007), *An Information Avalanche*, „IEEE Computer", vol. 40, no. 1, s. 104–105.

Ceri Stefano (2018), *On the role of statistics in the era of big data: A computer science perspective*, „Statistics & Probability Letters", no. 136, s. 68–72, https://doi.org/10.1016/j.spl.2018.02.019

ChallengeRocket.com (2018), *Warsaw AI Hackathon at Google Campus Warsaw*, https://challengerocket.com/warsaw-artificial-intelligence-hackathon-google-campus-warsaw (dostęp: 25.04.2018).

Chang Emily (2018), *Brotopia: Breaking up the Boys' Club of Silicon Valley*, Portfolio/Penguin, New York.

Chang Fay, Dean Jeffrey, Ghemawat Sanjay, Hsieh Wilson C., Wallach Deborah A., Burrows Mike, Chandra Tushar, Fikes Andrew, Gruber Robert E. (2006), *Bigtable: A Distributed Storage System for Structured Data*, [w:] *7th {USENIX} Symposium on Operating Systems Design and Implementation (OSDI)*, s. 205–218, https://research.google/pubs/pub27898/ (dostęp: 22.03.2018).

Chang Ray M., Kauffman Robert J., Kwon YoungOk (2014), *Understanding the paradigm shift to computational social science in the presence of big data*, „Decision Support Systems", no. 63, s. 67–80, https://doi.org/10.1016/J.DSS.2013.08.008

Chang Robert (2015), *Doing Data Science at Twitter. A reflection of my two year Journey so far. Sample size N = 1*, https://medium.com/@rchang/my-two-year-journey-as-a-data-scientist-at-twitter-f0c13298aee6 (dostęp: 29.08.2018).

Chang Winston, Luraschi Javier, Mastny Timothy (2019), *profvis: Interactive Visualizations for Profiling R Code*, https://cran.r-project.org/package=profvis (dostęp: 22.03.2018).

Charmaz Kathy (2009), *Teoria ugruntowana. Praktyczny przewodnik po analizie jakościowej*, Wydawnictwo Naukowe PWN, Warszawa.

Charmaz Kathy, Clarke Adele E. (red.) (2013), *Grounded Theory and Situational Analysis*, Sage, London.

Chemaly Soraya, Buni Catherine (2016), *The secret rules of the internet*, https://www.theverge.com/2016/4/13/11387934/internet-moderator-history-youtube-facebook-reddit-censorship-free-speech (dostęp: 5.01.2019).

Chen Catherine, Jiang Haoqiang (2018), *Important Skills for Data Scientists in China: Two Delphi Studies*, „Journal of Computer Information Systems", vol. 60(3), s. 287–296, https://doi.org/10.1080/08874417.2018.1472047

Chen Hao (2010), *Comparative Study of C, C++, C# and Java Programming Languages*, Vaasa University of Applied Sciences, Vassa.

Chen Hsinchun, Chiang Roger H.L., Storey Veda C. (2012), *Business Intelligence and Analytics: From Big Data to Big Impact*, „Management Information Systems Quarterly", vol. 36(4), s. 1165–1188, http://aisel.aisnet.org/misq/vol36/iss4/16 (dostęp: 22.03.2018).

Chen Min, Mao Shiwen, Liu Yunhao (2014), *Big Data: A Survey*, „Mobile Networks and Applications", vol. 19(2), s. 171–209, https://doi.org/10.1007/s11036-013-0489-0

Chojnowski Maciej (2020), *Biecek: SI? Musimy coś sobie wyjaśnić*, https://www.sztucznainteligencja.org.pl/biecek-si-musimy-cos-sobie-wyjasnic/ (dostęp: 26.04.2020).

Choudhury Tanzeem, Pentl Alex, Pentland Alex (2002), *The Sociometer: A Wearable Device for Understanding Human Networks*, https://dam-prod.media.mit.edu/x/files/tech-reports/TR-554.pdf (dostęp: 22.03.2018).

Chouldechova Alexandra (2017), *Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments*, „Big Data", vol. 5(2), s. 153–163, https://doi.org/10.1089/big.2016.0047

Ciechanowski Leon, Przegalińska Aleksandra, Magnuski Mikołaj, Gloor Peter (2018), *In the shades of the uncanny valley: An experimental study of human–chatbot interaction*, „Future Generation Computer Systems", vol. 92, s. 539–548, https://doi.org/10.1016/J.FUTURE.2018.01.055

Clark Liat (2012), *Google's Artificial Brain Learns to Find Cat Videos*, https://www.wired.com/2012/06/google-x-neural-network/ (dostęp: 21.01.2018).

Clark Stuart (2014), *Artificial intelligence could spell end of human race – Stephen Hawking*, „The Guardian", 2 grudnia https://www.theguardian.com/science/2014/dec/02/stephen-hawking-intel-communication-system-astrophysicist-software-predictive-text-type (dostęp: 14.02.2018).

Clarke Adele E. (1991), *Social Words / Arenas Theory as Organizational Theory*, [w:] D.R. Maines (red.), *Social Organization and Social Process. Essays in Honor of Anselm Strauss*, Aldine de Gruyter, New York, s. 119–158.

Clarke Adele E. (2003), *Situational Analyses: Grounded Theory Mapping After the Postmodern Turn*, „Symbolic Interaction", vol. 26(4), s. 553–576.

Clarke Adele E. (2005), *Situational Analysis. Grounded Theory After the Postmodern Turn*, Sage, London.

Clarke Adele E. (2015), *From Grounded Theory to Situational Analysis. What's New? Why? How?*, [w:] A.E. Clarke, C. Friese, R.S. Washburn (red.), *Situational Analysis in Practice. Mapping Research with Grounded Theory*, Left Coast Press Inc., Walnut Creek, s. 84–118.

Clarke Adele E., Casper Monica J. (1996), *From Simple Technology to Complex Arena: Classification of Pap Smears, 1917–90*, „Medical Anthropology Quarterly", vol. 10(4), s. 601–623, https://doi.org/10.1525/maq.1996.10.4.02a00120

Clarke Adele E., Friese Carrie (2007), *Situational Analysis: Going Beyond Traditional Grounded Theory*, [w:] K. Charmaz, A. Bryant (red.), *Handbook of Grounded Theory*, Sage, London, s. 694–743.

Clarke Adele E., Star Susan Leigh (2008), *The Social Worlds Framework: A Theory/Method Package*, [w:] Edward J. Hackett, O. Amsterdamska, M. Lynch, J. Wajcman (red.), *The Handbook of Science and Technology Studies*, The MIT Press, Cambridge–London, s. 113–158.

Clarke Adele E., Friese Carrie, Washburn Rachel S. (2015), *Introducing Situational Analysis*, [w:] A.E. Clarke, C. Friese, R.S. Washburn (red.), *Situational Analysis in Practice. Mapping Research with Grounded Theory*, Left Coast Press Inc., Walnut Creek, s. 11–75.

Clarke Adele E., Friese Carrie, Washburn Rachel S. (2017), *Situational Analysis: Grounded Theory After the Interpretive Turn*, Sage, Los Angeles.

Clarke Peter (2012), *Google neural network teaches itself to identify cats*, https://www.eetimes.com/document.asp?doc_id=1266579 (dostęp: 21.01.2018).

Class Central (2018), *Data Science Courses*, https://www.class-central.com/subject/data-science (dostęp: 21.02.2018).

Cleveland William S. (2001), *Data Science: an Action Plan for Expanding the Technical Areas of the Field of Statistics*, „International Statistical Review", vol. 69(1), s. 21–26, https://doi.org/10.1111/j.1751-5823.2001.tb00477.x

Codd Edgar F. (1970), *A relational model of data for large shared data banks*, „Communications of the ACM", vol. 13(6), s. 377–387, https://doi.org/10.1145/362384.362685

Collobert Ronan, Farabet Clement, Kavukcuoglu Koray, Chintala Soumith (2019), *What is Torch?*, http://torch.ch/ (dostęp: 15.07.2019).

Conway Drew (2010), *The Data Science Venn Diagram*, http://www.dataists.com/2010/09/the-data-science-venn-diagram/ (dostęp: 18.02.2018).

Conway Drew (2014), *Data science through the lens of social science*, „Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining – KDD '14", https://doi.org/10.1145/2623330.2630824

Cook Gary, Lee Jude, Kong Ada, Deans John, Johnson Brian, Jardim Elizabeth (2017), *Clicking Clean: Who Is Winning the Race To Build a Green Internet?*, Greenpeace Inc., https://storage.googleapis.com/planet4-international-stateless/2017/01/35f0ac1a-clickclean2016-hires.pdf (dostęp: 15.07.2019).

Courtland Rachel (2018), *Bias detectives: the researchers striving to make algorithms fair*, „Nature", vol. 558(7710), s. 357–360, https://doi.org/10.1038/d41586-018-05469-3

Crain Matthew (2018), *The limits of transparency: Data brokers and commodification*, „New Media and Society", vol. 20(1), https://doi.org/10.1177/1461444816657096

Crawford Kate (2021), *Atlas of AI: power, politics, and the planetary costs of artificial intelligence*, Yale University Press, New Haven.

Crawford Kate, West Sarah Myers, Whittaker Meredith (2019), *Discriminating Systems: Gender, Race and Power in AI*, AI Now Institute, New York, https://ainowinstitute.org/discriminatingsystems.pdf (dostęp: 15.07.2019).

Crawford Kate, Whittaker Meredith, Dobbe Roel, Fried Genevieve, Kaziunas Elizabeth, Mathur Varoon, West Sarah M., Richardson Rashida, Schultz Jason, Schwartz Oscar (2018), *AI Now Report 2018*, AI Now Institute, New York, https://ainowinstitute.org/AI_Now_2018_Report.pdf (dostęp: 15.07.2019).

Creemers Rogier (2015), *Planning Outline for the Construction of a Social Credit System (2014–2020)*, https://chinacopyrightandmedia.wordpress.com/2014/06/14/planning-outline-for-the-construction-of-a-social-credit-system-2014-2020/ (dostęp: 21.01.2019).

CrowdFlower (2017), *2017 Data Scientist Report*, https://visit.crowdflower.com/rs/416-ZBE-142/images/data-scientist-report-dec.pdf?mkt_tok=eyJpIjoiWXpNek5EQmtNalJsTkdWayIsInQiOiJPb29MV2JJdU81alRhbGh6OUVWcmt2

UWpibXJ3cG5pSlFrNUxlVUdwT2hna1VOOU5Gd2tMU3ZEWmhoTVVmVHRXNWFhMFM4eTI1dDJwbWRJczVoTVlnRjFkQjl4ekNmT (dostęp: 11.02.2018).

CS Department Toronto University (b.d.), *Geoffrey E. Hinton – Home Page*, http://www.cs.toronto.edu/~hinton/ (dostęp: 29.03.2018).

Cukier Kenneth, Mayer-Schönberger Victor (2014), *Big Data. Rewolucja, która zmieni nasze myślenie, pracę i życie*, Wydawnictwo MT Biznes, Warszawa.

Culkin John M. (1967), *A Shoolman's Guide to Marshall McLuhan*, „The Saturday Review", 18 marca, s. 51–53, 70–72.

Cyfryzacja KPRM (b.d.), *Robert Kroplewski*, https://www.gov.pl/web/cyfryzacja/robert-kroplewski1 (dostęp: 6.08.2020).

Czapska Martyna (2018), *RODO a sztuczna inteligencja*, http://lexrobotica.pl/2018/05/25/rodo-a-sztuczna-inteligencja/ (dostęp: 10.09.2018).

Czarnocka-Cieciura Marta, Migdał Piotr (2015), *TagOverflow*, https://github.com/stared/tagoverflow (dostęp: 9.12.2018).

Czarnowski Ireneusz, Krawiec Krzysztof, Mańdziuk Jacek, Stefanowski Jerzy (2018), *Raport z pierwszego Zjazdu Polskiego Porozumienia na Rzecz Rozwoju Sztucznej Inteligencji*, PP-RAI, Poznań, https://pp-rai.cs.put.poznan.pl/pp-rai-2018-raport.pdf (dostęp: 6.08.2020).

Czubkowska Sylwia (2018), *Dane, kłamstwa i wybory: Facebook wybiera Ci prezydenta*, „Gazeta Wyborcza", 24 marca.

Ćwiklak Dariusz (2017), *Wielki Brat szepcze do ciebie*, „Newsweek Polska", 26 czerwca, s. 73–75.

Dalton Craig M., Taylor Linnet, Thatcher Jim (2016), *Critical Data Studies: A dialog on data and space*, „Big Data & Society", vol. 3(1), https://doi.org/10.1177/2053951716648346

Dar Pranav (2018), *Python or R? Hadley Wickham and Wes McKinney are Building Platform Independent Libraries!*, https://www.analyticsvidhya.com/blog/2018/05/python-and-r-are-joining-hands-to-eliminate-platform-dependency/ (dostęp: 9.06.2018).

Darmach Krystian (2017), *Autoetnografia 2.0*, „Kultura i Społeczeństwo", nr 3(LXI), s. 87–101.

Data & Society (2018), *Data & Society Research Institute*, https://datasociety.net/ (dostęp: 30.04.2018).

DataCamp (2018), *DataCamp Scholarship for Women and Gender Minorities Application Form 2018*, https://docs.google.com/forms/d/e/1FAIpQLScPBQuNDqucnpQoMUhFAKcpFSXYskb_zWr5k8Uy3pPB6o0Uag/viewform (dostęp: 10.09.2018).

Data Science Association (2020), *Data Science Code of Professional Conduct*, http://datascienceassn.org/code-of-conduct.html (dostęp: 2.08.2020).

Data Science Warsaw (2018a), *Data Science Summit*, http://dssconf.pl/ (dostęp: 30.03.2018).

Data Science Warsaw (2018b), *Data Science Warsaw (Warszawa, Polska) | Meetup*, https://www.meetup.com/pl-PL/Data-Science-Warsaw/ (dostęp: 19.11.2018).

Davenport Thomas H., Patil D.J. (2012), *Data Scientist: The Sexiest Job of the 21st Century*, „Harvard Business Review", https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century (dostęp: 30.03.2018).

Davidson James, Livingston Blake, Sampath Dasarathi, Liebald Benjamin, Liu Junning, Nandy Palash, Van Vleet Taylor, Gargi Ullas, Gupta Sujoy, He Yu, Lambert Mike (2010), *The YouTube video recommendation system*, [w:] *Proceedings of the fourth ACM conference on Recommender systems – RecSys '10. New York*, ACM Press, New York, s. 293–296, https://dl.acm.org/doi/10.1145/1864708.1864770

Delapenha Lauren (2017), *42 Essential Quotes by Data Science Thought Leaders*, https://www.kdnuggets.com/2017/05/42-essential-quotes-data-science-thought-leaders.html (dostęp: 6.02.2018).

Denyer Simon (2016), *China's plan to organize its society relies on 'big data' to rate everyone*, „The Washington Post", 22 października.

Deptuła Jacek (2018), *Facebook dobrze wie, kto i jak będzie głosował*, „Dziennik Łódzki", 21 marca.

Desai Jules, Watson David, Wang Vincent, Taddeo Mariarosaria, Floridi Luciano (2022), *The epistemological foundations of data science: a critical analysis*, „SSRN Electronic Journal", https://doi.org/10.2139/ssrn.4008316

Devlin Josh (2018), *Want a Job in Data? Learn This*, https://www.dataquest.io/blog/why-sql-is-the-most-important-language-to-learn (dostęp: 20.04.2018).

DeZyre (2015), *Data Science Programming: Python vs R*, https://www.dezyre.com/article/data-science-programming-python-vs-r/128 (dostęp: 29.09.2018).

Diaz-Bone Rainer (2013), *Situationsanalyse – Strauss meets Foucault?*, „Forum Qualitative Sozialforschung", vol. 14(1), https://doi.org/10.17169/fqs-14.1.1928

Diebold Francis X. (2012), *On the Origin(s) and Development of "Big Data": The Phenomenon, the Term, and the Discipline*, http://www.ssc.upenn.edu/~fdiebold/papers/paper112/Diebold_Big_Data.pdf (dostęp: 29.09.2018).

Dijck José van (2014), *Datafication, dataism and dataveillance: Big Data between scientific paradigm and ideology*, „Surveillance and Society", vol. 12(2), s. 197–208, https://doi.org/10.24908/ss.v12i2.4776

dimlakgorkehgz (2019), *GUIs to save from typing R code*, https://alternativeto.net/list/2063/guis-to-save-from-typing-r-code/ (dostęp: 11.02.2019).

Dixon James (2010), *Pentaho, Hadoop, and Data Lakes*, https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/ (dostęp: 14.08.2019).

Dodge David (2018), *5 Reasons Python Programming is Perfect for Kids*, https://codakid.com/5-reasons-python-programming-is-perfect-for-kids/ (dostęp: 20.07.2019).

Donoho David (2015), *50 Years of Data Science*, „Journal of Computational and Graphical Statistics", vol. 26(4), s. 745–766, https://doi.org/10.1080/10618600.2017.1384734

Dopierała Renata (2013), *Prywatność w perspektywie zmiany społecznej*, Zakład Wydawniczy NOMOS, Kraków.

Drabas Tomasz, Lee Denny, Karau Holden (2017), *Learning PySpark*, Packt Publishing, Birmingham.

Draper Nora (2017), *Fail Fast: The Value of Studying Unsuccessful Technology Companies*, „Media Industries Journal", vol. 4(1), https://doi.org/10.3998/mij.15031809.0004.101

Drejewicz Szymon (2017), *Podcast Data Science po polsku*, https://soundcloud.com/szymon-drejewicz-184766256 (dostęp: 20.06.2017).

