Martin Braschler · Thilo Stadelmann
Kurt Stockinger *Editors*

Applied Data Science

Lessons Learned for the Data-Driven Business

Springer

Applied Data Science

Martin Braschler • Thilo Stadelmann •
Kurt Stockinger
Editors

Applied Data Science

Lessons Learned for the Data-Driven Business

Springer

Editors
Martin Braschler
Inst. of Applied Information Technology
ZHAW Zurich University
of Applied Sciences
Winterthur, Switzerland

Thilo Stadelmann
Inst. of Applied Information Technology
ZHAW Zurich University
of Applied Sciences
Winterthur, Switzerland

Kurt Stockinger
Inst. of Applied Information Technology
ZHAW Zurich University
of Applied Sciences
Winterthur, Switzerland

ISBN 978-3-030-11820-4 ISBN 978-3-030-11821-1 (eBook)
https://doi.org/10.1007/978-3-030-11821-1

Preface

In early 2013, the three editors of this volume were instrumental in founding the ZHAW Datalab, a Data Science Research Institute (DSRI)[1] at Zurich University of Applied Sciences.[2] "Big data" was big in the public press,[3] but DSRIs were still rare, especially in Europe. Both for our colleagues and us, it was the natural thing to do: joining forces to create synergies internally and demonstrate critical mass outwardly to ultimately facilitate better applied research projects (for which selecting project team members and acquiring funding becomes much easier in a larger group). The initial idea was to form a network of experts[4] that would engage in regular project-based collaborations without much administrative overhead. The goal of that partnership had been to perform applied research projects between academia and industry at the interface of structured and unstructured data analysis. The already existing strong partnership not only gave us confidence in the validity of the approach, it also made explicit the very need that led to the foundation of the ZHAW Datalab: the need for a concise description of what we were doing. Let us explain.

[1] A DSRI is a university-wide initiative integrating researchers from different disciplines predominantly occupied with a wide range of aspects surrounding the analysis of data.

[2] A snapshot of these beginnings is contained in Fig. 1. The founders of the ZHAW Datalab also published an early position paper (Stadelmann, Stockinger, Braschler, Cieliebak, Baudinot, Dürr and Ruckstuhl, "*Applied Data Science in Europe—Challenges for Academia in Keeping Up with a Highly Demanded Topic*", ECCS 2013). See also www.zhaw.ch/datalab

[3] The community that formed around the term was also very active, as can be seen, for example, in the history of meetings of the Swiss Big Data User Group (see https://www.meetup.com/swiss-big-data/). One of the editors (K.S.) even gave a talk at the first meetup of this group when it was still called "Swiss Hadoop User Group."

[4] A similar concept has been demonstrated by the Network Institute of the Vrije Universiteit Amsterdam (see http://www.networkinstitute.org/).

How We Became Data Scientists

Basically, we were able to capitalize on the emergence of the new data science field at exactly the right time. The ZHAW School of Engineering had been very successful in executing many applied research projects for more than 10 years prior to the inception of the ZHAW Datalab in 2013. Most of these projects were firmly "located" at the interfaces of the disciplines that today make up data science. In particular, computer scientists and statisticians joined forces and worked on problems integrating and analyzing structured and unstructured data. This was applied data science at work "par excellence." However, there was no "elevator pitch" for the kinds of problems we were working on together with our colleagues, no easy way to describe the ideas to funding agencies and prospective industry partners. If nothing less, the term "data science" delivered a concise description of what we perceived to be the field we were working in (Fig. 1).

One of the first joint activities within Datalab was to organize a workshop to perform a reality check on the potential of the topic of data science.[5] SDSI2014, the first Swiss Workshop on Data Science already exceeded our proudest expectation of attendees by a factor of 2 (see also Fig. 2); since then, the workshop has grown into a

Fig. 1 Five of the seven founders of the ZHAW Datalab in one of their first board meetings, with two of the editors (K.S. and T.S.) in the back row and the third (M.B.) taking the picture. The bottom row shows Gerold Baudinot (left), Andreas Ruckstuhl and Oliver Dürr (right), while Mark Cieliebak is missing (picture courtesy of T.S.)

[5]While a search conducted on LinkedIn for the terms "data scientist switzerland" returns more than 1500 hits as of early 2018, in 2013 it found only two persons (this credit goes to Violeta Vogel of PostFinance, and Daniel Fasel then of Swisscom: https://www.linkedin.com/in/violeta-vogel-3a556527/, https://www.linkedin.com/in/danielfasel/).

Fig. 2 Impressions from SDSI2014, the first Swiss Workshop on Data Science: Michael Natusch (left) delivers his insight into the core values of big data in front of parts of the audience (right; pictures courtesy of T.S.)

series of conferences that attracts a majority of the Swiss data science community. The growing interest in data science resulted in a significant increase of applied research projects that were initiated by the members of Datalab. Reflecting on the ever-growing number of people identifying themselves as data scientists and projects being described as data science projects led us to identify an additional need.

Why This Book Is Relevant

While data science builds on foundations from other disciplines, it is inherently an interdisciplinary and applied endeavor. The goal of data science is not only to work in one of its constituting sub-disciplines per se (e.g., machine learning or information systems), but to apply such methods and principles to build data products for specific uses cases that generate value from data. While very valuable textbooks exist on the individual subdisciplines,[6] the data science literature is missing a volume that acknowledges the applied science context of data science by systematically showing the connection between certain principles and methods, on the one end, and their application in specific use cases, on the other. One of the major goals of this book is to provide the reader with relevant lessons learned from applied data science projects at the intersection of academia and industry.

How to Read the Book

This book is organized into three parts: Part I pays tribute to the interdisciplinary nature of data science and provides a common understanding of data science terminology for readers with different backgrounds. The book is not a replacement for classical textbooks (i.e., it does not elaborate on fundamentals of certain methods

[6]See for example http://www.learndatasci.com/free-data-science-books/

and principles described elsewhere), but defines applied data science, the work of a data scientist, and the results of data science, namely, data products. Additionally, Part I sheds light on overarching topics such as legal aspects and societal risks through widely applied data science. These chapters are geared toward drawing a consistent picture of data science and are predominantly written by the editors themselves. We recommend the reader to work through the first four chapters in order.

Part II broadens the spectrum by presenting views and insights from diverse authors—some from academia, some from industry, some from Switzerland, some from abroad. These chapters describe a fundamental principle, method, or tool in data science by means of analyzing specific use cases and drawing concrete lessons learned from them. The presented case studies as well as the applied methods and tools represent the nuts and bolts of data science. The chapters in Part II can be read in any order, and the reader is invited to focus on individual chapters of interest.

Part III is again written from the perspective of the editors and summarizes the lessons learned of Part II. The chapter can be viewed as a meta study in data science across a broad range of domains, viewpoints and fields. Moreover, the chapter provides answers to the following question: What are the mission critical factors for success in different data science undertakings? Part III is written in a concise way to be easily accessible even without having read all the details of the case studies described in Part II.

Who Should Read the Book

While writing and editing the book, we had the following readers in mind: first, practicing data scientists in industry and academia who want to broaden their scope and enlarge their knowledge by assimilating the combined experience of the authors. Second, decision-makers in businesses that face the challenge of creating or implementing a data-driven strategy and who want to learn from success stories. Third, students of data science who want to understand both the theoretical and practical aspects of data science vetted by real case studies at the intersection of academia and industry.

Thank You

We thank you, the reader, for taking the time to learn from the collected insights described in this book. We as editors are university lecturers and researchers in our primary job; it is an immense pleasure and honor for us to be able to convey our insights. We are also very grateful for the trust and patience we received from our publisher, Springer, specifically impersonated by Ralf Gerstner. We want to thank our coauthors that contributed excellent work that is fundamental for making this

book a success. Finally, we thank our students, colleagues, and partners from the Datalab, the Master of Advanced Studies in Data Science Program, and the Swiss Alliance for Data-Intensive Services for providing the environment in which this book project (and some of the reported use cases) could flourish.

Specifically, I (Martin Braschler) thank my co-editors for consistently engaging and stimulating discussions, Vivien Petras and the team of the Berlin School of Library and Information Science at the Humboldt-Universität zu Berlin for hosting me during part of the period I worked on this book, my colleagues that I have collaborated with in past projects and who have thus informed my understanding of data science topics, and last but not least my family, who provides for me the much needed balance to life as a university teacher and researcher.

I (Thilo Stadelmann) thank my co-editors and Michael Brodie for helpful discussions and valuable lessons in collaboration. Thank you for your patience and collegiality. I learned a lot. Thanks go to Geri Baudinot for enabling the ZHAW Datalab and further developments by his vision, patronage, and mentorship. My final "thank-you" is best expressed with a quote adapted from Reuben Morgan: "*Freely you gave it all to me… great is the love, poured out for all, this is my god.*"

I (Kurt Stockinger) thank my wife Cinthia and my two little kids Luana and Lino for the ability to work on the book during calm evening hours—after having changed diapers and read several good night stories that did not contain data science topics.

Winterthur, Switzerland
Spring 2019

Martin Braschler
Thilo Stadelmann
Kurt Stockinger

Contents

Part I
Foundations

Chapter 1
Introduction to Applied Data Science

Thilo Stadelmann, Martin Braschler, and Kurt Stockinger

Abstract What is data science? Attempts to define it can be made in one (prolonged) sentence, while it may take a whole book to demonstrate the meaning of this definition. This book introduces data science in an applied setting, by first giving a coherent overview of the background in Part I, and then presenting the nuts and bolts of the discipline by means of diverse use cases in Part II; finally, specific and insightful lessons learned are distilled in Part III. This chapter introduces the book and provides an answer to the following questions: What is data science? Where does it come from? What are its connections to big data and other mega trends? We claim that multidisciplinary roots and a focus on creating value lead to a discipline in the making that is inherently an interdisciplinary, applied science.

1 Applied Data Science

It would seem reasonable to assume that many readers of this book have first really taken notice of the idea of "data science" after 2014. Indeed, while already used sparingly and with different meanings for a long time, widespread use of the term "data science" dates back to only 2012 or thereabouts (see Sect. 2). Of course, the "substance" of the field of data science is very much older and goes by many names. To attest to this fact, the institute at which the editors of this book are located has given itself the mission to "build smart information systems" already back in 2005. And the main fields of work of the respective editors (Information Retrieval, Information Systems/Data Warehousing, and Artificial Intelligence/Machine Learning) all have traditions that span back for decades. Still, a fundamental shift has been underway since 2012.

T. Stadelmann (✉) · M. Braschler · K. Stockinger
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stdm@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_1

This chapter traces this fundamental shift by giving a historical account of the various roots of data science.[1] However, what do we understand by the term data science? As for this book, we adopt a definition that attests to the breadth and history of the field, is able to discriminate it from predecessor paradigms, and emphasizes its connection to practice by having a clear purpose:

Data science refers to a unique blend of principles and methods from analytics, engineering, entrepreneurship and communication that aim at generating value from the data itself.

These principles and methods are diverse (e.g., spanning disciplines from IT to legal studies) and are applied to all kinds of data (from relational to multimedia) to explicitly achieve a specific end: added value. This makes data science inherently an interdisciplinary and applied science and connects the term closely with the definitions of a *data product* (an exploitable insight derived from the collected facts) and the *data scientist* (the one carrying out data science endeavors). All three terms will be more thoroughly treated, rooted, and discussed in Chaps. 2–4.

2 The History of Data Science

This section intends to give a concise answer to two questions: what is the connection between data *science* and business, especially in the presence of a massive media hype? And what fueled the availability of (and trust in) large-scale data analysis? A more complete overview of the most influential publications leading to data science as we know it is given by Press (2013).

2.1 Data Science, Business, and Hype

The understanding of data science as the field concerned with all aspects of making sense of data goes back to discussions in the scientific community that started with Tukey (1962)[2] and where summarized by Cleveland (2001) in requiring an independent scientific discipline in extension to the technical areas of the field of statistics. Notable mentions go to the foundation of the field of "knowledge discovery in databases" (KDD) (Fayyad et al. 1996) after the KDD workshop in 1989,[3] the first mentioning of "data science" in the title of a scientific conference in 1996

[1] Based on updated, translated, and considerably extended versions of (Stockinger and Stadelmann 2014; Stockinger et al. 2016).

[2] About the same time as Tukey used "data science" in reference to statistics, Peter Naur in Sweden used the term (interchangeably with "datalogy") to refer to computer science (Sveinsdottir and Frøkjær 1988).

[3] See https://www.kdnuggets.com/meetings/kdd89/index.html. Since 1995, the term "data mining" has risen to prominence: http://www.aaai.org/Conferences/KDD/kdd95.php

(Hayashi et al. 1996), and Leo Breiman's (2001) famous call to unite statistical and computational approaches to modeling data. These events lead to the foundation of the first scientific data science journals[4] in 2002 and the first data science research centers in 2007.[5]

However, widespread recognition beyond a scientific community, including the dynamics we see today, only started after certain contributions from business: Hal Varian tells the McKinsey Quarterly in 2009 that the "*sexy job of the next 10 years will be statisticians*" (Manyika 2009). This view broadens beyond statistics after the introduction of the term "data scientist" by Patil and Hammerbacher in 2008 during their collaboration at LinkedIn and Facebook (Patil 2011). Both felt the need for a new job description for their team members that, on the one hand, got deep engineering know-how and, on the other hand, were directly shaping the economic value of the company's core products: "*those who use both data and science to create something new.*" Earlier, Davenport and Harris (2007) and others had prepared the way for an acceptance of those terms (Smith 2011) by influential popular scientific books that continued to accompany the development of data science (Siegel 2013; Domingos 2015).

The two concepts—the field of data science as well as the job description of a data scientist—in their now popular form (Loukides 2010) together with their fame per se thus ultimately resulted from a need and development within businesses[6] (Patil 2011). The scientific discussion, once in a leading role, had difficulty to keep up with the dynamics of 2010–2015 and followed with some delay (Provost and Fawcett 2013; Stadelmann et al. 2013). It is currently accelerating again (see Brodie (2015b) and his chapters later in this book).

Omnipresent, however, since the adoption of the topic in mass media has been a hype [see some of its expressions, e.g., in Humby (2006) or Davenport and Patil (2012)] strong enough to provoke skepticism even in benevolent experts. While hype can lead to unreflected and hence bad decisions on all levels (from job choice to entrepreneurial and legislative agenda setting), it should not cloud the view on the real potential and challenges of data science:

- *Economic potential*: the McKinsey Global Institute estimates the net worth of the open data market alone to be three trillion dollars (Chui et al. 2014). A recent update explains that this potential is not realized yet, and certainly not overhyped

[4] See, e.g., http://www.codata.org/publications/data-science-journal (inaugurated 2002, relaunched 2015) and http://www.jds-online.com/ (since 2003).

[5] See, e.g., http://datascience.fudan.edu.cn/ (under the term "Dataology and Data Science").

[6] This anchoring of the modern understanding of data science more in business than in academia is the main reason for many of the references in this work pointing to blog posts and newspaper articles instead of scientific journals and conference papers. It reflects current reality while not making the point that academia is subordinate. Data science as a field and business sector is in need of the arranging, normative work of academia in order to establish solid methodical foundations, codes of conduct, etc. This book is meant as a bridge builder in this respect.

(Henke et al. 2016). Earlier, Manyika et al. (2011) estimated a total shortcoming of 1,90,000 new data scientists.

- *Societal impact*: data analytics affects medical care (Parekh 2015), political opinion making (Harding 2017; also Krogerus and Grassegger 2016 and the aftershocks of the US presidential election 2016 with regards to the involvement of the company Cambridge Analytica), and personal liberty (Li et al. 2015; see also Chap. 6 on risks and side effects).
- *Scientific influence*: data-intensive analysis as the fourth paradigm of scientific discovery promises breakthroughs in disciplines from physics to life sciences (Hey et al. 2009; see also Chap. 8 "on developing data science").

Hype merely exclaims that "*data is the new oil!*" and jumps to premature conclusions. The original quote continues to be more sensible: "*[. . .] if unrefined, it cannot really be used. It has to be changed [. . .] to create a valuable entity that drives profitable activity*" (Humby 2006). This already hints at the necessity of the precise and responsible work of a data scientist, guided by a body of sound principles and methods maintained within a scientific discipline. However, how did individual voices of "big data evangelists" grow into a common understanding of the power and usefulness of the resource of data by means of analytics?

2.2 Different Waves of Big Data

Data science has profound roots in the history of different academic disciplines as well as in science itself (see also the detailed discussion in the next chapter). The surge of large-scale science in the second half of the twentieth century, typified by facilities like CERN[7] or the Hubble Space Telescope,[8] is the direct enabler of data science as a paradigm of scientific discovery based on data. These facilities have arguably enabled the *first wave* of big data: a single experiment at CERN, for example, would generate hundreds of terabytes of data in just one second, if not for a hardware filter that would do a preselection of what to record.

Consequently, specific projects like RD45 (Shiers 1998) where launched already in the 1990s to manage these high volumes of data before any commercial database management system was able to host petabytes (Düllmann 1999).[9] This rise in scientific data volumes was not just due to technical ability but due to a change of paradigm (Hey et al. 2009): The first paradigm of science basically was to perform theoretical studies; the second paradigm added empiricism: the experimental

[7] See https://home.cern/ (the website of the web's birthplace).

[8] See http://hubblesite.org/

[9] Additionally, unlike with relational databases in industry at that time, the types of data to be stored for scientific experiments frequently comprised numerical data (such as temperature, velocity, or collision counts for particles), often stored in object-oriented database systems or images (e.g., from stars or galaxies), both at higher volumes and speed.

Fig. 1.1 Examples of the amount of data created every minute publicly on the web, as of September 2018 and compared to mid-2017. Adapted from: Domo (2017, 2018)

evaluation of theoretical hypotheses. Because of the complexity and expensiveness of large-scale scientific experiments like at CERN, computer simulations emerged as the third paradigm of scientific discovery (called computational science). Now, the fourth paradigm is data-intensive science: evaluating, for any given experiment, *all* the facts (i.e., the complete data set, not just sub-samples and hand-engineered features), combining them to *all possible* probabilistic hypotheses (see also Brodie's chapter on "what is data science" later).

Following large-scale science, the *second wave* of big data was triggered by internet companies[10] like Google, Yahoo, or Amazon at the beginning of the twenty-first century. In contrast to scientific data, web companies originally focused on managing text data; the continuing data explosion (see Fig. 1.1) is still fueled by additionally indexing more and more images and videos. Social media companies such as Facebook and LinkedIn gave individuals—instead of large-scale scientific facilities—the ability to contribute to the growth of data; this is regarded as the *third wave* of the data tsunami. Finally, the *fourth wave* is currently rolling up based on the rise of machine-generated data, such as log-files and sensor data on the Internet of things.

[10]All of the following three example companies have transformed themselves considerably since the times of the second wave (see https://abc.xyz/ and https://www.oath.com/ for Google and Yahoo, respectively; https://www.amazon.com/ still looks familiar, but compare http://www.visualcapitalist.com/jeff-bezos-empire-chart/).

Fig. 1.2 Snapshot from the preparations of a business road show in illustration of the hopes and dreams connected with the term "big data" as used by the public press around 2013 (picture courtesy of T.S.). The hopes and dreams have been seamlessly transferred to other wording as of the writing of this book. This is in stark contrast to the continuous scholarly work on big data discussed, e.g., by Valarezo et al. (2016) and the scientific community that adopted the same name (see, e.g., https://journalofbigdata.springeropen.com/)

3 Data Science and Global Mega Trends

The scientific and commercial development of data science has been accompanied by considerable buzz in the public press. In this section, we review the contemporary trends of big data, AI, and digitalization, respectively, and put the terms into the context of the professional discussion. Our goal is to disentangle the meaning of the terms as hype words from their scientific definition by showing discrepancies in public understanding from what experts refer to when using largely overlapping vocabulary, thus contributing to a successful dialog (Fig. 1.2).

3.1 Big Data

In 2013–2014, "big data" was the favorite buzzword in business: the new oil (McAfee et al. 2012) of the economy. It alluded exclusively to the technical origins of the term, namely, Laney's (2001) "3 Vs" (variety, velocity, and volume) as

attributes of the growing flood[11] of data: the amount, the number of sources, and the speed at which new data arrives is very large, that is, "big" (Soubra 2012). It is this technical definition that is also eponymous of the scientific sub-discipline of information systems research that encompasses the development of database management systems at scale.[12] Data science, on the contrary, is far from focusing exclusively on "big" in the sense of "large" data: while large itself is always a relative term as compared to the computational resources of the day,[13] data science is concerned with generating value from *all kinds of* data (see Braschler's later chapter on "small data").

There is a different, more economical than technological connotation to the public discussion on big data that conveys some meaning for the data scientist: it is best seen in a historic version of the Wikipedia article on big data from 2014.[14] In its "See also" section, the term "big data" is brought into relationship with terms like "big oil," "big tobacco," "big media," etc. The connotation is as follows: at least as much as from the description of the phenomenon of increased variety/velocity/volume, the term big data might stem from a whole economy's hope of having found the new oil (Humby 2006), of getting new business opportunities, and of launching the "next big thing" after the Internet and mobile revolution.[15] This explains why the term has been hyped a lot in business for several years. Additionally, this also expresses the fear of individuals being ushered into the hands of a "big brother," just as it is expressed as a present reality in the other "big*" terms from above.

The term "big data" up to here thus contributes a two-fold meaning to the professional discussion on data science: first, *technologically*, it gives a description of the growing state of data (and as such is a scientific sub-discipline of research in databases, information systems, and distributed systems). Second, *economically*, it expresses a hope for business opportunities and voices a subtle concern with respect to the attached dangers of this business. Both dimensions are worthy to be explored and have to be researched. The greatest potential of the term, however, may lie in pointing to the following *social* phenomenon[16]:

---

[11] In a variation of Naisbitt and Cracknell's (1984) famous quote on megatrends, Eric Brown (2014) said: "Today, we are drowning in data and starved for information."

[12] The top league of international database researchers meets under the umbrella of "very large data bases" (http://www.vldb.org/) since 1992.

[13] See, e.g., the seminal book by Witten et al. (1999) on "managing gigabytes" that was allegedly influential in building Google but would not qualify as discussing "big" a couple of years later. Similarly, the test collection at the first TREC conference, at 2 gigabytes of size, posed a considerable challenge to the participants in 1992 (Harman and Voorhees 2006).

[14] The following discussion is based on https://en.wikipedia.org/wiki/Big_data as of May 1, 2014. It is accessible via Wikipedia's history button.

[15] Dan Ariely expressed this in a Facebook post gone viral on January 6, 2013: "Big data is like teenage sex: everyone talks about it, nobody really knows how to do it, everyone thinks everybody else is doing it, so everybody claims they are doing it. . ." (see http://t.co/tREI1mRQ).

[16] This thought was first formulated by Michael Natusch in his Keynote at SDSl2014, the 1st Swiss Conference on Data Science (see also Fig. 1.1 in the preface to this book): https://www.zhaw.ch/en/research/inter-school-cooperation/datalab-the-zhaw-data-science-laboratory/sds2014/michael-natusch/

Table 1.1 AI hype at its worst. The metrics of the "Rocket AI launch party" prank at NIPS 2016 (Tez 2016)

RSVPs to party	316
CVs sent via email in advance	46
Well-known investors that got in touch to fund	5
Planning time	<8 h
Money spent	$79 for domain, $417 for alcohol, snacks and police fine
Estimated value of RocketAI	$\$O(10^7)$

People (and organizations) have changed their mindset in the last decade. Data is now regarded as being available (cheap if not for free) for virtually any aspect of the world, meaning that we can have facts for potentially any phenomenon on the planet. Additionally, we have the technology ready to automatically make use of these facts via the principles and methods of data science, for almost all existing business processes. This enables optimized (i.e., fact-based) decisions, which have a measurable value independent of the data having properties of up to *n* Vs (Vorhies 2014). This social value of "big data" thus is this: it refers to *a big change in thinking about the possibilities of data-driven automated decision making.*

3.2 Artificial Intelligence

The term "big data" was largely forsaken by the public press after circa 2 years of constant use (White 2015), but it did not leave a vacuum: unforeseen breakthroughs in deep learning technology as of 2012 ended the last AI winter[17] around 2015 (see also Stadelmann et al.'s later chapter on "deep learning in industrial practice") and directly turned it into the next "AI" hype. This unreasonable cycle of popularity can be traced in the open, too, for example, Simard et al. (2003) display the AI winter of the 2000s with the following quote: "[. . .] it was even pointed out by the organizers of the Neural Information Processing System (NIPS) conference that the term 'neural networks' in the submission title was negatively correlated with acceptance." On the other hand, the RocketAI story (Tez 2016) illustrates the peak of the hype: at the NIPS conference of 2016, expectations in neural networks where again so high that the joke of two PhD students of a "launch party" for a fake, fancy AI start-up attracted large unsolicited funding, applications, and attendees with minimum effort within a day (see Table 1.1).

AI in the public press as of 2018 mainly refers to the expectation to "do with computers anything that a human could do—and more." It often ascribes human-like properties to AI systems as reflected in larger parts of the discussion revolving

[17] See https://en.wikipedia.org/wiki/AI_winter

around terms like robots (Kovach 2017), digital assistants (Kremp 2018), self-driving cars (Roberts 2018), chatbots (Spout Social 2018), and neural networks as digital brains (Gruber 2017). What contributes to (if not even causes) the high, even inflated, expectations is the use of terms originally coined for intelligent living beings ("intelligence," "learning," "cognitive," "social"). Everybody has an intuitive understanding of what "intelligence" means in a human context and subconsciously ascribes said properties to the technical system.

The scientific community is not innocent of this dilemma, even if it tries to clarify things (Brooks 2017): in private communication,[18] one of the fathers of AI regretted coining the term "artificial intelligence" at the Dartmouth workshop in 1956 exactly for the outgrowths described above, mentioning that it would have been wiser (but maybe less successful) to have gone with the second proposed name for the field—"*complex computer applications.*" It is exactly this that defines the scientific discipline of AI today (Russell and Norvig 2010): a collection of diverse techniques, from efficient search algorithms to logic to various shades of machine learning, to solve tasks of everyday life that usually can only be solved by humans, in an attempt that *might look intelligent* from the outside. As such, the field of AI is not concerned with researching intelligence per se nor in reproducing it; but it delivers practical solutions to problems, for example, in business for many decades (Hayes-Roth et al. 1983), even with deep learning (LeCun et al. 1998).

3.3 Digitalization

Different technological and industry-specific trends[19] additionally get summarized under the unifying term "digitalization"[20] in the public discourse. The added value of this term per se can reasonably be questioned: things arguably became digital—digitized—since the IT revolution of businesses and societies in the last century. The modern use extends this mega trend by emphasizing increased interconnectedness (social networks) and automation. This trend is specifically enabled by data science, based on the availability of digital data ("big data" in the social sense above) and analytics technologies ("AI"). It spans almost all industry and societal branches, and hence the discussion not only involves data science professionals—technical people—but also sociologists, politicians, etc., with valuable contributions to the phenomenon at large.

The missing selectivity of the public use of "digitalization" and the abovementioned "buzz" words create a problem: experts and laypeople speak in

[18]The statement was orally witnessed at an AI planning conference in the 1990s by a colleague who wishes to remain anonymous.

[19]Compare terms like FinTech (Dapp et al. 2014), MedTech (MedTech Europe 2018), EdTech, etc., (Mayer 2016) as well as Industrie 4.0 (Kagermann et al. 2011).

[20]Not "digitization"—compare the article by Clerck (2017).

the same terms but mean different things. The new "AI algorithm" inside a company's product is potentially more statistics than AI, speaking in technical terms; the just purchased "big data platform" might likely refer to an analytics tool not specifically designed to handle large data (as would be the case if called such by a big data researcher); and digitalization changes our education, but likely not predominantly in the sense that we now have to replace human teachers, but by teaching students skills in handling a digitalized society (including skills in handling digital media and basic understandings of data science technology) (Zierer 2017).