Drury Benjamin J., Oliver Siy John, Cheryan Sapna (2011), *When Do Female Role Models Benefit Women? The Importance of Differentiating Recruitment From Retention in STEM*, „Psychological Inquiry", vol. 22(4), s. 265–269, https://doi.org/10.1080/1047840X.2011.620935

Dubrow Joshua Kjerulf, Tomescu-Dubrow Irina (2016), *The rise of cross-national survey data harmonization in the social sciences: emergence of an interdisciplinary methodological field*, „Quality & Quantity", vol. 50(4), s. 1449–1467, https://doi.org/10.1007/s11135-015-0215-z

Dunn Jeff (2016), *We put Siri, Alexa, Google Assistant, and Cortana through a marathon of tests to see who's winning the virtual assistant race – here's what we found*, https://www.businessinsider.com/siri-vs-google-assistant-cortana-alexa-2016-11?IR=T#how-do-i-say-where-is-the-library-in-spanish-38 (dostęp: 10.07.2019).

Dutcher Jennifer (2014), *Big Data Isn't a Concept – It's a Problem to Solve*, https://datascience.berkeley.edu/what-is-big-data/ (dostęp: 9.12.2016).

Dwoskin Elizabeth, Harwell Drew, Timberg Craig (2018), *Facebook had a closer relationship than it disclosed with the academic it called a liar*, https://www.washingtonpost.com/business/economy/facebook-had-a-closer-relationship-than-it-disclosed-with-the-academic-it-called-a-liar/2018/03/22/ca0570cc-2df9-11e8-8688-e053ba58f1e4_story.html?amp;utm_term=.e288973a7f9b&noredirect=on&utm_term=.a21d2a1e8 (dostęp: 6.02.2019).

Dyk David van, Fuentes Montse, Jordan Michael I., Newton Michael, Ray Bonnie K., Lang Duncan T., Wickham Hadley (2015), *ASA Statement on the Role of Statistics in Data Science*, http://magazine.amstat.org/blog/2015/10/01/asa-statement-on-the-role-of-statistics-in-data-science/ (dostęp: 7.02.2018).

Dzieciątko Mariusz, Spinczyk Dominik (2016), *Text mining. Metodyka, narzędzia, zastosowania*, Wydawnictwo Naukowe PWN, Warszawa.

Eagle Nathan (2005), *Machine Perception and Learning of Complex Social Systems*, praca doktorska, Massachusetts Institute of Technology, https://www.media.mit.edu/publications/machine-perception-and-learning-of-complex-social-systems/ (dostęp: 6.02.2019).

Eagle Nathan, Greene Kate (2014), *Reality Mining: Using Big Data to Engineer a Better World*, The MIT Press, Cambridge–London.

Eagle Nathan, Pentland Alex (2006), *Reality mining: sensing complex social systems*, „Journal Personal and Ubiquitous Computing", vol. 10(4), s. 255–268, https://doi.org/10.1007/s00779-005-0046-3

Economic Graph Team (2017), *LinkedIn's 2017 U.S. Emerging Jobs Report*, https://economicgraph.linkedin.com/research/LinkedIns-2017-US-Emerging-Jobs-Report (dostęp: 7.02.2018).

Eder Maciej (2013), *Mind your corpus: systematic errors in authorship attribution*, „Literary and Linguistic Computing", vol. 28(4), s. 603–614, https://doi.org/10.1093/llc/fqt039

Eder Maciej (2014), *Metody ścisłe w literaturoznawstwie i pułapki pozornego obiektywizmu – przykład stylometrii*, „Teksty Drugie", nr 2, s. 90–105.

Eder Maciej (2015), *Does size matter? Authorship attribution, small samples, big problem*, „Digital Scholarship in the Humanities", vol. 30(2), s. 167–182, https://doi.org/10.1093/llc/fqt066

Eder Maciej (2017), *Visualization in stylometry: Cluster analysis using networks*, „Digital Scholarship in the Humanities", vol. 32(1), s. 50–64, https://doi.org/10.1093/llc/fqv061

Eder Maciej, Rybicki Jan, Kestemont Mike (2016), *Stylometry with R: A Package for Computational Text Analysis*, „The R Journal", vol. 8(1), s. 107–121, https://doi.org/10.32614/RJ-2016-007

Elish M.C., Boyd Danah (2018), *Situating methods in the magic of Big Data and AI*, „Communication Monographs", vol. 85(1), s. 57–80, https://doi.org/10.1080/03637751.2017.1375130

Elliott Larry (2018), *Robots will take our jobs. We'd better plan now, before it's too late*, „The Guardian", 1 stycznia, https://www.theguardian.com/commentisfree/2018/feb/01/robots-take-our-jobs-amazon-go-seattle (dostęp: 14.02.2018).

Exxact (2019), *NVIDIA Data Science Workstations*, https://www.exxactcorp.com/NVIDIA-Data-Science-Workstations (dostęp: 12.09.2019).

Facebook (2018), *Yann LeCun – Facebook Research*, https://research.fb.com/people/lecun-yann/ (dostęp: 23.08.2018).

Feinberg Donald (2017), *Data Lake, Big Data, NoSQL – The Good, The Bad and The Ugly*, https://blogs.gartner.com/donald-feinberg/2017/07/29/oh-terms--use-technology-need-new-hype/ (dostęp: 6.08.2019).

Ferrucci David, Brown Eric, Chu-Carroll Jennifer, Fan James, Gondek David, Kalyanpur Aditya A., Lally Adam, Murdock J. William, Nyberg Eric, Prager John, Schlaefer Nico, Welty Chris (2010), *Building Watson: An Overview of the DeepQA Project*, „AI Magazine", vol. 31(3), s. 59–79, https://doi.org/10.1609/aimag.v31i3.2303

Feuerstein Steven (1996), *Advanced Oracle PL/SQL Programming with Packages*, O'Reilly & Associates, Inc., Sebastopol.

Fierro Miguel, Salvaris Mathew, Wu Tao (2017), *Lessons Learned From Benchmarking Fast Machine Learning Algorithms*, https://blogs.technet.microsoft.com/machinelearning/2017/07/25/lessons-learned-benchmarking-fast-machine-learning-algorithms/ (dostęp: 15.11.2019).

Fjeld Jessica, Achten Nele, Hilligoss Hannah, Nagy Adam Chistopher, Srikumar Madhulika (2020), *Principled Artificial Intelligence: Mapping Consensus*

in Ethical and Rights-based Approaches to Principles for AI, „SSRN Electronic Journal", https://ssrn.com/abstract=3518482 (dostęp: 6.08.2020).

Floridi Luciano (2012), *Big Data and Their Epistemological Challenge*, „Philosophy & Technology", vol. 25(4), s. 435–437, https://doi.org/10.1007/s13347-012-0093-4

Foreman John W. (2017), *Mistrz analizy danych. Od danych do wiedzy*, Wydawnictwo Helion, Gliwice.

Formulated.by (2018), *15 Data Science Slack Communities to Join*, https://towardsdatascience.com/15-data-science-slack-communities-to-join-8fac301bd6ce (dostęp: 5.09.2019).

Fox John (2009), *Aspects of the Social Organization and Trajectory of the R Project*, „The R Journal", no. 1 (December), s. 5–13.

Fox John, Leanage Allison (2016), *R and the Journal of Statistical Software*, „Journal of Statistical Software", vol. 73(2), s. 1–13, https://doi.org/10.18637/jss.v073.i02

Franceschi-Bicchierai Lorenzo (2018), *Why We're Not Calling the Cambridge Analytica Story a 'Data Breach'*, https://motherboard.vice.com/en_us/article/3kjzvk/facebook-cambridge-analytica-not-a-data-breach (dostęp: 5.02.2019).

Freeman Linton C. (2014), *The Development of Social Network Analysis – with an Emphasis on Recent Events*, [w:] *The SAGE Handbook of Social Network Analysis*, Sage Publications Ltd., London, s. 26–39.

Frenken Koen, Schor Juliet (2017), *Putting the sharing economy into perspective*, „Environmental Innovation and Societal Transitions", vol. 23, s. 3–10, https://doi.org/10.1016/j.eist.2017.01.003

Frey Carl Benedikt, Osborne Michael A. (2017), *The future of employment: How susceptible are jobs to computerisation?*, „Technological Forecasting and Social Change", vol. 114(1), s. 254–280, https://doi.org/10.1016/j.techfore.2016.08.019

Friedman Batya, Nissenbaum Helen (1996), *Bias in computer systems*, „ACM Transactions on Information Systems", vol. 14(3), s. 330–347, https://doi.org/10.1145/230538.230561

Fuller Abby (2019), *On Soft Talks and Being Technical – talk at Monktoberfest 2019*, https://www.youtube.com/watch?reload=9&v=cDgi4oaoIto (dostęp: 30.04.2020).

Fundacja Panoptykon (2019), *Prawo do wyjaśnienia w pytaniach i odpowiedziach*, https://panoptykon.org/prawo-do-wyjasnienia (dostęp: 5.09.2019).

Future of Life Institute (2017), *AI Principles*, https://futureoflife.org/ai-principles/ (dostęp: 7.02.2018).

Fürg Daniel (2017), *Curt Simon Harlinghausen: Der kulturelle Wandel findet oftmals nicht statt*, https://48forward.com/greencircle/curt-simon-harlinghausen/ (dostęp: 27.06.2020).

Galloway Scott (2018), *Wielka Czwórka. Ukryte DNA: Amazon, Apple, Facebook i Google*, Dom Wydawniczy Rebis, Poznań.

Gandomi Amir, Haider Murtaza (2015), *Beyond the hype: Big data concepts, methods, and analytics*, „International Journal of Information Management", vol. 35(2), s. 137–144, https://doi.org/10.1016/j.ijinfomgt.2014.10.007

Gardner Dan (2012), *Introduction*, [w:] R. Smolan, J. Erwitt (red.), *The human face of big data*, Against All Odds Productions, Sausalito, s. 14–15.

Garner Bennett (2018), *Why I Code in Python*, https://medium.com/@Bennett-Garner/why-i-code-in-python-a1e4012eb859 (dostęp: 24.07.2019).

Gągolewski Marek (2016), *Programowanie w języku R. Analiza danych. Obliczenia. Symulacje*, Wydawnictwo Naukowe PWN, Warszawa.

Gągolewski Marek, Bartoszuk Maciej, Cena Anna (2016), *Przetwarzanie i analiza danych w języku Python*, Wydawnictwo Naukowe PWN, Warszawa.

Gershon Ilana (2011), *Neoliberal Agency*, „Current Anthropology", vol. 52(4), s. 537–555, https://doi.org/10.1086/660866

Gersz Aleksandra (2018), *Wyciek danych z Facebooka wpłynął na wybory w Polsce?*, „Dziennik Łódzki", 21 marca.

Główny Urząd Statystyczny (2019), *Jak zapytać o dane*, https://stat.gov.pl/pytania-i-zamowienia/jak-zamowic-dane/ (dostęp: 6.08.2019).

Gogoi Namrata (2019), *How to Use AI Camera Features on Any Android Phone*, https://www.guidingtech.com/ai-camera-features-android-phone/ (dostęp: 10.07.2019).

Gogołek Włodzimierz (2016), *Rafinacja dużej skali zasobów sieciowych – Big Data. Dziennikarskie źródło informacji*, „Ekonomika i Organizacja Przedsiębiorstwa", nr 7(798), s. 12–18.

Goldstein Ira (2019), *What ! No GUI ? – Teaching A Text Based Command Line Oriented Introduction to Computer Science Course*, „Information Systems Education Journal", no. 17(February), s. 40–48.

Gontarz Andrzej (2019), *Rynek pracy. Od BigData i AI do BI*, „Enterprise Software Review", https://bigdatatechwarsaw.eu/report-job-market-for-data-professionals-from-big-data-and-ai-to-bi/ (dostęp: 6.11.2019).

Goodfellow Ian, Bengio Yoshua, Courville Aaron (2016), *Deep Learning*, http://www.deeplearningbook.org/ (dostęp: 5.11.2017).

Google (2012), *Google's R Style Guide*, https://google.github.io/styleguide/Rguide.xml (dostęp: 5.01.2018).

Google (2018), *Peter Norvig – Google AI*, https://ai.google/research/people/author205 (dostęp: 22.05.2018).

Google (2019), *Colaboratory – FAQ*, https://research.google.com/colaboratory/faq.html (dostęp: 23.07.2019).

Google Data Center 360° Tour (b.d.), https://www.youtube.com/watch?v=zDAYZU4A3w0 (dostęp: 7.01.2020).

Google Trends (2020a), *Big data, Machine learning, Data science, Artificial intelligence – Explore – Google Trends*, https://trends.google.com/trends/explore?date=2008-01-012020-09-30&q=%2Fm%2F0bs2j8q,%2Fm%2F01hyh_,%2Fm%2F0jt3_q3,%2Fm%2F0mkz&hl=en-US (dostęp: 5.10.2020).

Google Trends (2020b), *Big data, Machine learning, Data science – Explore – Google Trends*, https://trends.google.com/trends/explore?date=2008-01-012020--09-30&q=%2Fm%2F0bs2j8q,%2Fm%2F01hyh_,%2Fm%2F0jt3_q3&hl=en--US (dostęp: 5.10.2020).

Gorecki Jan, Yuan Jiaming (2019), *Database-like ops benchmark*, https://h2oai.github.io/db-benchmark/index.html (dostęp: 6.05.2019).

Gostkiewicz Michał (2017), *Chiny podłączą 1,3 miliarda ludzi i wielkie firmy do Matrixa. Prawdziwego. Czerwonej pigułki nie będzie*, http://weekend.gazeta.pl/weekend/1,152121,22621623,chiny-podlacza-1-3-miliarda-ludzi-i-wielkie-firmy-do-matrixa.html (dostęp: 18.06.2018).

Grace Katja, Salvatier John, Dafoe Allan, Zhang Baobao, Evans Owain (2018), *When Will AI Exceed Human Performance? Evidence from AI Experts*, „Journal of Artificial Intelligence Research", no. 62, s. 729–754

Granville Vincent (2014), *16 analytic disciplines compared to data science*, https://www.datasciencecentral.com/profiles/blogs/17-analytic-disciplines-compared (dostęp: 3.01.2017).

Greene Daniel, Hoffman Anna Lauren, Stark Luke (2019), *Better, Nicer, Clearer, Fairer: A Critical Assessment of the Movement for Ethical Artificial Intelligence and Machine Learning*, Hawaii International Conference on System Sciences (HICSS), Maui.

Grewal Paul (2018), *Suspending Cambridge Analytica and SCL Group From Facebook*, https://newsroom.fb.com/news/2018/03/suspending-cambridge-analytica/ (dostęp: 5.02.2019).

Groen Albin (2019), *Maximizing use of the terminal*, https://medium.com/@albingroen/maximizing-use-of-the-terminal-9b7b12ab5dd2 (dostęp: 11.09.2019).

Grolemund Garrett, Wickham Hadley (2011), *Dates and Times Made Easy with {lubridate}*, „Journal of Statistical Software", vol. 40(3), s. 1–25.

Grommé Francisca, Ruppert Evelyn, Cakici Baki (2018), *Data Scientists: A new faction of the transnational field of statistics*, [w:] H. Knox, D. Nafus (red.), *Ethnography for a data-saturated world*, Manchester University Press, Manchester, s. 33–61.

Grush Loren (2015), *Google engineer apologizes after Photos app tags two black people as gorillas*, https://www.theverge.com/2015/7/1/8880363/google-apologizes-photos-app-tags-two-black-people-gorillas?utm_source=Codecademy (dostęp: 26.02.2018).

Gualtieri Mike, Carlsson Kjell, Sridharan Srividya, Rerdoni Robert, Yunus Aldila (2018), *The Forrester WaveTM: Multimodal Predictive Analytics and Machine Learning Solutions, Q3 2018*, https://reprints.forrester.com/#/assets/2/1451/RES141374/reports (dostęp: 11.09.2019).

Gulipalli Pradeep (2019), *The Pareto Principle for Data Scientists*, https://www.kdnuggets.com/2019/03/pareto-principle-data-scientists.html (dostęp: 15.04.2019).

Guo Philip J. (2018), *Non-Native English Speakers Learning Computer Programming*, [w:] *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems – CHI '18. T. 2018-April*, ACM Press, New York, s. 1–14.

Gutierrez Sebastian (2014), *Data Scientists at Work: Sexy Scientists Wrangling Data And Begetting New Industries*, Appres, Berkeley.

Hale Jeff (2018), *The Most in Demand Skills for Data Scientists*, https://towardsdatascience.com/the-most-in-demand-skills-for-data-scientists-4a4a8db896db (dostęp: 25.10.2018).

Halevy Alon, Norvig Peter, Pereira Fernando (2009), *The Unreasonable Effectiveness of Data*, „IEEE Intelligent Systems", vol. 24(2), s. 8–12, https://doi.org/10.1109/MIS.2009.36

Hałas Elżbieta (2006), *Interakcjonizm symboliczny*, Wydawnictwo Naukowe PWN, Warszawa.

Hamideh (2015), *Do data scientists use Excel?*, https://datascience.stackexchange.com/questions/5443/do-data-scientists-use-excel (dostęp: 4.09.2018).

Hansen Steven (2017), *How Big Data Is Empowering AI and Machine Learning?*, https://hackernoon.com/how-big-data-is-empowering-ai-and-machine-learning-4e93a1004c8f (dostęp: 13.02.2018).

Harari Yuval Noah (2017), *Homo Deus: A Brief History of Tomorrow*, Vintage, London.

Harari Yuval Noah (2018), *21 lekcji na 21 wiek*, Wydawnictwo Literackie, Kraków.

Haratyk Karol, Biały Kamila, Gońda Marcin (2017), *Biographical meanings of work: the case of a Polish freelancer*, „Przegląd Socjologii Jakościowej", t. XIII, nr 4, s. 136–159, https://doi.org/10.18778/1733-8069.13.4.08

Harper Norma Wynn, Daane C.J. (1998), *Causes and Reduction of Math Anxiety in Preservice Elementary Teachers*, „Action in Teacher Education", vol. 19(4), s. 29–38, https://doi.org/10.1080/01626620.1998.10462889

Harris Harlan D., Murphy Sean Patric, Vaisman Marck (2013), *Analyzing The Analyzers: An Introspective Survey of Data Scientists and Their Work*, O'Reilly, Beijing–Cambridge.

Harris Jonathan (2012), *Data Driven*, [w:] R. Smolan, J. Erwitt (red.), *The human face of big data*, Against All Odds Productions, Sausalito, s. 200–203.

Hatalska Natalia (2021), *Wiek paradoksów: czy technologia nas ocali?*, Wydawnictwo Znak, Kraków.

Hatton Celia (2015), *China 'social credit': Beijing sets up huge system*, https://www.bbc.com/news/world-asia-china-34592186 (dostęp: 21.01.2019).

Henderson Peter, Islam Riashat, Bachman Philip, Pineau Joelle, Precup Doina, Meger David (2017), *Deep Reinforcement Learning that Matters*, https://arxiv.org/abs/1709.06560 (dostęp: 13.02.2018).

Henke Nicolaus, Levine Jordan, Mcinerney Paul (2018), *You Don't Have to Be a Data Scientist to Fill This Must- Have Analytics Role*, „Harvard Business Review", 5 lutego, https://hbr.org/2018/02/you-dont-have-to-be-a-data-scientist-to-fill-this-must-have-analytics-role (dostęp: 20.08.2018).

Hill Kashmir (2012), *How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did*, „Forbes", 16 lutego, https://www.forbes.com/sites/kashmirhill/2012/02/16/how-target-figured-out-a-teen-girl-was-pregnant-before-her-father-did/#6d733aa66668 (dostęp: 2.05.2018).

Hu Jane C. (2018), *So you think being called NIPS SUX? Here's how organizations handle unfortunate acronyms*, https://qz.com/1437829/from-isis-to-nips-how-organizations-handle-unfortunate-acronyms/ (dostęp: 12.02.2019).

Huet Ellen (2016), *The Humans Hiding Behind the Chatbots*, https://www.bloomberg.com/news/articles/2016-04-18/the-humans-hiding-behind-the-chatbots (dostęp: 6.01.2019).

Hughes Phil (1999), *An Interview with Guido van Rossum*, „Linux Journal", http://web.archive.org/web/20160310004241/http://www.linuxjournal.com/article/5028 (dostęp: 16.07.2019).

Hunter John D. (2007), *Matplotlib: A 2D Graphics Environment*, „Computing in Science & Engineering", vol. 9(3), s. 90–95, https://doi.org/10.1109/MCSE.2007.55

Hunter John D., Droettboom Michael (2012), *matplotlib*, [w:] A. Brown, G. Wilson (red.), *The Architecture of Open Source Applications*, Volume II: *Structure, Scale, and a Few More Fearless Hacks*, http://aosabook.org/en/matplotlib.html (dostęp: 6.01.2019).

Hyndman Rob (2014), *Am I a data scientist?*, https://robjhyndman.com/hyndsight/am-i-a-data-scientist/ (dostęp: 5.01.2018).

Ierusalimschy Roberto, Celes Waldemar, Figueiredo Luiz Henrique de (2019), *Lua*, https://www.lua.org/ (dostęp: 15.07.2019).

Ihaka Ross, Gentleman Robert (1996), *R: A Language for Data Analysis and Graphics*, „Journal of Computational and Graphical Statistics", vol. 5(3), s. 299–314, https://doi.org/10.2307/1390807

Inbar Ohad, Tractinsky Noam, Meyer Joachim (2007), *Minimalism in information visualization*, [w:] *Proceedings of the 14th European conference on Cognitive ergonomics invent! explore! – ECCE '07. New York*, ACM Press, New York, s. 185.

Is it a job offer for a Data Scientist?, http://smarterpoland.pl/index.php/2017/01/is-it-a-job-offer-for-a-data-scientist/ (dostęp: 21.02.2018).

Isaak Jim, Hanna Mina J. (2018), *User Data Privacy: Facebook, Cambridge Analytica, and Privacy Protection*, „Computer", vol. 51(8), s. 56–59, https://doi.org/10.1109/MC.2018.3191268

Ismail Nur Amie, Abidin Wardah Zainal (2016), *Data Scientist Skills*, „IOSR Journal of Mobile Computing & Application", vol. 03(04), s. 52–61, https://doi.org/10.9790/0050-03045261

IT Central Station (2019), *Data Science Platforms: Buyer's Guide and Reviews*, https://www.itcentralstation.com/landing/report-data-science-platforms (dostęp: 16.04.2019).

Iwasiński Łukasz (2016), *Społeczne zagrożenia danetyzacji rzeczywistości*, [w:] B. Sosińska-Kalata (red.), *Nauka o informacji w okresie zmian. Informatologia i humanistyka cyfrowa*, Wydawnictwo SBP, Warszawa, s. 135–146.

Iwasiński Łukasz (2017), *Przyczynek do rozważań nad suwerennością konsumenta w epoce danetyzacji i big data*, „Kultura – Historia – Globalizacja", nr 21, s. 119–133.

Jackson Michelle (2001), *Meritocracy, Education and Occupational Attainment: What Do Employers Really See as Merit?*, Working Paper, no. 3, Department of Sociology, University of Oxford, s. 1–24.

Jacobs Adam (2009), *The pathologies of big data*, „Communications of the ACM", vol. 52(8), s. 36–44, https://doi.org/10.1145/1536616.1536632

Jacyno Małgorzata (2007), *Kultura indywidualizmu*, Wydawnictwo Naukowe PWN, Warszawa.

Jain Varsha (2018), *Luxury: Not for Consumption but Developing Extended Digital Self*, „Journal of Human Values", vol. 24(1), s. 25–38, https://doi.org/10.1177/0971685817733570

Jarvis Jeremy (2014), *@jeremyjarvis: A data scientist is a statistician who lives in San Fransisco*, https://twitter.com/jeremyjarvis/status/428848527226437632 (dostęp: 7.12.2017).

Jemielniak Dariusz (2018), *Socjologia 2.0: o potrzebie łączenia big data z etnografią cyfrową, wyzwaniach jakościowej socjologii cyfrowej i systematyzacji pojęć*, „Studia Socjologiczne", nr 2(229), s. 7–29, https://doi.org/10.24425/122461

Jemielniak Dariusz (2019), *Socjologia internetu*, Wydawnictwo Naukowe Scholar, Warszawa.

Jemielniak Dariusz (2020), *Thick Big Data*, Oxford University Press, Oxford.

Joler Vladan, Crawford Kate (2018), *Anatomy of an AI System*, https://anatomyof.ai/img/ai-anatomy-publication.pdf (dostęp: 16.04.2019).

Jolly Eshin (2018), *Pymer4: Connecting R and Python for Linear Mixed Modeling*, „Journal of Open Source Software", vol. 3(31), s. 1–3, https://doi.org/10.21105/joss.00862

Jóźwiak Michał (2019), *Grzechomaty i księża-roboty, czyli o… Kościele przyszłości?*, https://misyjne.pl/grzechomaty-i-ksieza-roboty-czyli-o-kosciele-przyszlosci/ (dostęp: 26.08.2020).

Junco Pablo Ruiz (2017), *Data Scientist Personas: What Skills Do They Have and How Much Do They Make?*, https://www.glassdoor.com/research/data-scientist-personas/ (dostęp: 25.10.2018).

Jung Alexander (2019), *imgaug Documentation. Release 0.2.9*, https://buildmedia.readthedocs.org/media/pdf/imgaug/latest/imgaug.pdf (dostęp: 27.11.2018).

Juza Marta (2016), *Internet w życiu społecznym – nadzieje, obawy, krytyka*, „Studia Socjologiczne", nr 1(220), s. 199–221.

Kacperczyk Anna (2007), *Badacz i jego poszukiwania w świetle „Analizy Sytuacyjnej" Adele E. Clarke*, „Przegląd Socjologii Jakościowej", t. III, nr 2, s. 5–32.

Kacperczyk Anna (2011), *Obiekt graniczny (Boundary object)*, [w:] K.T. Konecki, P. Chomczyński (red.), *Słownik socjologii jakościowej*, Wydawnictwo Difin, Warszawa, s. 192.

Kacperczyk Anna (2014), *Autoetnografia – technika, metoda, nowy paradygmat? O metodologicznym statusie autoetnografii*, „Przegląd Socjologii Jakościowej", t. X, nr 3, s. 32–74.

Kacperczyk Anna (2016), *Społeczne światy. Teoria – empiria – metody badań: na przykładzie społecznego świata wspinaczki*, Wydawnictwo Uniwersytetu Łódzkiego, Łódź.

Kacperczyk Anna (2017), *Rozum czy emocje? O odmianach autoetnografii oraz epistemologicznych przepaściach i pomostach między nimi*, „Kultura i Społeczeństwo", nr 3(LXI) s. 127–148.

Kaczorowska-Spychalska Dominika (2019), *UŁ komentuje: Sztuczna inteligencja i biznes*, https://www.uni.lodz.pl/aktualnosc/szczegoly/ul-komentuje-sztuczna-inteligencja-i-biznes-zwiazek-prawie-doskonaly (dostęp: 4.07.2019).

Kaggle (2017a), *2017 Kaggle Machine Learning & Data Science Survey*, https://www.kaggle.com/kaggle/kaggle-survey-2017 (dostęp: 19.07.2019).

Kaggle (2017b), *2017: The State of Data Science & Machine Learning*, https://www.kaggle.com/surveys/2017 (dostęp: 27.03.2018).

Kaggle (2018a), *2018 Kaggle Machine Learning & Data Science Survey*, https://www.kaggle.com/kaggle/kaggle-survey-2018 (dostęp: 19.07.2019).

Kaggle (2018b), *Kaggle Days*, https://www.kaggledays.com/ (dostęp: 25.04.2018).

Kallenberg Michiel, Petersen Kersten, Nielsen Mads, Ng Andrew, Diao Pengfei, Igel Christian, Vachon Celine M., Holland Katharina, Winkel Rikke Rass, Karssemeijer Nico, Lillholm Martin (2016), *Unsupervised Deep Learning Applied to Breast Density Segmentation and Mammographic Risk Scoring*, „IEEE Transactions on Medical Imaging", vol. 35(5), s. 1322–1331, https://doi.org/10.1109/TMI.2016.2532122

Kane Michael J., Emerson John W., Weston Stephen (2013), *Scalable Strategies for Computing with Massive Data*, „Journal of Statistical Software", vol. 55(14), s. 1–19, https://doi.org/10.18637/jss.v055.i14

Kapur Manu, Rummel Nikol (2012), *Productive failure in learning from generation and invention activities*, „Instructional Science", vol. 40(4), s. 645–650, https://doi.org/10.1007/s11251-012-9235-4

karupakalas (2018), *IDE alternatives for R programming (RStudio, IntelliJ IDEA, Eclipse, Visual Studio)*, https://datascience.stackexchange.com/a/28853 (dostęp: 11.07.2019).

KDnuggets (b.d.), *About KDnuggets*, https://www.kdnuggets.com/about/index.html (dostęp: 5.01.2018).

Keras (2019), *Backend utilities*, https://keras.io/backend/ (dostęp: 17.07.2019).

Ketkar Nikhil (2017), *Deep Learning with Python*, Apress, Berkeley, https://link.springer.com/book/10.1007/978-1-4842-2766-4 (dostęp: 11.07.2019).

Kędzierski Robert (2018), *Niepokorni nie mogą nawet jeździć pociągiem. Pierwsze ofiary chińskiego systemu oceny obywateli*, http://next.gazeta.pl/next/7,156830,23474436,niepokorni-nie-moga-nawet-jezdzic-pociagiem-pierwsze-ofiary.html (dostęp: 18.06.2018).

King John, Magoulas Roger (2016), *2016 Data Science Salary Survey*, https://www.oreilly.com/radar/2016-data-science-salary-survey-results/ (dostęp: 11.07.2019).

Kitchin Rob (2014), *Big Data, new epistemologies and paradigm shifts*, „Big Data & Society", vol. 1(1), s. 1–12, https://doi.org/10.1177/2053951714528481

Kling Rob, Gerson Elihu M. (1978), *Patterns of Segmentation and Intersection in the Computing World*, „Symbolic Interaction", vol. 1(2), s. 24–43, https://doi.org/10.1525/si.1978.1.2.24

Knapik Rozalia (2018), *Sztuczny Bóg. Wizerunki technologicznej Osobliwości w (pop)kulturze*, Instytut Kultury Popularnej, Poznań.

Knox Hannah, Nafus Dawn (red.) (2018), *Ethnography for a data-saturated world*, Manchester University Press, Manchester.

Kobielus James (2017), *7 Ways to Get High-Quality Labeled Training Data at Low Cost*, https://www.kdnuggets.com/2017/06/acquiring-quality-labeled-training-data.html (dostęp: 5.02.2019).

Koch Therese (2018), *NIPS AI Conference to Continue Laughing about Nipples at the Expense of Women in Tech*, https://medium.com/@therese.koch1/nips-ai-conference-to-continue-laughing-about-nipples-at-the-expense-of-women-in-tech-8c0fa74b1ec4 (dostęp: 12.02.2019).

Kodołamacz.pl i Centrum Zarządzania Innowacjami i Transferem Technologii Politechniki Warszawskiej (2017), *Zawody Przyszłości #1*, http://kodolamacz.pl/zawodyprzyszlosci/ (dostęp: 11.06.2017).

Koloch Grzegorz, Grobelna Karolina, Zakrzewska-Szlichtyng Karolina, Kamiński Bogumił, Kaszyński Daniel (2017), *Intensywność wykorzystania danych w gospodarce a jej rozwój – analiza diagnostyczna*, Ministerstwo Cyfryzacji Rzeczypospolitej Polskiej, Warszawa, https://mc.bip.gov.pl/rok-2017/analiza-diagnostyczna-intesywnosc-wykorzystania-danych-w-gospodarce-a-jej-rozwoj.html (dostęp: 12.02.2019).

Komisja Europejska (2020), White Paper on Artificial Intelligence – A European approach to excellence and trust, COM(2020) 65 final.

Komisja Europejska (2021), Wniosek w sprawie rozporządzenia Parlamentu Europejskiego i Rady ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze Unii, COM(2021) 206 final.

Komisja Europejska (2022a), Akt o rynkach cyfrowych: gwarancja sprawiedliwych i otwartych rynków cyfrowych, https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/digital-markets-act-ensuring-fair-and-open-digital-markets_en (dostęp: 8.02.2022).

Komisja Europejska (2022b), Akt o usługach cyfrowych: bezpieczeństwo i odpowiedzialność uczestników interakcji w internecie, https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/digital-services-act-ensuring-safe-and-accountable-online-environment_pl (dostęp: 8.02.2022).

Komisja Europejska (2022c), *Doskonałość i zaufanie do sztucznej inteligencji*, https://ec.europa.eu/info/strategy/priorities-2019-2024/europe-fit-digital-age/excellence-trust-artificial-intelligence_pl (dostęp: 8.02.2022).

Konecki Krzysztof T. (2000), *Studia z metodologii badań jakościowych. Teoria ugruntowana*, Wydawnictwo Naukowe PWN, Warszawa.

Kopf Dan (2015), *Hadley Wickham, the Man Who Revolutionized R*, https://priceonomics.com/hadley-wickham-the-man-who-revolutionized-r/ (dostęp: 5.01.2018).

Kotuła Sebastian Dawid (2014), *Wstęp do Open Source*, Wydawnictwo Stowarzyszenia Bibliotekarzy Polskich, Warszawa.

Kowalski Jaroslaw, Biele Cezary, Krzysztofek Kazimierz (2019), *Smart Home Technology as a Creator of a Super-Empowered User*, [w:] W. Karwowski, T. Ahram (red.), *Advances in Intelligent Systems and Computing*, Springer International Publishing, Cham, s. 175–180, https://doi.org/10.1007/978-3-030-11051-2_27

Kozinets Robert V. (2003), *The Field behind the Screen: Using Netnography for Marketing Research in Online Communities*, „Journal of Marketing Research", vol. 39(1), s. 61–72, https://doi.org/10.1509/jmkr.39.1.61.18935

Kozyrkov Cassie (2018), *Top 10 roles in AI and data science*, https://medium.com/hackernoon/top-10-roles-for-your-data-science-team-e7f05d90d961 (dostęp: 23.10.2018).

Krawczyk Stanisław, Migdał Piotr (2011), *Zespół Aspergera, nauki ścisłe i kultura nerdów*, „Rocznik Kognitywistyczny", t. 5, s. 93–101, https://doi.org/10.4467/20843895RK.12.011.0415

Krawiec Krzysztof (2003), *Uczenie maszynowe i sieci neuronowe*, Wydawnictwo Politechniki Poznańskiej, Poznań.

Kriegel Alex, Trukhnov Boris M. (2011), *Discovering SQL: A Hands-On Guide for Beginners*, Wrox, Indianapolis.

Kromme Jeroen (2017), *Python & R vs. SPSS & SAS*, http://www.theanalyticslab.nl/2017/03/18/python-r-vs-spss-sas/ (dostęp: 26.10.2018).

Kroplewski Robert, Staniłko Jan, Ciesielski Michał, Flakiewicz Paweł, Jarzewski Andrzej, Kroszczyńska Elżbieta, Lubos Beata, Podgórska Anna, Pukaluk Michał, Pytko Tomasz, Romaniec Hubert, Wancio Agata, Stefaniak Sylwia, Zaczek Agata (2019), *Polityka Rozwoju Sztucznej Inteligencji w Polsce na lata 2019–2027*, Warszawa, https://www.gov.pl/attachment/0aa51cd5-b934-4bcb-8660-bfecb20ea2a9 (dostęp: 23.10.2018).

Kross Sean (2021), *postcards: Create Beautiful, Simple Personal Websites*, https://cran.r-project.org/package=postcards (dostęp: 26.10.2018).

Krzemiński Ireneusz (1986), *Symboliczny interakcjonizm i socjologia*, Państwowe Wydawnictwo Naukowe, Warszawa.

Krzysztofek Kazimierz (2010), *„Fragteracja" złożonych systemów społecznych – kilka pytań i hipotez badawczych*, „Studia i Prace Uniwersytetu Ekonomicznego w Krakowie", nr 13, s. 30–47.

Krzysztofek Kazimierz (2011), *W stronę maszyn społecznych. Jaka będzie socjologia, której nie znamy?*, „Studia Socjologiczne", nr 2(201), s. 123–145.