The missing selectivity in the use of terms cannot be removed from the discourse.[21] It is thus important for data professionals—data scientists—to understand what experts and laypeople mean and hear when speaking of such terms, in order to anticipate misunderstandings and confront unreasonable expectations.

4 Outlook

What is the future of data science? Data science as a discipline of study is still *in its infancy* (Brodie 2015a), and the principles and methods developed in its underlying disciplines have to be furthered in order to adapt to the phenomenon we called big data in the previous section. This maturing of data science will be addressed in two later companion chapters by Michael Brodie in Part II of the book.

From a business perspective, data science will continue to deliver value to most industries[22] by introducing possibilities for automation of repetitive tasks.[23] A recent overview of successful data-driven innovations in Switzerland, presented at the first "Konferenz Digitale Schweiz," showed the overall potential by demonstrating that the innovation depth in current data science projects is surprisingly low (Swiss Alliance for Data-Intensive Services 2017): one third of business innovations was achieved by deploying technology and processes that have been well-known for decades; another third was achieved by consulting companies on recent innovations in data-driven business models and technologies; and for only one third, applied

[21] Some experts suggest to use all the discussed terms synonymously for the sake of simplicity, e.g., Brodie in his later chapter on "developing data science" speaks of "AI/data science". While this might be appropriate in certain situations, maintaining proper attribution certainly helps in other situations by maintaining concise and effective communication. Using precise terminology prevents inflated expectations, describes true expertise, and gives guidance in where to find it (e.g., in what discipline).

[22] François Bancilhon put it frankly in his ECSS 2013 keynote: "*most industries will be disrupted*" (see http://www.informatics-europe.org/ecss/about/past-summits/ecss-2013/keynote-speakers.html).

[23] See Brooks (2017) for a counter argument on the hopes (or fears) that too many jobs could be automated in a nearer future. But as Meltzer (2014) points out, repetitive tasks even in jobs considered high-profile (e.g., medical diagnosis or legal advice) could be automated quite well: automation potential lies in repetitiveness per se, not the difficulty of the repetitive task.

research projects fostered the foundation for the business innovation. Part II of this book reports on numerous case studies emerging from that latter third.

While the benefit to businesses is fairly obvious and easily measurable in terms of profit, the effect of data science on our societies is much less well understood. Recent reports warn about potential blind spots in our core technologies (Rahimi and Recht 2017) and go as far as suggesting to treat AI technology similar to nuclear weapons in limiting access to research results (Brundage et al. 2018). The recent emergence of the word "techlash" might indicate society's first large-scale adverse reaction on the dawn of digitalization (Kuhn 2018). Clemens Cap explores such issues in his chapter towards the end of Part I of this book, while Widmer and Hegy shed some light on the legal space in which a data scientist operates.

The next three chapters will continue defining what data science, a data scientist, and a data product is, respectively. As stated in the preface, they are best read in order to get a coherent picture of the frame for this book. The remaining chapters can be approached in any order and according to personal interest or project need. A concise summary of all lessons learned will be presented in Part III. We intend this part to form best practices for applying data science that you will frequently refer to as you start your own professional data science journey.

References

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science, 16*(3), 199–231.

Brodie, M. L. (2015a). *The emerging discipline of data science*. Keynote at the 2nd Swiss Workshop on Data Science SDS|2015. Available May 3, 2018, from https://www.youtube.com/watch?v=z93X2k9RVqg

Brodie, M. L. (2015b). *Doubt and verify: Data science power tools*. Available March 23, 2018, from http://www.kdnuggets.com/2015/07/doubt-verify-data-science-power-tools.html

Brooks, R. (2017). The seven deadly sins of AI predictions. *MIT Technology Review*. Available March 28, 2018, from https://www.technologyreview.com/s/609048/the-seven-deadly-sins-of-ai-predictions/

Brown, E. D. (2014). *Drowning in data, starved for information*. Available March 27, 2018, from http://ericbrown.com/drowning-in-data-starved-for-information.htm

Brundage, M., Avin, S., Clark, J., Toner, H., Eckersley, P., Garfinkel, B., et al. (2018). The malicious use of artificial intelligence: Forecasting, prevention, and mitigation. *arXiv preprint arXiv:1802.07228*.

Chui, M., Farrell, D., & Jackson, K. (2014). *How government can promote open data*. Available March 23, 2018, from https://www.mckinsey.com/industries/public-sector/our-insights/how-government-can-promote-open-data

Clerck, J. (2017). Digitization, digitalization and digital transformation: The differences. *i-SCOOP*. Available March 23, 2018, from https://www.i-scoop.eu/digitization-digitalization-digital-transformation-disruption/

Cleveland, W. S. (2001). Data science: An action plan for expanding the technical areas of the field of statistics. *International Statistical Review, 69*(1), 21–26.

Dapp, T., Slomka, L., AG, D. B., & Hoffmann, R. (2014). *Fintech–the digital (r) evolution in the financial sector*. Frankfurt am Main: Deutsche Bank Research.

Davenport, T. H., & Harris, J. G. (2007). *Competing on analytics: The new science of winning*. Boston: Harvard Business Press.

Davenport, T. H., & Patil, D. (2012). *Data scientist: The sexiest job of the 21st century*. Available March 23, 2018, from http://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century/ar/1

Domingos, P. (2015). *The master algorithm: How the quest for the ultimate learning machine will remake our world*. New York: Basic Books.

Domo. (2017). *Data never sleeps 5.0*. Available March 23, 2018, from https://www.domo.com/learn/data-never-sleeps-5

Domo. (2018). *Data never sleeps 6.0*. Available October 9, 2018, from https://www.domo.com/learn/data-never-sleeps-6

Düllmann, D. (1999). Petabyte databases. *ACM SIGMOD Record, 28*(2), 506.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine, 17*(3), 37.

Gruber, A. (2017, January 17). Wenn Maschinen lernen lernen. *Spiegel Online*. Available May 10, 2018, from http://www.spiegel.de/netzwelt/web/kuenstliche-intelligenz-wenn-maschinen-lernen-lernen-a-1130255.html

Harding, C. (2017). *Digital participation – The advantages and disadvantages*. Available March 23, 2018, from https://www.polyas.de/blog/en/digital-democracy/digital-participation-advantages-disadvantages

Harman, D. K., & Voorhees, E. M. (2006). TREC: An overview. *Annual Review of Information Science and Technology, 40*(1), 113–155.

Hayashi, C., Yajima, K., Bock, H. H., Ohsumi, N., Tanaka, Y., & Baba, Y. (Eds.). (1996). *Data science, classification, and related methods: Proceedings of the fifth conference of the international federation of classification societies (IFCS-96)*, Kobe, Japan, March 27–30, 1996. Springer Science & Business Media.

Hayes-Roth, F., Waterman, D. A., & Lenat, D. B. (1983). *Building expert system*. Boston, MA: Addison-Wesley Longman.

Henke, N., Bughin, J., Chui, M., Manyika, J., Saleh, T., Wiseman, B., & Sethupathy, G. (2016). *The age of analytics: Competing in a data-driven world*. McKinsey Global Institute report.

Hey, T., Tansley, S., & Tolle, K. M. (2009). *The fourth paradigm: Data-intensive scientific discovery* (Vol. 1). Redmond, WA: Microsoft Research.

Humby, C. (2006, November). *Data is the new Oil!*. ANA Senior marketer's summit, Kellogg School. http://ana.blogs.com/maestros/2006/11/data_is_the_new.html

Kagermann, H., Lukas, W. D., & Wahlster, W. (2011). Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution. *VDI nachrichten, 13*, 11.

Kovach, S. (2017). *We talked to Sophia – The AI robot that once said it would 'destroy humans'*. Tech Insider youtube video. Available May 10, 2018, from https://www.youtube.com/watch?v=78-1MlkxyqI

Kremp, M. (2018, May 9). Google Duplex ist gruselig gut. *Spiegel Online*. Available May 10, 2018, from http://www.spiegel.de/netzwelt/web/google-duplex-auf-der-i-o-gruselig-gute-kuenstliche-intelligenz-a-1206938.html

Krogerus, M., & Grassegger, H. (2016). Ich habe nur gezeigt, dass es die Bombe gibt. *Das Magazin*, (48–3). Available May 11, 2018, from https://www.tagesanzeiger.ch/ausland/europa/Ich-habe-nur-gezeigt-dass-es-die-Bombe-gibt/story/17474918

Kuhn, J. (2018). "Techlash": Der Aufstand gegen die Tech-Giganten hat begonnen. *Süddeutsche Zeitung*. Available April 3, 2018, from http://www.sueddeutsche.de/digital/digitalisierung-techlash-der-aufstand-gegen-die-tech-giganten-hat-begonnen-1.3869965

Laney, D. (2001). 3D data management: Controlling data volume, velocity and variety. *META Group Research Note, 6*(70).

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

Li, R., Lu, B., & McDonald-Maier, K. D. (2015). Cognitive assisted living ambient system: A survey. *Digital Communications and Networks, 1*(4), 229–252.

Loukides, M. (2010). *What is data science?* Available March 23, 2018, from https://www.oreilly.com/ideas/what-is-data-science

Manyika, J. (2009). Hal Varian on how the Web challenges managers. *McKinsey Quarterly*. Available March 23, 2018, from https://www.mckinsey.com/industries/high-tech/our-insights/hal-varian-on-how-the-web-challenges-managers

Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., & Byers, A. H. (2011). *Big data: The next frontier for innovation, competition, and productivity*. Available March 23, 2018, from https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation

Mayer, M. (2016). *Fintech? Edtech? Adtech? Duriantech? – The 10 buzziest startup sectors*. Available March 23, 2018, from https://techsauce.co/en/startup-2/fintech-edtech-adtech-duriantech-the-10-buzziest-startup-sectors/

McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D. J., & Barton, D. (2012). Big data: The management revolution. *Harvard Business Review, 90*(10), 60–68. Available March 23, 2018, from https://hbr.org/2012/10/big-data-the-management-revolution

MedTech Europe. (2018). The European medical technology industry in figures 2018. *MedTech Europe Brochure*. Available May 10, 2018, from http://www.medtecheurope.org/EU-medtech-industry-facts-and-figures-2017

Meltzer, T. (2014). Robot doctors, online lawyers and automated architects: The future of the professions? *The Guardian.* Available March 28, 2018, from https://www.theguardian.com/technology/2014/jun/15/robot-doctors-online-lawyers-automated-architects-future-professions-jobs-technology

Naisbitt, J., & Cracknell, J. (1984). *Megatrends: Ten new directions transforming our lives (No. 04; HN59. 2, N3.)*. New York: Warner Books.

Parekh, D. (2015). *How big data will transform our economy and our lives*. Available March 23, 2018, from http://techcrunch.com/2015/01/02/the-year-of-big-data-is-upon-us/

Patil, D. (2011). *Building data science teams*. Available March 23, 2018, from http://radar.oreilly.com/2011/09/building-data-science-teams.html

Press, G. (2013). *A very short history of data science*. Available March 23, 2018, from https://www.forbes.com/sites/gilpress/2013/05/28/a-very-short-history-of-data-science

Provost, F., & Fawcett, T. (2013, March). Data science and its relationship to big data and data-driven decision making. *Big Data, 1*(1), 51–59.

Rahimi, A., & Recht, B. (2017). Reflections on Random Kitchen Sinks. Acceptance speech for Test of Time Award at NIPS 2017. Available March 28, 2018, from http://www.argmin.net/2017/12/05/kitchen-sinks/

Roberts, D. (2018, May 9). Here's how self-driving cars could catch on. *Vox article*. Available May 10, 2018, from https://www.vox.com/energy-and-environment/2018/5/8/17330112/self-driving-cars-autonomous-vehicles-texas-drive-ai

Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Upper Saddle River, NJ: Pearson Education.

Shiers, J. (1998). Building a multi-petabyte database: The RD45 project at CERN. In *Object databases in practice* (pp. 164–176). Upper Saddle River, NJ: Prentice Hall.

Siegel, E. (2013). *Predictive analytics: The power to predict who will click, buy, lie, or die*. Hoboken, NJ: Wiley.

Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. *ICDAR, 3*, 958–962.

Smith, D. (2011). *"Data Science": What's in a name?* Available March 27, 2018, from http://blog.revolutionanalytics.com/2011/05/data-science-whats-in-a-name.html.

Soubra, D. (2012). *The 3Vs that define Big Data*. Available March 23, 2018, from www.datasciencecentral.com/forum/topics/the-3vs-that-define-big-data

Spout Social. (2018). The complete guide to chatbots in 2018. *Sprout blog*. Available May 10, 2018, from https://sproutsocial.com/insights/topics/chatbots/

Stadelmann, T., Stockinger, K., Braschler, M., Cieliebak, M., Baudinot, G. R., Dürr, O., & Ruckstuhl, A. (2013, August). Applied data science in Europe – Challenges for academia in keeping up with a highly demanded topic. In *European computer science summit ECSS 2013*. Amsterdam: Informatics Europe.

Stockinger, K., & Stadelmann, T. (2014). Data Science für Lehre, Forschung und Praxis. *HMD Praxis der Wirtschaftsinformatik, 51*(4), 469–479.

Stockinger, K., Stadelmann, T., & Ruckstuhl, A. (2016). Data Scientist als Beruf. In D. Fasel, & A. Meier (Eds.), *Big data*. Edition HMD. https://doi.org/10.1007/978-3-658-11589-0_4.

Sveinsdottir, E., & Frøkjær, E. (1988). Datalogy—The Copenhagen tradition of computer science. *BIT Numerical Mathematics, 28*(3), 450–472.

Swiss Alliance for Data-Intensive Services. (2017). *Digitization & innovation through cooperation*. Glimpses from the Digitization & Innovation Workshop at "Konferenz Digitale Schweiz". Available March 28, 2018, from https://www.data-service-alliance.ch/blog/blog/digitization-innovation-through-cooperation-glimpses-from-the-digitization-innovation-workshop

Tez, R.-M. (2016). *Rocket AI: 2016's most notorious AI launch and the problem with AI hype*. Blog post. Available May 10, 2018, from https://medium.com/the-mission/rocket-ai-2016s-most-notorious-ai-launch-and-the-problem-with-ai-hype-d7908013f8c9

Tukey, J. W. (1962). The future of data analysis. *The Annals of Mathematical Statistics, 33*(1), 1–67.

Valarezo, U. A., Pérez-Amaral, T., & Gijón, C. (2016). Big data: Witnessing the birth of a new discipline. *Journal of Informatics and Data Mining, 1*(2).

Vorhies, W. (2014). *How many "V's" in big data? The characteristics that define big data.* Available March 23, 2018, from https://www.datasciencecentral.com/profiles/blogs/how-many-v-s-in-big-data-the-characteristics-that-define-big-data

White, A. (2015). *The end of big data – It's all over now.* Available March 23, 2018, from https://blogs.gartner.com/andrew_white/2015/08/20/the-end-of-big-data-its-all-over-now/

Witten, I. H., Moffat, A., & Bell, T. C. (1999). *Managing gigabytes: Compressing and indexing documents and images*. San Francisco, CA: Morgan Kaufmann.

Zierer, K. (2017, December 27). Warum der Fokus auf das digitale Klassenzimmer Unfug ist. *Spiegel Online*. Available May 10, 2018, from http://www.spiegel.de/lebenundlernen/schule/digitales-klassenzimmer-die-schueler-muessen-wieder-in-den-mittelpunkt-a-1181900.html#ref=meinunghpmobi

Chapter 2
Data Science

Martin Braschler, Thilo Stadelmann, and Kurt Stockinger

Abstract Even though it has only entered public perception relatively recently, the term "data science" already means many things to many people. This chapter explores both top-down and bottom-up views on the field, on the basis of which we define data science as "a unique blend of principles and methods from analytics, engineering, entrepreneurship and communication that aim at generating value from the data itself." The chapter then discusses the disciplines that contribute to this "blend," briefly outlining their contributions and giving pointers for readers interested in exploring their backgrounds further.

1 Introduction

"Data science" is a term that has entered public perception and imagination only since the first half of the decade. Even in the expert community, fundamental treatments such as "What Is Data Science?" (Loukides 2010) were first published as recently as 2010. Yet, the substance of what constitutes data science has been built up for much longer. An attempt to define the term "data science" can follow either a top-down or a bottom-up philosophy. On the one hand, looking "top-down," data science is the research field that studies mechanisms and approaches necessary to generate value and insights from data, enabling the building of data products. Importantly, a "data product" is not just a product "dealing" with data, but it is a product deriving its value from the data and producing data itself (Loukides 2010). On the other hand, adopting the "bottom-up" view, data science is an interdisciplinary research field (Stockinger et al. 2015) that adopts a new, holistic way of exploiting data, looking beyond single aspects such as how to store data, or how to access it. It follows that we need to integrate competencies from many older disciplines of study: technical-mathematical disciplines such as "computer science" and "statistics," but also disciplines such as "entrepreneurship" and "art."

M. Braschler (✉) · T. Stadelmann · K. Stockinger
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: bram@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_2

No one view, top-down or bottom-up, is superior. In fact, there was and is considerable disagreement exactly where the boundary of data science is to be drawn (Warden 2011). Rather than engage in this "war of definitions," we think it is helpful to view the different approaches as complementary. For our own work in talking to audiences as diverse as students, colleagues, and business partners, we found the following definition most helpful, and thus adopt it for this book on applied data science (Stadelmann et al. 2013):

> Data science refers to a unique blend of principles and methods from analytics, engineering, entrepreneurship and communication that aims at generating value from the data itself.

What makes this phrasing stand out for us is threefold:

1. The definition distinguishes data science well from preceding paradigms: it is not equal to its individual parts, such as analytics, engineering, etc. (or their sub-disciplines, such as AI, algorithms, or statistics), that is, no single sub-discipline "owns" data science. Nor is it simply the sum of these parts, that is, it does not include any sub-discipline entirely. Instead, it refers to a *unique blend of principles and methods* from them. We arrived at this conclusion through intensive collaboration between computer scientists and statisticians at the ZHAW School of Engineering and are convinced that it holds generally. Unlike e-science and other domain-specific paradigms, data science is universal in applying to all kinds of data and application areas. Finally, unlike in data mining, which concentrates on exploratory data analysis, there is a clear goal in data science to *generate value from data*. Reflecting on the top-down view given above, the data product guides this value generation.
2. The definition connects science to practice: by emphasizing data science as an applied science (encompassing *entrepreneurship*, having the goal of *generating value*), the important aspect of it being "grounded in reality"[1] is highlighted. Again, the applicability of this has been verified many times over in our work in applied research and development over the past years, and distinguishes data science from some of the fundamental work done in the constituting disciplines (e.g., establishing the laws of probability remains a fundamental result in math, not data science).
3. The definition testifies to the breadth and history of the field: the *unique blend of methods and principles* explicitly acknowledges that data science is "standing on the shoulders of giants" instead of "reinventing the wheel." It also acknowledges the fact that it is more than an umbrella term: the blend we refer to is unique, and not just a universal collection, but a tailored selection of relevant methods, principles, and tools from the constituting disciplines.

The remainder of this chapter will trace these "giants" and their contributions to data science.

[1] See also Brodie's later chapter "on developing data science".

Critically, to turn "data science" into more than a label or a hype (see, e.g., Davenport and Patil 2012), and rather into a real discipline on its own right, a re-thinking of the whole process of leveraging data is necessary, from data acquisition all the way to the building of a "data product." Data science is more than an umbrella term precisely because it not only allows to bundle all the individual disciplines from the constituting fields but the term also finally allows a convenient way to express this idea of working at the so far uncovered interfaces of the different subfields. Data science is very much about creating synergies. The remainder of this chapter will highlight clearly that data science is an *applied* and *interdisciplinary* endeavor: the case studies covered in Part II could not be feasible otherwise and would suffer greatly from the lack of a concise, accurate term to describe them.

2 Applied Data Science

When discussing the "added value" of combining traditional academic disciplines such as statistics and computer science, but also economics, the notion of "generating value from data" stands out. Data becomes a product—inherently making data science an applied science. On the other hand, an endeavor becomes *scientific* if it examines a phenomenon by use of the scientific method[2] with the goal to gain knowledge. It becomes *applied science* if the scientific method is applied not just to any phenomenon but to problems that arise in "everyday life" and the solution of which directly improves issues at home, at work, in business, or in the society at large.

The distinction between basic and applied research thus is the origin of the research question—the phenomenon or hypothesis to consider. The majority of research in data science is applied, being directly motivated by use cases; and it can be argued that without this demand from use cases the more fundamental questions (e.g., how to scale the developed methods to all relevant domains[3]) would basically not arise. In turn, use cases provide a means to test hypotheses arising from purely fundamental work—much like test set examples in machine learning help evaluating the generalization capabilities of an established (trained) model.

More use cases than ever await solutions due to more data than ever being available to more actors than at any time in history[4]—but if one believes in the age-old saying "knowledge is power," leveraging that data becomes ever more pressing, lest the competition might glean insight from it first. Many companies discover that they are in fact more data-driven than they may have previously

[2]The scientific method refers to the cycle of theory formation and experimentation introduced by Newton—see https://en.wikipedia.org/wiki/Scientific_method

[3]Refer also to Brodie's later chapter on "what is data science?".

[4]See, e.g., https://insidebigdata.com/2017/02/16/the-exponential-growth-of-data/

perceived, and that, as their respective fields of business transform in the information age, they need to "activate" their data if they want to continue to prosper. This shift had already been seen previously in data-intensive academic fields, such as physics and astronomy, and thus there is much that industry can learn from earlier endeavors.

3 Interdisciplinarity in Data Science

The key to becoming a business player in today's supercharged "online market" is the ability to build the necessary "data products." Successful data science projects often capitalize on the interface between industry and science by relying, on the one hand, on a successful interpretation of the use case and the customer needs, coupled with an attractive, effective design, and on the other hand on building on top of the right, state-of-the art techniques and tools.

It would be a tall order to cover all the many diverse disciplines for such a project in equal depth. In practice, this is not necessary in every undertaking. Ideally, a team of data scientists bundles the required skills, with the individual team members having different profiles—more on the question of what makes a successful data scientist can be found in the following chapter. Importantly, one could argue that a fair amount of fascination for the field of data science derives precisely from this bridging of business and engineering aspects.

Fig. 2.1 A tag cloud compiled from the tags that researchers at the ZHAW Datalab use to describe their research (produced with the generator available at https://www.jasondavies.com/wordcloud/)

Figure 2.1 shows the different academic subfields that data scientists at the ZHAW Datalab[5] use to describe their main lines of research. The figure, a tag cloud or word cloud (Bateman et al. 2008), nicely illustrates the sheer diversity of (sub-)disciplines that contribute to data science. We will in the following sections briefly describe the most important of the contributing academic fields, starting with the technical-mathematical disciplines, and extending to the additional fields that have to be covered to truly produce "data products."

3.1 Computer Science

Computer science, the academic discipline that covers all aspects related to the design ("hardware") and use ("software") of computers (Aho and Ullman 1992), is a frequent first career path for data scientists. This is on the basis of the two subfields of data-processing algorithms (Knuth 1968) and information systems (Ramakrishnan and Gehrke 2002). The former is the study of the way that computers process data: algorithms describe how computers carry out tasks. The latter deals with storage, handling, and processing of large amounts of (digital) data—something that is impossible without the use of computers. Processing and handling data stands at the core of every (digital) computer. Starting with the introduction of the von Neumann architecture in 1945 (von Neumann 1993), even the instructions for the computer are handled equally to the data it processes—both are stored in the same form, namely, in the volatile main memory and on external storage. Everything is thus "data" from a computer science perspective. However, not all aspects of computer science are of equal importance to data science: aspects such as design of hardware, or software engineering, take backseat to those research lines directly addressing data and information:

- The storage of data: here, mainly research on database systems (Silberschatz et al. 1997), that is, the persistent storage of structured data, is relevant. Classically, the "relational model" for databases (Codd 1970) has been the main approach for storing structured data for a long time. However, in the context of big data systems, new exciting developments are also pertinent, such as NoSQL databases (Stonebraker 2010).
- The handling of data: mostly tools-driven. Scripting languages, such as Python (van Rossum and Drake 2003) or Perl (Wall et al. 1999) are often used for "data wrangling," that is, the transformation of data between various formats or representations.
- The processing or accessing of data and information: here, in addition to algorithmic work that we will treat below under the umbrella of artificial intelligence, the most important research subfields are data warehousing (for structured data)

[5] See the preface of this book for more information on the ZHAW Datalab.

(Chaudhuri and Dayal 1997; Inmon 2005) and Information Retrieval (for unstructured data) (Schütze et al. 2008). Data warehousing is mostly concerned with the methods to arrange data for efficient and effective analysis, where information retrieval extends the research on accessing unstructured textual or multimedia data to questions on how to interpret the data to satisfy information needs by the users.

Much of the subfields listed above can be subsumed under the heading "information systems." It should be noted here that the names of these subfields are somewhat plagued by inconsistent use of the terms "data" and "information." Often, they deal in actuality with both aspects—with the "raw" data and with information, that is, the data coupled with an interpretation (Bellinger et al. 2004).

3.2 Statistics

While computer science delivers the tools to store, process, and query the data, which is the "fuel" of data science, statistics is at the core of the academic fields that support the transformation of data into value, for example, in the form of insight or decisions (Wilcox 2009). When consulting common definitions of the field of statistics, some of the same boxes we mentioned for information systems are ticked: statistics deals, much like information systems, with the collection and organization of data. However, the viewpoint is a fundamentally different one: while in information systems, we refer to the storage and processing of data at large in the "mechanical sense," here we have the focus on the selection and organization of data in the mathematical sense. This collection is the precursor to analysis, interpretation, and presentation of data. Statistics provides tools to describe properties of data sets ["descriptive statistics" (Holcomb 1997)] as well as drawing conclusions from data sets that are a sample of a larger population ["inferential statistics" (Wasserman 2013)]. Crucially, statistics provides the tools (tests) to verify hypotheses as to the relationship between variables and data sets and provides a different angle to work done in computer science on machine learning.

3.3 Artificial Intelligence

Artificial intelligence (AI) (Russell and Norvig 2010), and especially its branch machine learning, is typically treated as a subfield of computer science, and sits nicely at the intersection of computer science, statistics, and several other disciplines. AI generally studies solutions to complex problems arising in areas such as human perception, communication, planning, and action (Luger 2008). Most relevant for data science, but not uniquely so, is the branch of machine learning that studies algorithms that can "learn" from data, based on pattern recognition in data

sets (Bishop 2007). There is potential in increasingly combining this with logic-based AI that reasons over ontologies.[6] Ideally, the learning improves performance on a given task without the need for a human to program an explicit solution (Samuel 1959). This is both attractive in cases where such an explicit solution is very complex or if the task deals with constantly changing parameters.

Supervised (machine) learning is based on providing the learning algorithm pairs of possible inputs and their corresponding outputs. Based on this "training data," a desired mapping function or algorithm is learned. A key problem in this context is potential overfitting, that is, if the learning process picks up undesired artifacts present in the training data that are not representative of the larger population of possible inputs. More fundamentally, suitable training data may be difficult and costly to obtain. In unsupervised (machine) learning, the aim is to find hidden structure in data sets without the use of training labels.

3.4 Data Mining

Another subfield straddling the boundaries of computer science and statistics is data mining. The term is used in somewhat different ways, with many different forms of information processing being labelled as data mining (Witten et al. 2016). Generally, it applies principles from machine learning, statistics, and data visualization, where the goal is the detection of previously unknown patterns in data.[7] The differentiation to data science lies in the focus on the extracted patterns themselves, whereas data science covers a broader range of topics already beginning at data collection with the explicit goal of a data product in the end. Data mining can thus be thought of as a predecessor paradigm to interdisciplinary work on data by applying fundamental results from, for example, the analytics fields of statistics or machine learning.