Krzysztofek Kazimierz (2012), *Big Data Society. Technologie samozapisu i samopokazu: ku humanistyce cyfrowej*, „Transformacje", nr 1–4(72–75), s. 223–257.

Krzysztofek Kazimierz (2017), *Kierunki ewaluacji technologii cyfrowych w działaniu społecznym. Próba systematyzacji problemu*, „Studia Socjologiczne", nr 1, s. 195–224.

Krzysztofek Kazimierz (2018), *Prévoir – Savoir – Pouvoir, czyli od przewidywania do wiedzy i władzy*, „Stan Rzeczy", nr 14, s. 17–39.

Kuhn Thomas S. (2001), *Struktura rewolucji naukowych*, Fundacja Aletheia, Warszawa.

Kulisiewicz Marcin, Kazienko Przemysław, Szymanski Boleslaw K., Michalski Radosław (2018), *Entropy Measures of Human Communication Dynamics*, „Scientific Reports", vol. 8(1), 15697, https://doi.org/10.1038/s41598-018-32571-3

Kuncewicz Łukasz (2019), *You're ready to become a Data Scientist if…*, https://www.linkedin.com/posts/kuncewicz_youre-ready-to-become-a-data-scientist-if-activity-6566757953188311040-Ie3C (dostęp: 22.09.2019).

Kurczewska Joanna (1997), *Technokraci i ich świat społeczny*, Wydawnictwo Instytutu Filozofii i Socjologii PAN, Warszawa.

Kurzweil Ray (2016), *Nadchodzi Osobliwość. Kiedy człowiek przekroczy granice biologii*, Kurhaus Publishing, Warszawa.

Kuźba Michał, Biecek Przemysław (2020), *What Would You Ask the Machine Learning Model? Identification of User Needs for Model Explanations Based on Human-Model Conversations*, http://arxiv.org/abs/2002.05674 (dostęp: 26.10.2020).

Kwiatkowska Agnieszka (2017) *„Hańba w Sejmie" – zastosowanie modeli generatywnych do analizy debat parlamentarnych*, „Przegląd Socjologii Jakościowej", t. XIII, nr 2, s. 82–109.

Kwon Ohbyung, Lee Namyeon, Shin Bongsik (2014), *Data quality management, data usage experience and acquisition intention of big data analytics*, „International Journal of Information Management", vol. 34(3), s. 387–394, https://doi.org/10.1016/j.ijinfomgt.2014.02.002

Laney Douglas (2001), *3-D data management: Controlling data volume, velocity and variety*, http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf (dostęp: 14.03.2016).

Lapowsky Issie (2017), *What Did Cambridge Analytica Really Do for Trump's Campaign?*, https://www.wired.com/story/what-did-cambridge-analytica-really-do-for-trumps-campaign/ (dostęp: 7.02.2019).

Lapowsky Issie (2019), *How Cambridge Analytica Sparked the Great Privacy Awakening*, https://www.wired.com/story/cambridge-analytica-facebook-privacy-awakening/ (dostęp: 17.03.2019).

Lardinois Frederic (2015), *As Kubernetes Hits 1.0, Google Donates Technology To Newly Formed Cloud Native Computing Foundation*, https://techcrunch.com/2015/07/21/as-kubernetes-hits-1-0-google-donates-technology-to-newly-formed-cloud-native-computing-foundation-with-ibm-intel-twitter-and-others/ (dostęp: 5.05.2019).

Larose Daniel T. (2005), *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons, Hoboken.

Larose Daniel T. (2012), *Metody i modele eksploracji danych*, Wydawnictwo Naukowe PWN, Warszawa.

Larson Jeff, Mattu Surya, Kirchner Lauren, Angwin Julia (2016), *How We Analyzed the COMPAS Recidivism Algorithm*, https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm (dostęp: 30.04.2018).

Lasek Mirosława (2002), *Data Mining. Zastosowania w analizach i ocenach klientów bankowych*, Oficyna Wydawnicza „Zarządzanie i Finanse", Warszawa.

Lasek Mirosława (2007), *Metody Data Mining w analizowaniu i prognozowaniu kondycji ekonomicznej przedsiębiorstw: zastosowania SAS Enterprise Miner*, Wydawnictwo Difin, Warszawa.

Lassiter Luke Eric (2005), *The Chicago Guide to Collaborative Ethnography*, The University of Chicago Press, Chicago–London.

Latour Bruno (2013), *Technologia jako utrwalone społeczeństwo*, „AVANT. Pismo Awangardy Filozoficzno-Naukowej", nr 4(1), s. 17–48.

Lazer David, Pentland Alex, Adamic Lada, Aral Sinan, Barabási Albert-László, Brewer Devon, Christakis Nicholas, Contractor Noshir, Fowler James, Gutmann Myron, Jebara Tony, King Gary, Macy Michael, Roy Deb, Van Alstyne Marshall (2009), *Computational Social Science*, „Science", no. 323, s. 721–723.

Le Quoc V., Ranzato Marc'Aurelio, Monga Rajat, Devin Matthieu, Chen Kai, Corrado Greg S., Dean Jeff, Ng Andrew (2011), *Building high-level features using large scale unsupervised learning*, http://arxiv.org/abs/1112.6209 (dostęp: 23.02.2020).

LearnDataSci (2018), *Top Data Science Online Courses in 2018*, https://www.learndatasci.com/best-data-science-online-courses/ (dostęp: 21.02.2018).

LeCun Yann (b.d.), *Biographical Sketch*, http://yann.lecun.com/ex/bio.html (dostęp: 12.09.2018).

LeCun Yann, Bottou Léon, Bengio Yoshua, Haffner Patrick (1998), *Gradient-based learning applied to document recognition*, „Proceedings of the IEEE", vol. 86(11), s. 2278–2324, https://doi.org/10.1109/5.726791

LeCun Yann, Boser B., Denker J.S., Henderson D., Howard R.E., Hubbard W., Jackel L.D. (1989), *Backpropagation Applied to Handwritten Zip Code Recognition*, „Neural Computation", vol. 1(4), s. 541–551, https://doi.org/10.1162/neco.1989.1.4.541

Lee Adrian (2016), *Geoffrey Hinton, the 'godfather' of deep learning, on AlphaGo*, https://www.macleans.ca/society/science/the-meaning-of-alphago-the-ai-program-that-beat-a-go-champ/ (dostęp: 23.11.2018).

Lee Honglak, Pham Peter, Largman Yan, Ng Andrew (2009), *Unsupervised feature learning for audio classification using convolutional deep belief networks*, [w:] Y. Bengio, D. Schuurmans, J.D. Lafferty, C.K.I. Williams, A. Culotta (red.), *Advances in Neural Information Processing Systems 22*, Curran Associates, Inc., Vancouver, s. 1096–1104.

Leek Jeffrey (2016), *How to be a modern scientist*, http://leanpub.com/modern-scientist (dostęp: 12.09.2018).

Lerman Rachel, O'Brien Matt (2019), *Google's privacy push gets a mixed reception*, „Washington Times", 8 maja, https://www.washingtontimes.com/news/2019/may/8/googles-privacy-promises-dont-sway-many-experts/ (dostęp: 20.05.2019).

Levy Steven (2019), *Cambridge Analytica, Whistle-Blowers, and Tech's Dark Appeal*, https://www.wired.com/story/cambridge-analytica-whistle-blowers-and-techs-dark-appeal/ (dostęp: 15.10.2019).

Lewis-Kraus Gideon (2017), *Budowniczowie wieży Googel*, „Przekrój", nr 2(3557), s. 151–162.

Li Jun (2008), *Ethical Challenges in Participant Observation: A Reflection on Ethnographic Fieldwork*, „The Qualitative Report", vol. 13(1), s. 100–115.

Li Michael, Paczuski Paul (2017), *Ranked: 15 Python packages for Data Science*, http://blog.thedataincubator.com/wp-content/uploads/2017/04/Ranked--15-Python-Packages-for-Data-Science.pdf (dostęp: 20.05.2019).

Linden Gregory D., Jacobi Jennifer A., Benson Eric A. (2001), *Collaborative recommendations using item-to-item similarity mappings*, United States Patent US-6266649-B1, https://image-ppubs.uspto.gov/dirsearch-public/print/downloadPdf/6266649 (dostęp: 15.10.2019).

Lohr Steve (2009), *For Today's Graduate, Just One Word: Statistics*, „The New York Times", 5 sierpnia, http://www.nytimes.com/2009/08/06/technology/06stats.html?module=ArrowsNav&contentCollection=Technology&action=keypress&region=FixedLeft&pgtype=article (dostęp: 15.02.2018).

López Gustavo, Quesada Luis, Guerrero Luis A. (2018), *Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces BT – Advances in Human Factors and Systems Interaction*, [w:] I.L. Nunes (red.), *International Conference on Applied Human Factors and Ergonomics*, Springer International Publishing, Cham, s. 241–250.

Lorenz Chris (2012), *If You're So Smart, Why Are You under Surveillance? Universities, Neoliberalism, and New Public Management*, „Critical Inquiry", vol. 38(3), s. 599–629, https://doi.org/10.1086/664553

Lorenzi Jean-Herve, Berrebi Mickaël (2019), *Przyszłość naszej wolności. Czy należy rozmontować Google'a i kilku innych?*, Państwowy Instytut Wydawniczy, Warszawa.

Loukides Mike (2010), *What is data science?*, https://www.oreilly.com/ideas/what-is-data-science (dostęp: 8.09.2016).

Lowrie Ian (2016), *Caring for Computers: How Russian Data Scientists Refashion Their Laptops*, „Anthropology Now", vol. 8(2), s. 25–33, https://doi.org/10.1080/19428200.2016.1202578

Lowrie Ian (2017), *Algorithmic rationality: Epistemology and efficiency in the data sciences*, „Big Data & Society", vol. 4(1), s. 1–13, https://doi.org/10.1177/2053951717700925

Lowrie Ian (2018), *Becoming a real data scientist. Expertise, flexibility and lifelong learning*, [w:] H. Knox, D. Nafus (red.), *Ethnography for a data-saturated world*, Manchester University Press, Manchester, s. 62–81.

Lutyński Jan (1974), *Uwagi na temat typologii wywiadów*, maszynopis.

Lyotard Jean-François (1997), *Kondycja ponowoczesna: raport o stanie wiedzy*, Fundacja Aletheia, Warszawa.

Mac Ryan (2018), *Cambridge Analytica Data Scientist Aleksandr Kogan Wants You To Know He's Not A Russian Spy*, https://www.buzzfeednews.com/article/ryanmac/facebook-cambridge-analytica-aleksandr-kogan-not-spy (dostęp: 1.02.2019).

Majek Karol (2020), *Polskie blogi o sztucznej inteligencji i analizie danych*, https://deepdrive.pl/polskie-blogi-o-sztucznej-inteligencji-i-analizie-danych/ (dostęp: 20.04.2020).

Mallan Kerry Margaret, Singh Parlo, Giardina Natasha (2010), *The challenges of participatory research with 'tech-savvy' youth*, „Journal of Youth Studies", vol. 13(2), s. 255–272, https://doi.org/10.1080/13676260903295059

Manhart Klaus (1996), *Artificial Intelligence Modelling: Data Driven and Theory Driven Approaches*, [w:] K.G. Troitzsch, U. Mueller, G.N. Gilbert, J.E. Doran (red.), *Social Science Micro Simulation*, Springer, Berlin, s. 416–431, https://doi.org/10.1007/978-3-662-03261-9_19

Mannes John (2017), *Geofferey Hinton was briefly a Google intern in 2012 because of bureaucracy*, https://techcrunch.com/2017/09/14/geoffrey-hinton-was-briefly-a-google-intern-in-2012-because-of-bureaucracy/ (dostęp: 23.11.2018).

Manyika James, Chui Michael, Brown Brad, Bughin Jacques, Dobbs Richard, Roxburgh Charles, Hung Byers Angela (2011), *Big data: The next frontier for innovation, competition, and productivity*, McKinsey Global Institute, Seattle, https://www.mckinsey.com/~/media/mckinsey/business%20functions/mckinsey%20digital/our%20insights/big%20data%20the%20next%20frontier%20for%20innovation/mgi_big_data_full_report.pdf (dostęp: 5.11.2018).

Marcus George E. (1995), *Ethnography in/of the World System: The Emergence of Multi-Sited Ethnography*, „Annual Review of Anthropology", no. 24, s. 95–117.

Marczuk Piotr, Mieczkowski Piotr, Calini Leonardo, Paszcza Bartosz (2019), *Iloraz sztucznej inteligencji. Potencjał AI w polskiej gospodarce*, Fundacja Digital Poland, Warszawa, https://www.digitalpoland.org/assets/publications/iloraz-sztucznej-inteligencji/iloraz-sztucznej-inteligencji-edycja-2-2019.pdf (dostęp: 23.11.2018).

Marr Bernard (2016), *What Is The Difference Between Artificial Intelligence And Machine Learning?*, „Forbes", 6 grudnia, https://www.forbes.com/sites/bernardmarr/2016/12/06/what-is-the-difference-between-artificial-intelligence-and-machine-learning/#23890af2742b (dostęp: 14.02.2018).

Maslow Abraham H. (1966), *The Psychology of Science. A Reconnaissance*, Gateway Editions, South Bend.

Mason Hilary (2010), *A Taxonomy of Data Science*, http://www.dataists.com/2010/09/a-taxonomy-of-data-science/ (dostęp: 30.07.2018).

masterindatascience (2018), *Top 23 Schools with Data Science Master's Programs*, https://www.mastersindatascience.org/schools/23-great-schools-with-masters-programs-in-data-science/ (dostęp: 21.02.2018).

Matloff Norman (2017), *Norm Matloff's answer to Will Python take over R?*, https://www.quora.com/Will-Python-take-over-R/answer/Norm-Matloff (dostęp: 8.09.2018).

Matloff Norman (2020), *TidyverseSkeptic: An opinionated view of the Tidyverse «dialect» of the R language*, https://github.com/matloff/TidyverseSkeptic#readme (dostęp: 23.10.2020).

Matplotlib (2018), *Matplotlib: Python plotting – Matplotlib 3.0.2 documentation*, https://matplotlib.org/3.0.2/index.html (dostęp: 16.09.2018).

McCarthy John, Minsky Marvin L., Rochester Nathaniel, Shannon Claude E. (1955), *Darmouth AI Project Proposal*, Darmouth University, Darmouth, http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html (dostęp: 23.10.2020).

McCulloch Gretchen (2019), *Coding Is for Everyone – as Long as You Speak English*, https://www.wired.com/story/coding-is-for-everyoneas-long-as-you-speak-english/ (dostęp: 10.09.2019).

McKinney Wes (2010), *Data Structures for Statistical Computing in Python*, [w:] S. van der Walt, J. Millman (red.), *Proceedings of the 9th Python in Science Conference*, Python in Science Conference, Austin, s. 51–56, https://doi.org/10.25080/MAJORA-92BF1922-00A

McKinney Wes (2011), *pandas: a Foundational Python Library for Data Analysis and Statistics*, [w:] *PyHPC 2011: Python for High Performance and Scientific Computing*, PyHPC, Seattle, https://www.dlr.de/sc/Portaldata/15/Resources/dokumente/pyhpc2011/submissions/pyhpc2011_submission_9.pdf (dostęp: 23.10.2020).

McKinney Wes (2012), *Python for Data Analysis. Data Wrangling With Pandas, NumPy, And IPython*, O'Reilly, Sebastopol.

McKinney Wes (2018), *About – Wes McKinney*, http://wesmckinney.com/pages/about.html (dostęp: 4.12.2018).

Meetup (b.d.), https://secure.meetup.com/meetup_api (dostęp: 19.07.2019).

Merity Stephen (2017), *Bias is not just in our datasets, it's in our conferences and community*, https://smerity.com/articles/2017/bias_not_just_in_datasets.html (dostęp: 10.02.2019).

Merritt Jonathan (2018), *Sztuczna inteligencja zagrożeniem dla chrześcijaństwa?*, „Miesięcznik Znak", nr 762, s. 52–57.

Michel Jean Baptiste, Shen Yuan Kui, Aiden Aviva Presser, Veres Adrian, Gray Matthew K., Pickett Joseph P., Hoiberg Dale, Clancy Dan, Norvig Peter, Orwant Jon, Pinker Steven, Nowak Martin A., Lieberman Aiden Erez (2011), *Quantitative Analysis of Culture Using Millions of Digitized Books*, „Science", vol. 331(6014), s. 176–182, https://doi.org/10.1126/science.1199644

Microsoft (2019), *Microsoft R Open: The Enhanced R Distribution*, https://mran.microsoft.com/open (dostęp: 19.07.2019).
Microsoft Azure (2019), *Azure Databricks*, https://azure.microsoft.com/pl-pl/services/databricks/ (dostęp: 3.06.2019).
Microsoft Research (2019a), *People: danah boyd*, https://www.microsoft.com/en-us/research/people/dmb/ (dostęp: 11.02.2019).
Microsoft Research (2019b), *People: Kate Crawford*, https://www.microsoft.com/en-us/research/people/kate/#!publications (dostęp: 11.02.2019).
Migdał Piotr (2014), *Symmetries and self-similarity of many-body wavefunctions*, http://arxiv.org/abs/1412.6796 (dostęp: 11.02.2019).
Migdał Piotr (2016), *From Science to Data Science, a Comprehensive Guide for Transition*, https://www.kdnuggets.com/2016/04/data-science-comprehensive-guide-transition.html (dostęp: 3.02.2018).
Migdał Piotr (2017), *After PyData Warsaw 2017*, https://p.migdal.pl/2017/11/15/after-pydata-warsaw-2017.html (dostęp: 1.12.2017).
Migdał Piotr, Jakubanis Rafał (2018), *Keras or PyTorch as your first deep learning framework*, https://deepsense.ai/keras-or-pytorch/ (dostęp: 17.07.2019).
Miller Justin J. (2013), *Graph database applications and concepts with Neo4j*, „Proceedings of the Southern Association for Information Systems Conference, Atlanta", no. 2324, s. 141–147.
Miller Tim (2019), *Explanation in artificial intelligence: Insights from the social sciences*, „Artificial Intelligence", no. 267, s. 1–38, https://doi.org/10.1016/J.ARTINT.2018.07.007
Miloslavskaya Natalia, Tolstoy Alexander (2016), *Big Data, Fast Data and Data Lake Concepts*, „Procedia Computer Science", no. 88, s. 300–305, https://doi.org/10.1016/j.procs.2016.07.439
Mims Christopher (2018), *Who Has More of Your Personal Data Than Facebook? Try Google*, „The Wall Street Journal", 22 kwietnia, https://www.wsj.com/articles/who-has-more-of-your-personal-data-than-facebook-try-google-1524398401 (dostęp: 9.05.2018).
Minelli Michael, Dhiraj Ambiga, Chambers Michele (2013), *Big Data, Big Analytics. Emerging business intelligence and analytic trends for today's businesses*, John Wiley & Sons, New Jersey.
Ministerstwo Cyfryzacji RP (2018a), *RODO – Informator*, https://www.gov.pl/cyfryzacja/rodo-informator (dostęp: 12.02.2019).
Ministerstwo Cyfryzacji RP (2018b), *Założenia do strategii AI w Polsce. Plan działań Ministerstwa Cyfryzacji*, https://www.gov.pl/documents/31305/436699/Założenia_do_strategii_AI_w_Polsce_-_raport.pdf/a03eb166-0ce5-e53c-52a4-3bfb903edf0a (dostęp: 12.02.2019).
Misal Disha (2018), *PyTorch vs Keras: Who Suits You The Best*, „Analytics India Magazine", https://www.analyticsindiamag.com/pytorch-vs-keras-who-suits-you-the-best/ (dostęp: 19.02.2019).

Mohamed Shakir, Png Marie-Therese, Isaac William (2020), *Decolonial AI: Decolonial Theory as Sociotechnical Foresight in Artificial Intelligence*, „Philosophy & Technology", vol. 33(4), s. 659–684, https://doi.org/10.1007/s13347-020-00405-8

Molnar Christoph (2018), *iml: An R package for Interpretable Machine Learning*, „Journal of Open Source Software", vol. 3(26), 786, https://doi.org/10.21105/joss.00786

Morrison Daniel R. (2016), *Mapping to Make Sense of Messy Worlds*, „Symbolic Interaction", vol. 39(3), s. 519–521, https://doi.org/10.1002/symb.237

Mortimer Steven (2018), *Most Starred R Packages on GitHub*, https://stevenmortimer.com/most-starred-r-packages-on-github/ (dostęp: 11.07.2018).

Morzy Tadeusz (2013), *Eksploracja danych. Metody i algorytmy*, Wydawnictwo Naukowe PWN, Warszawa.

Mökander Jakob, Juneja Prathm, Watson David S., Floridi Luciano (2022), *The US Algorithmic Accountability Act of 2022 vs. The EU Artificial Intelligence Act: what can they learn from each other?*, „Minds and Machines", no. 32, s. 751–758, https://doi.org/10.1007/s11023-022-09612-y

Mróz Maciej (2020), *Nowy renesans, czyli SI po ludzku*, https://www.sztucznainteligencja.org.pl/nowy-renesans-czyli-si-po-ludzku/ (dostęp: 10.03.2020).

Muenchen Robert A. (2014), *Why R is Hard to Learn*, http://r4stats.com/articles/why-r-is-hard-to-learn/ (dostęp: 21.03.2018).

Muenchen Robert A. (2019), *The Popularity of Data Science Software*, http://r4stats.com/articles/popularity/ (dostęp: 1.07.2018).

Muûlu Kıvanç, Brun Yuriy, Holmes Reid, Ernst Michael D., Notkin David (2012), *Speculative analysis of integrated development environment recommendations*, „ACM SIGPLAN Notices", vol. 47(10), s. 669–682, https://doi.org/10.1145/2398857.2384665

Müller Kirill, Walthert Lorenz (2020), *styler: Non-Invasive Pretty Printing of R Code*, https://cran.r-project.org/package=styler (dostęp: 10.03.2020).

Myoo Sidney (2013), *Ontoelektronika*, Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków.

Naur Peter (1974), *Concise Survey of Computer Methods*, Studentlitteratur, Lund.

Neff Gina, Tanweer Anissa, Fiore-Gartland Brittany, Osburn Laura (2017), *Critique and Contribute: A Practice-Based Framework for Improving Critical Data Studies and Data Science*, „Big Data", vol. 5(2), s. 85–97, https://doi.org/10.1089/big.2016.0050

Netflix (2019), *How Netflix's Recommendations System Works*, https://help.netflix.com/en/node/100639 (dostęp: 9.07.2019).

NeurIPS (2018), *Neural Information Processing Systems Foundation. Code of Conduct*, https://neurips.cc/public/CodeOfConduct (dostęp: 12.02.2019).

Neuwirth Erich (2014), *RColorBrewer: ColorBrewer Palettes*, https://cran.r-project.org/package=RColorBrewer (dostęp: 9.07.2019).

Newton Casey (2019), *The Trauma Floor. The secret lives of Facebook moderators in America*, https://www.theverge.com/2019/2/25/18229714/cognizant-facebook-content-moderator-interviews-trauma-working-conditions-arizona (dostęp: 26.02.2019).

Ng Andrew (2017), *AI is the new electricity*, [w:] *AI Frontiers. Applied Deep Learning*, Santa Clara, https://nov2017.aifrontiers.com/#speakers (dostęp: 16.03.2018).

Ng Andrew (2018a), *About – Andrew Ng*, https://www.andrewng.org/about/ (dostęp: 21.01.2018).

Ng Andrew (2018b), *Yann LeCun Interview – Foundations of Convolutional Neural Networks*, https://www.coursera.org/lecture/convolutional-neural-networks/yann-lecun-interview-4PnfT (dostęp: 23.10.2018).

Ng Andrew (b.d.), *Andrew Ng's Home page*, http://ai.stanford.edu/~ang/original-Homepage.html (dostęp: 21.01.2018).

Ng Andrew, Widom Jennifer (2014), *Origins of the Modern MOOC (xMOOC)*, [w:] F.M. Hollands, D. Tirthali (red.), *MOOCs: excpectations and reality. Full report*, Columbia University, New York, s. 34–47.

Ng Andrew Y., Zheng Alice X., Jordan Michael I. (2001), *Stable algorithms for link analysis*, [w:] *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval – SIGIR '01*, ACM Press, New York, s. 258–266.

Nguyen Clinton (2016), *China might use data to create a score for each citizen based on how trustworthy they are*, https://www.businessinsider.com/china-social-credit-score-like-black-mirror-2016-10?IR=T (dostęp: 21.01.2019).

Nicolaus Henke, Bughin Jacques, Chui Michael, Manyika James, Saleh Tamim, Wiesman Bill, Sethupathy Guru (2016), *The age of analytics: Competing in a data-driven world*, https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/the-age-of-analytics-competing-in-a-data-driven-world (dostęp: 21.01.2018).

Niedbalski Jakub (red.) (2014), *Metody i techniki odkrywania wiedzy. Narzędzia CAQDAS w procesie analizy danych jakościowych*, Wydawnictwo Uniwersytetu Łódzkiego, Łódź.

Nissenbaum Helen (2001), *How Computer Systems Embody Values*, „IEEE Computer", no. 120, s. 118–119.

Norén Laura (2018), *Ethics of Data Science*, syllabus, New York University, New York, https://cdt.org/wp-content/uploads/2018/07/Ethics-of-Data-Science.pdf (dostęp: 14.12.2018).

Northpointe Inc. (2012), *Practitioners Guide to COMPAS*, http://www.northpointeinc.com/files/technical_documents/FieldGuide2_081412.pdf (dostęp: 3.04.2017).

Norvig Peter (1987), *A Unified Theory of Inference for Text Understanding*, University of California, Berkeley.

Norvig Peter (2001), *Teach Yourself Programming in Ten Years*, http://norvig.com/21-days.html (dostęp: 13.04.2018).

Norvig Peter (2012), *Colorless Green Ideas Learn Furiously: Chomsky and the Two Cultures of Statistical Learning*, „Significance", vol. 9(4), s. 30–33, https://doi.org/10.1111/j.1740-9713.2012.00590.x

Norvig Peter (2018), *Peter Norvig – Resume*, http://norvig.com/resume.html (dostęp: 4.03.2018).

Nowosad Jakub (2019), *Elementarz programisty. Wstęp do programowania używając R*, Wydawnictwo Space A, Poznań.

NumFOCUS (2018), *NumFOCUS: Open Code = Better Science*, https://numfocus.org/ (dostęp: 3.12.2018).

Nunns James (2017), *How Python rose to the top of the data science world*, „Computer Business Review", https://techmonitor.ai/technology/data/python-rose-top-data-science-world (dostęp: 2.10.2018).

Nurczyk Ewelina, Ramza Barbara (2017), *Zawody przyszłości: Data Scientist*, „Kariera w Finansach i Bankowości" nr 2017/2018, s. 26–30.

NYU Center for Data Science (2013), *Yann LeCun Appointed Director of NYU Center for Data Science*, https://cds.nyu.edu/yann-lecun-appointed-director-of-nyu-center-for-data-science/ (dostęp: 19.09.2018).

O'Connor Brendan O., Bamman David, Smith Noah A. (2011), *Computational Text Analysis for Social Science: Model Assumptions and Complexity*, „Second Workshop on Computational Social Science and Wisdom of the Crowds" (NIPS 2011), s. 1–8, http://people.cs.umass.edu/~wallach/workshops/nips2011css/papers/OConnor.pdf (dostęp: 2.10.2018).

O'Neil Cathy (2013), *On Being a Data Skeptic*, O'Reilly, Sebastopol.

O'Neil Cathy (2017), *Broń matematycznej zagłady. Jak algorytmy zwiększają nierówności i zagrażają demokracji*, Wydawnictwo Naukowe PWN, Warszawa.

O'Neil Cathy, Schutt Rachel (2015), *Badanie danych: raport z pierwszej linii działań*, Wydawnictwo Helion, Gliwice.

Obem Anna (2018), *Nowa afera, stare wyzwania*, https://cyfrowa-wyprawka.org/aktualnosci/nowa-afera-stare-wyzwania (dostęp: 24.04.2018).

Obem Anna (2020), *Co możesz zrobić po obejrzeniu Social Dilemma (poza wyrzuceniem telefonu)?*, https://panoptykon.org/spoleczny-dylemat (dostęp: 12.10.2020).

OECD (2015), *Does Math Make You Anxious?*, „PISA in Focus", no. 02, s. 1–4, https://doi.org/10.1787/5js6b2579tnx-en

Ohlhorst Frank J. (2013), *Big Data Analytics: Turning Big Data into Big Money*, Wiley, New York.

Oliphant Travis E. (2006), *A guide to NumPy*, Trelgol Publishing, USA.

Oliphant Travis E. (2007), *Python for Scientific Computing*, „Computing in Science & Engineering", vol. 9(3), s. 10–20.

Oliphant Travis E., Manduca Armando, Ehman Richard L., Greenleaf James F. (2001), *Complex-valued stiffness reconstruction for magnetic differential equation*, „Magnetic Resonance in Medicine", no. 45, s. 299–310, https://doi.org/10.1002/1522-2594(200102)45:2<299::AID-MRM1039>3.0.CO;2-O

Olszewski Adrian (2017), *Will Python take over R?*, https://www.quora.com/Will-Python-take-over-R/answer/Adrian-Olszewski- (dostęp: 18.09.2018).

Olszewski Adrian (2018), *Why do so many statisticians not want to become data scientists? Why are they not interested in Big Data?*, https://www.quora.com/Why-do-so-many-statisticians-not-want-to-become-data-scientists-Why-are-they-not-interested-in-Big-Data/answer/Adrian-Olszewski-1 (dostęp: 8.09.2018).

Onalytica (2017), *Big Data 2017: Top 100 Influencers And Brands*, http://www.onalytica.com/wp-content/uploads/2017/05/Onalytica-Big-Data-Top-100-Influencers-and-Brands.pdf (dostęp: 5.01.2019).

Onalytica (2018), *Data Science: Top 100 Influencers, Brands & Publications*, http://www.onalytica.com/wp-content/uploads/2018/04/Onalytica-Data-Science-Top-100-Influencers-Brands-and-Publications.pdf (dostęp: 5.01.2019).

Ooms Jeroen (2014), *The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects*, http://arxiv.org/abs/1403.2805 (dostęp: 5.01.2019).

Oord Aaron van den, Dieleman Sander, Schrauwen Benjamin (2013), *Deep content-based music recommendation*, [w:] C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K.Q. Weinberger (red.), *Advances in Neural Information Processing Systems 26*, Curran Associates, Inc., Lake Tahoe, s. 2643–2651, http://papers.nips.cc/paper/5004-deep-content-based-music-recommendation.pdf (dostęp: 5.01.2019).

Orsini Lauren (2014), *Why Python Makes A Great First Programming Language*, ReadWrite, https://readwrite.com/2014/07/08/what-makes-python-easy-to-learn/ (dostęp: 25.07.2019).

Osowski Stanisław (2006), *Sieci neuronowe do przetwarzania informacji*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa.

Osowski Stanisław (2013), *Metody i narzędzia eksploracji danych*, Wydawnictwo BTC, Warszawa.

Ozimek Adam (2017), *The Paradox Of Robots Taking All Our Jobs*, „Forbes", 28 października, https://www.forbes.com/sites/modeledbehavior/2017/10/28/the-paradox-of-robots-taking-all-our-jobs/#1f8e16501274 (dostęp: 14.02.2018).

Ożóg Maciej (2009), *Transgresje panoptykonu. Nadzór w dobie technologii cyfrowych*, „Kultura Współczesna", nr 2(60), s. 14–30.

Ożóg Maciej (2018), *Życie w krzemowej klatce*, Wydawnictwo Uniwersytetu Łódzkiego, Łódź.

Paharia Rajat (2014), *Lojalność 3.0: jak zrewolucjonizować zaangażowanie klientów i pracowników dzięki big data i rywalizacji*, Wydawnictwo MT Biznes Ltd., Warszawa.

Paliszkiewicz Joanna (2018), *Kreowanie w mediach społecznościowych wizerunku kandydata do pracy na przykładzie portalu LinkedIn*, „Zarządzanie Zasobami Ludzkimi", nr 5(124), s. 79–91.

Paprocki Wojciech (2016), *Cyfryzacja gospodarki i społeczeństwa – szanse i wyzwania dla sektorów infrastrukturalnych*, [w:] W. Paprocki, J. Grajewski, J. Pieriegud (red.), *Cyfryzacja gospodarki i społeczeństwa*, Europejski Kongres Finansowy, Gdańsk, s. 39–57.

Pascnau Razovan, Patraucean Victoria, Precup Doina (2018), *Transylvanian Machine Learning Summer School (TMLSS). Summary of the first edition*, https://drive.google.com/file/d/1Bcuzv9MM-U3CG_QLA63zzGB1E3zfjUgQ/view (dostęp: 25.07.2019).

Paszke Adam, Gross Sam, Chintala Soumith, Chanan Gregory, Yang Edward, DeVito Zachary, Lin Zeming, Desmaison Alban, Antiga Luca, Lerer Adam (2017), *Automatic differentiation in PyTorch*, 31st Conference on Neural Information Processing Systems (NIPS 2017), https://openreview.net/pdf?id=BJJsrmfCZ (dostęp: 14.02.2018).

Paulson Linda Dailey (2007), *Developers Shift to Dynamic Programming Languages*, „Computer", vol. 40(2), s. 12–15.

Pavlicek Antonin, Sudzina Frantisek, Malinova Ludmila (2017), *Impact of Gender and Personality Traits (Bfi-10) on Tech Savviness*, „Idimt-2017 – Digitalization in Management, Society and Economy", no. 46, s. 195–199.

Pebesma Edzer (2018), *Simple Features for R: Standardized Support for Spatial Vector Data*, „The R Journal", vol. 10(1), s. 439–446, https://doi.org/10.32614/RJ-2018-009

Pedregosa Fabian (2015), *Feature extraction and supervised learning on fMRI: from practice to theory*, praca doktorska, Université Pierre et Marie Curie, https://tel.archives-ouvertes.fr/tel-01100921 (dostęp: 25.07.2019).

Pedregosa Fabian (2018), *About me*, http://fa.bianp.net/pages/about.html (dostęp: 4.12.2018).

Pedregosa Fabian, Bach Francis, Gramfort Alexandre (2014), *On the Consistency of Ordinal Regression Methods*, „Journal of Machine Learning Research", vol. 18(55), s. 1–35, http://jmlr.org/papers/v18/15-495.html (dostęp: 25.07.2019).

Pedregos, Fabian, Leblond Rémi, Lacoste-Julien Simon (2017), *Breaking the Nonsmooth Barrier: A Scalable Parallel Method for Composite Optimization*, 31st Conference on Neural Information Processing Systems (NIPS 2017), https://proceedings.neurips.cc/paper/2017/file/072b030ba126b2f4b2374f342be9ed44--Paper.pdf (dostęp: 4.12.2018).

Pedregosa Fabian, Varoquaux Gaël, Gramfort Alexandre, Michel Vincent, Thirion Bertrand, Grisel Olivier, Blondel Mathieu, Prettenhofer Peter, Weiss Ron, Dubourg Vincent, Vanderplas Jake, Passos Alexandre, Cournapeau David, Brucher Matthiew, Perrot Matthieu, Duchesnay Édouard (2011), *Scikit-learn: Machine Learning in {P}ython*, „Journal of Machine Learning Research", no. 12, s. 2825–2830.

Peng Roger D. (2011a), *Reproducible Research in Computational Science*, „Science", vol. 334(6060), s. 1226–1227, https://doi.org/10.1126/science.1213847

Pentland Alex (2009), *Reality Mining of Mobile Communications: Toward a New Deal on Data*, [w:] INSEAD (red.), *The Global Information Technology Report.*

Mobility in a Networked World, World Economic Forum, Genewa, s. 75–80, http://hd.media.mit.edu/wef_globalit.pdf (dostęp: 25.07.2019).