3.5 Additional Technical Disciplines

There are multiple additional technical disciplines that contribute to the "umbrella field" of data science. We want to make a special note of some of these: on the one hand, there is business intelligence (Chen et al. 2012), which stands at the interface between technical aspects and management and aims to leverage the data to provide a view on business operations. On the other hand, there are several disciplines that

[6]Pointed out, e.g., by Emmanuel Mogenet from Google, at the Zurich Machine Learning & Data Science Meetup in February 2017, talking about combining subsymbolic (machine learning) approaches with symbolic (logic-based) AI.

[7]Rather than detection of the data itself. One could thus argue that the term is unfortunate, and an alternative along the lines of "mining on data" would be more appropriate.

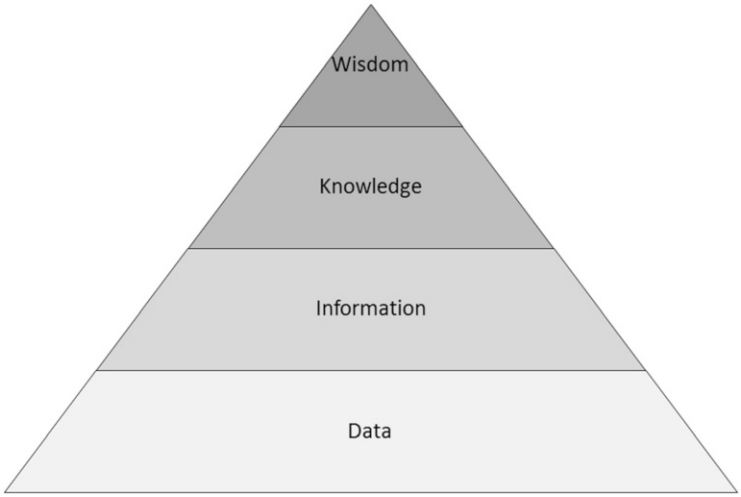

Fig. 2.2 The knowledge pyramid

dissolved from AI in the last decades and now form their own communities, usually centered around separate types of data: this includes speech as well as natural language processing, which deals with processing natural (human) language by computer (Manning and Schütze 1999; Deng and Yu 2014); computer vision, which deals with images and video[8]; or pattern recognition, which deals with the problem of automatizing human perception tasks (Duda et al. 2001).

3.6 The "Knowledge Discovery in Databases (KDD)" Process

An alternative viewpoint of data science can be taken by referencing the "knowledge discovery in databases (KDD)" process (Fayyad et al. 1996). Data typically sits at the bottom of a "knowledge pyramid," illustrated in Fig. 2.2 (Frické 2009). Simply put, data can be viewed as a collection of codes that can be stored, organized, and accessed. Only if a (contextual) meaning is affixed to data does it become information. When information is then analyzed and interpreted, and new inferences are drawn, the result is knowledge. This is, as we have previously pointed out, the goal of data science, insofar as value is created through new knowledge.

On a conceptual level, the KDD process models this progression from data to knowledge through a series of stages[9]:

[8]Since 2012, the field of computer vision has been largely changed by recent research on deep neural networks, see, e.g., Goodfellow et al. (2016) and LeCun (2013).

[9]Compare also the cross-industry standard process for data mining (Shearer 2000).

1. Data is first selected (measured, recorded, then initially stored)
2. Then it is pre-processed (noise removal, filling in of missing values, outlier detection, etc.);
3. It is transformed into a form suitable for analysis (often a 2D table format, after including feature selection or transformations like Fourier transform)
4. Then it is analyzed (by statistical or machine learning methods to find patterns of correlation).
5. Finally, the result of potentially multiple analyses is interpreted/evaluated by a human (or human-devised decision mechanism).

These five stages again tie in nicely with the technical disciplines discussed so far. An alternative rendering of the KDD process, that puts a slightly different emphasis on the different stages, aligning it more with the disciplines, could thus read:

1. Data recording
2. Data wrangling (including data cleaning and storage, i.e., in databases or information retrieval systems)
3. Data analysis (including statistics, artificial intelligence, and data mining)
4. Data visualization and/or interpretation
5. Decision making

Data science as an interdisciplinary academic field goes far beyond only technical-mathematical aspects as covered by disciplines such as computer science or statistics. We like the quote by Hilary Mason, who concisely described data science as "statistics, computer science, domain expertise, and what I usually call 'hacking'" (Woods 2012). The chapter has not covered the "domain expertise" bit so far, but such expertise is crucial for understanding the unique value proposition of treating data science as a unified pursuit of leveraging data. "Domain expertise" both addresses the need for knowledge of the different domains that the data originates from (legal domain, medical domain, etc.), but also more generally the possession of non-technical skills such as arts and entrepreneurship. The KDD process as outlined above culminates in data visualization/interpretation and decision making, which both heavily rely on non-technical expertise as outlined below.

3.7 Data or Information Visualization

At the interface between computer science (computer graphics, see, e.g., Hughes et al. 2013) and arts (see below) lies data or information visualization (Ware 2012). In both cases, the goal is a rendering of the data to support better human understanding. When choosing the term "data visualization," more weight is given to the aspect of raw data visualization (Tufte 2001), whereas "information visualization" stresses more on the aspect of supporting interpretation of the data through

visualization.[10] Often the terms are used (rather confusingly) interchangeably. In both cases, the visualization is a communication tool: large amounts of data are either compressed visually into a rendering that can provide an overview or are rendered to be explored interactively, with the user zooming in and out of the data to discover the needed information. The rendering of the results of data analysis is crucial to feed the KDD stages of interpretation or decision making—data has to be rendered in such a way that the desired information or knowledge becomes prominent.

3.8 Arts

The renderings of data or information visualization often combine usefulness with attractive presentation—giving the resulting graphics a new, artistic dimension. The term "new digital realism" is used—data being the medium to visualize what exists "but cannot be seen" (Sey 2015). The analogy to "realism" in classical arts, such as painting, implies that reality is rendered "as it is"—without emotional interpretation or subjective frames. This is of course a tall order, insofar as any visualization consciously puts certain aspects of the data in the forefront, and thus influences subsequent interpretation. In its most pure form, art in the context of data science may pursue the finding of "beauty,"[11] not value, in data.

3.9 Communication

A totally different, yet important artistic aspect of data science lies in the general challenge of the communication of results. Data products—any findings, any value in the data—rely on interpretations, and this discipline and their proponents need the skills to effectively and truthfully communicate well to stakeholders. We will explore this angle of "communication as a skill" more in the next chapter, where we discuss the profile of a successful data scientist.

3.10 Entrepreneurship

Our definition of data science puts the data product in the center: the data product leverages data to produce value and insight (see Chap. 4). The design of a data product is much more than a technical exercise, or even an exercise in leveraging

[10] See also the emerging genre of data journalism: https://en.wikipedia.org/wiki/Data_journalism

[11] For some examples, see Pickover (2001).

domain expertise in the narrow sense. Questions of how to frame the value proposition of the product, how to identify the right audience, and how to find the matching business model arise. These directly address the business-savvy of the data scientist. Generally speaking, a successful data scientist does well to display a degree of entrepreneurial spirit: opportunities to seize value have to be anticipated, based on a deep enough understanding of all three of the following aspects: the technical possibilities, the potential in the data itself, and the need of some "customer."

4 Value Creation in Data Science

The discussion of entrepreneurship brings the overall exposition on data science as a new field of academic study full circle. Whether data science is perceived as an amalgamation of different disciplines, essentially harvesting the synergies of combining technical foundations of computer science with analysis insight from statistics and extending these skills to domain expertise and business-savvy, or whether data science is more seen as a holistic approach to leverage the value of data: the results of applied data science projects typically culminate in data products, that is, products that derive their value from the data they are built on. Data products often come in the form of data-driven services. Consequently, the discipline of service science (Spohrer 2009), which is concerned with, among other things, service innovation, and sits at the intersection of business and information technology, also contributes to the development of data products (see more details in Chap. 4).

5 Conclusions

In closing the chapter, we want to reflect again on the idea of data science both as a field of study in its own right and as an umbrella term that allows to describe interdisciplinary endeavors at the interfaces of the disciplines covered above. As stated in the introduction, these two views can be interpreted as "top-down" and "bottom-up." Both views are complementary and enhance the insight into the nature of data science. By covering the various (sub-)disciplines, the bottom-up view of data science as a "unique blend" of the disciplines is represented well. As regards the top-down view, the essence of data science undertakings is the creation of value from data (or information). That thought is not necessarily new. If we look at older fields of study that nowadays contribute to data science projects, such as information retrieval, then similar themes emerge: by making access to relevant information possible, Information Retrieval "turns information useful." Is data science therefore really more than the sum of its parts? Or could this creation of value take place under different, potentially even more specific, labels in all cases? Or, put differently, if we

start with the concept of data science and then remove all the (sub-)disciplines in this chapter, will we be left with something meaningful?

The last of these questions borders on the philosophical, and the answer is probably influenced by where we draw the boundaries of the disciplines. But it may be precisely at the interfaces of the disciplines, or even in the gaps between them, that data science is an enabler for new concepts. The rewards for venturing into these spaces between disciplines and finding new, exciting combinations may be greater than ever. The case studies in the chapters of Part II are a nice testament of the diversity of research questions or business cases that can be pursued.

References

Aho, A. V., & Ullman, J. D. (1992). *Foundations of computer science*. New York: Computer Science Press.

Bateman, S., Gutwin, C., & Nacenta, M. (2008). Seeing things in the clouds: The effect of visual features on tag cloud selections. In *Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia* (pp. 193–202). Pittsburgh: ACM.

Bellinger, G., Castro, D., & Mills, A. (2004). *Data, information, knowledge, and wisdom*. http://www.Systems-thinking.org/dikw/dikw.htm

Bishop, C. M. (2007). *Pattern recognition and machine learning*. New York: Springer.

Chaudhuri, S., & Dayal, U. (1997). An overview of data warehousing and OLAP technology. *ACM SIGMOD Record, 26*(1), 65–74.

Chen, H., Chiang, R. H., & Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS Quarterly, 36*, 1165–1188.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM, 13*(6), 377–387.

Davenport, T. H., & Patil, D. (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.

Deng, L., & Yu, D. (2014). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing, 7*(3–4), 197–387.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification* (2nd ed.). Wiley.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine, 17*(3), 37.

Frické, M. (2009). The knowledge pyramid: A critique of the DIKW hierarchy. *Journal of Information Science, 35*(2), 131–142.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning*. Cambridge: MIT Press.

Holcomb, Z. C. (1997). *Fundamentals of descriptive statistics*. London: Routledge.

Hughes, J. F., Van Dam, A., Foley, J. D., McGuire, M., Feiner, S. K., Sklar, D. F., & Akeley, K. (2013). *Computer graphics: Principles and practice* (3rd ed.). Boston: Addison Wesley Professional.

Inmon, W. H. (2005). *Building the data warehouse*. Indianapolis: Wiley.

Knuth, D. E. (1968). *The art of computer programming: Fundamental algorithms*. Reading: Addison-Wesley.

LeCun, Y. (2013). *Hi Serge*. Google+ post. Available May 23, 2018, from https://plus.google.com/+YannLeCunPhD/posts/gurGyczzsJ7

Loukides, M. (2010). *What is data science*. Available June 12, 2018, from https://www.oreilly.com/ideas/what-is-data-science

Luger, G. F. (2008). *Artificial intelligence: Structures and strategies for complex problem solving* (6th ed.). Boston: Pearson.
Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. Cambridge: MIT Press.
Ramakrishnan, R., & Gehrke, J. (2002). *Database management systems* (3rd ed.). New York: McGraw Hill.
Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Upper Saddle River, NJ: Pearson Education.
Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development, 3*(3), 210–229.
Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge: Cambridge University Press.
Sey, M. (2015). *Data visualization design and the art of depicting reality*. https://www.moma.org/explore/inside_out/2015/12/10/data-visualization-design-and-the-art-of-depicting-reality/
Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing, 5*(4), 13–22.
Silberschatz, A., Korth, H. F., & Sudarshan, S. (1997). *Database system concepts* (Vol. 4). New York: McGraw-Hill.
Spohrer, J. (2009). Editorial column—Welcome to our declaration of interdependence. *Service Science, 1*(1), i–ii. https://doi.org/10.1287/serv.1.1.i.
Stadelmann, T., Stockinger, K., Braschler, M., Cieliebak, M., Baudinot, G., Dürr, O., & Ruckstuhl, A. (2013, August). *Applied data science in Europe: Challenges for academia in keeping up with a highly demanded topic*. European Computer Science Summit, ECSS 2013, Informatics Europe, Amsterdam.
Stockinger, K., Stadelmann, T., Ruckstuhl, A. (2015). Data Scientist als Beruf. Big Data – Grundlagen, Systeme und Nutzungspotenziale (Edition HMD, 59–81). Berlin: Springer.
Stonebraker, M. (2010). SQL databases v. NoSQL databases. *CACM, 53*(4), 2010.
Tufte, E. R. (2001). *The visual display of quantitative information* (2nd ed.). Cheshire, CT: Graphics Press.
Van Rossum, G., & Drake, F. L. (2003). *An introduction to python*. Bristol: Network Theory.
Von Neumann, J. (1993). First draft of a report on the EDVAC. *IEEE Annals of the History of Computing, 15*(4), 27–75.
Wall, L., Christiansen, T., & Schwartz, R. L. (1999). *Programming perl*. Sebastopol, CA: O'Reilly & Associates.
Warden, P. (2011). http://radar.oreilly.com/2011/05/data-science-terminology.html
Ware, C. (2012). *Information visualization: Perception for design*. San Francisco: Elsevier.
Wasserman, L. (2013). *All of statistics: A concise course in statistical inference*. Berlin: Springer Science & Business Media.
Wilcox, R. R. (2009). *Basic statistics: Understanding conventional methods and modern insights*. Oxford: Oxford University Press on Demand.
Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data mining: Practical machine learning tools and techniques*. San Francisco: Morgan Kaufmann.
Woods, D. (2012). Bitly's Hilary Mason on "what is a data scientist?" *Forbes Magazine*. https://www.forbes.com/sites/danwoods/2012/03/08/hilary-mason-what-is-a-data-scientist/

Chapter 3
Data Scientists

Thilo Stadelmann, Kurt Stockinger, Gundula Heinatz Bürki, and Martin Braschler

Abstract What is a data scientist? How can you become one? How can you form a team of data scientists that fits your organization? In this chapter, we trace the skillset of a successful data scientist and define the necessary competencies. We give a disambiguation to other historically or contemporary definitions of the term and show how a career as a data scientist might get started. Finally, we will answer the third question, that is, how to build analytics teams within a data-driven organization.

1 Introduction

Reading contemporary press, one can come under the impression that data scientists are a rare (Harris and Eitel-Porter 2015), almost mythical, species,[1] able to save companies by means of wonderworking (Sicular 2012) if only to be found (Columbus 2017). This chapter answers three questions: What is a data scientist? How to become a data scientist? And, how to build teams of data scientists? (see also Stockinger et al. 2016). Answering these questions will help companies to have realistic expectations toward their data scientists, will help aspiring data scientists to plan for a robust career, and will help leaders to embed their data scientists well.

What is a data scientist? As of spring 2018 the ZHAW Datalab, that is, the data science research institute (DSRI) of the Zurich University of Applied Sciences, has more than 70 affiliated colleagues that "professionally work on or with data on a daily basis."[2] The lab includes different kinds of researchers, ranging from computer

[1] The British recruiting firm Sumner & Scott was looking for a "*Seriously Fabulous Data Scientist*" on behalf of a customer from the game industry in April 2018.

[2] See http://www.zhaw.ch/datalab for a description of the lab, its statutes, and a list of associates.

T. Stadelmann (✉) · K. Stockinger · M. Braschler
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stdm@zhaw.ch

G. Heinatz Bürki
Swiss Alliance for Data-Intensive Services, Thun, Switzerland

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_3

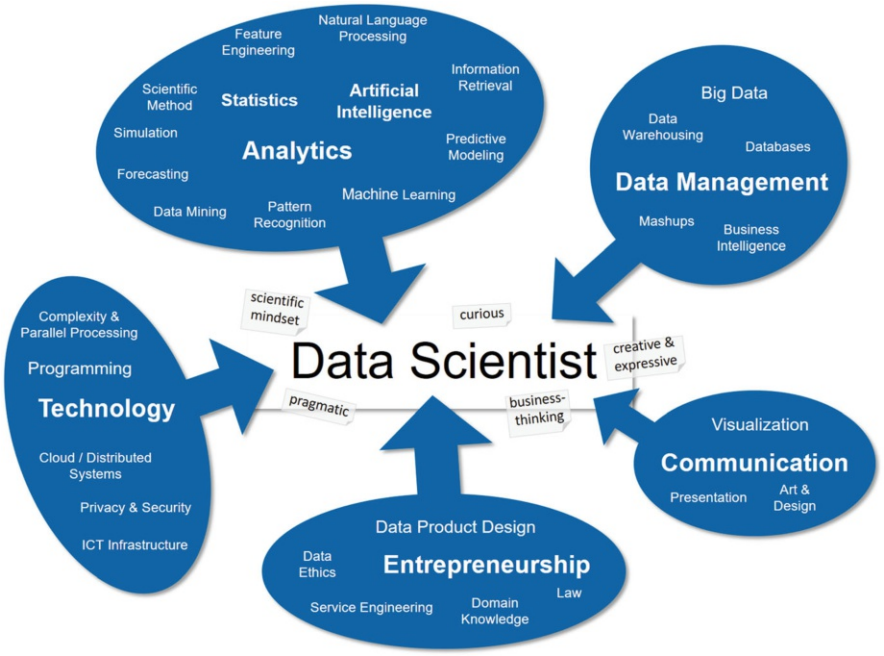

Fig. 3.1 The definition of a data scientist by means of personal qualities (gray labels) and skills (in blue bubbles) as spanned up by this unique cut from several scientific (sub-)disciplines. Revised and extended from Stadelmann et al. (2013)

scientists doing analyses with machine learning to domain experts in medical imaging or quantitative finance, to lawyers working on data protection law. Can these colleagues be considered as data scientists? From what we know, not all of these colleagues call themselves primarily data scientists.

However, what is a data scientist? Going beyond the trivial definition of data scientists being those who conduct data science, we can approach the definition by sketching the set of skills and qualities of a data scientist as two layers. Figure 3.1 contains these two layers of information: First, the blue bubbles show the contributions of several competence clusters to the skill set of the data scientist. Second, the gray labels attached to the data scientist in the center show important qualities of the personality that are paramount for a data scientist's professional success. While the academic (sub-)disciplines underlying these competence clusters were treated in the previous chapter, we will explore their relations to the work of a data scientist in conjunction with the character traits in the next section. In Sect. 3, we will disambiguate the definition of a data scientist from historical and contemporary alternative meanings. In Sect. 4, we will show career paths toward data science and finally discuss how to build effective data science teams in Sect. 5.

2 The Data Scientist's Set of Skills and Qualities

Data scientists are T-shaped people (Guest 1991). They have broad interdisciplinary skills (the crossbar of the "T") and at the same time they have deep expertise (the T's stem) in a much narrower area of this skill set. This section will look at the crossbar and related soft skills, while Sect. 4 will look at the origins of the stem.

The blue areas in Fig. 3.1 show competence clusters within the data scientist's set of skills. The appearing terms have been selected due to their high likelihood of being important in the daily work on almost any project of a data scientist, in the following sense: in any project, some skills from any bubble will likely be needed, that is, some method(s) from the "Analytics" cluster but not all of them. We make no claim as regards to the completeness of this term set. Let us have a look at the individual clusters in more detail.

Technology and Data Management Handling data well is crucial enough for any data scientist to make it a top-level competence cluster. Data management includes, but is not limited to, big data technologies, databases and respective query languages like SQL. A background in extract-transform-load processes for data integration and the fundamentals of relational databases are relevant for many data science projects. The technology cluster includes various other areas from computer science such as the application and handling of software systems. Programming skills are paramount for a data scientist, however, not in the sense of large-scale software development, but in the form of scripting, for example, for data wrangling tasks. Combining small scripts in the spirit of the UNIX command line with each other (Peek et al. 1993) allows for rapid prototyping as well as repeatability of experiments.[3] It possibly also helps for executing analyses in different, even distributed environments.

Analytics Skills in analytics, especially machine learning, are one of the core competencies of a data scientist to extract knowledge from data. The two main approaches to analytical methods come from the fields of statistics (Wasserman 2013) and artificial intelligence (Russell and Norvig 2010); the two fields often provide different individual approaches to similar methods. While discussions arising from these differences in viewpoint are challenging for any practitioner in a data science team, they are also a source for mutual interdisciplinary understanding, and hence are very valuable. This has been analyzed thoroughly by Breiman (2001).

Entrepreneurship Data scientists are not only responsible for the implementation of an analytical solution for a given problem. Rather, they additionally need entrepreneurial skills to ask the right questions with respect to business cases, business models, and the consequences of the data products on the business and society at

[3]Experiments that are controlled purely by scripts are repeatable by archiving the code as well as data together with the results. They can be developed rapidly by re-using scripts from other projects (which is easier when every script serves exactly one purpose and uses a simple file-based API, as UNIX shell programs do) and automatizing parameter search.

large. This includes building up subject matter expertise in the application areas of the data product at hand, and the appreciation of the personal ethical responsibility. As many questions in data science touch on fundamental issues in privacy, data scientists must have knowledge of legal frameworks of their operating environments.

Communication Being responsible for the complete analytical workflow, data scientists personally communicate their results to (senior) management. Needed skills thus range from targeted presentation to information visualization in order to convey complex matters and conclusions in concise ways. It is questionable[4] if the creation of (graphical user interfaces for) web services for the final customers of data products should be a core part of the data science skill set.

The second layer of information in Fig. 3.1 shows personality trait labels attached to the data scientist. Being more part of a person's character than the skill set, it seems a bit unfair to require them for a job as widespread as a data scientist. On the other hand, it is a matter of fact that certain jobs fit specific people (Fux 2005). So what is the impact of these qualities on a practitioner's work?

Creativity and Expressiveness Both traits help in giving convincing presentations of the data scientist's work for internal stakeholders and potential customers. Creativity reaches even farther in also being a necessity for creating novel results. This plays into the next point.

Curiosity and Scientific Mindset Curiosity pairs well with enthusiasm. A scientific mindset will balance utter positivism with basing one's hopes and findings on facts through critical hypothesizing, thorough experimentation,[5] and precise formulation of results. Doubt and amazement are both important ingredients for novel solutions.

Business Thinking Thinking economically helps to have a clear goal in mind, on several levels: it contributes to not losing sight of the complete development process of a data product when concentrating, for example, on the analytical challenges at hand; it also helps in allocating resources to the various steps in a project and weigh options in order to produce business-relevant results. This will ultimately drive the success of analytics endeavors, since most stakeholders (in businesses, research, or society) will be convinced not by the coolness of the engineering, but by actually helpful results delivered on time and on budget.

[4]We see this better placed in the hands of software engineers; nevertheless, being able to build rapid prototypes in the way presented in the "Developing Data Products" course by the Johns Hopkins University (see https://www.coursera.org/learn/data-products) is an interesting additional skill for any practicing data scientist.

[5]The scientific method of theory formation (thinking) and collecting empirical evidence (experimenting) in a closed loop is directly applicable in the daily work of a data scientist. See a longer exposition of this thought in Stadelmann (2017) and an extension in a later chapter by Brodie ("On developing data science").

Pragmatism The quality to do rapid prototyping and quick experimentation cannot be underestimated. The analytical work of a data scientist is inherently empirical, and having a drive toward experimenting and getting one's hands dirty with code and messy data is paramount in making progress in many projects. A special sort of pragmatism with respect to coding—specifically, to be able to abstain from undue perfectionism in software engineering in early project phases in favor of "hacking"[6]—and system design (specifically, to use simple scripts in a command-line like fashion) helps in keeping efficiency high in usually very complex tool landscapes.

3 Disambiguation

The following paragraphs deal with disambiguating the definition of a data scientist as presented above from other meanings used previously or contemporary.

3.1 The History of a Job Description

Probably the first one publicly speaking about data scientists was Jeff Wu (1997), who suggested to use the term as a replacement for "statistician." The modern use presented in the previous section arguably emerged out of discussions between DJ Patil and Jeff Hammerbacher on how to call their team members at Facebook and LinkedIn, respectively, that were closer to product development than a usual "research scientist," but with more technical skills than a typical "business analyst" (Patil 2011). Previously, this profile had been called "deep analytical talent" in a noteworthy report from the McKinsey Global Institute (Manyika et al. 2011) and was famously rendered in graphic form by Drew Conway (2010) in his "data science Venn diagram." The diagram conveyed the notion that a data scientist works at the intersection of hacking, math (or statistics), and substantive expertise, thereby discriminating it from traditional research (no hacking), pure machine learning (no subject matter expertise), and a "danger zone" (no math). Patil and Hammerbacher added that their data scientists should also be able to communicate their own results to business owners on top of the deep engineering know-how embraced also by Conway.

The following years saw a race to ever more elaborate versions of the skill set of a data scientist, packed into Venn-like diagrams (Taylor 2014). In very short time,

[6]Take this with a grain of salt: while we plead for hacking to facilitate rapid prototyping, especially for our audience with a background in computer science, we certainly know about the importance of careful software engineering for production-ready code. See also Zinkevich (2018) for good advice for the latter (and the former).

deep analytical talent was inflated to unicorns (Press 2015), marketed toward C-level executives as the ones finally being able to *"align IT and business"* (Jones 2014). Additionally, expectations toward technical skills grew as well. For example, Brodie (2015a) pointed out the importance of data curation that involves the work on and with large IT systems at scale in preparation of the actual data analysis.[7]

Moreover, data scientists were supposed to carry huge responsibilities due to the disruptive potential of the paradigm of data-driven decision-making (Needham 2013). This raised the requirement on them to make the attached risks of their work explicit, for example, by attaching common measures of correctness, completeness, and applicability to data science results (Brodie 2015a) such as confidence intervals for all quantitative results.[8] The necessity for some measures to this effect becomes apparent when regarding analysis results from higher-dimensional data: in dimensions beyond three, human intuition even of experts fails completely, known under the term "curse of dimensionality" (Bellman 1961). Accordingly, the audience in a presentation of respective results could be easily and unintentionally mislead (Colclough 2017),[9] drawing fatal business decisions from misinterpretations of analytical results.

However, an informal survey amongst the ca. 190 participants of the SDS|2015 conference[10] revealed that only about 50% of the practicing data scientists apply any counter-measures against misinterpretation or illusory certainty—also because this is not required of them by their customers (internally or externally). However, a data scientist is a scientist: this means following sound scientific practice to not let one's own biases and presuppositions overrule experimentally established facts (Brodie 2015b).

3.2 *Insightful Debates*

Two additional debates provide insight on what can or cannot be expected from a modern data scientist:

First, the trend in the mid-2010s to make data science results and careers more easily accessible for a larger number of people (and customers) who might not have formal education in computer science, math, or statistics. As a side effect, the profile

[7]Such systems are used to find, gather, integrate, and manage potentially heterogeneous data sources. This adds up to about 80% of the daily work of a practicing data scientist (Brodie 2015a).

[8]However, the debate in the *Journal of Basic and Applied Social Psychology* on the removal of p-values from all published articles because of a theoretical invalidity of the null hypothesis significance testing procedure (Trafimow and Marks 2015) shows: reporting confidence intervals per se is no panacea as it "suffers from an inverse inference problem" as well.

[9]Colclough (2017) notes that just putting a data visualization on a slide often brings credibility to its statement, no matter the content of the visualization nor its correctness.

[10]See https://www.zhaw.ch/en/research/inter-school-cooperation/datalab-the-zhaw-data-science-laboratory/sds2015/

of the profession might dilute as the work of a data scientist is reduced to the operation of self-service Business Intelligence (BI) tools. On the other side of the same medal, complex and scientifically unsolved problems like social media monitoring (Cieliebak et al. 2014) are promised to get solved at the push of a button. While natural language processing has certainly made progress to the point of applicability in many cases, it is not solved in general—and which business owner can distinguish his very special use case that requires a great deal of generality from the superficially similar demonstration that works in a quite constrained environment?[11]

Seen in relationship with the above-mentioned responsibility of a data scientist for potential good or harm *at scale*, this development might be considered dangerous. It needs certain skills to draw correct conclusions from data analytics results; it is thus important to keep the science as an important ingredient in the data scientist. Business analytics is a part and not a superset of data science; vice versa, not all data science challenges could and should be approached using readymade BI tool boxes or BI engineers.[12] This leads over to a second debate:

Second, there is a notion of data scientists "type A" and "type B."[13] While "type A" are basically trained statisticians that have broadened their field ("data science for people"), "type B" have their roots in programming and contribute stronger to code and systems in the backend (B for "build," "data science for software"). So, two of the main influences for data science as an interdisciplinary field—computer science and statistics—are taken apart again to emphasize a less interdisciplinary profile.[14]

Seen from the viewpoint of interdisciplinarity, which is a key concept for the role of a data scientist, this (and similar) distinctions between mono-disciplinary rooted types of data scientists are useless. The whole point of interdisciplinarity, and by extension of the data scientists, is for proponents to think outside the box of their original disciplines (which might be statistics, computer science, physics, economics, or something completely different) and acquire skills in the neighboring disciplines in order to tackle problems outside of intellectual silos. Encouraging practitioners to stay in their silos, as suggested by the A/B typology, is counterproductive, as it is able to quench this spirit of out-of-the-box thinking.

The debate, however, is well suited in that it challenges the infamous—and often unrealistic—"unicorn" description of a data scientist who is supposed to be an "expert in each and everything." A concept that addresses the same concern but

[11] See http://xkcd.com/1425/. While the described phenomenon might be easy to solve in the year 2018, a contemporary example would be chatbots.

[12] The same applies to automated machine learning, although such systems have a real value for certain applications.

[13] Jaokar (2016) refers to a quora post by Michael Hochster for the origins: https://www.quora.com/What-is-data-science

[14] T-shaped people will have their roots—their depth—mostly in one field; hence, the problem described here arises not from different expressions of the T-shape per se, but from specifically differentiating what lead to the notion of a data scientist in the first place: combining computer science and statistical know-how (see Sect. 3.1).

arrives at different conclusions is the one of data scientists "Type I" and "Type II."[15] "Type II" data scientists are managers, concerned with hiring and leading data practitioners and having a more high-level view of data sciences' potentials and workings. On the other hand, "Type I" data scientists know how to "do the stuff" technically. This opens up the way to combined curricula for manager-type people and technically oriented people (different roles) while not compromising the interdisciplinary profile of either of them.

On the other hand, the attempt to isolate sub-types of a "Type I" comes down to merely re-labeling traditional job titles like statistician, business analyst, BI specialist, data miner, database engineer, software engineer, et cetera. If these titles fit the role, that is, accurately describe the skill set and breadth of the job description, they are still appropriate and very precise. If the job, however, requires the broader experience of a data scientist—the crossbar of the T instead of just the stem—this could be indicated using the proper description of data scientist. Problems arise if an expected but missing crossbar experience leads to weakening the credibility of the discipline of data science.

4 Starting a Data Science Career

If data scientists are interdisciplinary by nature with T-shaped skill profiles, trying to define what a data scientist is comes down to giving bounds on the width of that T's crossbar (how much interdisciplinary experience is necessary?) and the height of the T's pole (to what degree is this a specialist in some subset of skills?). The following bounds are subjective (as being based on personal experience), but can serve in giving guidelines as to what to expect from a senior data scientist, with our definition of "coverage" following in the next but one paragraph.

As for the crossbeam of the "T", a senior data scientist should cover a majority—we guesstimate ca. 80%—of the terms on the competency map in Fig. 3.1, distributed over all five of the blue competence clusters. This usually means that individual senior data scientists should be firmly anchored in one of these clusters and have a solid understanding in at least two others without avoiding the balancing act between the technical-analytical and entrepreneurial-communicative hemispheres. Thus, the necessary interdisciplinary breadth of knowledge is ensured without calling upon mythical beasts.

The intended "covering" means that the data scientist should have an understanding of the respective terms (e.g., "Natural Language Processing" within the "Analytics" cluster, or "Law" within "Entrepreneurship") deep enough to recognize certain opportunities or potential issues arising in the current project from this

[15]The naming itself is not important. The concept has been incorporated into the academic MSc Data Science program of the University of Sheffield (see https://www.sheffield.ac.uk/postgraduate/taught/courses/sscience/is/data-science) and seems to go back to Tierney (2013).

domain, and in case of doubt they can involve an expert. This level of understanding is usually gained by the equivalent of working hands on with the topic for a limited time or doing a typical one semester introductory course, that is, it is not expert-level knowledge. The necessary skills can be trained, given a disposition to quantitative, complex, and technical matters.

Regarding the stem of the "T", a typical career starts with undergraduate studies, for example, in statistics, computer science, or another data-intensive science. From there, interdisciplinary skills can be built either by a data science master's degree, hands-on collaborations with other disciplines, or continuing education. If personal interests suggest a closer look into research, a PhD is a good option, but not all education has to be formal.[16] In our experience, it is more important to show a good track record of projects one was engaged with in order to qualify for advertised data scientist positions. Projects in this regard is a loose term—included are also personal projects or those that are part of course work. What counts is the demonstration of gained experience, for example, by cultivated personal GitHub pages or blogs, published research articles, or by contributions to publicly available products. Certificates themselves are not sufficient due to them becoming more and more omnipresent among candidates.

A data science curriculum—whether offered by any institution in the higher education sector, or self-composed—should address the following three levels (measured in terms of distance to actual cases studies that could be solved). The content of the *business* layer is close to the case study that needs to be grasped in detail by the data scientist. This influences the choice of *algorithms* in the next layer but is more or less independent from the technical *infrastructure* in layer 3.

1. *Business*:
   (a) Visualization and communication of results
   (b) Privacy, security, and ethics
   (c) Entrepreneurship and data product design
2. *Algorithms*:
   (a) Data mining and statistics
   (b) Machine learning
   (c) Information Retrieval and natural language processing
   (d) Business intelligence and visual analytics
3. *Infrastructure*:
   (a) Databases, data warehouses, and information systems
   (b) Cloud computing and big data technology

[16]Especially in the context of data and computer science, online courses like the ones offered by Coursera (https://www.coursera.org/specializations/jhu-data-science) or Udacity (https://www.udacity.com/nanodegree) have a good credibility.

Ideally, such a curriculum considers this intimate connection between the application of data science in actual cases studies, on the one hand, and the fundamentals of data science like details of methods, on the other hand, already in the coursework. This can be achieved by connecting the relevant theory with project work in certain problem domains, for example, predictive maintenance (industry), medical imaging (health), algorithmic trading (finance), churn prediction (marketing), speech processing (technology), building control (energy), etc. These case studies run cross to all three layers from above.

We see the analytical aspects as central to any data science education: machine learning, statistics, and the underlying theories have to be solidly mastered by any data scientist in order to decide on feasibility and perform impact assessment. These skills—the "deep analytical talent" or "deep engineering know-how" as it has been called by various thought leaders—are the ones most deeply learned early on in one's vocational career (i.e., better studied thoroughly for years than acquired using a crash course). They are also the ones that host the greatest potential both in terms of risks and opportunities when unleashed on the world. Data scientists thus owe a responsible mastership of the engineering aspects to their environment.

5 Building Data Science Teams

Finding a senior—mature, "complete"—data scientist as described in the previous section might be difficult for an employer. Even if it was not so, it is advisable to let data scientists work in teams where the crossbows of the respective team members' T's overlap considerably, but the poles dig into different territory of the skill set map (Olavsrud 2015). This way, not only can the less wide crossbars of less senior data scientists be integrated, but the full potential of interdisciplinarity can be leveraged. How should such teams be embedded into the organization?

Executive-level support for establishing data and analytics as a strategic capability is one of the key success factors for enabling a company to do data-driven, automated decision-making. We will look at the following two main aspects of building data science teams[17]:

1. Shape an adequate operational model for the organization's advanced analytics capabilities and associated governance.
2. Identify data-driven use cases that have a big impact on the business and therefore the most added value.

[17]Other aspects are highlighted, e.g., by Stanley and Tunkelang (2016).

5.1 Operational Models for Advanced Analytics

Three main operational models exist for building a common data-driven culture in an organization (Hernandez et al. 2013). For an organization to decide for a specific one, this operational model has to align with the enterprise strategy first. Second, the complexity and maturity of the enterprise regarding data-driven decision-making is relevant. The choice of model thus depends on the organization's structure, size, and experience in this topic. The three models are as follows:

(a) *Centralized unit within the IT or finance department*
The structure of such an organization is simple and focuses on allocating limited resources to strategic topics. The typical enterprise choosing this model already has mature reporting and analytics capabilities, with both the IT and the finance department having already acquired the necessary skills. The centralized unit thus has the technical prerequisites of the IT or statistical knowledge of the finance department because of this previous work and provides the expertise to the business units. This model fits well to most small and simple organizations.

(b) *Cross-business unit with data scientists*
Again, experienced data scientists belong to a centralized group, where they are responsible for analytical models, standards, and best practices. But these data scientists establish contacts to domain experts or even other analytical groups in the business units, as all business units have mature basic analytical skills. This model can be seen as a "hub and spokes" approach compared to the purely central model (a). It fits to moderately more complex organizations that see data as a core competitive advantage.

(c) *Decentralized data science teams in several business units*
Here, every business unit engages its own data science team because the necessary business knowledge is domain specific. This business-specific analytical knowledge is significant to succeed. This model thus fits to highly complex, mostly large organizations with autonomous business units.

Due to its low requirements, the *operational model (b)* has the potential to be broadly implemented in practice. For its successful realization, it is relevant to consider which resources are available in the data science teams, such as data science skills, technology, and data with sufficient quality. In the beginning, a central interdisciplinary team consisting of experts with different deep focuses such as machine learning, natural language processing, or spatial analysis is formed. Business domain experts support these data scientists to implement high-quality business-related solutions. Scripting capabilities are among the core competencies, as they come into play in all phases of solution creation, from data extraction and transformation to system integration and finally building interactive dashboards for the user.

A good collaboration with the IT department is essential to ensure the work with an analytical sandbox results in high-quality prototypes and products. A framework "from pilot to production" and defined architectures are decisive for becoming

sustainably successful. Crucially, the unit should be supported by an adequate governance. A *steering board* assists with strategic decisions and work prioritization. An internal *data science meetup* presents exciting use cases to interested employees. Additionally, a close relation to renowned local universities is beneficial to learn about the newest methodologies and remain at the state of the art.

5.2 Data-Driven Use Cases

To start with the data-driven journey, organizations need to identify their crucial challenges with the most impact on their business. Then, analytics can support the process of finding a solution toward a new or updated product or service.

To spot relevant use cases, enterprises often get input from the market via the support of different consultants. Another opportunity is visiting industry-related conferences.[18] Design thinking approaches with cross-disciplinary teams, consisting of business people and data scientists, additionally help to detect use cases with strategic impact. A significant collection of key use cases can inspire an enterprise for the further journey. For the use case prioritization, a framework based on two dimensions is usually applied: estimated business benefits vs. the effort of investment in time or money (or complexity). The result is a roadmap of prioritized, high-value use cases together with the anticipated benefit, and consequently, it is possible to define quick wins: new problem-solving approaches that could be implemented very quickly. In addition, this method allows for the efficient identification of the most critical and therefore most valuable use cases for the company. By considering all existing resources, the use case portfolio can be managed well.

After use case prioritizations, it is helpful to start the first pilot with the support of an excited business sponsor. In the future, he or she can be designated as an ambassador for the new analytical approach. The CRISP-DM approach (Wirth and Hipp 2000) is often adopted in data science projects. When the first phase of piloting confirms the benefits for the business, a handover to the IT department helps to sustainably maintain the solution. Finally and step by step, the results and the knowledge about the new methodologies conquer the daily business (a more detailed overview of the creation of data products is presented in the next chapter).

[18]For example, Strata (https://conferences.oreilly.com/strata) and Predictive Analytics World (https://www.predictiveanalyticsworld.com/) are internationally renowned.

6 Summary

We have presented the modern data scientist as a truly and inherently interdisciplinary professional with deep analytical and engineering know how, able to think entrepreneurially and to communicate results in various appealing ways. No matter if one wants to hire or to become a data scientist—there are two pitfalls attached to this definition of a data scientist:

1. The danger of *canonization*: unicorns, as data science all-rounders are often called, do not exist. Any attempt to become or find one are headed for disappointment. The solution is to acknowledge that a senior data scientist should have a reasonable understanding of the majority of the data science skill set map (the crossbeam in "T-shaped people"), while going deep in only a very restricted area of the map (the stem of the "T"). Rather than chasing unicorns, one should view data science as teamwork, with the team of data scientists together covering the skill set map with complementary specializations.
2. The danger of *trivialization*: as finding data scientists becomes harder and being one becomes more profitable, there are natural market tendencies to dilute the skill profile and misuse the fashionable name for conventional job descriptions. Both may lead to wrong decisions due to mishandled complexity.

We presented a data science career as one rooted in one of many potential undergraduate degrees (computer science, industrial mathematics, statistics, or physics are prime candidates) that lays a solid disciplinary foundation (likely connected with the stem of this data scientist's "T"). On this, a data science master's degree can be pursued, or skills can be extended through other, more informal ways of continuing education in order to establish the crossbeam and the personal specialization.

Finally, we gave insights into the development of data science teams. Three operational models for advanced analytics depending on the organization's structure, size, and experience were presented. Besides associated governance, the exploitation of strategic use cases is key to be sustainably successful.

References

Bellman, R. (1961). *Curse of dimensionality. Adaptive control processes: A guided tour*. Princeton, NJ: Princeton University Press.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science, 16*(3), 199–231.

Brodie, M. L. (2015a). *The emerging discipline of data science – Principles and techniques for data-intensive analysis*. Keynote talk the 2nd Swiss Workshop on Data Science SDS|2015. Retrieved April 4, 2018, from https://www.youtube.com/watch?v=z93X2k9RVqg

Brodie, M. L. (2015b). Doubt and verify: Data science power tools. *KDnuggets*. Retrieved April 5, 2018, from https://www.kdnuggets.com/2015/07/doubt-verify-data-science-power-tools.html

Cieliebak, M., Dürr, O., & Uzdilli, F. K. (2014). *Meta-classifiers easily improve commercial sentiment detection tools*. LREC.

Colclough, A. (2017). *When data visualization goes wrong and numbers mislead*. Retrieved April 4, 2018, from https://www.dwrl.utexas.edu/2017/12/29/when-data-visualization-goes-wrong/

Columbus, L. (2017). *IBM predicts demand for data scientists will soar 28% by 2020*. Retrieved April 5, 2018, from https://www.forbes.com/sites/louiscolumbus/2017/05/13/ibm-predicts-demand-for-data-scientists-will-soar-28-by-2020/#6cf2d57e7e3b

Conway, D. (2010). *The data science Venn diagram*. Retrieved April 4, 2018, from http://www.dataists.com/2010/09/the-data-science-venn-diagram/ (figure available from http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram).

Fux, S. J. (2005). *Persönlichkeit und Berufstätigkeit: Theorie und Instrumente von John Holland im deutschsprachigen Raum, unter Adaptation und Weiterentwicklung von Self-directed Search (SDS) und Position Classification Inventory (PCI)*. Cuvillier Verlag.

Guest, D. (1991, September 17). The hunt is on for the Renaissance Man of computing. *The Independent* (London). Quoted by Retrieved April 4, 2018, from https://wordspy.com/index.php?word=t-shaped

Harris, J. G., & Eitel-Porter, R. (2015). Data scientists: 'As rare as unicorns'. *The Guardian*. Retrieved April 5, 2018, from https://www.theguardian.com/media-network/2015/feb/12/data-scientists-as-rare-as-unicorns

Hernandez, J., Berkey, B., & Bhattacharya, R. (2013). *Building an analytics-driven organization*. Retrieved June 9, 2018, from https://www.accenture.com/dk-en/~/media/Accenture/Conversion-Assets/DotCom/Documents/Global/PDF/Industries_2/Accenture-Building-Analytics-Driven-Organization.pdf

Jaokar, A. (2016). How to become a (Type A) data scientist. *KDnuggets*. Retrieved April 4, 2018, from https://www.kdnuggets.com/2016/08/become-type-a-data-scientist.html

Jones, A. (2014). Data science skills and business problems. *KDnuggets*. Retrieved April 4, 2018, from http://www.kdnuggets.com/2014/06/data-science-skills-business-problems.html

Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., & Byers, A. H. (2011). *Big data: The next frontier for innovation, competition, and productivity*. Retrieved March 23, 2018, from https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation

Needham, J. (2013). *Disruptive possibilities – How big data changes everything*. O'Reilly Media. ISBN 978-1-449-36567-7.

Olavsrud, T. (2015). Don't look for unicorns, build a data science team. *CIO*. Retrieved May 28, 2018, from https://www.cio.com/article/3011648/analytics/dont-look-for-unicorns-build-a-data-science-team.html

Patil, D. (2011). *Building data science teams*. Retrieved April 4, 2018, from http://radar.oreilly.com/2011/09/building-data-science-teams.html

Peek, J., O'Reilly, T., & Loukides, M. (1993). *UNIX power tools*. Sebastopol, CA: Oreilly & Associates Incorporated.

Press, G. (2015). *The hunt for unicorn data scientists lifts salaries for all data analytics professionals*. Retrieved May 28, 2018, from https://www.forbes.com/sites/gilpress/2015/10/09/the-hunt-for-unicorn-data-scientists-lifts-salaries-for-all-data-analytics-professionals/

Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). New Jersey: Pearson Education.

Sicular, S. (2012). The quest for data scientists. *The Australian Business Review*. Retrieved April 5, 2018, from https://www.theaustralian.com.au/business/business-spectator/the-quest-for-data-scientists/news-story/eab27147e92d0011520f5adb32010e43

Stadelmann, T. (2017). Science, applied. Die wissenschaftliche Methode im Kern des Produktentwicklungsprozesses. *AIssays blog*. Retrieved April 6, 2018, from https://stdm.github.io/Science-applied/

Stadelmann, T., Stockinger, K., Braschler, M., Cieliebak, M., Baudinot, G. R., Dürr, O., & Ruckstuhl, A. (2013, August). Applied data science in Europe – Challenges for academia in

keeping up with a highly demanded topic. In *European computer science summit ECSS 2013*. Amsterdam: Informatics Europe.

Stanley, J., & Tunkelang, D. (2016). Doing data science right – Your most common questions answered. *First Round Review*. Retrieved June 11, 2018, from http://firstround.com/review/doing-data-science-right-your-most-common-questions-answered/

Stockinger, K., Stadelmann, T., & Ruckstuhl, A. (2016). Data Scientist als Beruf. In D. Fasel, & A. Meier (Eds.), *Big Data*. Edition HMD. https://doi.org/10.1007/978-3-658-11589-0_4.

Taylor, D. (2014). *Battle of the data science Venn diagrams*. Retrieved April 4, 2018, from http://www.prooffreader.com/2016/09/battle-of-data-science-venn-diagrams.html

Tierney, B. (2013, March 22). Type I and Type II data scientists. *Oralytics Blog*. Retrieved April 4, 2018, from http://www.oralytics.com/2013/03/type-i-and-type-ii-data-scientists.html.

Trafimow, D., & Marks, M. (2015). Editorial. *Basic and Applied Social Psychology, 37*(1), 1–2. https://doi.org/10.1080/01973533.2015.1012991.

Wasserman, L. (2013). *All of statistics: A concise course in statistical inference*. New York: Springer Science & Business Media.

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining* (pp. 29–39).

Wu, J. (1997). *Statistics = data science*? Inaugural lecture at university of michigan, Ann Arbor. Retrieved April 4, 2018, from https://www2.isye.gatech.edu/~jeffwu/presentations/datascience.pdf

Zinkevich, M. (2018). *Rules of machine learning: Best practices for ML engineering*. Retrieved April 6, 2018, from https://developers.google.com/machine-learning/rules-of-ml/

Chapter 4
Data Products

Jürg Meierhofer, Thilo Stadelmann, and Mark Cieliebak

Abstract Data science is becoming an established scientific discipline and has delivered numerous useful results so far. We are at the point in time where we begin to understand what results and insights data science can deliver; at the same time, however, it is not yet clear how to systematically deliver these results for the end user. In other words: how do we design data products in a process that has relevant guaranteed benefit for the user? Additionally, once we have a data product, we need a way to provide economic value for the product owner. That is, we need to design data-centric business models as well.

In this chapter, we propose to view the all-encompassing process of turning data insights into data products as a specific interpretation of service design. This provides the data scientist with a rich conceptual framework to carve the value out of the data in a customer-centric way and plan the next steps of his endeavor: to design a great data product.

1 Introduction

Analytics[1] provides methodologies and tools to generate insights from data. Such insights may be *predictive*, for example, a traffic forecast, a recommendation for a product or a partner, or a list of customers who are likely to react positively to a marketing campaign (Siegel 2013). Insights may also be *descriptive*, that is, providing us with a better understanding of a current or past situation, for example, our company's performance right now or during the previous month. Insights will probably in any case be *actionable*, for example, by enabling a smart controller to

[1]We use the word "analytics" throughout this chapter to refer to those methods and tools from data science that pertain directly to analyzing, mining, or modeling the data: Statistical methods, machine learning algorithms, the application of data management tools, etc.

J. Meierhofer (✉) · T. Stadelmann · M. Cieliebak
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: juerg.meierhofer@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_4

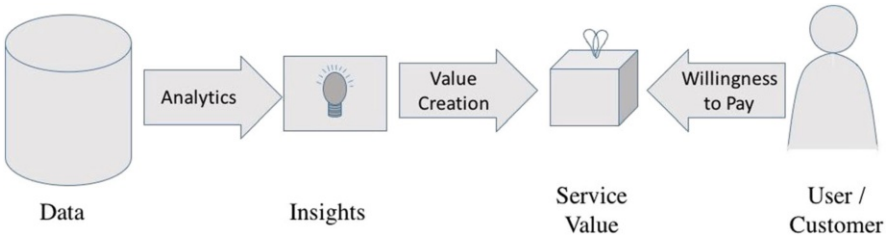

Fig. 4.1 The value chain of a data product

drive a car, operate a building control system, or regulate electricity production according to market demands. In an extension of the purely exploratory paradigm of data mining, a data scientist purposefully plans to build such data insights that benefit the user (Veeramachaneni 2016). This automatically moves the *result* to the center of the analytics process.

But do these kinds of insights already make up a data product? To find the answer, we go back to the definition of a product (Kottler 2003): a product provides a set of benefits for which the customer has a willingness to return a value, typically in the form of money. Thus, insights generated from data can be considered a data product if there are "users" willing to give back value for these insights. The user may be an external customer (e.g., a "consumer") or a user in an organization, for example, inside the company. The value given back may be in the form of a financial payment, but not necessarily (there are other dimensions of value like emotional or social value (Jagdish et al. 1991), or the collected data, e.g., health data from wearables, search patterns, etc.). This is illustrated in the complete value chain of a data product (see Fig. 4.1).

In other words: in order to have a data product, we need to design insights generating relevant benefits for which users pay. Service science provides us with concepts to solve this problem: according to Lusch and Vargo (2014), a service is defined as the application of competences (knowledge and skills) for the benefit of another entity. With respect to data products, the application of competences refers to the competence of applying data science—the "*unique blend of skills from analytics, engineering & communication aiming at generating value from the data itself*" (Stadelmann et al. 2013).

Therefore, a data product is defined as the application of data science competences to provide benefit to another entity. This makes perfect sense if we substitute "data science" for its original definition cited above, thus resulting in:

> A data product is defined as the application of a unique blend of skills from analytics, engineering & communication aiming at generating value from the data itself to provide benefit to another entity.

Data products are a subset of services (every data product meets the definition of a service, but not every service is a data product). Therefore, the concepts and methods of service science and service design can be applied to systematically design data products. This rounds off earlier work of defining a data product as the result of

value-driven analysis that generates added value out the analyses of the underlying data (Loukides 2010). There is a vast field of application examples available for added value generated by analysis. Siegel (2013), for instance, provides an extensive list of 182 examples grouped in the 9 categories: (1) family and personal life; (2) marketing, advertising, and the web; (3) financial risk and insurance; (4) healthcare; (5) law enforcement and fraud detection; (6) fault detection, safety, and logistical efficiency; (7) government, politics, nonprofits, and education; (8) human language understanding, thought, and psychology; (9) workforce: staff and employees. There are also other literature sources providing similar application examples with different groupings, for example, Marr (2016).

In the next section, we provide a very short introduction to general service design before explaining the specific characteristics of applying it to the design of data products. We then identify the gap between current service design and the development of data products, and subsequently propose a framework specific for data product design. We conclude by a discussion of the essential building block of each data product—the data itself, and how to potentially augment it—and a review of the current state of the field, including an outlook to future work.

2 Service Design

Service design starts from the user perspective, which means understanding the tasks and challenges the user faces in his context. Customer insight research methods such as depth interviews, participant observation or shadowing, service safari, focus groups, cultural probes. etc. (Polaine et al. 2013), serve to understand the user in his context. The value proposition design framework (Osterwalder et al. 2014) describes a practical template to map the customer jobs, pains, and gains, which together constitute the so-called customer profile (see right hand side of Fig. 4.2). The customer jobs are challenges and tasks that the user needs to tackle and solve. The pains are factors that annoy the user during his job, and the gains provide the benefits that the customer aims at. For the design of the data product, features fitting

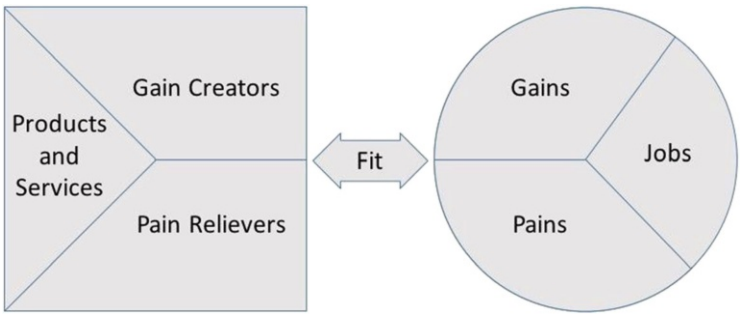

Fig. 4.2 Fit of value proposition (left) with customer needs (right) (Osterwalder et al. 2014)

with the customer jobs, pains, and gains need to be designed (right hand side of Fig. 4.2). In this context, it is very important to note that service design systematically considers also non-functional customer jobs, for example, emotional or social jobs (Osterwalder et al. 2014; Smith and Colgate 2007).

Additionally, we can apply many more of the useful tools for service design, like customer journey mapping, emotion curve, service blueprinting, service ecosystem design, etc. (Polaine et al. 2013; Stickdorn and Schneider 2010).

A word about terminology: the term "service design for customers" often evokes the connotation of consumer services. However, according to the concept of service-dominant logic, the so-called customer generically may be any person getting benefits of the service (Lusch and Vargo 2014). The human being may well be a consumer, but also an employee getting support for doing his job, a citizen getting support for his everyday life, or also an individual representing a societal stake.