Pentland Alex, Heibeck Tracy (2008), *Honest Signals. How They Shape Our World*, The MIT Press, Cambridge.

Pérez Fernando, Granger Brian E. (2007), *IPython: A System for Interactive Scientific Computing*, „Computing in Science and Engineering", vol. 9(3), s. 21–29, https://doi.org/10.1109/MCSE.2007.53

Pérez Fernando, Granger Brian E. (2015), *Project Jupyter: Computational Narratives as the Engine of Collaborative Data Science*, http://archive.ipython.org/JupyterGrantNarrative-2015.pdf (dostęp: 4.12.2018).

Peterson Brian G., Carl Peter (2020), *PerformanceAnalytics: Econometric Tools for Performance and Risk Analysis*, https://cran.r-project.org/package=PerformanceAnalytics (dostęp: 25.07.2019).

Petrov Dmitry (2017), *How A Data Scientist Can Improve His Productivity – Data Version Control*, https://blog.dataversioncontrol.com/how-a-data-scientist-can-improve-his-productivity-730425ba4aa0 (dostęp: 9.08.2019).

Piatetsky Gregory (2017), *Python overtakes R, becomes the leader in Data Science, Machine Learning platforms*, https://www.kdnuggets.com/2017/08/python-overtakes-r-leader-analytics-data-science.html (dostęp: 3.08.2018).

Piatetsky Gregory (2018a), *Data Scientist – best job in America, 3 years in a row*, https://www.kdnuggets.com/2018/01/glassdoor-data-scientist-best-job-america-3years.html (dostęp: 3.02.2018).

Piatetsky Gregory (2018b), *Here are the most popular Python IDEs / Editors*, https://www.kdnuggets.com/2018/12/most-popular-python-ide-editor.html (dostęp: 17.07.2019).

Piatetsky Gregory (2018c), *Python eats away at R: Top Software for Analytics, Data Science, Machine Learning in 2018: Trends and Analysis*, https://www.kdnuggets.com/2018/05/poll-tools-analytics-data-science-machine-learning-results.html/2 (dostęp: 25.11.2018).

Piatetsky Gregory (2018d), *The 6 components of Open-Source Data Science / Machine Learning Ecosystem; Did Python declare victory over R?*, https://www.kdnuggets.com/2018/06/ecosystem-data-science-python-victory.html (dostęp: 25.03.2019).

Piotrowski Andrzej (1998), *Ład interakcji. Studia z socjologii interpretatywnej*, Wydawnictwo Uniwersytetu Łódzkiego, Łódź.

Polanyi Michael (2005), *Personal Knowledge. Towards a Post-Critical Philosophy*, Routledge, London.

Pontificia Accademia per la Vita (2020a), *renAIssance – Rome Call for AI Ethics*, http://www.academyforlife.va/content/pav/en/events/intelligenza-artificiale.html (dostęp: 10.03.2020).

Pontificia Accademia per la Vita (2020b), *Rome Call for AI Ethics*, Vatican, https://www.romecall.org/wp-content/uploads/2022/03/RomeCall_Paper_web.pdf (dostęp: 5.03.2020).

Portmess Lisa, Tower Sara (2015), *Data barns, ambient intelligence and cloud computing: the tacit epistemology and linguistic representation of Big Data*, „Ethics and Information Technology", vol. 17(1), s. 1–9, https://doi.org/10.1007/s10676-014-9357-2

Postman Neil (2004), *Technolpol. Triumf techniki nad kulturą*, Wydawnictwo Muza, Warszawa.

Powers David M.W. (2007), *Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation*, Flinders University, Adelaide, http://david.wardpowers.info/BM/Evaluation_SIETR.pdf (dostęp: 10.03.2020).

Prendki Jennifer (2018), *Before you launch your machine learning model, start with an MVP*, https://venturebeat.com/2018/11/24/before-you-launch-your-machine-learning-model-start-with-an-mvp/ (dostęp: 30.03.2019).

Press Gil (2018), *The Brute Force Of IBM Deep Blue And Google DeepMind*, „Forbes", 7 stycznia, https://www.forbes.com/sites/gilpress/2018/02/07/the-brute-force-of-deep-blue-and-deep-learning/#741dad3249e3 (dostęp: 10.07.2019).

Princeton Alumni Weekly (2012), *John D. Hunter '90*, https://paw.princeton.edu/memorial/john-d-hunter-'90 (dostęp: 4.10.2018).

Prokulski Łukasz (2017), *Ankieta – wyniki (i jak je podsumować w R)*, https://blog.prokulski.science/index.php/2017/10/17/ankieta-wyniki/ (dostęp: 14.11.2017).

Prokulski Łukasz (2019), *Analiza ofert pracy. Ile jest pracy dla ludzi zajmujących się analizą danych i machine learningiem?*, https://blog.prokulski.science/index.php/2019/02/18/analiza-ofert-pracy/#more-2093 (dostęp: 12.06.2019).

Provost Foster, Fawcett Tom (2013), *Data Science and its Relationship to Big Data and Data-Driven Decision Making*, „Big Data", vol. 1(1), s. 51–59, https://doi.org/10.1089/big.2013.1508

Przanowski Karol (2014), *Credit Scoring w erze Big Data*, Oficyna Wydawnicza Szkoła Główna Handlowa w Warszawie, Warszawa.

Przegalińska Aleksandra (2014), *Nawroty bylejakości*, „dwutygodnik.com", nr 10(144).

Przegalińska Aleksandra (2015), *Jeśli nie neoluddyzm, to co?*, „Res Publica Nova", nr 3(221), https://publica.pl/teksty/jesli-nie-neoluddyzm-to-co-52814.html (dostęp: 14.11.2017).

Przegalińska Aleksandra (2016a), *Ciało i umysł – wstęp kuratorski*, [w:] E. Drygalska (red.), *Interfejsy, kody, symbole. Przyszłość komunikowania*, Miasto Przyszłości/Laboratorium Wrocław, Wrocław, s. 11–19.

Przegalińska Aleksandra (2016b), *Istoty wirtualne. Jak fenomenologia zmieniała sztuczną inteligencję*, Towarzystwo Autorów i Wydawców Prac Naukowych Universitas, Kraków.

Przegalińska Aleksandra (2016c), *Konflikt generacyjny wśród botów: między ELIZĄ, Cleverbotem a Tay*, [w:] V. Kuś (red.), *Live'Bot*, E-naukowiec, Bydgoszcz, s. 10–14.

Przegalińska Aleksandra, Ciechanowski Leon, Magnuski Mikołaj, Gloor Peter (2018), *Muse Headband: Measuring Tool or a Collaborative Gadget?*, [w:]

F. Grippa, J. Leitão, J. Gluesing, K. Riopelle, P. Gloor (red.), *Collaborative Innovation Networks: Building Adaptive and Resilient Organizations*, Springer International Publishing, Cham, s. 93–101.

Przybyłowska Ilona (1978), *Wywiad swobodny ze standaryzowaną listą poszukiwanych informacji i możliwości jego zastosowania w badaniach socjologicznych*, „Przegląd Socjologiczny", nr 30, s. 53–63.

Puschmann Cornelius, Burgess Jean (2014), *Metaphors of big data*, „International Journal of Comunication", no. 8, s. 1690–1709.

PwC (2015), *What's next for the 2017 data science and analytics job market?*, https://www.pwc.com/us/en/publications/data-science-and-analytics.html (dostęp: 10.02.2018).

PyData (2018), *PyData Warsaw 2018*, https://pydata.org/warsaw2018/ (dostęp: 14.08.2018).

PyLadies (2019), *About: PyLadies*, https://www.pyladies.com/about/ (dostęp: 12.02.2019).

Pyszczuk Artur (2011), *Programowanie w Bashu, czyli jak pisać skrypty w Linuksie*, https://www.arturpyszczuk.pl/files/bash/bash.pdf (dostęp: 5.06.2019).

Python Software Foundation (2019), *Search results PyPI*, https://pypi.org/search/?c=Intended+Audience+%3A%3A+Science%2FResearch&o=-created&q=&page=1 (dostęp: 16.07.2019).

R Foundation (2019), *CRAN – Contributed Packages*, https://cran.r-project.org/web/packages/ (dostęp: 16.07.2019).

R-Ladies (2019), *About us – R-Ladies Global*, https://rladies.org/about-us/ (dostęp: 12.02.2019).

Radomski Andrzej, Bomba Radosław (2013), *Zwrot cyfrowy w humanistyce. Internet, nowe media, kultura 2.0*, http://www.sbc.org.pl/dlibra/docconten-t?id=77488 (dostęp: 4.02.2018).

Raji Inioluwa Deborah, Smart Andrew, White Rebecca N., Mitchell Margaret, Gebru Timnit, Hutchinson Ben, Smith-Loud Jamila, Theron Daniel, Barnes Parker (2020), *Closing the AI accountability gap*, [w:] *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, ACM, New York, s. 33–44.

Rajpurkar Pranav, Irvin Jeremy, Ball Robyn L., Zhu Kaylie, Yang Brandon, Mehta Hershel, Duan Tony, Ding Daisy, Bagul Aarti, Langlotz Curtis P., Patel Bhavik N., Yeom Kristen W., Shpanskaya Katie, Blankenberg Francis G., Seekins Jayne, Amrhein Timothy J., Mong David A., Halabi Safwan S., Zucker Evan J., Ng Andrew, Lungren Matthew P. (2018), *Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists*, „PLOS Medicine", vol. 15(11), e1002686, https://doi.org/10.1371/journal.pmed.1002686

Raschka Sebastian (2018), *Python. Uczenie maszynowe*, Wydawnictwo Helion, Gliwice.

Rath Matthias (2018), *Data Science – die neue Leitwissenschaft?*, [w:] T. Knubben, E. Schols, U. Braun (red.), *Weltkulturatlas. Kultur in Zeit der Globalisierung. Daten, Geschichten, Grafiken*, av Edition, Stuttgart, s. 21–37.

Ratner Alexander, Bach Stephen H., Ehrenberg Henry, Fries Jason, Wu Sen, Ré Christopher (2017), *Snorkel: rapid training data creation with weak supervision*, „Proceedings of the VLDB Endowment", vol. 11(3), s. 269–282, https://doi.org/10.14778/3157794.3157797

Raymond Eric S. (2001), *The cathedral and the bazaar*, [w:] E.S. Raymond (red.), *In the Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, O'Reilly, Sebastopol, s. 16–64.

Reitz Kenneth (2018), *Python Guide Documentation Release 0.0.1*, https://buildmedia.readthedocs.org/media/pdf/python-guide/latest/python-guide.pdf (dostęp: 12.02.2019).

RENOIR (2018), *RENOIR Project*, http://renoirproject.eu/ (dostęp: 21.02.2018).

Reshama Shaikh (2018), *Why Women Are Flourishing In R Community But Lagging In Python*, https://reshamas.github.io/why-women-are-flourishing-in-r-community-but-lagging-in-python/ (dostęp: 12.12.2018).

Rexer Analytics (2018), *Data Science Survey*, http://www.rexeranalytics.com/data-science-survey.html (dostęp: 1.02.2018).

Ribeiro Marco Tulio, Singh Sameer, Guestrin Carlos (2016), *'Why Should I Trust You?' Explaining the Predictions of Any Classifier*, [w:] *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, s. 1135–1144.

Richardson Rashida, Schultz Jason, Crawford Kate (2019), *Dirty Data, Bad Predictions: How Civil Rights Violations Impact Police Data, Predictive Policing Systems, and Justice*, „New York University Law Review Online", styczeń, s. 192–233.

Robertson Adi (2019), *The 5 biggest announcements from Facebook's F8 developer conference keynote*, https://www.theverge.com/2019/4/30/18524068/facebook-f8-2019-keynote-highlights-summary-news-feed-messenger-instagram-oculus (dostęp: 20.05.2019).

Rodak Olga (2017), *Twitter jako przedmiot badań socjologicznych i źródło danych społecznych: perspektywa konstruktywistyczna*, „Studia Socjologiczne", nr 3(226), s. 209–236.

Rogozhnikov Alex (2016), *Jupyter (IPython) notebooks features*, https://arogozhnikov.github.io/2016/09/10/jupyter-features.html (dostęp: 22.07.2019).

Rossum Guido van (1997), *Comparing Python to Other Languages*, https://www.python.org/doc/essays/comparisons/ (dostęp: 16.07.2019).

Rothenberger Lea, Fabian Benjamin, Arunov Elmar (2019), *Relevance of Ethical Guidelines for Artificial Intelligence – a Survey and Evaluation*, „European Conference on Information Systems", maj, s. 1–11.

RStudio (2018a), *About us*, https://www.rstudio.com/about/ (dostęp: 26.07.2018).

RStudio (2018b), *RStudio – RStudio Products*, https://www.rstudio.com/products/rstudio/ (dostęp: 27.11.2018).

RStudio Team (2016), *RStudio: Integrated Development Environment for R*, http://www.rstudio.com/ (dostęp: 27.11.2018).

RStudio Team (2017), *RStudio IDE: Cheat Sheet*, https://www.rstudio.org/links/ide_cheat_sheet (dostęp: 27.11.2018).

Rumelhart David E., Hinton Geoffrey E., Williams Ronald J. (1986), *Learning representations by back-propagating errors*, „Nature", vol. 323(6088), s. 533–536, https://doi.org/10.1038/323533a0

Russel Stuart J., Norvig Peter (2009), *Artificial Intelligence: A Modern Approach*, Prentice Hall, New Jersey.

Rybicki Jan, Eder Maciej (2011), *Deeper Delta across genres and languages: do we really need the most frequent words?*, „Literary and Linguistic Computing", vol. 26(3), s. 315–321, https://doi.org/10.1093/llc/fqr031

Ryciak Norbert (2019), *Danetyka, czyli o polskim tłumaczeniu Data Science*, https://www.kodolamacz.pl/blog/data-science-po-polsku/ (dostęp: 22.01.2020).

Sadowski Jathan (2019), *When data is capital: Datafication, accumulation, and extraction*, „Big Data & Society", vol. 6(1), 205395171882054, https://doi.org/10.1177/2053951718820549

Saltz Jeff (2019), *Ethics in Data Science Projects: Current Practices and Perceptions*, [w:] *Proceedings of the 27th European Conference on Information Systems (ECIS), Stockholm & Uppsala, Sweden, June 8–14, 2019*, Research-in-Progress Papers, no. 68, https://aisel.aisnet.org/ecis2019_rip/68 (dostęp: 27.11.2019).

Sato Kaz (2016), *How a Japanese cucumber farmer is using deep learning and TensorFlow*, https://cloud.google.com/blog/products/gcp/how-a-japanese-cucumber-farmer-is-using-deep-learning-and-tensorflow?authuser=0 (dostęp: 14.06.2019).

Satyaseel Harshit (2018), *Most Popular Python Libraries for Data Science in 2018*, https://www.technotification.com/2018/09/popular-python-libraries-data-science.html (dostęp: 1.10.2018).

Savage Mike, Halford Susan (2017), *Speaking Sociologically with Big Data: Symphonic Social Science and the Future for Big Data Research*, „Sociology", vol. 51(6), s. 1132–1148, https://doi.org/10.1177/0038038517698639

Saxena Ashutosh, Sun Min, Ng Andrew (2009), *Make3D: Learning 3D Scene Structure from a Single Still Image*, „IEEE Transactions on Pattern Analysis and Machine Intelligence", vol. 31(5), s. 824–840, https://doi.org/10.1109/TPAMI.2008.132

Schmidhuber Jürgen (2015), *Deep learning in neural networks: An overview*, „Neural Networks", no. 61, s. 85–117, https://doi.org/10.1016/j.neunet.2014.09.003

Schmidt Eric, Rosenberg Jonathan, Eagle Alan (2014), *Jak działa Google*, Wydawnictwo Insignis Media, Kraków.

Schulze Elizabeth (2019), *40% of A.I. start-ups in Europe have almost nothing to do with A.I., research finds*, https://www.cnbc.com/2019/03/06/40-percent-of-ai-start-ups-in-europe-not-related-to-ai-mmc-report.html (dostęp: 2.04.2019).

Schutten Gerrit-Jan, Chan Chung-hong, Leeper Thomas J., Foster John, Oller Sergio, Hester Jim, Watts Stephen, Katossky Arthur, Malavin Stas, Garmonsway Duncan, Mahmoudian Mehrad, Kerlogue Matt, Steuer Detlef, Lauer Michal (2020), *readODS: Read and Write ODS Files*, https://cran.r-project.org/package=readODS (dostęp: 1.03.2021).

Schwartz Mattathias (2017), *Facebook User Data Harvested by Cambridge Analytica*, https://theintercept.com/2017/03/30/facebook-failed-to-protect-30-million-users-from-having-their-data-harvested-by-trump-campaign-affiliate/ (dostęp: 1.02.2019).

scikit-learn (2018), *About us – scikit-learn 0.20.1 documentation*, https://scikit-learn.org/stable/about.html#citing-scikit-learn (dostęp: 4.07.2018).

SciPy.org (b.d.), https://scipy.org/index.html (dostęp: 13.08.2018).

Scriptol.com (2006), *List of Hello World Programs in 200 Programming Languages*, https://www.scriptol.com/programming/hello-world.php (dostęp: 15.07.2019).

Sefala Raesetje, Gebru Timnit, Moorosi Nyalleng, Klein Richard (2021), *Constructing a Visual Dataset to Study the Effects of Spatial Apartheid in South Africa*, [w:] *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, https://openreview.net/forum?id=WV0waZ-z9dTF (dostęp: 13.08.2018).

Sekercioglu Eser (2018), *For data science, which will die first: R or Python?*, https://www.quora.com/For-data-science-which-will-die-first-R-or-Python/answer/Eser-Sekercioglu (dostęp: 8.09.2018).

Sharif Bonita, Maletic Jonathan I. (2010), *An Eye Tracking Study on camelCase and under_score Identifier Styles*, [w:] *2010 IEEE 18th International Conference on Program Comprehension*, IEEE, Braga, s. 196–205, http://ieeexplore.ieee.org/document/5521745/ (dostęp: 13.08.2018).