3 The Gap Toward Data Product Design

Keeping in mind our definition of data products (the application of data science competences to provide benefit to another entity), the service design approach discussed so far clearly satisfies the second part of that definition, that is, providing benefit to another entity. However, there is still a gap w.r.t. to the application of data science competencies: service design per se does not systematically consider using analytics competences to bring forth benefits for the customer. In cases where the respective data is available, leveraging analytics capabilities in service design (i.e., doing data product design) generally yields more value to the customer and in return more revenue to the provider.

Two scenarios are conceivable—enhancing existing or creating completely new services:

1. First, we may assume that an existing product or service is effective in meeting the customer needs but could do this *more efficiently* if insights from data were used. For example, assume a service giving advice to customers when to replace existing factory equipment (machines). Leveraging data about the status of the old machines (i.e., condition monitoring) as well as forecasted production volumes, market evolution, etc., the service can become much more efficient and more effective. In this scenario, an existing solution becomes more efficient and is provided with higher quality.
2. Second, by leveraging data science, we can find *completely different and new products* which are much more effective in meeting the customer needs. Although new data products do not create new customer needs[2] (the fundamental

[2]There is often the belief that technology can create new customer needs, which is only true at a superficial level. If we dig deeper in the hierarchy of customer needs, which we do in service design, we find underlying needs which are given by the customers' tasks.

underlying motivations and needs of customers have been there before, often not at the conscious level), the new data products may provide completely new and previously inconceivable ways to satisfy those needs. For example, we may develop a configurable music player that continually evaluates data about the context and situation of the user via a connection to his smartphone and adapts the playlist to meet the circumstances, smoothly adapting to events like new music releases or sensed moods and environmental conditions.

Designing the resulting data products requires methodologies that go beyond those covered by the service design literature. Meierhofer and Meier (2017) propose an approach to data product design which we are going to discuss in the following section.

4 Bridging the Gap (Then and Now)

From the previous discussion we see that service design provides us with a framework to systematically design products that generate relevant benefits for the customer. These benefits could be quantitatively or qualitatively higher if the potential of data was leveraged.

However, data scientists made the experience in recent years that insights generated by sophisticated analytics algorithms are often not properly adopted or undervalued by the users: the insights may be considered technology driven, not relevant for the user, or simply not trusted by experts (Finlay 2014; Veeramachaneni 2016). Hence, there is a gap between analytics results and value creation. This gap needs to be bridged in order to exploit the potential of data products (see Fig. 4.3).

Of course, many excellent data products available today show that this gap can be bridged: the examples of Siegel (2013) in nine different industries have already been mentioned. Such cases, in which insights from data are developed into data products that fit with the customer needs, might be successful because of the situative combination of good ideas: interdisciplinary teams formed by so-called "T-shaped-people"[3] (Stickdorn and Schneider 2010) (i.e., by the ideal profile of a data scientist) may be sufficiently creative to exploit the potential of analytics while deriving a value proposition that is consequently driven by the customer needs. However, a more systematic methodology for the development process is desirable.

First approaches for systematic data product design have been presented in the literature after Loukides (2011) pointed out that "...the products aren't about the data; they're about enabling their users to do whatever they want, which most often has little to do with data." Howard et al. (2012) then suggested the so-called drivetrain approach that we will briefly review below. Recently, Scherer et al.

[3]The horizontal part of the T-shape refers to the broad skills in a large field like data science, with additional depth in a specific sub-field, e.g., service design or analytics (the vertical part).

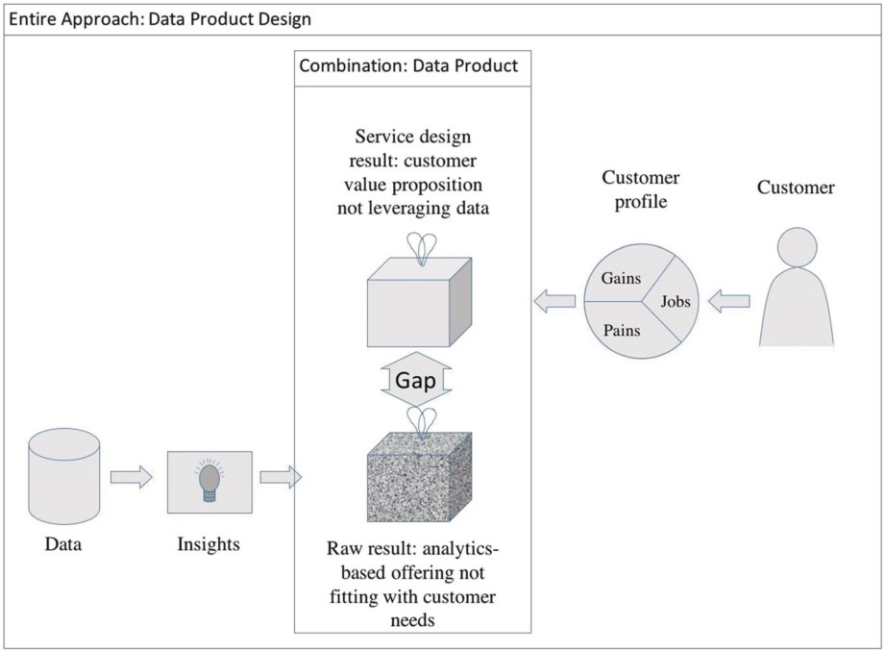

Fig. 4.3 Data products bridging the gap between analytics and service design

(2016) presented another approach on how to use data analytics for identifying user patterns.

The 4-stage drivetrain process starts with the definition of the goal: Which of the user's needs shall be addressed next? Let us assume for a moment the example of "web search"—finding information on the web, based on keyword queries. The second step is then the identification of the levers that the data scientist can set to reach this goal. This may, for example, be a new analytics idea, as has been the case with the "PageRank" algorithm within Google's then new answer to the web search example created above: it is based on the idea that the number of incoming and outgoing links to web pages (so-called off-page data) contain information about its relevance with respect to a query. The third step consists of collecting the necessary data sources to enable setting the identified levers of the previous step. In the Google example, this data was collected by the company in their search index. The data may thus already be available internally. However, the combination of internal and external data has great potential for (and often holds the key to) the realization of new analytics ideas. For this reason, the question of how to design good data products is closely linked with knowledge of the international data market as well as of the open data movement and respective options: publicly available datasets may at least augment one's internal data, as the next section will show. The fourth step finally involves building analytical models, as the options of which modeling technique to apply are to a large extent predetermined by the previous three steps.

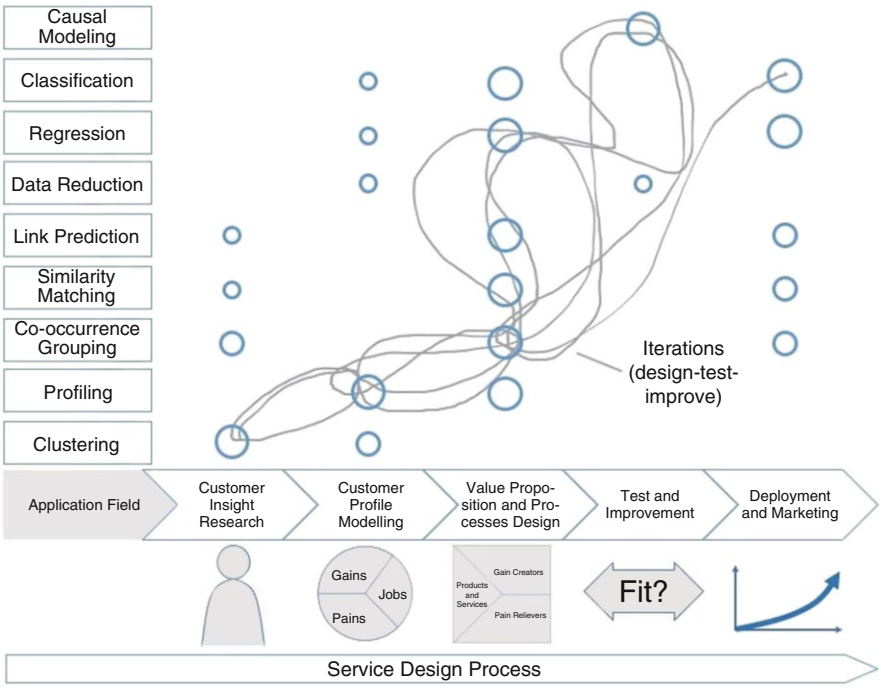

Fig. 4.4 Methodological bridge between the service design process and data analytics tools

The drivetrain approach is the distillate of the lessons learned of hundreds of publicly held data science competitions. While capturing indispensable knowledge, it is still quite abstract, being more descriptive than prescribing next actions: it serves well as a model to conceptualize a successful data product design project in retrospect but is hard to use as a model to decide on the next concrete step.

To overcome this weakness and provide an all-encompassing approach for data product design, we propose to cover all phases of the service design process and additionally exploit the full spectrum of data analytics methods and tools as much as possible, in a way that allows for planning ahead. We use the framework shown in Fig. 4.4.:

- The horizontal axis depicts the stages of a typical service design process: for a given application field (e.g., "customer searches and purchases a new product of our portfolio") we start the process with collecting data about potential users or customers ("customer insight research"), then build the customer profile (jobs, pains, gains), followed by the phases for designing the value proposition and the service processes. In the next phase, we test the fit between value proposition and the customer profile and improve our solution in several iterations (indicated by the squiggles in the figure). In the last step, we bring our new data product to the market (deployment and marketing).

- The vertical axis in Fig. 4.4 shows a structure of several data analytics methods w.r.t. their potential to provide benefits for data products (raw results according to Fig. 4.3). The terms on the vertical axis (from "clustering" to "causal modeling") stand for fundamental data analytics methods (or tools, use cases) according to Provost and Fawcett (2013).
- The dots in the matrix framework of Fig. 4.4 indicate in which stage of the data product design process which data analytics tool can be typically applied. Larger dots qualitatively indicate a stronger value contribution in the corresponding combination.

As practical service design cases often do not follow the service design process in a linear way from left to right in Fig. 4.4, we exemplarily discuss the matrix in Fig. 4.4 by a case study according to Meierhofer (2017). This example is in the application field of customer service representatives (agents) in a company providing consumer services. The goal is to provide employee support services to the agents in order to make inconvenient tasks easier for the employees, to reduce sources of errors, and to increase efficiency. Such tasks may be, for example, to detect the relevant contact reason of the customer in real-time (e.g., while having the customer on the phone) and find related contacts of the same customer in the past. For instance, a customer may have contacted the firm for various matters several times before and this time has a complaint for one specific topic, which makes it difficult for the agent to dig the details relevant for this complaint out of the contact history in a short time. Or the customer may call because of a specific question concerning his customized product instantiation or his customized contract. It is likely that the company has the answer ready in its data base in the form of a documented solution of a similar problem in the past. An individual agent can impossibly keep all these cases in mind or find them while talking to a customer.

Data and technical tools, e.g., records of past customer interactions as well as algorithms for speech-to-text and natural language processing, are assumed to help in this process and provide benefits to the agents. Finding the relevant nuggets in the bulk of past customer contacts, which often are documented as unstructured data, can be heavily supported by such analytics tools. Hence, this case study starts from the perspective of data and technology according to Meierhofer and Meier (2017) instead of a precise understanding of the user's jobs, pains, and gains. It can be considered a technology push approach.

In traditional service engineering procedures, the project would deploy as follows:

- An interdisciplinary project team is set up consisting of (a) analytics specialists, (b) IT specialists in the company-specific CRM system, and complemented by (c) business process specialist of the customer service department.
- In a requirements engineering process, the required features for the agent support tool are elaborated and then stripped down to a feasible set in the framework of the project constraints (cost, time, quality).
- The tool is implemented, technically tested, and deployed to the users. This last step includes training as well as change management aspects in order to convince

the agents of the benefits of the new tool. This development and deployment phase would typically span over several months and result in high resources costs.

Unfortunately, this procedure often turns out not to be effective in the sense that the tool delivered after the long development period does not solve relevant jobs, pains, or gains of the users. As a consequence, the users consider the tool irrelevant and are not ready to invest the energy to get sufficiently familiar with it in order to leverage at least some benefit. This is the point where cultural change management comes in to get the agents to use the new tool, which is often not successful. At the end, the project may be considered a disappointment.

To circumvent this problem, best-practice approaches have come up in the recent years tackling the problem from a design perspective in combination with agile methodologies. The challenge to support the agents in their daily work would consequently start by understanding and modelling the agents jobs, pains, and gains. Next, a value proposition would be developed which helps the agents to do their job, overcome the pains, and increase their gains. However, this procedure would typically miss out the potential of the new possibilities in analytics, which may be assumed in the fields of mining data (e.g., past customer interaction records) or process automation (e.g., speech recognition). As a consequence, for the case study described above, an agent support tool may be built which turns out to be useful for the agents, for example, by providing search tools for similar problems, but could possibly provide much more benefit by systematically applying analytics.

Now, applying the new data product design scheme shown in Fig. 4.4, we proceed as follows:

- To start, remembering that we have a technology-driven case, we elaborate a map of the data-driven assets available which we assume to provide benefits for the given problem statement. In this case, this is:
  - Generating a layout of insights that can be gained from past customer interactions. The data of closed customer contacts, which is stored in records in the CRM tool, is mined and interpreted by data scientists in co-creation with process experts of the customer service department.
  - Exploring the possibilities of natural language processing and speech-to-text conversion in the context of the agents' work with a CRM system (e.g., the environment of the use case, the languages applied, the acoustical environment, the real-time requirements, etc.).

  This collection of the data-based value contributions as a starting position corresponds to tackling the problem from the left-hand side in Fig. 4.1 and to establishing the vertical axis in Fig. 4.4.
- In the next step, we develop the horizontal axis of Fig. 4.4 and proceed with understanding the agents' jobs, pains and gains. To do so, we research insights about their jobs, pains, and gains by shadowing a qualitative sample of agents in their daily job (i.e., accompanying the agents as an observer). A practical tool to do this can be found in the "a day in the life of" concept: accompanying a person during a typical day and observing what she does, where she struggles or needs

too much time or energy for doing a job, and where she gets the desired output in a satisfactory manner.

This qualitative customer insight research step is complemented by a quantitative analysis of process data found in the agent workflow tool. The process steps found completed in past customer interactions are stored with their timestamp as well as the type of process step and free text remarks entered by the agent. This analysis backs up the qualitative insights about the jobs, pains, and gains found so far, and eventually verifies or falsifies the hypotheses.

- This collection of agent data also enables the potential segmentation of the agents into different profiles (so-called "personas" in the service design terminology) by clustering approaches. Based on this, different profiles of agents can be described. If the analysis yields different relevant profiles with clear differences in the pains and gains (the jobs are assumed to be the same in the same job context), the service for the agents needs to be developed with different flavors depending on the profile.
- Next, we tackle the task of developing the actual service for the agents, which means developing the value proposition (left-hand side of Fig. 4.2). In this step, we now make use of the collection of the data-based value contributions that we prepared at the start of our technology-driven approach. We confront the elaborated agents' jobs, pains, and gains with those value contributions differentiated according to the customer profile. This step yields the following outcomes:
  - There are jobs, pains, or gains to which we can responded by the given data-based value contributions. For example, finding similar cases in the past may be supported by similarity matching of the current case description with past descriptions by means of Information Retrieval methods.
  - There are jobs, pains, or gains for which we do not have a data-based value contribution. This situation takes us to making additional data sources accessible or to solving the problems by non-data-based means. For instance, it would be very helpful for the agents to get an indication of the customers' current emotional tension and the evolution of this in the past. We may not have sufficient data of the past cases to detect this reliably and may suggest a conversational script for the agent to find this out while talking to the customer.
  - There are data-based value contributions for which we do not have a corresponding job, pain, or gain (yet). In this case, we may find that the particular data has no value for our problem. Or, alternatively, we may find a way to utilize the data for solving the problem in a new way which was not seen before. Example: for a given customer enquiry, we may have data indicating that other users already had the same problem before, but the solution could not be standardized enough to generate a script for the agents for solving future problems. However, we can leverage this information to create a user support community and defer users whose problems have sufficient similarity to this community for peer-to-peer problem solving.

- The new service for supporting the agents (i.e., the value proposition) designed in this way is developed in several prototyping steps and tested with a sample of agents. These tests reveal technical bugs, but much more important, make transparent whether our hypothesis on the agents' jobs, pains, and gains as well as the corresponding value proposition are validated or falsified. If falsified, we introduce an additional iteration and adapt our service to test it again until we find a sufficient fit of our solution with the problem.
- Finally, when we deploy the new tool to the entire group of our customer service representatives, we measure how the tool is used by collecting data from the process workflows and the CRM data records. We detect where the solutions can be improved and enter the continuous improvement process.

5 The Essential Building Block of a Data Product

We finally turn our attention to the essential building block that distinguishes a data product from universal services: the supporting data and its analysis. Here, we focus on the data sources, since methods and technologies for data analytics are covered in detail in several other chapters of this book.

A data product can only be as good as its supporting data. While this statement might sound trivial at first sight, it has enormous impact on the design of a data product: if the underlying data is unreliable, all efforts to get high-quality analytics results and creating value to the customer must fail. Here, "unreliable" includes various types of issues, for example, incomplete or faulty data entries, unavailability, legal issues, etc. Hence, careful selection of appropriate data sources is an important step in the development of a data product.

Many data products are based on internal data. This data is proprietary to the data service provider, who often has full authority over its content, data format, access rules, licenses etc., which makes it comparably[4] easy and straightforward to incorporate it in a data product. However, there are still some reasons why internal data might not be used for a data product:

1. It is *personal* data, that is, "all information relating to an identified or identifiable person" (FACH 1992); this could be, for instance, customer profiles, phone call history or transcripts, customer feedback, etc. All personal data is subject to privacy regulations, which vary significantly from country to country. For instance, in the USA any data that might be traced back to an identifiable person is considered private and, thus, protected. When Netflix, a video-on-demand service, released a dataset of movie ratings of its users, researchers were able to

[4]Numerous hardships are attached to the process of extracting, transforming, and loading (ETL) even internal data into a form that is amenable for analytics. The topic of (automatic) data integration and corresponding engineering efforts toward a data warehouse is huge. For the sake of this chapter, however, we will assume the respective organization has already taken care of it.

identify individual persons within this supposedly anonymized data, thus, forcing Netflix to withdraw the dataset (Singel 2009).
2. The data is *confidential*, for example, emails, internal documents, meeting minutes, etc., and an automated data product might unwittingly reveal such information to the customers.
3. The data was *intended for a purpose* different from the data product: for instance, the "Principle of Earmarking" in German and European data protection regulations explicitly prohibits usage of personal data for any other than the intended purpose without consent.

As a way out, data products may augment internal data with additional external sources to provide maximum benefit to the user. There exist literally hundreds of thousands of external datasets that are available for download[5] (static) or via an "Application Programming Interface (API)" (dynamic). Thus, the question often is not *if* a useful dataset exists, but *where* to find it in the vast expanse of accessible data sources. To this end, *data marketplaces*, such as datahub, Amazon AWS Datasets, or Microsoft Azure Marketplace, come into play, which are useful in three major ways: they are a central point of discoverability and comparison for data, along with indicators of quality and scope; they handle the cleaning and formatting of the data, so that it is ready for use (this step, also known as data wrangling or data munging, can take up to 80% in a data science project (Lohr 2014); and they offer an economic model for broad access to data that would otherwise prove difficult to either publish or consume.

On the other hand, there exists a vast amount of *open data*, which is ever-increasing since more and more governments, research institutions, and NGOs are adapting open data strategies. These data include, for instance, human genome sequences, historic weather data, or voting results of the Swiss National Council. Data collections such as data.gov (USA), open-data.europa.eu (European Union), or data.gov.uk (United Kingdom) contain thousands of public datasets (see Chap. 14 on the usage of open data). While most of these datasets are stand-alone, *Linked Open Data* (LOD) provides methods to interlink entities within open datasets. Linked data, which goes back to Tim Berners-Lee (2006), uses a Uniform Resource Identifier (URI) for each entity, and RDF triples to describe relations between these entities. This allows machines to traverse the resulting graph, which contains nowadays billions of entities, and collect required information automatically.

Once the underlying data of the data product is clear, it can be collected, pre-processed, combined, and analyzed to provide the desired service to the customer. Since most data products rely on data that changes over time, it is important to track the data sources closely, because API's can be updated, data formats may change, or entire data sources may vanish completely. Only then it can be ensured that the data product works reliable and to the benefits of the customer.

[5] See, for example, http://cooldatasets.com/

6 Discussion and Conclusions

We have reviewed the state of the art in data product design and concluded that up to now no systematic approach has been presented that allows for planning the next steps in designing a product based on data insights specifically to the needs of a certain customer. We suggested to extend the methodology found in the discipline of service science by concrete ideas on how and where to invoke certain analytics methods and tools. We argued that using the methodology, and hence vocabulary, of service-dominant logic and service design gets data scientists a long way toward such a development processing on the broad range of possible data science use cases, not just in typical "service business" settings. In Fig. 4.4, we presented a concise but all-encompassing framework of how to develop data products from a user-centric point of view, including suggestions of typically helpful analytics methods and tools per design phase. Finally, we gave pointers to potential external data sources to enhance the essential building block of each data product—its underlying data collection.

We see data product design as a discipline that is still in its infancy.[6] Its core and borders are still very much under development:

- While one of the first university-level courses on the topic mentions to ". . .focus on the statistical fundamentals of creating a data product that can be used to tell a story about data to a mass audience" and then focuses on technical details in building web applications (Caffo 2015), others are based on a curriculum that focuses on service design, leaving analytics aspects to other modules (Stockinger et al. 2016).
- While the drivetrain approach has been too abstract to guide new design endeavors, our approach is conceptually nearer to certain kinds of applications and thus may in practice be more difficult to apply to a problem of, say, the internal control of a machine (where no user is directly involved) than in marketing (although it really is generally applicable).

We thus see our presented approach as a contribution to an ongoing discussion: all data scientists need, besides deep analytics know-how, the business-related skills to not just design a successful algorithm, but to think through a whole product. This is the all-encompassing process we have sketched above. For the engineering-heavy data scientist, who daily mangles data and thinks in terms of algorithms, this may seem far away: she is more involved in CRISP-DM-like processes (Shearer 2000) to assemble the smaller parts of the final solution. But these smaller parts are then treated as black boxes within the all-encompassing data product design process as outlined above.

In this sense, the data product design approach presented here is not the process to create each data insight (smaller part). It is the packaging of one or many of these into "publishing" form through optimization, wrapping, and finally marketing.

[6] Borrowing a phrase from Michael M. Brodie that he frequently relates to data science as a whole.

Future investigation has to answer the question of how to bring both processes into one conceptual framework: the "internal" CRISP-DM-like data insight creations, and the "external" data product design wrapper.

References

Berners-Lee, T. (2006). Linked data. *Blog post*. https://www.w3.org/DesignIssues/LinkedData.html

Caffo, B. (2015). *Developing data products*. MOOC at Johns Hopkins University, Coursera. https://www.coursetalk.com/providers/coursera/courses/developing-data-products

Federal Assembly of the Swiss Confederation. (1992). *Federal act of 19 June 1992 on data protection (FADP)*, paragraph 3. https://www.admin.ch/opc/en/classified-compilation/19920153/index.html

Finlay, S. (2014). *Predictive analytics, data mining and big data: Myths, misconceptions and methods*. Basingstoke: Palgrave Macmillan.

Howard, J., Zwemer, M., & Loukides, M. (2012). *Designing great data products – the drivetrain approach: A four-step process for building data products*. Retrieved March 28, 2012, from https://www.oreilly.com/ideas/drivetrain-approach-data-products

Jagdish, N., Sheth, B., & Newman, I. (1991). Why we buy what we buy: A theory of consumption values. *Journal of Business Research, 22*(2), 159–170.

Kottler, P. (2003). *Marketing management*. Upper Saddle River, NJ: Prentice Hall.

Lohr, S. (2014). For big-data scientists, 'janitor work' is key hurdle to insights. *The New York Times*, blog post. https://mobile.nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html

Loukides, M. (2010). *What is data science?*, blog post. https://www.oreilly.com/ideas/what-is-data-science

Loukides, M. (2011). *The evolution of data products*. O'Reilly Media. ISBN 978-1-449-31651-8.

Lusch, R. F., & Vargo, S. L. (2014). *Service-dominant logic*. Cambridge: Cambridge University Press.

Marr, B. (2016). *Big data in practice: how 45 successful companies used big data analytics to deliver extraordinary results*. Chichester: Wiley. ISBN 978-1-119-23138-7 (hbk).

Meierhofer, J. (2017). Service value creation using data science. In E. Gumesson, C. Mele, & F. Polese (Eds.), *Service dominant logic, network and systems theory and service science: Integrating three perspectives for a new service agenda*. Rome: Youcanprint Self-Publishing.

Meierhofer, J., & Meier, K. (2017). From data science to value creation. In *Proceedings of International Conference on Exploring Service Science* 2017.

Osterwalder, A., Pigneur, Y., Bernarda, G., & Smith, A. (2014, November). *Value proposition design*. Hoboken, NJ: Wiley.

Polaine, A., Løvlie, L., & Reason, B. (2013). *Service design – From insight to implementation*. Brooklyn, NY: Rosenfeld Media.

Provost, F. P., & Fawcett, T. (2013). *Data science for business*. Sebastopol, CA: O'Reilly and Associates.

Scherer, J. O., Kloeckner, A. P., Duarte Ribeiro, J. L., Pezzotta, G., & Pirola, F. (2016). Product-service system (PSS) design: Using design thinking and business analytics to improve PSS design. *Procedia CIRP, 47*, 341–346.

Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *J Data Warehousing, 5*, 13–22.

Siegel, E. (2013). *Predictive analytics – The power to predict who will click, buy, lie, or die*. Hoboken, NJ: Wiley. ISBN 978-1-118-35685-2.

Singel, R. (2009). Netflix spilled your brokeback mountain secret, lawsuit claims. *Wired*, blog post. https://www.wired.com/2009/12/netflix-privacy-lawsuit

Smith, J. B., & Colgate, M. (2007). Customer value creation: A practical framework. *Journal of Marketing Theory and Practice, 15*(1), 7–23.

Stadelmann, T., Stockinger, K., Braschler, M., Cieliebak, M., Baudinot, G., Dürr, O., & Ruckstuhl, A. (2013). *Applied data science in Europe – Challenges for academia in keeping up with a highly demanded topic*. European computer science summit ECSS 2013. Amsterdam: Informatics Europe.

Stickdorn, M., & Schneider, J. (2010). *This is service design thinking*. Amsterdam: BIS.

Stockinger, K., Stadelmann, T., & Ruckstuhl, R. (2016). Data Scientist als Beruf. *Big Data*. Edition HMD, Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-11589-0_4.

Veeramachaneni, K. (2016, December 7). *Why you're not getting value from your data science*. Harvard Business School Publishing Corporation. https://hbr.org/2016/12/why-youre-not-getting-value-from-your-data-science

Chapter 5
Legal Aspects of Applied Data Science

Michael Widmer and Stefan Hegy

Abstract Data scientists operate in a legal context and the knowledge of its rules provides great benefit to any applied data science project under consideration, in particular with view to later commercialization. Taking into account legal aspects early on may prevent larger legal issues at a subsequent project stage. In this chapter we will present some legal topics to provide data scientists with a frame of reference for their activities from a legal perspective, in particular: (1) comments on the qualification and protection of "data" from a legal perspective, including intellectual property issues; (2) data protection law; and (3) regulatory law. While the legal framework is not the same worldwide and this chapter mainly deals with Swiss law as an example, many of the topics mentioned herein also come up in other legislations.