Sharp Sight (2016), *Data Science Crash Course*, http://www.sharpsightlabs.com/ (dostęp: 5.11.2017).

Sharp Sight (2018), *R vs Python... which to learn for data science*, https://www.sharpsightlabs.com/blog/r-vs-python/ (dostęp: 14.08.2018).

Shaw Zed A. (2014), *Learn Python the Hard Way: a very simple introduction to the terrifyingly beautiful world of computers and code*, Addison-Wesley, Donnelley.

Shead Sam (2018), *'NIPS' AI Conference Changes Name, Kind Of*, „Forbes", 21 listopada, https://www.forbes.com/sites/samshead/2018/11/21/nips-ai-conference-changes-name-kind-of/#8020bc73bc15 (dostęp: 10.02.2019).

Shiab Nael (2015), *On the Ethics of Web Scraping and Data Journalism*, https://gijn.org/2015/08/12/on-the-ethics-of-web-scraping-and-data-journalism/ (dostęp: 5.02.2018).

Shibutani Tamotsu (1955), *Reference Groups as Perspectives*, „American Journal of Sociology", vol. 60(6), s. 562–569, https://doi.org/10.1086/221630

Silaparasetty Vinita (2018), *Python vs R for Machine Learning*, [w:] *International Conference on Security*, Garden City University, Bangalore, s. 1–19.

Silge Julia, Robinson David (2018), *Text Mining with R: A Tidy Approach*, O'Reilly, Boston, https://www.tidytextmining.com/ (dostęp: 10.02.2019).

Silver David, Hubert Thomas, Schrittwieser Julian, Antonoglou Ioannis, Lai Matthew, Guez Arthur, Lanctot Marc, Sifre Laurent, Kumaran Dharshan, Graepel Thore, Lillicrap Timothy, Simonyan Karen, Hassabis Demis (2018), *A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play*, „Science", vol. 362(6419), s. 1140–1144, https://doi.org/10.1126/science.aar6404

Silver David, Schrittwieser Julian, Simonyan Karen, Antonoglou Ioannis, Huang Aja, Guez Arthur, Hubert Thomas, Baker Lucas, Lai Matthew, Bolton Adrian, Chen Yutian, Lillicrap Timothy, Hui Fan, Sifre Laurent, Driessche George van den, Graepel Thore, Hassabis Demis (2017), *Mastering the game of Go without human knowledge*, „Nature", vol. 550(7676), s. 354–539, https://doi.org/10.1038/nature24270

Silver David, Huang Aja, Maddison Chris J., Guez Arthur, Sifre Laurent, Van Den Driessche George, Schrittwieser Julian, Antonoglou Ioannis, Panneershelvam Veda, Lanctot Marc, Dieleman Sander, Grewe Dominik, Nham John, Kalchbrenner Nal, Sutskever Ilya, Lillicrap Timothy, Leach Madeleine, Kavukcuoglu Koray, Graepel Thore, Hassabis Demis (2016), *Mastering the game of Go with deep neural networks and tree search*, „Nature", vol. 529(7587), https://doi.org/10.1038/nature16961

Silverman David (2010), *Prowadzenie badań jakościowych*, Wydawnictwo Naukowe PWN, Warszawa.

Simonite Tom (2018), *AI Researchers Fight Over Four Letters: NIPS*, https://www.wired.com/story/ai-researchers-fight-over-four-letters-nips/ (dostęp: 12.02.2019).

Singularity University (2018) *Peter Norvig – Mentor – Singularity University*, https://su.org/mentors/peter-norvig/ (dostęp: 1.08.2018).

Skeet Jon (2010), *Writing the perfect question*, https://codeblog.jonskeet.uk/2010/08/29/writing-the-perfect-question/ (dostęp: 9.09.2019).

Skura Małgorzata, Lisicki Michał (2018), *Gen liczby: jak dzieci uczą się matematyki?*, Wydawnictwo Mamania, Warszawa.

Slowikowski Kamil (2019), *ggrepel: Automatically Position Non-Overlapping Text Labels with 'ggplot2'*, https://cran.r-project.org/package=ggrepel (dostęp: 9.09.2019).

Smart Francis (2013), *Export R Results Tables to Excel – Please don't kick me out of your club*, https://www.r-bloggers.com/2013/08/export-r-results-tables-to-excel-please-dont-kick-me-out-of-your-club/ (dostęp: 28.10.2017).

Somers James (2017), *Is AI Riding a One-Trick Pony?*, „MIT Technology Review", 29 września, https://www.technologyreview.com/s/608911/is-ai-riding-a-one-trick-pony/ (dostęp: 23.11.2018).

Somers James (2018), *The Scientific Paper Is Obsolete. Here's What's Next*, https://www.theatlantic.com/science/archive/2018/04/the-scientific-paper-is-obsolete/556676/ (dostęp: 12.08.2018).

Sopyła Krzysztof (2017), *Podsumowanie ankiety Data Science Polska 2017*, https://ksopyla.com/data-science/podsumowanie-ankiety-data-science-polska-2017/ (dostęp: 10.10.2017).

Sopyła Krzysztof (2019), *Dlaczego porzuciłem Tensorflow na rzecz Pytorch*, https://ksopyla.com/machine-learning/pytorch-vs-tensorflow/ (dostęp: 19.03.2019).

Sorensen Chris (2017), *How U of T's 'godfather' of deep learning is reimagining AI*, https://www.utoronto.ca/news/how-u-t-s-godfather-deep-learning-reimagining-ai (dostęp: 23.11.2018).

Sotrender (2018), *Sotrender. No-bullshit analytics. Analyze and optimize your marketing over social media*, https://www.sotrender.com/pl/team/ (dostęp: 1.12.2018).

Soubra Diya (2012), *The three Vs that define big data*, http://www.datasciencecentral.com/forum/topics/the-3vs-that-define-big-data (dostęp: 5.04.2017).

Spector Alfred, Norvig Peter, Petrov Slav (2012), *Google's hybrid approach to research*, „Communications of the ACM", vol. 55(7), s. 34–37, https://doi.org/10.1145/2209249.2209262

Spinellis Diomidis (2005), *Version control systems*, „IEEE Software", vol. 22(5), s. 108–109, https://doi.org/10.1109/MS.2005.140

Staniak Mateusz, Biecek Przemysław (2018), *Explanations of model predictions with live and breakDown packages*, http://arxiv.org/abs/1804.01955 (dostęp: 1.12.2019).

Staniak Mateusz, Biecek Przemyslaw (2019), *The Landscape of R Packages for Automated Exploratory Data Analysis*, http://arxiv.org/abs/1904.02101 (dostęp: 1.12.2019).

Star Susan Leigh, Griesemer James R. (1989), *Institutional Ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907–39*, „Social Studies of Science", vol. 19(3), s. 387–420, https://doi.org/10.1177/030631289019003001

StatCounter (2019), *Desktop Operating System Market Share Worldwide*, https://gs.statcounter.com/os-market-share/desktop/ (dostęp: 18.09.2019).

Strauss Anselm L. (1978), *A Social World Perspective*, „Studies in Symbolic Interaction", no. 1, s. 119–28.

Strauss Anselm L. (1984), *Social Worlds and Their Segmentation Processes*, [w:] N. Denzin (red.), *Studies in Symbolic Interaction*, JAI Press, Greenwitch, s. 123–139.

Strauss Anselm L. (1993), *Continual Permutations of Action*, Aldine de Gruyter, New York.

Strauss Anselm L., Corbin Juliet (1990), *Basics of Qualitative Research. Grounded Theory Procedures and Techniques*, Sage, Newbury Park–London–New Delhi.

Striphas Ted (2015), *Algorithmic culture*, „European Journal of Cultural Studies", vol. 18(4–5), s. 395–412, https://doi.org/10.1177/1367549415577392

Strong Alexus (2018), *Why I'm Learning Python in 2018*, https://dimitris-livas.squarespace.com/blog/2018/1/22/why-im-learning-python-in-2018 (dostęp: 22.01.2019).

Sumbul Michel (2014), *Big Data problematic*, https://whatsbigdata.be/category/big-data-overview/ (dostęp: 30.01.2017).

Sumpter David J. (2019), *Osaczeni przez liczby: o algorytmach, które kontrolują nasze życie: od Facebooka i Google'a po fake newsy i bańki filtrujące*, Copernicus Center Press, Kraków.

Surma Jerzy (2017), *Cyfryzacja życia w erze big data. Człowiek, biznes, państwo*, Wydawnictwo Naukowe PWN, Warszawa.

Surowiecki James (2017), *Chill: Robots Won't Take All Our Jobs*, https://www.wired.com/2017/08/robots-will-not-take-your-job/ (dostęp: 14.02.2018).

switchup (2018), *2018 Best Data Science Bootcamps*, https://www.switchup.org/rankings/best-data-science-bootcamps (dostęp: 21.02.2018).

System Wspomagania Analiz i Decyzji GUS (2019), *Miasta największe pod względem liczby ludności*, http://swaid.stat.gov.pl/Dashboards/Miasta największe pod względem liczby ludności.aspx (dostęp: 11.06.2019).

Szahaj Andrzej (2004), *Zniewalająca moc kultury. Artykuł i szkice z filozofii kultury, poznania i polityki*, Wydawnictwo Uniwersytetu Mikołaja Kopernika, Toruń.

Szeliga Marcin (2017), *Data science i uczenie maszynowe*, Wydawnictwo Naukowe PWN, Warszawa.

Szpunar Magdalena (2012), *Nowe-Stare Medium*, Wydawnictwo Instytutu Filozofii i Socjologii Polskiej Akademii Nauk, Warszawa.

Szpunar Magdalena (2013), *Wokół koncepcji gatekeepingu. Od gatekeepingu tradycyjnego do technologicznego*, [w:] I.S. Fiut (red.), *Idee i Myśliciele: medialne i społeczne aspekty filozofii*, Wydawnictwa Akademii Górniczo-Hutniczej, Kraków, s. 56–61.

Szpunar Magdalena (2018), *Kultura algorytmów*, „Zarządzanie w Kulturze", nr 19(1), s. 1–10, https://doi.org/10.4467/20843976ZK.18.001.8493

Szpunar Magdalena (2019), *Kultura algorytmów*, Wydawnictwo ToC, Kraków.

Szreder Mirosław (2015a), *Big data wyzwaniem dla człowieka i statystyki*, „Wiadomości Statystyczne", nr 8(651), s. 1–11.

Szreder Mirosław (2015b), *Czy statystyka pozwala lepiej zrozumieć świat?*, „Polityka", 12 maja.

Szreder Mirosław (2016) *Złudzenie Big Data*, „Tygodnik Powszechny", nr 10(3478), s. 50–51.

Szreder Mirosław (2018a), *O algorytmach Big Data (na marginesie książki Cathy O'Neil pt. „Broń matematycznej zagłady. Jak algorytmy zwiększają nierówności i zagrażają demokracji", Wydawnictwo Naukowe PWN, 2017)*, „Wiadomości Statystyczne", nr 2(681), s. 78–81.

Szreder Mirosław (2018b), *Populacja małych frakcji*, „Polityka", 16 stycznia.

Sztandar-Sztanderska Karolina, Kotnarowski Michał, Zieleńska Marianna (2021), *Czy algorytmy wprowadzają w błąd? Metaanaliza algorytmu profilowania bezrobotnych stosowanego w Polsce*, „Studia Socjologiczne", nr 1(240), s. 89–115, https://doi.org/10.24425/sts.2021.136280

Sztompka Piotr (2007), *Socjologia. Analiza społeczeństwa*, Wydawnictwo Znak, Kraków.

Szumakowicz Eugeniusz (2000), *Sztuczna inteligencja – problem czy pseudoproblem?*, [w:] E. Szumakowicz (red.), *Granice sztucznej inteligencji. Eseje i studia*, Wydawnictwo Politechniki Krakowskiej, Kraków, s. 11–42.

Szupiluk Ryszard (2013), *Dekompozycje wielowymiarowe w agregacji predykcyjnych modeli data mining*, Oficyna Wydawnicza Szkoła Główna Handlowa w Warszawie, Warszawa.

Szymielewicz Katarzyna (2017), *Algorytm zwycięstwa*, https://panoptykon.org/wiadomosc/algorytm-zwyciestwa (dostęp: 9.04.2018).

Szymielewicz Katarzyna, Obem Anna (2020), *Sztuczna inteligencja non-fiction*, https://panoptykon.org/sztuczna-inteligencja-non-fiction (dostęp: 25.07.2020).

Śledziewska Katarzyna, Włoch Renata (2020), *Gospodarka cyfrowa. Jak nowe technologie zmieniają świat*, Wydawnictwa Uniwersytetu Warszawskiego, Warszawa.

Taddeo Mariarosaria, Floridi Luciano (2018), *How AI can be a force for good*, „Science", vol. 361(6404), s. 751–752, https://doi.org/10.1126/science.aat5991

Tadeusiewicz Ryszard (2018), *Prof. dr hab. inż. Ryszard Tadeusiewicz – Dossier w pigułce*, https://www.uci.agh.edu.pl/uczelnia/tad/dossier.php (dostęp: 5.12.2018).

Tadeusiewicz Ryszard, Mikrut Zbigniew (1978), *Identyfikacja i modelowanie neuronu na maszynie cyfrowej*, „Archiwum Automatyki i Telemechaniki", nr 3(23), s. 345–356.

Tarkowski Alek, Paszcza Bartosz, Mileszyk Natalia (2020), *„AlgoPolska": 11 postulatów, których realizacja pozwoli uniknąć mrocznych wizji rodem z „Black Mirror*, https://klubjagiellonski.pl/2019/12/09/algopolska-11-tez-ktorych-realizacja-pozwoli-uniknac-nam-mrocznych-wizji-rodem-z-black-mirror/ (dostęp: 5.01.2020).

Tatman Rachael (2019), *Six steps to more professional data science code*, https://www.kaggle.com/rtatman/six-steps-to-more-professional-data-science-code (dostęp: 29.09.2019).

Taylor David (2016), *Battle of the Data Science Venn Diagrams*, https://www.kdnuggets.com/2016/10/battle-data-science-venn-diagrams.html4 (dostęp: 14.12.2017).

Taylor Linnet, Dencik Lina (2020), *Constructing Commercial Data Ethics*, „Technology and Regulation", vol. 2, s. 1–10, https://doi.org/10.26116/techreg.2020.001

TechAmerica (2012), *Demystifying big data: A practical guide to transforming the business of Government*, http://www.techamerica.org/Docs/fileManager.cfm?f=techamerica-bigdatareport-final.pdf (dostęp: 5.03.2017).

The Neural Information Processing Systems Foundation Board of Trustees (2018), *From the Board: Changing our Acronym*, https://nips.cc/Conferences/2018/News (dostęp: 12.02.2019).

The pandas project (2018), *pandas: Python Data Analysis Library*, http://pandas.pydata.org/about.html (dostęp: 4.12.2018).

Thieme Nick (2018), *R Generation*, „Significance", vol. 15(4), s. 14–19, https://doi.org/10.1111/j.1740-9713.2018.01169.x

Thomas Suzanne L., Nafus Dawn, Sherman Jamie (2018), *Algorithms as fetish: Faith and possibility in algorithmic work*, „Big Data & Society", vol. 5(1), s. 1–11, https://doi.org/10.1177/2053951717751552

Tocci Jason (2009), *Geek cultures: Media and identity in the digital age*, praca doktorska, University of Pennsylvania, http://repository.upenn.edu/dissertations/AAI3395723/ (dostęp: 12.02.2019).

Tomanek Krzysztof, Bryda Grzegorz (2015), *Odkrywanie postaw dydaktyków zawartych w komentarzach studenckich. Analiza treści z zastosowaniem słownika klasyfikacyjnego*, „Przegląd Socjologiczny", nr 64(4), s. 51–81.

Tomanek Krzysztof, Bryda Grzegorz (2017), *Metodyka dla analizy treści w projektach stosujących techniki text mining i rozwiązania CAQDAS piątej generacji*, „Przegląd Socjologii Jakościowej", t. XIII, nr 2, s. 128–143.

Torres J.C. (2018), *The future of smartphone cameras is AI*, https://www.slashgear.com/the-future-of-smartphone-cameras-is-ai-15530789/ (dostęp: 10.07.2019).

Törnberg Petter, Törnberg Anton (2018), *The limits of computation: A philosophical critique of contemporary Big Data research*, „Big Data & Society", vol. 5(2), s. 1–12, https://doi.org/10.1177/2053951718811843

Troszczyński Marek, Wawer Aleksander (2017), *Czy komputer rozpozna hejtera? Wykorzystanie uczenia maszynowego (ML) w jakościowej analizie danych*, „Przegląd Socjologii Jakościowej", t. XIII, nr 2, s. 62–80.

Trzpiot Grażyna (2017), *Rozumienie Data Science*, [w:] G. Trzpiot (red.), *Statystyka a Data Science*, Wydawnictwo Uniwersytetu Ekonomicznego w Katowicach, Katowice, s. 6–30.

Tufte Edward R. (1983), *The Visual Display of Quantitative Information*, Graphics Press, Cheshire.

Turner Anna, Zieliński Marcin W., Słomczyński Kazimierz M. (2018), *Google big data: charakterystyka i zastosowanie w naukach społecznych*, „Studia Socjologiczne", nr 4(231), s. 49–71, https://doi.org/10.24425/122482

UNECE (2014), *How big is Big Data? Exploring the role of Big Data in Official Statistics*, http://www1.unece.org/stat/platform/display/bigdata/How+big+is+-Big+Data (dostęp: 5.06.2017).

Uri Therese (2015), *The Strengths and Limitations fo Using Situational Analysis Grounded Theory as Research Methodology*, „Journal of Ethnographic & Qualitative Research", vol. 10(1), s. 135–151, https://www.depts.ttu.edu/education/our-people/Faculty/additional_pages/duemer/epsy_5382_class_materials/2019/The_Strengths_and_Limitations_of_Using_SituationalAnalysis_Grounded_Theory_as_Research_Methodology_Uri_2015.pdf (dostęp: 10.07.2019).