1 Introduction and Background Information

Data science is inherently an applied science (Stadelmann et al. 2013) aiming at generating value from the data itself; thus, law is among the many disciplines to be taken into account in data scientists' activities. Considering legal aspects already at the outset of the development of data products may help address and minimize last minute legal issues, which would not only be frustrating but in many cases also costly. An example could be the development of a business solution for a financial service provider that neglects certain regulatory requirements stated by the financial supervisory authority. In this event, the regulatory barrier would impede a successful implementation, irrespective of increased efficiency standards or the potential commercial value of such solution. Implementing the regulatory requirements into an already existing product at the end may require substantial work and involves additional costs. If the legal Dos and Don'ts had been properly outlined and taken into consideration from the outset—either with or without the client being

M. Widmer · S. Hegy (✉)
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: wime@zhaw.ch; hegy@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_5

involved—the project would possibly have been successful without further changes. Also, understanding and taking into account legal concerns from the start may make companies more willing to conduct (big) data analytics, while currently some companies are still reluctant to do so, partly because of legal and information security concerns (KPMG and Bitkom research 2016).

While this chapter cannot provide an in-depth review of all legal aspects, it is meant to provide data scientists with a certain awareness for important issues they may encounter in the "legal minefield" of data science, if only to seek assistance from a legal department or legal advisor at the right time. Since the number of potential legal issues is extensive, this contribution will outline some of these legal topics to provide data scientists a frame of reference, in particular: (1) comments on "data" from a legal standpoint, including intellectual property issues (*see* Sect. 2 below); (2) data protection law (*see* Sect. 3); and (3) regulatory law (*see* Sect. 4). While the legal framework is not the same worldwide and this chapter mainly deals with Swiss law as an example, many of the topics mentioned also come up in other legislations. For example, the draft for a revised Swiss Data Protection Act mirrors many of the provisions of the General Data Protection Regulation of the EU. Therefore, the issues raised below will likely also come up in a similar manner in other jurisdictions. However, data scientists have to be aware that if their activities concern a multitude of jurisdictions (in particular, if they are acting cross-border) they will have to determine which laws apply and may even have to take into account various legal systems.

2 "Data" from a Legal Standpoint: Goods, Intellectual Property, and Unfair Competition Law

2.1 Introductory Comments

It is obvious that "data" is at the core of data science. Consequently, data scientists should always consider how the rights to data affect their activities, for example, whether they have obtained the necessary rights concerning the data they work with and how such rights may affect the results of their work. If they have not obtained the necessary rights, their activities may infringe third parties' rights and the exploitation, use, or even publication of the results of their activities may be adversely affected (cf. Oxenham 2016, p. 16).

2.2 Ownership of Data as Ownership of Goods?

While the legal qualification of "data" is important to the activities of data scientists, the term is not entirely clear in all respects. The question of "ownership" of data, how the commercial purposes (e.g., transfer of data) may be reached from a legal

perspective and how data should be qualified were extensively discussed in Switzerland. With respect to personal data some argue that a right similar to an intellectual property right should be introduced (Flückiger 2013, p. 864) or that data should be treated similar to goods (Eckert 2016b, pp. 247 et seq.) (Eckert 2016a). Others are against the introduction of ownership rights and propose contractual solutions of the issues arising in this context (Benhamou and Tran 2016, pp. 572 et seq.).[1] In short: how data should be qualified from a legal perspective is still somewhat disputed.[2] However, data are not "goods," and as long as the data is not protected by copyrights, other intellectual property rights, or unfair competition laws (*see* below), the only remaining solution is to solve any issues arising, for example, concerning how certain data should be used, with contracts—although this may not always be possible.

Consequently, while it is agreed that data often is a commercial asset that is widely being sold and licensed, the legal discussion in this respect is still somewhat open. Data scientists have to keep in mind that the above-mentioned issues will have an impact on their activities and have to be taken into account.

But irrespective of how the abstract "data" is qualified from a legal perspective, one has to remember that the data *carriers*, that is, the tangible media on which the data is stored, legally qualify as "goods." The ownership, etc., of such data carriers will be treated like any other goods, so they can, for example, be owned and sold.

2.3 Copyrights

In addition to the protection of the legal ownership of the "data carriers," on which the data is stored, the abstract "data" will in certain cases be protected under other legal concepts. For example, certain data may be protected by copyrights and neighboring rights.

With regard to the European Union (EU), several directives are in place to harmonize essential copyright rights throughout the member states. Reference is made in particular to the Directive on the harmonization of certain aspects of copyright and related rights in the information society (InfoSoc Directive; 2001/29/EC) and the Directive on the legal protection of computer programs (Software Directive; 2009/24/EC) (cf. European Commission n.d.). In Switzerland, Article 2 para. 2 Swiss Copyright Act (CA) provides that literary and artistic intellectual creations with an individual character constitute "works" and are protected, irrespective of their value or purpose. Computer programs also qualify as "works" in this sense (Article 2 para. 3 CA). "Collections" are protected as works in their own right insofar as they are intellectual creations with individual character with regard to their selection and arrangement (Article 4 para. 1 CA). However, this does not

[1] Further details concerning this discussion as well as alternative solutions, cf. Thouvenin (2017).

[2] With regard to the international discussion, cf., for example, Ritter and Mayer (2018).

preclude that works included in a collection may continue to be protected individually (Article 4 para. 2 CA).

Where data science is concerned, the question will arise whether certain data is copyright protected or not. While the threshold for a qualification as a work is not very high (e.g., original texts, pictures, or music are protected by copyright), mere ideas are not copyright protected and this question will have to be answered on a case-by-case basis. Depending on the data set, which is intended to form the basis of the data scientist's work, either some (or none) of the data therein may be protected as an individual work, part of the data may constitute a collection, or even the entire data set may constitute a collection.[3]

To the extent that certain data do constitute a work or a collection in the sense of the CA, the CA grants the author of such a work the *exclusive right* thereto and the *exclusive right* to decide whether, when, and how his work is used (Article 9 para. 1 and Article 10 para. 1 CA). This includes, in particular, the right to make copies of the work, such as data carriers, to offer, transfer, and otherwise distribute copies of the work as well as to decide whether (Article 10 para. 2 CA), when, and how the work may be altered or used to create a derivative work or may be included in a collection (Article 11 para. 1 CA).

In the context of data science, this means that copying data which constitutes a work or collection infringes on the exclusivity rights of the author. However, one could consider licensing or buying the copyrights to such data. Moreover, infringements of copyrights could be *justified* (e.g., by consent) or could fall within the scope of a copyright exemption, such as private use in enterprises, public administrations, institutions, commissions, and similar bodies for internal information and documentation (Article 19 CA).

However, it has to be noted that where data is available publicly on the Internet, this does not automatically mean that the potentially existing copyrights have been abandoned or implied consent to copying of such data has been given [a *different view is held by* Weber (2014, p. 22)]. Also, on the Internet it is not always clear whether the person making the work available publicly is actually the right owner. This has to be kept in mind by data scientists, in particular when creating data products, even if they are meant to mainly include or use publicly available data from the Internet.

In addition, also the tools of data scientists or the result of their activities may be protected by copyright, such as computer programs. However, individual algorithms and short computer programs that lack any complexity are not protected under Swiss copyright law (*see* Cherpillod 2012, n. 64).

Finally, there is the question, whether computer-generated works can be subject to copyright protection. Traditionally, the protection of computer-generated works by copyright was not an issue. Computers and computer programs were considered

[3]For example, the Swiss Federal Supreme Court found that the Swiss Compendium of Medicines did *not* reach the required individual character and was not protected by copyright (BGE 134 III 166).

to be merely tools that supported the creative process, very much like a brush in the hands of an artist. But, with the rapid development of machine learning software, a subset of artificial intelligence, some computer programs are no longer only supportive tools; they actually take autonomous decisions with regard to the creative process—free of human intervention (Guadamuz 2017). Therefore, it is subject to debate whether creations generated through machine learning qualify as "intellectual creations" and enjoy copyright protection. A broader concept of copyright protection in this field may help protect the results of the work of data scientists in certain cases.

2.4 Database Right Sui Generis

Furthermore, in the EU, there is another right that may serve to protect certain data. The Directive on the legal protection of Databases (Directive 96/9/EC) provides for an exclusive specific right for database producers. The holder of such database rights may prohibit the extraction and/or re-utilization of the whole or certain parts of a database. The rights are valid for 15 years and are meant to protect the investment of time, money, and effort into such database, irrespective of whether the database is in itself innovative ("non-original" databases). Thus, it applies even if the database is not protected by copyrights. (The Directive harmonized also copyright law applicable to the structure and arrangement of the contents of databases ("original" databases).)

While this sui generis database right covers databases, which would not be protected under regular copyright, one has to keep in mind that it only applies if there has been a qualitatively and/or quantitatively substantial investment in either the obtaining, verification, or presentation of the contents. Moreover, the right only prevents extraction and/or re-utilization of the whole or of a substantial part, evaluated qualitatively and/or quantitatively, of the contents of that database (Article 7 para. 1 Directive 96/9/EC). If only insubstantial parts of the database are concerned, repeated and systematic extraction and/or re-utilization are still prohibited under certain circumstances (Article 7 para. 5 Directive 96/9/EC).

In many cases, the activities of data scientists will not be so extensive as to infringe these rights. However, whenever data scientists obtain databases, this so-called sui generis database right should be taken into account and it should be determined whether it applies to their specific case.

As above with copyrights, the result of the activities of data scientists (or interim results) may fall under the Directive 96/9/EC and enjoy the respective protection. This may help in the commercial exploitation of these results (e.g., as part of a licensing of such database rights to third parties).

2.5 Unfair Competition Law

In Switzerland, the legal protection of such databases is not as clear. Switzerland has not introduced a database right similar to the one of the EU mentioned above. While some databases will be covered by copyright (if the selection and arrangement of the data is an intellectual creation with individual character) even extensive data collections or databases will not constitute a copyrightable work if the respective system is simple. In any case, the copyright protection enjoyed by a collection will be limited to the structure of the database (e.g., selection and arrangement), if the respective data itself is not protected by copyright (Arpagaus 2013, n. 35). Therefore, it is important to examine how other legal concepts may be used to obtain a certain legal protection of such databases:

Article 5 let. c Unfair Competition Act (UCA) provides that it constitutes unfair competition to take the marketable results of work of another person by means of technical reproduction processes without an adequate effort of one's own and to exploit them "as such." The question is whether—and, if so, under what circumstances—this provision could be used to protect at least certain databases.

Databases, even ones that are not protected by copyright, can constitute a "marketable result of work of another person" in the sense of Article 5 let. c UCA. Downloading such a database from the Internet or otherwise copying it constitutes a "technical reproduction process." Consequently, some of the requirements for protection under Article 5 let. c UCA will be met in many cases.

The issue then becomes whether the exploitation of such a database was made "*without an adequate effort*," This will have to be considered with respect to the specific facts on a case by case basis. One will have to take into account a comparison between the effort of the first mover and the effort of the potential infringer—and also, whether the first mover already had an opportunity to amortize its efforts at the time of reproduction and exploitation.

In addition, the marketable results have to be exploited "*as such*." While this term is very vague, the Federal Supreme Court has taken this to mean that it requires not only that the reproduction of the result is made directly, but that it also must be exploited directly. *Direct reproduction* requires that the technical process directly implicates the original, while *direct exploitation* would require a commercial or professional use of the result in competition without an adequate own effort of the infringer.

Often, data scientists will not "take" the entire (or large part) of an existing database or another marketable result of a third party merely by means of a technical reproduction process, but will put more effort into their work, in particular use at least a combination of data. In addition, data scientists will often not directly exploit databases without an adequate own effort. The entire idea of data science is to apply one's own (adequate) effort to a set of data and create a new data product, which obviously goes far beyond the original set of data (Weber 2014, p. 21).

Therefore, Article 5 let. c UCA will often not be applicable in the context of data science. Nevertheless, it is important to keep this provision in mind to be able to avoid its pitfalls on a case by case basis.

2.6 Manufacturing and Trade Secrets

In addition to the provisions outlined above, certain data that data scientists want to use may be protected as manufacturing or trade secrets. While protection by Article 5 UCA as outlined in the section above does not require the data to be secret, Article 6 UCA protects *manufacturing and trade secrets* if they have become known as the result of spying or otherwise have been obtained unlawfully. In such cases, exploiting manufacturing or trade secrets or disclosing them to third parties constitutes unfair competition.

Moreover, the disclosure of manufacturing or trade secrets by a person who is under a statutory or contractual obligation not to reveal such facts, as well as exploiting manufacturing or trade secrets disclosed in such a way, is punishable under Swiss criminal law (Article 162 Swiss Criminal Code).

While data scientists in many cases will not intentionally violate manufacturing or trade secrets, they should still try to make sure that the data they use does not contain and violate such secrets.

3 Data Protection/Privacy

3.1 Background

While it may well be that only companies will survive that rigorously exploit (big) data, one should not forget that data science and data exploitation must not lead to an infringement of privacy rights (Polzer 2013, p. 6). Data protection and privacy are protected by the Swiss constitution as fundamental constitutional rights. Data protection laws are meant to specify the constitutional rights of privacy. Those data protection laws also have to be taken into account in the field of data science.[4]

Swiss data protection law is mainly set forth in the Federal Act on Data Protection of June 19, 1992 (DPA), and the Swiss Federal Ordinance to the Federal Act on Data Protection of June 14, 1993 (DPO). In the EU, the General Data Protection

[4]This contribution is not meant to be a full-blown introduction into data protection/privacy laws and the following will concentrate on a limited number of data protection law issues, which may have particular importance for data scientists.

Regulation (GDPR)[5] has entered into force and will apply as of May 25, 2018. The Swiss DPA is currently under revision and it is expected that it will be strongly influenced by the GDPR, in particular because cross-border data transfers are daily business.[6]

3.2 Personal Data

Swiss data protection laws only deal with the processing of personal data. Obviously, not all data is personal data. Under Swiss law, personal data is defined as "all information relating to an identified or identifiable person" (Article 3 let. a DPA). A person is considered to be identifiable if identification is possible without undue efforts and one has to expect that this will possibly be done (Rudin 2015, n. 10).

While this definition seems clear, there is a large spectrum between data that is clearly connected to an identifiable person and data that cannot in any way be re-identified.

De-identification of data generally is used to denominate a process of "removing or obscuring any personally identifiable information from individual records in a way that minimizes the risk of unintended disclosure of the identity of individuals and information about them" (Nelson 2015, p. 12). Therefore, de-identified data may theoretically still be linked to individuals, for example, using a code, algorithm, or pseudonym.

The definition of "**pseudonymization**" in the GDPR is somewhat different: "*'pseudonymisation' means the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organizational measures to ensure that the personal data are not attributed to an identified or identifiable natural person*" (Article 4(5) GDPR).

Anonymization on the other hand is a process of data de-identification leading to data where individual records cannot be linked back to an individual as they do not include the required translation variables to do so. Consequently, anonymized data, as it is often used in data science, is generally not subject to the DPA. De-identification may also be sufficient to exclude data from the scope of DPA, if the re-identification is not possible without undue efforts or if one does not have to expect that this will possibly be done.

However, data scientists should be aware that the process of anonymization or de-identification of data, which currently constitutes personal data, does, in itself,

[5]Regulation (EU) 2016/679 of the European Parliament and of the Council of April 27, 2016, on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC.

[6]The present contribution is based on the status of legislative proceedings as of February 23, 2018.

constitute the processing of personal data and, thus, is subject to the DPA. Only the result of the anonymization (and possibly of the de-identification) is no longer relevant from a perspective of data protection laws.

Also, there is no guarantee that de-identification and/or anonymization completely precludes re-identification of the data subject. On the contrary: in particular in connection with Big Data, if there is a large amount of data, re-identification of the data subject becomes more likely and possible (Baeriswyl 2013, p. 15).[7] Once such re-identification becomes possible "without undue efforts" and one has to expect that this will possibly be done, the data becomes personal data, and the DPA applies. Consequently, if one has sufficient data to conduct such re-identification, one will have to comply with the DPA (Weber 2014, p. 20). So, while the process of re-identification itself constitutes a data processing that is relevant under the Swiss DPA, one has to be aware that the DPA becomes applicable already at an earlier stage, that is, once re-identification is "possible without undue efforts and one has to expect that this will possibly be done" (Baeriswyl 2014, pp. 50–52). If personal data is generated by accident, nevertheless, Swiss data protection laws would apply. Finally, even if the data scientist provides de-identified and/or anonymized data to third parties, data protection laws will have to be complied with, if the data scientist has to expect that re-individualization will take place (Baeriswyl 2014, p. 53).[8] This is an issue that should be further analyzed in joint research activities conducted by IT specialists and legal scholars.

Thus, the boundary between personal data and other data is somewhat vague, in particular because of the technical developments; data that cannot be re-individualized today may well become related to an identifiable person tomorrow, and, thus, become personal data (FDJP 2016, p. 43).

Consequently, even anonymization or de-individualization of the respective data does not completely exclude that data protection laws will be applicable to the activities of a data scientist. This is true irrespective of whether the data is used only internally in a data product or whether it is visible also externally and irrespective of the effect of the data product on the data subject concerned (e.g., whether you use the data for personalized pricing or to achieve better usability of a software for the data subject).

[7] With regard to de-identification, re-identification, alternative approaches, and use-cases, cf. Narayanan et al. (2016).

[8] In this context one may also point to the US Federal Trade Commission's (FTC) 2012 report *Protecting Consumer Privacy in an Era of Rapid Change: Recommendations for Businesses and Policymakers*, in which the FTC takes the position that "data is not 'reasonably linkable' to the extent that a company: (1) takes reasonable measures to ensure that the data is de-identified; (2) publicly commits not to try to re-identify the data; and (3) contractually prohibits downstream recipients from trying to re-identify the data" (retrieved February 14, 2018, from https://www.ftc.gov/reports/protecting-consumer-privacy-era-rapid-change-recommendations-businesses-policymakers).

3.3 Privacy by Design

Sometimes, legal developments are outpaced by technological developments. Data protection laws try to address this issue by provisions concerning "privacy by design"—the GDPR as well as the draft for a revision of the DPA. Privacy by design is an approach which takes privacy into account already in the phase of designing a product or a data analysis.

The idea of including this principle into the relevant laws is that law and technology should complement each other and that technologies, which already take privacy into account, are necessary to help implement data protection laws (Legislative Message DPA 2017, p. 7029). Technology may be used to enhance data security and, at the same time, the level of protection of personal data (Kiss and Szöke 2015, p. 323).

In the draft to the revised DPA, the principle requires that technical and organizational measures have to be set up in order for the data processing to meet the data protection regulations. It has to be considered from as early as the planning stage. The purpose is to achieve that systems for data processing are engineered (from a technological and organizational perspective) from the beginning in a way that they comply with data protection principles (Legislative Message DPA 2017, p. 7029).

While this is rather vague, there are already certain reports and principles that can be used when trying to determine what "privacy by design" requires. Some guidance can be found, for example, in the following "7 foundational principles" of privacy by design (Cavoukian 2011):

1. Proactive not reactive, preventive not remedial

   The privacy by design approach aims to identify, anticipate, and prevent privacy invasive events before they arise. It does not wait for privacy risks to materialize, nor does it offer remedies in case a privacy breach occurs.

2. Privacy as the default

   The default settings deliver the maximum degree of privacy. No action is required by the individual in order to protect their privacy.

3. Privacy embedded into design

   Privacy is integral to the system, without diminishing functionality. It becomes an essential component of the core functionality being delivered.

4. Full functionality—positive-sum, not zero-sum

   Privacy by design accommodates all legitimate interests and objectives in a positive-sum "win-win" manner. It avoids the pretense of false dichotomies, such as privacy vs. security, demonstrating that it is possible to have both.

5. End-to-end-security—full lifecycle protection

   Privacy must be protected by strong security measures throughout the entire lifecycle of the data involved; from the cradle to the grave.

6. Visibility and transparency—keep it open

   The data subject is made fully aware of the personal data being collected, and of the purpose(s). Moreover, the component parts and operations remain visible and transparent.

7. Respect for user privacy—keep it user-centric

   Privacy measures are consciously designed around the interests and needs of individual users.

In addition, the European Union Agency for Network and Information Security has addressed the issue in its report "Privacy and Data Protection by Design—from policy to engineering," which tries to bridge the gap between the law and the available technologies. It can also provide further insight into this issue and is certainly a good reference for data scientists.

3.4 Privacy by Default

While "privacy by default" is listed as one of the "7 foundational principles" of privacy by design above, this principle is also explicitly mentioned in the GDPR as well as the draft for a revision of the DPA.

The respective legal provisions require that it is ensured by suitable settings that by default only such personal data are processed that are required for the respective purpose of the processing. The "default setting" is the setting that is automatically given or applied to a software application, computer program, or device, if not altered or customized by the user.

In other words, the respective data processing should—as a default—be as privacy friendly as possible, except if the data subject changes the default settings (Legislative Message DPA 2017, p. 7030), for example, to obtain additional functionalities. Such settings have to enable the data subject to make its own choices concerning privacy to a certain extent.

3.5 Automated Individual Decisions

Another provision in data protection law which could substantially affect the activities of data scientists concerns "automated individual decisions." The GDPR as well as the draft for a revision of the DPA restrict automated individual decision making under certain circumstances. The GDPR states that the "data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her" (Article 22 para. 1 GDPR). The draft for a revision of the DPA provides that a data subject has to be informed "of a decision which is taken exclusively on the basis of an automated processing, including profiling, and

which has legal effects on the data subject or affects him significantly" (Legislative Message DPA 2017, pp. 7056 et seq.).

This may lead to substantial difficulties in data science, in particular in cases where individual decisions are taken by algorithms. However, the GDPR only covers decisions based "*solely*" on automated processing while the term used in the draft for a revised DPA is "*exclusively.*" So how should the term "solely" in the GDPR or "exclusively" in the draft to the revised DPA be interpreted?

While one could argue that it is already sufficient if a human was included at the end of the process to formally "make the decision," this would defy the purpose of the legal provisions. Rather, it should only be considered that the decision is not based *solely* or *exclusively* on automated processing, if a person "actively exercises any real influence on the outcome of a particular decision-making process" (Bygrave 2000) and actively assesses the result of the automated processing before this person takes the decision also formally.

There are, however, also exceptions to this requirement. One important exception is that the provisions will not apply if the automated process was based on the data subject's explicit consent. According to the GDPR, the data subjects must be provided with information not only of the *existence* of such automated decision making, but also of the *logic involved* and the *significance* and *envisaged consequences* of such processing for the data subject (Article 13(2)(f) GDPR), which also is a necessary foundation for a valid consent.

However, explaining (and understanding) what goes on, for example, in a neural network in terms of a generated outcome (i.e., why is this case decided that way?) is a difficult task, even for an expert (cf. Dong et al. 2017; Stock and Cisse 2017). It will be substantially more difficult to try and explain such issues (or other algorithms) to an average data subject. In particular, if one cannot easily trace the precise path of a neural network to a final answer, the description of automated decisions is open to interpretation. This difficulty may also affect the issue of validity of a data subject's consent, since such consent not only has to be freely given, specifically and unambiguously, but also has to be made on an "*informed*" basis (Article 4 (11) GDPR). And even in cases of valid consent, the data subjects will still have to be informed and the data subjects will have the (1) right to obtain human intervention; (2) right to express their point of view; (3) right to obtain an explanation of the decision reached; and (4) right to challenge the decision (Recital 71 GDPR).

Since algorithms are an important means of governing data streams, assessments of how an automated decision will affect the data subject may have to be made on a regular basis. However, this seems to be an impossible reality, should automated decisions become the norm (Naudits 2016).

3.6 Self-Regulation

In addition, the GDPR as well as the draft for a revision of the DPA to a certain extent provide that self-regulation shall have some legal effects. Self-regulation is

generally considered to be the regulation in a field, for example, an industry, by its own members often using, for example, standards or codes of conduct (such as the Code of Conduct of the German insurance industry),[9] as opposed to legislation set forth by the government.

There are various kinds of self-regulation (i.e., regulation by private, non-governmental entities). Autonomous self-regulation is solely based on the initiative of the private players, while initiated self-regulation is based on private activities initiated by governmental impulses. In some cases, government may even try to steer self-regulation and, thus, achieve a regulated autonomy (regulated self-regulation), such as in the case of data protection.

While there are some disadvantages to self-regulation (e.g., lack of transparency, democratic deficit, putting private and commercial interests over public interests), some of these disadvantages can be addressed in regulated self-regulation. Moreover, self-regulation can also have many advantages: It can avoid further governmental interventions and legal regulations; self-regulation generally is closer to actual practice and the involved parties can introduce their technical expertise. In addition, self-regulation generally is more flexible than governmental regulation and it is easier to react to (technical) changes. Finally, self-regulation can contribute to the good reputation of the field concerned (Widmer 2003, pp. 20–22).

The GDPR as well as the draft for a revision of the DPA introduce the possibility of regulated self-regulation. Article 40 GDPR provides that associations and other bodies representing categories of controllers or processors may prepare codes of conduct for the purpose of specifying the application of the GDPR in certain aspects. Such codes would then have to be submitted to the supervisory authority, which shall approve it, if it complies with the GDPR. In cases where a code of conduct concerns processing activities in several member states, the supervisory authority must, before approval, submit it to the European Data Protection Board for an opinion. If it approves, the European Commission must review the code and, if it also approves, publish it.

Such codes of conducts can be used not only to facilitate cross-border data transfers, but also help to set forth and demonstrate compliance, in particular with regard to security risks of data processing (*see*, e.g., Articles 24, 28, and 32 as well as Recitals 77 and 81 GDPR). Codes of conduct are particularly fit to address legal questions for specific industries, but also other questions of data protection, such as the requirements of privacy by design or privacy by default in specific fields (Bergt 2016, p. 671).

The revision of the DPA goes less far than the GDPR in this aspect. It provides that professional and business associations whose statutes entitle them to defend the economic interests of their members, as well as federal bodies, may submit a code of conduct to the supervisory authority. Thereupon, the authority shall comment on the submitted code and publish its opinion. However, note that the interested parties

[9] Gesamtverband der Deutschen Versicherungswirtschaft (2012). This code is currently under revision due to GDPR adaptations.

cannot derive any rights from a positive opinion or a waiver of an opinion. Nevertheless, in case of a positive opinion from the supervisory authority, it can be assumed that behavior in line with the submitted code of conduct does not entail any administrative measures (Legislative Message DPA 2017, pp. 7034–7035).

Taking this possibility into account, it may make sense, for example, for associations in the field of data science to consider initiating and/or participating in self-regulation projects concerning certain issues, which affect their activities. Not only will this give them the possibility to more closely have an effect on the regulation which concerns them and to mitigate the risks from vague legal provisions, it may also give them the possibility to more quickly influence how (technical) changes of their field are approached from a legal perspective and may even help prevent further sector-specific data protection laws.

4 Regulatory Aspects

Data science obviously does not take place in a vacuum. The application of data science to particular fields and the creation of new data products from a legal perspective will also have to take into account the context of the specific industry data science is applied to. In many industries, there are substantial regulatory requirements that have to be met, not to mention sector-specific data protection provisions to be taken into account. Creating new products without concern to such regulatory frameworks may result in commercial nonstarters or expensive rectifications before commercialization.

Space constraints hinder us from exhaustively listing and explaining such regulatory frameworks for all fields to which data science may be applied. Suffice it to say that among many others, the following fields are particularly regulated and such sector-specific laws will have to be taken into account: banking and finance, insurance, pharmaceutical sector, health care, and telecommunications.

5 Conclusion

In this chapter, we have outlined a number of legal issues that can affect the activities of data scientists.