Vaidhyanathan Siva (2018), *Antisocial media. Jak Facebook oddala nas od siebie i zagraża demokracji*, Grupa Wydawnicza Foksal, Warszawa.

Van Camp Jeffrey (2019), *8 Best Smart Speakers in 2019: Alexa, Google Assistant, Siri, Cortana | WIRED*, https://www.wired.com/story/best-smart-speakers/ (dostęp: 10.07.2019).

Vanderplas Jake (2016), *Conda: Myths and Misconceptions*, https://jakevdp.github.io/blog/2016/08/25/conda-myths-and-misconceptions/ (dostęp: 22.07.2019).

Vazquez Favio (2017), *Data version control with DVC. What do the authors have to say?*, https://towardsdatascience.com/data-version-control-with-dvc-what-do-the-authors-have-to-say-3c3b10f27ee (dostęp: 3.08.2018).

Vicknair Chad, Macias Michael, Zhao Zhendong, Nan Xiaofei, Chen Yixin, Wilkins Dawn (2010), *A comparison of a graph database and a relational database: a data provenance perspective*, „Proceedings of the 48th Annual Southeast Regional Conference on ACM SE", https://doi.org/10.1145/1900008.1900067

Vopson Melvin M. (2020), *The information catastrophe*, „AIP Advances", vol. 10(8), 085014, https://doi.org/10.1063/5.0019941

Votta Fabio (2018), *@favstats: Our @hadleywickham, who art at RStudio, Hallowed be thy name. Thy functions run, thy unit tests...*, https://twitter.com/favstats/status/1043836510880116738 (dostęp: 23.01.2019).

Wachter Sandra, Mittelstadt Brent, Floridi Luciano (2017), *Why a Right to Explanation of Automated Decision-Making Does Not Exist in the General Data Protection Regulation*, „International Data Privacy Law", vol. 7(2), s. 76–99, https://doi.org/10.1093/idpl/ipx005

Wagner Ben (2018), *Ethics as an Escape from Regulation: From ethics-washing to ethics-shopping?*, [w:] E. Bayamlioglu, I. Baraliuc, L.A.W. Janssens, M. Hildebrandt (red.), *Being Profiling: Cogitas Ergo Sum. 10 Years of Profiling the European Citizen*, Amsterdam University Press, Amsterdam, s. 84–90.

Walne Zgromadzenie Delegatów Polskiego Towarzystwa Socjologicznego (2012), *Kodeks etyki socjologa*, http://pts.org.pl/wp-content/uploads/2016/04/kodeks.pdf (dostęp: 4.04.2017).

Wang Tricia (2013), *Why Big Data Needs Thick Data*, https://medium.com/ethnography-matters/why-big-data-needs-thick-data-b4b3e75e3d7 (dostęp: 7.05.2018).

Wang Tricia (2016), *The human insights missing from big data*, https://www.ted.com/talks/tricia_wang_the_human_insights_missing_from_big_data#t-9755 (dostęp: 7.05.2018).

Warsaw.AI (2020), *Przemysław Biecek*, https://warsaw.ai/speaker/przemyslaw-biecek/ (dostęp: 5.08.2020).

Watson Samuel (2018), *Samuel S. Watson's answer to Who will eventually win the Python versus R debate in data science?*, https://www.quora.com/Who-will-eventually-win-the-Python-versus-R-debate-in-data-science/answer/Samuel-S-Watson-1 (dostęp: 8.09.2018).

Watson Sara M. (2015) *Data is the New "__"*, http://dismagazine.com/discussion/73298/sara-m-watson-metaphors-of-big-data/ (dostęp: 23.04.2018).

Wąsowski Michał (2018), *Mark Zuckerberg pokazał „prawdziwą twarz" na przesłuchaniach w Kongresie USA*, https://businessinsider.com.pl/firmy/strategie/jak-poradzil-sobie-mark-zuckerberg-na-przesluchaniu-senatu-usa/b1yjyfg (dostęp: 9.05.2018).
WDI18 (2018), *Warszawskie Dni Informatyki 2018: Data Science by Data Science Warsaw*, https://warszawskiedniinformatyki.pl/ (dostęp: 11.05.2018).
Weber Max (1989), *Polityka jako zawód i powołanie*, Wydawnictwo Znak, Kraków.
What is the difference between Terminal, Console, Shell, and Command Line? (2014), https://askubuntu.com/questions/506510/what-is-the-difference-between-terminal-console-shell-and-command-line (dostęp: 13.10.2019).
White John (2012), *Julia, I Love You*, http://www.johnmyleswhite.com/notebook/2012/03/31/julia-i-love-you/ (dostęp: 15.07.2019).
White Tom (2015), *Hadoop: The Definitive Guide*, O'Reilly, Sebastopol.
Why R? Foundation (2018), http://whyr.pl/ (dostęp: 11.05.2018).
Wickham Hadley (2010), *A Layered Grammar of Graphics*, „Journal of Computational and Graphical Statistics", vol. 19(1), s. 3–28, https://doi.org/10.1198/jcgs.2009.07098
Wickham Hadley (2014a), *Advanced R*, Chapman and Hall/CRC Press, New York.
Wickham Hadley (2014b), *Data science: how is it different to statistics?*, http://bulletin.imstat.org/2014/09/data-science-how-is-it-different-to-statistics/ (dostęp: 3.09.2018).
Wickham Hadley (2014c), *Tidy Data*, „Journal of Statistical Software", vol. 59(10), s. 1–23, https://doi.org/10.18637/jss.v059.i10
Wickham Hadley (2017), *tidyverse: Easily Install and Load the "Tidyverse"*, https://cran.r-project.org/package=tidyverse (dostęp: 11.05.2018).
Wickham Hadley (2018a), *Hadley Wickham*, http://hadley.nz/ (dostęp: 16.01.2018).
Wickham Hadley (2018b), *profr: An Alternative Display for Profiling Information*, https://cran.r-project.org/package=profr (dostęp: 3.09.2019).
Wickham Hadley (2018c), *scales: Scale Functions for Visualization*, https://cran.r-project.org/package=scales (dostęp: 3.09.2019).
Wickham Hadley (2018d), *You can't do data science in a GUI*, https://www.youtube.com/watch?v=cpbtcsGE0OA (dostęp: 27.11.2018).
Wickham Hadley (2019), *Letter To A Young Maintainer – talk at Monktoberfest 2019*, https://www.youtube.com/watch?v=1K7u5hkciLI (dostęp: 30.04.2020).
Wickham Hadley, Grolemund Garrett (2017), *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*, O'Reilly, Beijing–Boston–Farnham–Sebastopol–Tokyo, https://r4ds.had.co.nz (dostęp: 27.11.2018).
Wickham Hadley, Grolemund Garrett (2018), *Język R. Kompletny zestaw narzędzi dla analityków danych*, Wydawnictwo Helion, Gliwice.
Wickham Hadley, Averick Mara, Bryan Jennifer, Chang Winston, McGowan Lucy, François Romain, Grolemund Garrett, Hayes Alex, Henry Lionel, Hester Jim, Kuhn Max, Pedersen Thomas, Miller Evan, Bache Stephan, Müller Kirill,

Ooms Jeroen, Robinson David, Seidel Dana, Spinu Vitalie, Takahashi Kohske, Vaughan Davis, Wilke Claus, Woo Kara, Yutani Hiroaki (2019), *Welcome to the Tidyverse*, „Journal of Open Source Software", vol. 4(43), 1686, https://doi.org/10.21105/joss.01686

Wikipedia (b.d.), *Data scientist*, https://pl.wikipedia.org/wiki/Data_scientist (dostęp: 13.02.2018).

Wilke Claus O. (2019), *cowplot: Streamlined Plot Theme and Plot Annotations for 'ggplot2'*, https://cran.r-project.org/package=cowplot (dostęp: 30.04.2020).

Will Josh (2012), *@josh_wills: Data Scientist (n.): Person who is better at statistics than any software engineer and better at software engineering than any statistician*, https://twitter.com/josh_wills/status/198093512149958656 (dostęp: 7.12.2017).

Williams Alex (2017), *Will Robots Take Our Children's Jobs?*, „The New York Times", 11 grudnia, https://www.nytimes.com/2017/12/11/style/robots-jobs-children.html (dostęp: 14.02.2018).

Williams Laurie, Kessler Robert, Cunningham Ward, Jeffries Ron (2000), *Strengthening the case for pair programming*, „IEEE Software", vol. 17(4), s. 19–25, https://doi.org/10.1109/52.854064

Williams Wendy M., Ceci Stephen J. (2015), *National hiring experiments reveal 2:1 faculty preference for women on STEM tenure track*, „Proceedings of the National Academy of Sciences", vol. 112(17), s. 5360–5365, https://doi.org/10.1073/pnas.1418878112

WiMLDS (2018), http://wimlds.org/ (dostęp: 26.09.2018).

Winner Langdon (1980), *Do Artifacts Have Politics?*, „Daedalus", vol. 109(1), s. 121–136.

Witkowska Dorota (2002), *Sztuczne sieci neuronowe i metody statystyczne. Wybrane zagadnienia finansowe*, Wydawnictwo C.H. Beck, Warszawa.

Wolpert David H., Macready William G. (1997), *No free lunch theorems for optimization*, „IEEE Transactions on Evolutionary Computation", vol. 1(1), s. 67–82, https://doi.org/10.1109/4235.585893

Wong Gillian, Chin Josh (2016), *China's New Tool for Social Control: A Credit Rating for Everything*, „The Wall Street Journal", 28 listopada, https://www.wsj.com/articles/chinas-new-tool-for-social-control-a-credit-rating-for-everything-1480351590 (dostęp: 21.01.2019).

Wong Julia Carrie (2018), *Mark Zuckerberg apologises for Facebook's 'mistakes' over Cambridge Analytica*, „The Guardian", https://www.theguardian.com/technology/2018/mar/21/mark-zuckerberg-response-facebook-cambridge-analytica (dostęp: 5.02.2019).

Woodgate Rob (2019), *What Is Slack, and Why Do People Love It?*, https://www.howtogeek.com/428046/what-is-slack-and-why-do-people-love-it/ (dostęp: 4.09.2019).

Wrenn Mary V. (2015), *Agency and neoliberalism*, „Cambridge Journal of Economics", vol. 39(5), s. 1231–1243, https://doi.org/10.1093/cje/beu047

Wu Jeff (1997), *Statistics = Data Science?*, https://www2.isye.gatech.edu/~jeffwu/presentations/datascience.pdf (dostęp: 11.11.2017).

Xie Yihui (2016), *Bookdown: Authoring Books and Technical Documents with R Markdown*, Chapman and Hall/CRC, New York.

Xie Yihui, Allaire J.J., Grolemund Garrett (2018), *R Markdown: The Definitive Guide*, Chapman and Hall/CRC, Boca Raton.

Xie Yihui, Thomas Amber, Hill Alison P. (2022), *blogdown: Creating Websites with R Markdown*, https://bookdown.org/yihui/blogdown/ (dostęp: 4.09.2019).

Yair Gad (2007), *Meritocracy*, [w:] G. Ritzer (red.), *The Blackwell Encyclopedia of Sociology*, John Wiley & Sons, Ltd., Oxford, https://doi.org/10.1002/9781405165518.wbeosm082

Yau Nathan (2009), *Rise of the Data Scientist*, https://flowingdata.com/2009/06/04/rise-of-the-data-scientist/ (dostęp: 12.12.2017).

Yegulalp Serdar (2017), *Facebook brings GPU-powered machine learning to Python*, https://www.infoworld.com/article/3159120/facebook-brings-gpu-powered-machine-learning-to-python.html (dostęp: 17.07.2019).

Youtie Jan, Porter Alan L., Huang Ying (2016), *Early social science research about Big Data*, „Science and Public Policy", vol. 44(1), s. 1–10, https://doi.org/10.1093/scipol/scw021

Zagórna Anna (2020), *Unia wypuściła „Białą księgę sztucznej inteligencji"*, https://www.sztucznainteligencja.org.pl/unia-wypuscila-biala-ksiege-sztucznej-inteligencji/ (dostęp: 23.02.2020).

Zaharia Matei, Franklin Michael J., Ghodsi Ali, Gonzalez Joseph, Shenker Scott, Stoica Ion, Xin Reynold S., Wendell Patrick, Das Tathagata, Armbrust Michael, Dave Ankur, Meng Xiangrui, Rosen Josh, Venkataraman Shivaram (2016), *Apache Spark: a unified engine for big data processing*, „Communications of the ACM", vol. 59(11), s. 56–65, https://doi.org/10.1145/2934664

Zaród Marcin (2018), *Aktorzy-sieci w kolektywach hakerskich w Polsce*, praca doktorska, Uniwersytet Warszawski, Warszawa.

Zawistowska Alicja (2013) *„Płeć matematyki". Zróżnicowania osiągnięć ze względu na płeć wśród uzdolnionych uczniów*, „Studia Socjologiczne", nr 3(210), s. 75–95.

Zeileis Achim, the R community. Contributions (fortunes, or code) by Hothorn Torsten, Dalgaard Peter, Ligges Uwe, Wright Kevin, Maechler Martin, Brinchmann Halvorsen Kjetil, Hornik Kurt, Murdoch Duncan, Bunn Andy, Brownrigg Ray, Bivand Roger, Graves Spencer, Lemon Jim, Kleiber Christian, Reiner David L., Gunter Berton, Koenker Roger, Berry Charles, Schwartz Marc, Dewey Michael, Bolker Ben, Dunn Peter, Goslee Sarah, Blomberg Simon, Venables Bill, Rau Roland, Petzoldt Thomas, Turner Rolf, Leeds Mark, Charpentier Emmanuel, Evans Chris, Sonego Paolo, Ehlers Peter, Steuer Detlef, Galili Tal, Snow Greg, Ripley Brian D., Sumner Michael, Winsemius David, Andronic Liviu, Diggs Brian, Stigler Matthieu, Friendly Michael, Eddelbuettel Dirk, Heiberger Richard M., Burns Patrick, Menne Dieter, Vries Andrie de, Rowlingson Barry, Lancelot Renaud, Weylandt R. Michael, Skoien Jon Olav, Morneau Francois,

Unwin Antony, Wiley Joshua, Therneau Terry, Hanson Bryan, Singmann Henrik, Szoecs Eduard, Passolt Gregor, Nash John C. (2016), *fortunes: R Fortunes*, https://cran.r-project.org/package=fortunes (dostęp: 17.07.2019).

Zhang Amy X., Muller Michael, Wang Dakuo (2020), *How do Data Science Workers Collaborate? Roles, Workflows, and Tools*, http://arxiv.org/abs/2001.06684 (dostęp: 23.02.2020).

Zhao Shuai, Talasila Manoop, Jacobson Guy, Borcea Cristian, Aftab Syed Anwar, Murray John F. (2018), *Packaging and Sharing Machine Learning Models via the Acumos AI Open Platform*, [w:] *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, IEEE, Orlando, s. 841–846, https://ieeexplore.ieee.org/document/8614160/ (dostęp: 17.07.2019).

Zuboff Shoshana (2015), *Big other: Surveillance Capitalism and the Prospects of an Information Civilization*, „Journal of Information Technology", vol. 30(1), s. 75–89, https://doi.org/10.1057/jit.2015.5

Zuboff Shoshana (2019), *The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power*, PublicAffairs, New York.

Żulicki Remigiusz (2016), *Big Data – nowa wiedza?*, [w:] M. Maciąg, F. Polakowski (red.), *Postęp cywilizacyjny – stan obecny i perspektywy*, Wydawnictwo Naukowe TYGIEL, Lublin, s. 19–31.

Żulicki Remigiusz (2017), *Potencjał Big Data w badaniach społecznych*, „Studia Socjologiczne", nr 3(226), s. 176–207.

Żulicki Remigiusz (2019), *Pułapki myślowe data-driven. Krytyka (nie tylko) metodologiczna*, „Marketing i Rynek", nr 8(XXVI), s. 3–14, https://doi.org/10.33226/1231-7853.2019.8.1

Żulicki Remigiusz, Żytomirski Michał (2020), *Próba wypracowania metodologii pomiaru baniek filtrujących w wyszukiwarce Google*, „Zarządzanie Mediami", nr 8(3), s. 243–257, https://doi.org/10.4467/23540214ZM.20.034.12052

Spis rysunków

Spis tabel

Czy sztuczna inteligencja pozbawia nas pracy? Algorytmy przejmują władzę nad światem? Czy big data sprawia, że jesteśmy bezustannie inwigilowani, a ogromna ilość danych zastępuje ekspertów i naukowców? Cokolwiek sądzimy na te tematy, jedno jest pewne – istnieje heterogeniczne środowisko ludzi zajmujących się tzw. „sztuczną inteligencją" czy tzw. „big data" od strony technicznej oraz metodologicznej. Pole ich działania nazywane jest data science, a oni – data scientists.

Publikacja to pierwsza monografia socjologiczna dotycząca data science i pierwsza praca w naukach społecznych, w której data science zostało zbadane jako społeczny świat w rozumieniu Adele E. Clarke. Podejście to pozwala spojrzeć na data science, nazwane dekadę wstecz w „Harvard Business Review" „najseksowniejszym zawodem XXI wieku", zarówno z perspektywy jego uczestników, jak i z lotu ptaka, w relacji do akademii, biznesu, prawa, mediów czy polityki.

Remigiusz Żulicki jest adiunktem w Katedrze Metod i Technik Badań Społecznych w Instytucie Socjologii na Wydziale Ekonomiczno-Socjologicznym Uniwersytetu Łódzkiego. Zajmuje się socjologią cyfrową i metodologią badań społecznych. Entuzjasta otwartego oprogramowania i otwartej nauki. Niniejsza książka powstała na podstawie jego rozprawy doktorskiej, obronionej z wyróżnieniem w 2021 roku.

WYDAWNICTWO UNIWERSYTETU ŁÓDZKIEGO

wydawnictwo.uni.lodz.pl
ksiegarnia@uni.lodz.pl
(42) 665 58 63

Książka dostępna również jako e-book

ISBN 978-83-8331-110-4