It seems clear that data carriers should be treated as goods from a legal point of view, and that copyright protects some data or data collections. Also, in the EU databases are to a certain extent protected by a sui generis right (in Switzerland no such right exists) and, in addition, unfair competition law also prevents certain abuse of data. Data scientists should be aware that the legal discussion in this respect has not yet caught up and is still open. Moreover, data scientists should always consider how rights to data affect their activities, for example, whether they have obtained the

necessary rights concerning the data they work with and how such rights may affect the results of their work.

Data protection law is certainly a legal field that data scientists should be aware and have some knowledge of. While many activities of data scientists will not necessarily involve personal data, the risk of re-identification—and its impact on the qualification of data as "personal data"—must always be considered. Among many others, "privacy by design" and "privacy by default" are some of the provisions that have to be taken into account already early in the process of developing a data product. In addition, rules concerning "automated individual decisions" often are of concern to data science. However, it remains to be seen how they will play out in the future. Self-regulation may be one way to address some of the vagueness and uncertainties of data protection laws from the perspective of specific fields of data science and—if effective—may also help to mitigate the legal risks and preclude the potential perception of a need for further sector-specific legislation.

Finally, data scientists should always remain aware that the application of data science to specific fields may also lead to the application of certain industry-specific regulation. It is important that data scientists obtain at least a broad overview of such industry-specific laws and consider their effect on their activities and potential data products already at the beginning of a project.

References

Arpagaus, R. (2013). Commentary on Art. 5 UCA. In R. Hilty & R. Arpagaus (Eds.), *Basler Kommentar, Bundesgesetz gegen den Unlauteren Wettbewerb (UWG)*. Basel: Helbing Lichtenhahn.

Baeriswyl, B. (2013). "Big Data" ohne Datenschutz-Leitplanken. *Digma, 13*(1), 14–18.

Baeriswyl, B. (2014). Big Data zwischen Anonymisierung und Re-Individualisierung. In R. H. Weber & F. Thouvenin (Eds.), *Big data und Datenschutz – Gegenseitige Herausforderungen* (pp. 45–59). Zürich: Schulthess.

Benhamou, Y., & Tran, L. (2016). Circulation des biens numériques: De la commercialisation à la portabilité. *Sic!* (11), 571–591.

Bergt, M. (2016). Verhaltensregeln als Mittel zur Beseitigung der Rechtsunsicherheit in der Datenschutz-Grundverordnung. *Computer und Recht*, (10), 670–678.

Bygrave, L. (2000). Automated profiling, minding the machine: Article 15 of the EC data protection directive and automated profiling. *Computer Law and Security Review, 7*(4). Retrieved February 14, 2018, from http://www6.austlii.edu.au/cgi-bin/viewdoc/au/journals/PLPR/2000/40.html

Cavoukian, A. (2011). *Privacy by design, The 7 foundational principles, Implementation and mapping of fair information practices*. Retrieved February 14, 2018, from https://www.ipc.on.ca/wp-content/uploads/Resources/pbd-implement-7found-principles.pdf

Cherpillod, I. (2012). Commentary on Art. 2 CA. In B. Müller & R. Oertli (Eds.), *Stämpflis Handkommentar SHK, Urheberrechtsgesetz (URG)* (2nd ed.). Bern: Stämpfli.

Dong, Y., Su, H., Zhu, J., & Bao, F. (2017). *Towards interpretable deep neural networks by leveraging adversarial examples*. Retrieved February 14, 2018, from https://arxiv.org/abs/1708.05493

Eckert, M. (2016a). Digitale Daten als Wirtschaftsgut: Besitz und Eigentum an digitalen Daten. *SJZ, 112*(11), 265–274.

Eckert, M. (2016b). Digitale Daten als Wirtschaftsgut: Digitale Daten als Sache. *SJZ, 112*(10), 245–249.

European Commission. (n.d.). *The EU copyright legislation*. Retrieved February 14, 2018, from https://ec.europa.eu/digital-single-market/en/eu-copyright-legislation

FDJP. (2016). *Erläuternder Bericht zum Vorentwurf für das Bundesgesetz über die Totalrevision des Datenschutzgesetzes und die Änderung weiterer Erlasse zum Datenschutz*. Retrieved February 14, 2018, from https://www.bj.admin.ch/dam/data/bj/staat/gesetzgebung/datenschutzstaerkung/vn-ber-d.pdf

Flückiger, A. (2013). L'autodétérimination en matière de données personnelles: un droit (plus si) fundamental à l'ère digitale ou un nouveau droit de propriété? *AJP, 10*(6), 837–864.

Gesamtverband der Deutschen Versicherungswirtschaft. (2012). *Verhaltensregeln für den Umgang mit personenbezogenen Daten durch die deutsche Versicherungswirtschaft*. Retrieved February 14, 2018, from https://www.gdv.de/resource/blob/23938/8db1616525e9a97326e2f2303cf42bd5/download-code-of-conduct-data.pdf

Guadamuz, A. (2017). Artificial intelligence and copyright. *WIPO Magazine* (5), pp. 14–19.

Kiss, A., & Szöke, G. L. (2015). Evolution or revolution? Steps forward to a new generation of data protection regulation. In S. Gutwirth, R. Leenes, & P. de Hert (Eds.), *Reforming European data protection law* (pp. 311–331). Dordrecht: Springer.

KPMG & Bitkom Research. (2016). *Mit Daten Werte Schaffen – Study 2016*. Retrieved February 14, 2018, from https://home.kpmg.com/de/de/home/themen/2016/06/mit-daten-werte-schaffen.html

Legislative Message DPA. (2017). *Botschaft zum Bundesgesetz über die Totalrevision des Bundesgesetzes über den Datenschutz und die Änderung weiterer Erlasse zum Datenschutz* (BBl 2017, pp. 6941–7192).

Narayanan, A., Huey, J., & Felten, E. W. (2016). A precautionary approach to big data privacy. In S. Gutwirth, R. Leenes, & P. De Hert (Eds.), *Data protection on the move* (pp. 357–385). Dordrecht: Springer.

Naudits, L. (2016, August 2). *The right not to be subject to automated decision-making: The role of explicit consent*. Retrieved February 14, 2018, from https://www.law.kuleuven.be/citip/blog/the-right-not-to-be-subject-to-automated-decision-making-the-role-of-explicit-consent

Nelson, G. S. (2015). *Practical implications of sharing data: A primer on data: Privacy, anonymization, and de-identification* (Paper 1884–2015). Chapel Hill, NC: ThotWave Technologies. Retrieved February 14, 2018, from https://support.sas.com/resources/papers/proceedings15/1884-2015.pdf

Oxenham, S. (2016). Legal maze threatens to slow data science. *Nature, 536*(7614), 16–17.

Polzer, G. (2013). Big Data – Eine Einführung. *Digma, 13*(1), 6–10.

Ritter, J., & Mayer, A. (2018). Regulating data as property: A new construct for moving forward. *Duke Law & Technology Review, 16*(1), 220–277.

Rudin, B. (2015). Commentary on Art. 3 DPA. In B. Baeriswyl & K. Pärli (Eds.), *Stämpflis Handkommentar SHK, Datenschutzgesetz (DSG)*. Stämpfli: Zürich/Bern.

Stadelmann, T., Stockinger, K., Braschler, M., Cieliebak, M., Baudinot, G., Dürr, O., & Ruckstuhl, A. (2013). *Applied data science in Europe, challenges for academia in keeping up with a highly demanded topic*. Presented at the European Computer Science Summit 2013, Amsterdam, Netherlands. Retrieved February 14, 2018, from http://www.informatics-europe.org/images/ECSS/ECSS2013/slides/ECSS2013-Stadelmann-paper.pdf

Stock, P., & Cisse, M. (2017). *ConvNets and ImageNet beyond accuracy: Explanations, bias detection, adversarial examples and model criticism*. Retrieved February 14, 2018, from https://arxiv.org/abs/1711.11443

Thouvenin, F. (2017). Wem gehören meine Daten? Zu Sinn und Nutzen einer Erweiterung des Eigentumsbegriffs. *SJZ, 113*(2), 21–32.

Weber, R. H. (2014). Big data: Rechtliche perspektive. In R. H. Weber & F. Thouvenin (Eds.), *Big Data und Datenschutz – Gegenseitige Herausforderungen* (pp. 17–29). Zürich: Schulthess.

Widmer, M. (2003). *Das Verhältnis zwischen Medienrecht und Medienethik*. PhD thesis, University of Zürich, Stämpfli, Bern.

Chapter 6
Risks and Side Effects of Data Science and Data Technology

Clemens H. Cap

Abstract In addition to the familiar and well-known privacy concerns, there are more serious general risks and side effects of data science and data technology. A full understanding requires a broader and more philosophical look on the defining frames and on the goals of data science. Is the aim of continuously optimizing decisions based on recorded data still helpful or have we reached a point where this mind-set produces problems? This contribution provides some arguments toward a skeptical evaluation of data science. The underlying conflict has the nature of a second order problem: It cannot be solved with the rational mind-set of data science as it might be this mind-set which produces the problem in the first run. Moreover, data science impacts society in the large—there is no laboratory in which its effects can be studied in a controlled series of experiments and where simple solutions can be generated and tested.

1 Introduction

Data science has been defined by Braschler et al. (2019) as the *unique blend of skills from analytics, engineering, and communication, aiming at generating value from the data itself*. Data technology may be regarded as the *method of collecting data and deducing empirical and statistical models and making decisions thereon* with the help of algorithms. We focus on two aspects: Personal data, where data science affects the privacy of the connected persons, and model deduction, which may influence our way to do science and to make decisions. We perceive data science as an applied science, as a technology, and as a decision mechanism. Thus, technology assessment seems a reasonable thing to do. Risks connected with personal data and social "transformatory" risks, that is, resulting in changes of society, seem the foremost.

C. H. Cap (✉)
Universität Rostock, Rostock, Germany
e-mail: clemens.cap@uni-rostock.de

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_6

This task is difficult and urgent. Data technology is not yet broadly deployed in society, therefore the possible risks have not yet shown up in a larger number of concrete cases. They are not yet well understood and thus seem frightening. In nuclear technology or aviation, over 60 years of research and a tradition of detailed accident analysis produce a different situation and allow more clear statements. Nevertheless, assessment of data science is important and must not be delayed: Data technology is a pervasive technology and risks of a ubiquitous infrastructure are difficult to avoid should there be significant negative side effects. Data science infrastructure will soon have been integrated tightly into many products, workflows, and systems. While energy production can be, and has occasionally been, converted from nuclear to non-nuclear technology, data science applications are difficult to stop once they have been deployed, due to their universal integration into technical systems and social processes. The debate is emotionalized, risks affect everybody, leakage and data theft scandals heat up the discussion, cyber-wars and surveillance allude to fear, and much money can be made by industry.

Finally, it is not clear how to conduct systematic data science risk analysis. Side effects are not of a biological nature and cannot be studied in a lab; polls provide reliable answers only after (irreversible?) social change has been completed; philosophical debates on human values may be appropriate but employ methods which, unfortunately, rarely are taken seriously by the core target groups: data scientists, company owners, and policy makers.

2 Main Risks and Side Effects

Security and privacy issues are the foremost category of problems commonly associated with data science. As they are well discussed we shall only provide a brief overview. The Whitehouse Report on Big Data[1] provides a wealth of case studies in the areas of access to credit and employment, higher education, and criminal justice and connects them with possible though avoidable flaws in the big data process. Other authors raise more fundamental questions[2] or focus on the possibility of classification (Dwork and Mulligan 2013) and a resulting loss of different perspectives, individual autonomy, and democratic functioning of society.

The first trouble is *abuse of personal data.* It frequently leads to decisions which are not in the interest of the person involved. A well-known example is travel booking. Depending on the web browsing history, cookie pricing attempts to make the most profit from a customer. Every available information on social and financial stratum, past spending habits, and even computer brands is translated to offers and booking modalities which are optimal for the selling agency. Customers who do not complete the booking are followed through social media, search engines, and even email. Service providers employ all kinds of sensors, from fitness trackers

[1] See Executive Office of the President (2016).

[2] For example, Kree and Earle (2013).

to car black-boxes, to gather information on how to best make money from the observed person. Even smart phones owned by the customer are employed as mobile selling platforms of their manufacturers. To obtain the required platform control, software technologies are used to set up walled gardens preventing every escape of their owner. See Cap (2016) for a wealth of further examples.

Often *data theft* increases the risks of data abuse. We also face the problem of *incorrect data* and *data stickiness*, that is, situations where wrong or no longer valid data affect the life of an individual, which is unable to dissociate itself from data collections or even data errors in the past.

Given the wealth of deductions which can be made on persons from their data, some critics even question whether the dignity and autonomy of man would allow for a concept of a third party "*owning*" personal data of somebody else (Hofstetter 2016).

The *digital panopticon* is a further aspect, which originated in the surveillance debate. It leverages a thesis originally from Bentham (Warriar et al. 2002): A person feeling watched by an anonymous observer is likely to adhere to the ethical standards fantasized for the observer. This concept is further reflected in the mindset of Google whose former CEO Eric Schmidt suggested that "[if] you [had] something that you [didn't] want anyone to know, [. . .] you shouldn't be doing it in the first place." According to critics this statement demonstrates a complete lack of understanding of the concept of privacy (Esguerra 2009).

The **infrastructure risk** points out an important modality of data technology. Applications require a wide deployment of data sensors. Internet of Things experts speculate on more than 50 billion networked devices by 2020 (Hosain 2016). Data technology penetrates workflows, decision processes, and business plans. It promises convenience and optimized decisions. Ultimately, society finds itself in circumstances so nicely described by the sorcerer's apprentice.[3] When the technology has been deployed, it is extremely difficult to stop it or even reduce it—for technical, social, and economic reasons. Even if data science might guarantee the best of all possible worlds, we should be careful as a society when setting up an infrastructure where this final result is granted without a possibility of a later intervention.

Example: On a recent plane travel, the author was asked to have his boarding pass scanned as precondition for buying bottled water. Leaving aside duty-free and tax requirements, which could have been satisfied by manual ticket inspection or a higher price, the infrastructure problem was that the sales assistant could not even open the drawer of the cash register without a scan of personal data. Recently the Amazon Echo voice device behaved as the literal sorcerer's apprentice: A 6-year-old Texan girl was chatting with the device of her parents and asked it to order a dollhouse—which promptly was delivered. When the story was reported by a TV station, the reporter said: "I love the little girl, saying 'Alexa order me a dollhouse'." Devices in many a viewer's living room heard this command and placed orders for dollhouses (Nichols 2017).

[3]Famous German poem "Der Zauberlehrling" by Johann Wolfgang von Goethe, in which the control on an initially helpful spirit is hopelessly lost.

A disruptive change of science: In traditional science, theoreticians develop mental models which then are put to the test by observation. The models are cognitive constructions of the mind and do not constitute "reality"—although the act of confusing models with reality often helps a scientist to improve models: The earth is not flat but until the development of better astronomic instruments this model was helpful; the earth is not a sphere either, but only advanced questions really required the model of a rotational ellipsoid. The earth is not an ellipsoid either. All these mental models are "wrong" but helpful in the sense that they provide the physicist with constructions for "understanding" the world. Ultimately, in quantum physics the attempts to model observations by "machinery" are believed to fail. We recall Richard Feynman: "If you think you understand quantum mechanics, you don't understand it." Still the physicists' minds heavily and successfully use imagined machinery[4] since these cognitive tools fit the human mind.

Data science replaces this machinery by empirically validated models. In the optimal scenario, it drops theory and delivers the "best" numeric description of billions of experiments. This approach is hard to beat empirically. Why bother for an explanation if black-box predictions match myriads of experiments? This may be particularly attractive in complex system science such as medicine. Why bother to develop an explanatory description of a disease if a computer can diagnose and treat it much better?

Example: Anderson (1989) describes how a neural network can learn to control an inverted pendulum without prior knowledge of dynamics. The algorithm produced sets of real numbers as connection weights which interpolate complex functions with sufficient precision—it does not produce any "understanding" of the "learned" problem. These weights heavily depend on the structure of the network and on the randomization throughout the learning process.

Data science produces a solution for a problem (e.g., treating a disease), an abstract mathematical model for a complex object (e.g., an inverted pendulum), and in a few cases even additional insight into correlations and statistical mechanisms—it usually does not provide the mental models a human will use for "understanding" a system. We can argue that our mental models are incorrect, since modelling employs complexity reduction. However, the human mode of understanding our world and communicating about it is exactly in those "wrong" but vivid and demonstrative mental models which are close enough to the human mind. A disruptive transformation of science which replaces the human researcher by an algorithm is not desirable. The offer is, of course, tempting, but maybe we should reject it.

A GIS research project at the University of Zurich demonstrates a more refined approach (Schönholzer 2017): Studies indicate that repeated use of navigation systems weakens the sense of orientation of the user. Thus, the group now studies

[4]Feynman also acknowledges this. "I see vague pictures of Bessel functions [. . .] and dark brown x's flying around. And I wonder what the hell it must look to the students." See Root-Bernstein and Root-Bernstein (1999).

how the interaction of a user with a navigation system should be restructured to avoid superfluous use of the system and prevent further degradation of human capabilities. An interesting paradox arises: The more we develop helpful tools the more our natural skills degenerate. This is well known from other areas: An increasingly immobile life-style, for example, calls for regular compensation on the treadmill. With an increasing number of smart assistants and with data science applications taking over our decisions, we will face such paradoxes more and more often. Which mechanisms will prevent our cognitive decline? How reasonable are technical tools which, when used, destroy our abilities? Can this effect be counteracted successfully, as the Zurich GIS research project intends to do? Would it be more reasonable to use these tools less often? Do we have the necessary self-discipline? What protects us on a larger scale from first using such tools, then showing signs of degeneration, thus requiring such tools and finally becoming dependent on machines and associated business cases to manage our lives?

A **replacement of humans by algorithms** is closer than we might be aware of. We witness the trend in autonomous cars; financial trading already is dominated by algorithms and not only the allocation (Park et al. 2014) but also the selection (Miller-Merell 2012) of human resources will soon be taken over by computers. The latter means computers will decide which humans get a job, where and why. The pattern of replacing humans by machines was seen in the first industrial revolution, where it was for physical tasks. While a replacement of humans for repetitive, routine, dangerous, and boring tasks seems fine, we might cross a fundamental boundary when human *deciders* are systematically replaced by algorithms. Even if all involved parties benefit from better decisions, the issue at stake is the loss of core values of meaning of human life. While not everybody might agree that meaningful work is one of the core purposes of our existence, possible alternative worlds are not very attractive. Scenarios comprise dystopias such as "Brave New World" by Aldous Huxley, where the purpose of life is reduced to consumption and instantaneous satisfaction of needs, or more recently and drastically by the science fiction movie series Matrix, where humans are the mere appendix of a world governed by machines. Even if the outcome were a true paradise: Which effects would drive human evolution and prevent degeneration? What would we enjoy in our lives when it is no longer reward or success that sweetens labor? Can mankind exist for long being served by robots, regularly aroused by stimulating drugs, a kind of "soma" as described by Huxley?

Reverse and meta risk assessment: A thorough technology assessment also raises the following questions: What could possibly go wrong if risks are not analyzed correctly? What if they are communicated incorrectly to deciders or to the general public? What if they are perceived as larger than they "really" are?

A negative image of data science in public primarily affects the data science profession; there may be consequences for the acceptance of data technology products at large; ultimately, regulatory and legislative processes may damage the industry as such. It is important to recognize the differences between a "true" risk evaluation and a debate on the perceived risks (namely, negative *image*).

There certainly is the risk of rejection of data science due to a perceived abuse. It affects society at large through the loss of possible benefits. The trend to data avoidance or to a more sparing use of data[5] is promoted by privacy activists and may lead to insufficient data bases and incorrect models. The discussion therefore must also focus on the opposite question: May an individual claim the right to withhold data for research, thereby damaging the right of society to deduce possibly important results on these data? The Charta of Digital Rights[6] gives an affirmative answer to this question in Article 11(2).

Chosen ignorance is also an aspect to be discussed as part of a reverse risk assessment. Although being very skeptical toward possible side effects and naïve data science enthusiasm, the author neither considers nor suggests complete abstinence from data science. The famous quote of nuclear bomb physicist Edward Teller can guide us (Shattuck 1997): "There is no case where ignorance should be preferred to knowledge—especially if the knowledge is terrible." The allusion to nuclear weapon technology is not an over exaggeration of the author: Hannes Grassegger and Mikael Krogerus (Grassegger and Krogerus 2016) use this metaphor to speculate on the impact of data science on Brexit and 2016 US presidential elections. They cite[7] a psychologist on his application of data science to psychometric data: "I did not build the bomb. I only pointed out its existence." This is similar to the apologetic position taken by some physicists toward the discovery of nuclear chain reactions as basis for atomic bombs.

Data science applied incorrectly: In this article we presume that data science is done correctly and will not consider the risks of bad data science. They are said to comprise wrong results, bad analytics, bad data, and associated costs (Marr 2015).

3 Important Aspects

Data science and its core applications can be described as technology of optimization. Its ultimate vision is irresistible: *Observe everything, determine the best model automatically and provide us with the optimal answer to our decision problems.*

Deep Blue[8] is an impressive success of artificial intelligence. The system consistently wins chess against human opponents. While it provides for the pride of its programmer, ultimately it destroys the magic of the game of chess.

IBM's Watson is known as winner of the game-show Jeopardy! The impression this success made on the public amounts to a framing error, since of course, but

[5]This can be expressed more precisely in German with the hard to translate terms of "Datensparsamkeit" and "Datenvermeidung".

[6]Charter of Digital Fundamental Rights of the European Union. See https://digitalcharta.eu/wp-content/uploads/2016/12/Digital-Charta-EN.pdf

[7]English translation by the author.

[8]See Wikipedia article on Deep Blue: https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer)

contrary to wide-spread belief, success in Jeopardy! is not so much connected with "knowledge" or even cognitive reasoning but rather with mere storage of many facts and clever engineering. Game-shows where single line answers must be given in a short period of time and decide on winning or losing dangerously misrepresent the ability to store facts as "knowledge" or "intelligence." The weakness of Watson's statistical reasoning is demonstrated impressively by its famous mistake that Toronto was a city in the USA (O'Connor 2011), although formally it is not completely wrong as there are several Torontos in the USA.

Watson, of course, is capable of "reading" medical research papers at speeds much higher than a human doctor. Direct comparisons with human specialists provide promising headlines for patients, especially when it comes to rare diseases or to overlooking symptoms (Galeon 2016). Claims, however, that Watson soon might be the best doctor in the world (Friedman 2014) are dangerously misleading. We are reminded of the famous surprise of Weizenbaum (1976) when psychologists started to discuss therapeutic benefits of his ELIZA program, which was intended as a study in pattern matching and later was attributed human-like feelings by observers.

We argue that the debate is not about better data science, smarter algorithms, and faster processors but that it is affected by a fundamental framing error. From a doctor, a patient also requires support in sufferings, pain reduction, moral support, sharing in the desperate feelings, or other forms of human help, which cannot be outsourced to a machine.

Contemporary medicine, despite impressive success, already gets this wrong often by reducing patients to columns of data, to coded diagnoses, and to amounts of medicine to be taken or operations to be done. The human part of the helping profession gets lost over its scientific success. As the medical system happily makes money with this framing error, it remains uncorrected. The process of dying becomes less and less visible in a world of single-person households. The gap widens between the original task of medicine (to help patients in their sufferings) and sad effects of optimization (to provide legal proof that everything possible has been done and was properly billed; to prolong the life of the patient irrespective of personal wishes and life quality). While the topic is more delicate than our short and one-sided perspective, we must acknowledge the problem and the ongoing public debate on the issue.[9] We witness a serious side effect caused by the social and economic reactions to a one-sided scientific approach, which does not solve the original problem as well as its scientific proponents claim.

Contemporary data science may be regarded figuratively as the endeavor to close the gap between the best treatment for cancer and Watson calling Toronto a city in the USA. We shall consider the philosophical position that this attempt is futile at its roots for other than data science reasons. The remainder of this section provides some arguments for this end.

[9] See, for example, Borasio (2016) or Thöns (2016).

4 The Battle Field: Individual Freedom Versus Institutional Optimization

In a world of global optimization, the concept of freedom becomes meaningless. The model of human existence (grow up, educate yourself, find your place in society, provide some meaningful service to your peers, learn to cope with the inevitable sufferings, and die a dignified death) is destroyed. The data scientific endpoint of human evolution is the wheel chair where a neuro interface reads out the mental states of the occupant and provides all the decisions and actions, food and drugs, movement and entertainment, guaranteeing the "best" possible world for a user, who has never met the challenges of his ancestors, who has not learned to cope with difficulties, who has never met the bitter sweet teaching of failures due to an ever-optimized life.

Of course, data scientists are not working explicitly toward this dystopian target! The mechanism is more delicate and well-illustrated by a thought of C. S. Lewis (1972): "Of all tyrannies, a tyranny sincerely exercised for the good of its victims may be the most oppressive. It would be better to live under robber barons than under omnipotent moral busybodies." However, they construct decision algorithms fostering the illusion of "the best."

The debate on digital nudging and the political process ("selling" the "best" options to the constituents without debate as being without alternative[10]) demonstrates that such a development already is taking place in the political arena. The economic sector is more advanced. The business case of the free and informed individual, negotiating the best deal on a level playing field, has been lost. The consumer faces an anonymous digital opponent, which [sic[11]] knows his or her habits, preferences, past choices, financial and mental capabilities, friends and likes, and more. A user interface and a choice of language which is optimized down to minuscule psychometric details influences the emotions and optimizes the maximum financial benefit which can be made from this user. There is no possibility for the individual to deal fairly with an opponent which "owns" billions of psychometric profiles of past shopping interactions and employs optimized persuasion technologies. For example, a project at the University of Liechtenstein aims at discovering those design modifications in an online poll which are best to produce a specific bias in the poll.[12] There is no economic incentive to change this problematic trend.

Our digital future can be described as a feudalistic society where the owners of the "land" (data and algorithms) are the landlords and the data subjects work as their slaves. In the age of enlightenment, Kant taught us to use our own minds and ultimately reject undue dominance over our thinking. In the age of data processing

[10] An astonishingly large number of proposed legislation in the German Bundestag contains the phrase: "Alternatives: None". Decisions are no longer open to democratic debate in a mindset of optimization.

[11] "Which" and not "who": The opponent is a machine, algorithm or web portal, not a human.

[12] See https://www.uni.li/de/thema/information-systems/digital-nudging/digital-nudging-1

we urgently need to regain control over our own data in order not to end up as digital slaves of algorithms and a few anonymous institutions controlling them to our "benefit" (Cap 2017).

Attempts for solutions come in different degrees of practicality. A philosophical approach will promote a new age of digital enlightenment. It will comprise a renewed understanding of the value of freedom and of the importance of liberal-minded ideologies (in the European interpretation of the word, not in the sense of US politics). While academically appealing, this is completely insufficient for practical purposes and needs further implementation in education, legislation, and political measures. Some approaches are outlined by Cap (2016). An important aspect may be to educate the consumer that it is not in his or her monetary interest to offer personal data to companies, since every information on the consumer gets translated into profit-maximization strategies. For example, in B2C commerce the mechanism of cookie pricing makes the ultimate price dependent on search histories, product interests, and past shopping activities of the consumer. If this abuse of asymmetric information relationships between buyer and merchant is made more transparent to the public, market, legal, and governance mechanisms might produce a buyer reaction which reduces this abuse. Manifestos and chartas[13] and similar activities may further raise awareness.

5 The Mistake: Choice of an Incorrect Frame

Framing is the process of selecting a mindset for the semantic interpretation of a concept. The choice of a frame is at the core of every evaluation. For example, taxes may be framed as "heavy burden" or as "valuable contribution to society." The art of "convincing" or, in an alternative frame, "manipulation" often boils down to the choice of a frame (Wehling 2016) suitable to the specific intentions.

Frames which are commonly used to define science may be the naïve "finding out a so-called truth" or, more elaborated, "falsifying hypotheses." Data science technologies and their applications may be described in the frame of an "empirically validated optimal choice." This framing provides a setting which can never, rationally, be rejected. Why would anyone dislike what is best[14] for him? A rejection seems particularly absurd when it is based on empirical evidence which, rooted in world wide data collection, cannot realistically be falsified; when a margin of statistical error can be provided, and is sufficiently small; and when the decision is made by a computer which, per wide-spread belief, cannot err. A critical mind might

[13]User Data Manifesto 2.0 https://userdatamanifesto.org/, the European Digital Charta https://digitalcharta.eu/, the Swiss manifest for digital democracy http://digital-manifest.ch/, or the digital manifesto in Helbing et al. (2015).

[14]We leave aside for a moment the question of who may choose the target function for optimization. This leads to a likewise important debate, which we do not pursue at this place.

become skeptical at a choice of a defining frame which claims immunity against every criticism.

Many jokes "work" by employing a sudden and unexpected shift of the interpretational frames and many human tragedies are caused by sticking to an inappropriate frame. In an ancient Greek allegory King Midas wishes that everything he touches turns into gold. He realizes that he has chosen a wrong frame when not only his furniture but also his wine and his wife turn into solid gold. The defining frame for data science applications (empirically validated optimal choice) contradicts the reasonable frames for human existence and does not go well with concepts of humanity. Which mechanisms stop our society to make the same error as the figurative King Midas, when confronted with the promises of data science? It is, in fact, an interesting paradox. While everybody wants to lead a good life, the perspective of doing so by following the decisions of a machine ultimately is dehumanizing.

Human life is about empathy, about dealing with imperfection and coping with the sometimes-painful limitations of an often-absurd existence. The fundamental strengths of a human being are the ability to cope with this situation, by giving meaning to our life. Almost all productions of human culture, from music to literature and from astronomy to physics, are witnesses of a more or less successful coping with this situation. The frames of "optimal decisions" or of "maximization of profits" may be helpful in a few particular situations, however too broad an adherence to these frames or too successful an implementation of them turns the understanding of human values upside down.

Let us give a vivid example: How would the stereotypic data scientists fall in love? Would match-making algorithms choose their partners? As the efficiency of current algorithms in this field is subject of controversial debates (Tierny 2013), let us conduct this as a thought experiment! How would they *make* themselves fall in love with the person selected by the machine? Would they *really* fall in love or fall for an illusion? What if they felt more for a person the algorithm explicitly warns them of? The western concept of love as individual spontaneous attraction, which hopefully might grow into stable and trusted relationships, conflicts with eastern cultures where partners for life are selected by parents. Which target function should drive the selection process? Who will decide on that? The cultural conflicts between the stereotypic eastern and western partnering processes are replaced by questions on algorithmic parameters. The resolutions of these conflicts form the basis for many a personal fate and, ultimately, collective cultural development. Do we want to settle these conflicts by law, by tradition, or by individual decision? Do we agree to have them settled by Moore's law (Waldrop 2016), when by the technological coincidence fast computers and "intelligent" algorithms provide us with "optimal" choices?

Maybe the frame of optimization is not appropriate for many areas of human existence. If we draw this conclusion, why should we tolerate a creeping—and creepy—development toward this?

In theory and in a free society, the individual may choose differently. In practice, such a divergence faces constant pressure on those individuals who have not led the

best life available to them by ignoring to comply with suggestions of the machine.[15] Collectively, many people are not reflecting their lives in philosophical depth but rather follow the simple, nudged choices attractively offered via interfaces optimized to the effect desired by the operator. This trend is not isolated but transforms society into a form where these options of divergence are no longer offered and freedom of choice ultimately vanishes.

Again, the argument must not be parsed one-sidedly. Optimization and improvement *are* necessary; dangerous is the systematic, unreflected, and globalized acceptance of it as a singular dominant trend which is implemented in all processes, products, and work-flows.

6 The Second-Order Problem of Science and Data Science

Second-order problems[16] are problems which cannot be solved with the problem-solving behavior available to the system in which they arise. They are usually met with one or more of the following reactions: Denial that the problem exists, attempts to solve an unsolvable problem, or increasing those efforts which caused the problem in the very beginning (known as "more of the same"—paradox). The common aspect, unfortunately, is: The more a system tries to solve the problem, the bigger it grows the problem. Best intentions combined with an inability to recognize the paradox ultimately constitute the second order problem. A successful solution needs a second order approach and requires deliberation outside of the framework of the original problem. Often this relates to shifting an interpretational frame.

Unfortunately, very successful first-order problem solutions often tend to produce second-order problems. This is particularly true if large-scale systemic effects of first-order solutions are ignored or if only a single methodological approach is taken. The spiral of violence is a phenomenon well known from domestic abuse, and from cultural conflicts up to nuclear armament: If force is an accepted (first order) answer to violence between partners of similar power, this may lead to a spiral of ever-increasing violence with devastating destruction on both sides. The appropriate answer is to refuse using the first-order problem "solution" at the very beginning of the conflict.

By the same reasoning, dominance of the catholic church as single and dominant authority for explaining the world produced a crisis out of which in 1500–1800 AD numerous religious conflicts arose and modern science as a new form of survival evolved. Science and ratio as a method proved to be extremely successful and enabled impressive technological progress and welfare. The unilateral emphasis of

[15] For a more detailed elaboration, see Han (2010).

[16] See Watzlawik et al. (1979) for a systematic approach and Watzlawik (2005) or Watzlawik (2009) for a layman approach to the problem.

science and technology, however, also produced numerous problems. Climate change, destruction of our natural habitat, or the debates on post-truth politics are indicators thereof. Independently of where one stands with respect to these concrete debates, one cannot avoid acknowledging the existence of such a conflict in public debates. Even traditional scientific communities realize that and publish on various forms of inner-scientific problems, which come by the names of replication crisis or publication crisis.[17]

Of course, this certainly does not mean that the scientific approach is flawed and that we should return, for example, to the Bible as the source for scientific understanding—although social, political, and pseudo-scientific movements are growing which promote this goal. It should, however, be taken as an indication that the scientific method may have reached limits in the sense of a second-order problem. More lab experiments and more analyzed data might not be the correct mind-set which helps us out of the crisis. If one is willing to accept this hypothesis, data science points into the wrong direction. It is a first-order solution which is particularly good in aggravating the problem with its "more of the same" strategy.

We now provide a less philosophical, more concrete, well-known example from communication technology: In a world of printed letters, an answer within weeks was acceptable and gave the recipient sufficient time for a well-tuned reply. Then email was invented, leaving only days for an answer. The efficiency of the email solution was so high that ultimately half-day answers were expected. The traditional format of a letter with salutation and complimentary closings was perceived as a burden. Currently even faster formats, such as social network lifelines, messengers, and twitter-like forms of communication, are replacing email. The mere possibility of real-time like answers, often also the expectation thereof, produced a new communicative situation which does not leave room for thinking between the actions. The effects of a twittering president of a super-power on international diplomacy and on stock exchange prices can currently be witnessed.

The original problem—too large latency in communication by postal mail—has been solved too well and has in turn produced a second-order problem. For this, society currently pursues some unsuited first-order solutions. For example, digital non-natives are accused of not coping sufficiently with the new speed at the work place. This line of reasoning may be correct but is not helpful. Despite their ability of fast tweeting, the digital native generation has acquired their own deficiencies: Numerous studies describe a frightening loss of medium- and long-term attention, of the ability to understand, read, and appropriately react on emotions of other people, and even very significantly drop in *the* core human value, that is, empathy.[18]

[17] See, for example, Saltelli and Funtowicz (2017). The main reasons they give—and document with a wide range of references—comprise almost all properties frequently found in second-order problems, such as denial of the problem, being a victim of one's own success, no reaction to changes in systemic boundary conditions, flawed incentives leading to misallocations of resources, and more.

[18] A meta-analysis on 72 samples of 13,737 American college students demonstrates this empirically and provides a wealth of pointers into further literature. It identifies as reasons the changes in

Using electronic means of communication less frequently is not a solution either as this amounts to a voluntary decision of being less efficient than one's competitor. A suitable second-order solution remains to be found. The financial system exhibited similar systemic issues since the Lehman crisis,[19] likewise a medical system which focuses on economic efficiency instead of patient relief, as outlined above.

The author is convinced that the essential risks and side-effects of data science have a similar second-order nature: Data science is driven by the vision of the ultimately optimized scientific model, and data technology is driven by the dream of best economic results and most efficient applications. There is no first-order argument why this should be wrong! However, in the end, there is a problem with human values. As is the nature of a second-order problem, the situation fails to be understood with the tools of data science itself.

Starting with technology assessment and ending with Lehman, or the proverbial rise and fall of human values, may seem a helpless exaggeration and a much too big arena for our analysis. However, too small and too detailed a perspective is usually at the roots of second-order problems: Systemic effects are neglected, wide range effects are dismissed as far-fetched, and arguments outside of a narrow scope are perceived as irrelevant or methodologically flawed. Instead, the discourse of analysis should be widened, the pursued goals should be questioned, and the methods and scopes criticized and readjusted. This will not be done by data science itself: Why should data scientists limit themselves in their research, why should data technology companies hold back their run for economic success? It looks like we *do* have a second-order problem at hand.

Striving for improvement and optimal solutions is at the core of human development. However, deviating behavior, individual preferences, and even deficiency is human as well. The proper balance of these two aspects has successfully guided human development for centuries. Data science can destroy this balance.

7 Conclusion

This text fails to deliver technical solutions as they are usually expected in technical papers. Its goal is to raise awareness for a difficult, if not paradoxical, situation through intentionally pointed and painful metaphors. There is a wealth of proposals for quick fixes in other publications,[20] but these are merely band-aid and hide the issues at stake. The data scientist expecting a short "recipe" on how to do things

media and communication technology as well as an increased expectation of success and human optimality. See Konrath et al. (2011).

[19]For a description why financial systems failed in the 2008 crisis not because they ignored best practice but because they *followed* established governance, and for a description of the collective blindness in recognizing self-serving governance mechanisms, see Turnbull (2016).

[20]See, for example, Hofstetter (2016) or Cap (2016).

"right" will be frustrated, as the message is a criticism on one of the mind-sets of data technology: Ubiquitous analysis for pervasive optimization.

The privacy debate produced the "right to be left alone" (Brandeis and Warren 1890–1891) as a consequence of human dignity. Of course, the right should be balanced and must not be understood as an appeal to turn a collaborative society into hermits.

The data science debate must focus on the "*right to be different*" as a "*right to deviate*" from what is considered the optimal choice—whether this choice has been obtained by a majority consensus, by scientific methods, or calculated by an algorithm. Tolerance and the protection of minorities are the values to be saved. This goal is beyond what a particular scientific discipline can achieve and it culminates in the paradoxical insight that an optimal world is bad, or that there is no such thing like a best or true frame for understanding our existence.

The particularly problematic aspect is, of course, that the normative character of objectivized statements and optimized processes is an essential feature for the scientific and technological success of the last 300 years. So how would a "right to deviate" from established norms be implemented without destroying the beneficial aspects of such norms?

We might realize the danger of a slippery slope with a thought experiment. It is well known that algorithms may turn racist by learning a bias from observing humans. Bornstein (2017) describes the case of an algorithm for bail decisions in criminal cases, which learned to discriminate against Afro-American people based on police behavior. But what if an algorithm developed a preference or bias from "facts" "alone"? Would we be willing to accept predictions if they violated our sense of racial fairness? How would we deal with obvious violations of anti-discrimination laws by algorithms? Would we twist the facts? Would we legislate which facts may enter the algorithm? Would we forbid the use of algorithms even though a statistical analysis could tell us the objective "damage" which this prohibition would produce? Of course, every decision system will lead to some form of decision. Only a human sense of fairness produces the distinction between unethical bias and fair prediction, but this evaluation varies in time. It is in no way objective or scientific but reflects our values and depends on cultural, political, and economic conditions.

New and due to data science is our ability to quantify this human sense of fairness. An algorithmic answer seems available to this question "If we dropped our fairness towards group X and allowed our algorithms to discriminate against them then, on a world-wide scale, this would allow an increase in productivity of Y units and in security of Z units." The next question only seems logical. Which combination of X, Y and Z is acceptable from an ethical and economical point of view? Who is going to decide which point of view shall be used? Who will implement the answers and how?

We might follow Edward Teller (1998) who suggested "[that] we must learn to live with contradictions, because they lead to deeper and more effective understanding." The fall of man in physics was the discovery of the nuclear chain reaction, the fall of man in computing may prove to be the discovery of data science methods.

This analogue may point us to a solution. Politicians and generals in the cold war had a hard time learning that in a world with multiple nuclear armies a conflict cannot be reasonably won by using atomic bombs. The ultimate solution is clear: Know the bomb but do not use it!

Similarly, we are in the process of understanding that an economic system depending on percental growth leads to exponential development and exhausts available resources. While this is often abused as a romanticized argument in ideologic discussions, it also is a simple mathematical phenomenon. Although the author is fascinated by the mathematical elegance of science and of data scientific methods, it is his conviction that an indiscriminate application of data science may produce more problems than it solves. This is particularly true if the methods are applied on data related to humans, on human behavior, with a goal of optimizing costs and processes where humans are closely involved or in a situation where our human understanding of complex world phenomena are involved. Contrary to the observation on percental growth, this conviction is not a mathematical phenomenon but an assumption based on a personal observation of human nature and greed, so it is difficult to reach an agreement on it with formal reasoning alone. The short form of a solution is somewhat like in nuclear war technology: Study data science but do not apply it!

As in nuclear technology we need a debate on the details. We have learned to use nuclear engineering to make war, to produce energy, to diagnose, and to treat health problems—and we have been working on an understanding which applications are acceptable. Data science needs a similar debate. I expect this debate to be long and tedious. I am not sure which force will prove stronger—greed, rationality, or humanitarianism.

References

Anderson, C. W. (1989). Learning to control an inverted pendulum using neural networks. *IEEE Control System Magazine, 9*(3), 1989.

Borasio, G. (2016). *Selbstbestimmt sterben*. München: DTV Verlagsgesellschaft.

Bornstein, A. M. (2017). Are algorithms building the new infrastructure of racism? *Nautilus, 55*. http://nautil.us/issue/55/trust/are-algorithms-building-the-new-infrastructure-of-racism.

Brandeis, L., & Warren, S. (1890–1891). The right to privacy. *Harvard Law Review, 4*, 193–220.

Braschler, M., Stadelmann, T., & Stockinger, K. (2019). Data science. In M. Braschler, T. Stadelmann, & K. Stockinger (Eds.), *Applied data science – Lessons learned for the data-driven business*. Berlin: Springer.

Cap, C. H. (2016). Verpflichtung der Hersteller zur Mitwirkung bei informationeller Selbstbestimmung. In M. Friedewald, J. Lamla, & A. Roßnagel (Eds.), *Informationelle Selbstbestimmung im digitalen Wandel*. Wiesbaden: Springer Vieweg, DuD-Fachbeiträge.

Cap, C. H. (2017). Vertrauen in der Krise: Vom Feudalismus 2.0 zur Digitalen Aufklärung. In M. Haller (Ed.), *Öffentliches Vertrauen in der Mediengesellschaft*. Köln: Halem Verlag.

Dwork, C., & Mulligan, D. (2013, September). It's not privacy, and it's not fair. *Stanford Law Review*. https://www.stanfordlawreview.org/online/privacy-and-big-data-its-not-privacy-and-its-not-fair/

Esguerra, R. (2009, December 10). *Google CEO Eric Schmidt dismisses the importance of privacy*. Electronic Frontier Foundation. https://www.eff.org/de/deeplinks/2009/12/google-ceo-eric-schmidt-dismisses-privacy

Executive Office of the President. (2016, May). Big data: A report on algorithmic systems, opportunity, and civil rights. *The Whitehouse*. https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/2016_0504_data_discrimination.pdf

Friedman, L. (2014, April 22). IBM's Watson supercomputer may soon be the best doctor in the world. *Business Insider*. http://www.businessinsider.com/ibms-watson-may-soon-be-the-best-doctor-in-the-world-2014-4

Galeon, D. (2016, October 28). IBM's Watson AI recommends same treatment as doctors in 99% of cancer cases. *Futurism*. https://futurism.com/ibms-watson-ai-recommends-same-treatment-as-doctors-in-99-of-cancer-cases/

Grassegger, H., & Krogerus, M. (2016, December 3). Ich habe nur gezeigt, dass es die Bombe gibt. *Das Magazin* Nr. 48. https://www.dasmagazin.ch/2016/12/03/ich-habe-nur-gezeigt-dass-es-die-bombe-gibt/

Han, B.-C. (2010). *Müdigkeitsgesellschaft*. Berlin: Matthes & Seitz.

Helbing, D., et al. (2015). *Eine Strategie für das digitale Zeitalter*, 12.11.2015, Spektrum Verlag, http://www.spektrum.de/kolumne/eine-strategie-fuer-das-digitale-zeitalter/1376083

Hofstetter, Y. (2016). *Sie wissen alles: Wie Big Data in unser Leben eindringt und warum wir um unsere Freiheit kämpfen müssen*. München: Penguin.

Hosain, S. Z. (2016). Reality check: 50B IoT devices connected by 2020 – Beyond hype and into reality. *RCRWirelessNews*, 28th June 2016. http://www.rcrwireless.com/20160628/opinion/reality-check-50b-iot-devices-connected-2020-beyond-hype-reality-tag10

Konrath, S., O'Brien, E., & Hsing, C. (2011). Changes in dispositional empathy in American college students over time. *Personality and Social Psychology Review, 15*(2), 180–198.

Kree, I., & Earle, J. (2013, September 2013). Prediction, preemption, presumption: How big data threatens big picture privacy. *Stanford Law Review*. https://www.stanfordlawreview.org/online/privacy-and-big-data-prediction-preemption-presumption/

Lewis, C. S. (1972). *God in the dock. Essays on theology and ethics*. Grand Rapids: Eerdmans Publishing.

Marr, B. (2015, September 28). *The biggest risks of big data*. http://www.datasciencecentral.com/profiles/blogs/the-biggest-risks-of-big-data

Miller-Merell, J. (2012, September 27). Hiring by algorithm. The new self-checkout of HR. *SmartRecruiters Blog*. https://www.smartrecruiters.com/blog/hiring-by-algorithim-the-new-self-checkout-of-hr-2/

Nichols, S. (2017, January 7). TV anchor says live on-air 'Alexa, order me a dollhouse'. *The Register*. https://m.theregister.co.uk/2017/01/07/tv_anchor_says_alexa_buy_me_a_dollhouse_and_she_does/

O'Connor, A. (2011, February 15). Watson dominates Jeopardy but stumbles over geography. *New York Times*. https://artsbeat.blogs.nytimes.com/2011/02/15/watson-dominates-jeopardy-but-stumbles-over-geography/

Park, J., Seo, D., et al. (2014). *Practical human resource allocation in software projects using genetic algorithm*. In SEKE 2014, pp. 688–694. https://pdfs.semanticscholar.org/c317/a294cb7995f5ca7799ab1f7f878e7b0a4749.pdf

Root-Bernstein, R., & Root-Bernstein, M. (1999). *Sparks of genius – The thirteen thinking tools of the world's most creative people*. Boston: Houghton Mifflin.

Saltelli, A., & Funtowicz, S. (2017). What is science's crisis really about? *Futures, 91*, 5–11.

Schönholzer, F. (2017). Digitale Pfadfinder. *UZH News*, 14.11.2017. http://www.news.uzh.ch/de/articles/2017/Digitale-Pfadfinder.html

Shattuck, R. (1997). Forbidden knowledge: From prometheus to pornography. *Harvest Book*. Here quoted from Wikiquote https://en.wikiquote.org/wiki/Edward_Teller

Teller, E. (1998). Science and morality. *Science, 280*(5367), 1200–1201.

Thöns, M. (2016). *Patient ohne Verfügung: Das Geschäft mit dem Lebensende*. München: Piper.

Tierny, J. (2013, February 11). A match made in the code. *The New York Times*. http://www.nytimes.com/2013/02/12/science/skepticism-as-eharmony-defends-its-matchmaking-algorithm.html

Turnbull, S. (2016). Defining and achieving good governance. In G. Aras & C. Ingley (Eds.), *Corporate behavior and sustainability*. New York: Gower.

Waldrop, M. (2016). The chips are down for Moore's law. *Nature, 530*, 144–147.

Warriar, L., Roberts, A., & Lewis, J. (2002). 2002- *Surveillance – An analysis of Jeremy Bentham Michel Foucault and their present day relevance.* http://studymore.org.uk/ybenfou.htm

Watzlawik, P. (2005). *Vom Schlechten des Guten oder Hekates Lösungen*. München: Piper.

Watzlawik, P. (2009). *Anleitung zum Unglücklichsein*. München: Piper.

Watzlawik, P., et al. (1979). *Lösungen. Zur Theorie und Praxis menschlichen Wandels*. Bern: Hans Huber.

Wehling, E. (2016). *Politisches Framing: Wie eine Nation sich ihr Denken einredet – und daraus Politik macht*. Köln: Halem.

Weizenbaum, J. (1976). *Computer power and human reason: From judgement to calculation*. New York: Freeman.

Part II
Use Cases

Chapter 7
Organization

Martin Braschler, Thilo Stadelmann, and Kurt Stockinger

Abstract Part II of this book represents its core—the nuts and bolts of applied data science, presented by means of 16 case studies spanning a wide range of methods, tools, and application domains.

We organize the individual chapters in the following way, based on their main focus:

Chapters 8–10 present **fundamentals** that cut across many case studies:

- Brodie gives a detailed account of his opinionated view on the current state of data science as a science (Chap. 8). He also presents a development model based on research-development virtuous cycles for projects as well as the discipline as a whole that is grounded in reality (Chap. 9).
- Christen et al. then present a sound and practical guideline for ethical considerations in analytics projects (Chap. 10).

Chapters 11–16 focus on **methods and tools** within case studies:

- Ruckstuhl and Dettling (Chap. 11) and Stadelmann et al. (Chap. 12) present work on discipline-specific approaches and methodological contributions to data science from a *statistical* and *deep learning*-based viewpoint, respectively.
- Braschler gives a detailed exposition of the challenges of small data collections for *Information Retrieval* in Chap. 13.
- *Visual storytelling* is exemplified by Ackermann and Stockinger in Chap. 14.
- A tutorial on the mutual dependencies and benefits between data science and *computer security* is given by Tellenbach et al. in Chap. 15.
- Finally, Rettig et al. explain the architecture of a *big data stream processing system* based on a specific anomaly detection example (Chap. 16).

M. Braschler (✉) · T. Stadelmann · K. Stockinger
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: bram@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_7

Finally, Chapters 17–23 focus on the **applications** themselves:

- *Production*: Hollenstein et al. discuss the reduction of product complexity in industrial production (Chap. 17).
- *Commerce*: Geiger and Stockinger describe market monitoring by means of a carefully designed data warehouse architecture (Chap. 18); whereas Ott et al. report on demand planning by forecasting and how to evaluate its success (Chap. 20).
- *Health*: Leidig and Wolffe show how to predict disease spread in a population using data mining on mobile phone data (Chap. 19); personal health data management facilitated by good governance and IT architecture is discussed by Bignens and Hafen (Chap. 22); and finally, the complete cycle of medical image analysis is described by Mader (Chap. 23).
- *Finance*: Risk assessment using big data infrastructure is the focus of Chap. 21 by Breymann et al.

A more structured overview of the contents to each chapter (e.g., listed by methods applied, tools, discipline-specific viewpoints, stage in the knowledge discovery in databases (KDD) process, etc.) is provided in Part III of this book, serving as an index to the chapters of this part.

Chapter 8
What Is Data Science?

Michael L. Brodie

Abstract Data science, a new discovery paradigm, is potentially one of the most significant advances of the early twenty-first century. Originating in scientific discovery, it is being applied to every human endeavor for which there is adequate data. While remarkable successes have been achieved, even greater claims have been made. Benefits, challenge, and risks abound. The science underlying *data science* has yet to emerge. Maturity is more than a decade away. This claim is based firstly on observing the centuries-long developments of its predecessor paradigms—empirical, theoretical, and Jim Gray's *Fourth Paradigm of Scientific Discovery* (Hey et al., The fourth paradigm: data-intensive scientific discovery Edited by Microsoft Research, 2009) (aka eScience, data-intensive, computational, procedural)—and secondly on my studies of over 150 data science use cases, several data science-based startups, and, on my scientific advisory role for Insight (https://www.insight-centre.org/), a Data Science Research Institute (DSRI) that requires that I understand the opportunities, state of the art, and research challenges for the emerging discipline of data science. This chapter addresses essential questions for a DSRI: *What is data science*? *What is world-class data science research*? A companion chapter (Brodie, *On Developing Data Science, in Braschler et al. (Eds.), Applied data science – Lessons learned for the data-driven business, Springer 2019*) addresses the development of data science applications and of the data science discipline itself.

1 Introduction

What can data science do? *What characteristics distinguish data science from previous scientific discovery paradigms*? *What are the methods for conducting data science*? *What is the impact of data science*? This chapter offers initial answers to these and related questions. A companion chapter (Brodie 2019) addresses the

M. L. Brodie (✉)
Computer Science and Artificial Intelligence Laboratory, Massacheusetts Institute of Technology, Cambridge, USA
e-mail: michael@michaelbrodie.com

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_8

development of data science as a discipline, as a methodology, as well as data science research and education. Let us start with some slightly provocative claims concerning data science.

Data science has been used successfully to accelerate the discovery of probabilistic outcomes in many domains. Piketty's (2014) monumental result on wealth and income inequality was achieved through data science. It used over 120 years of sporadic, incomplete, observational economic data, collected over 10 years from all over the world (Brodie 2014b). What is now called *computational economics* was used to establish the correlation, with a very high likelihood (0.90), that wealth gained from labor could never keep up with wealth gained from assets. What made front page news worldwide was a second, more dramatic correlation that there is a perpetual and growing wealth gap between the rich and the poor. This second correlation was not derived by data analysis but is a human interpretation of Piketty's data analytic result. It contributed to making *Capital in the Twenty-First Century* the best-selling book on economics, but possibly the least read. Within a year, the core result was verified by independent analyses to a far greater likelihood (0.99). One might expect that further confirmation of Piketty's finding would be newsworthy; however, it was not, as the more dramatic rich-poor correlation, while never analytically established, had far greater appeal. This illustrates the benefits and risks of data science.

Frequently, due to the lack of evidence, economic theories fail. Matthew Weinzierl, a leading Harvard University economist, questions such economic modelling in general saying, "that the world is too complicated to be modelled with anything like perfect accuracy" and "Used in isolation, however, it can lead to trouble" (Economist 2018b). Reputedly, Einstein said: "Not everything that counts can be counted. Not everything that's counted, counts." The hope is that data science and computational economics will provide theories that are fact-based rather than based on hypotheses of "expert" economists (Economist 2018a) leading to demonstrably provable economic theories, that is, what really happened or will happen. This chapter suggests that this hope will not be realized this year.

Many such outcomes[1] have led to verified results through methods outside data science. Most current data analyses are domain specific, many even specific to classes of models, classes of analytical methods, and specific pipelines. Few data science methods have been generalized outside their original domains of application, let alone to all domains (to illustrated in a moment). A rare and excellent exception is a generic scientific discovery method over scientific corpora (Nagarajan et al. 2015) generalized from a specific method over medical corpora developed for drug discovery (Spangler et al. 2014) that is detailed later in the chapter.

[1]Not Piketty's, since computational economics can find *what* might have happened—patterns, each with a given likelihood—but lacks the means of establishing causal relationships, that is, establishing *why*, based solely on observational data.

It is often claimed that data science will transform conventional disciplines. While transformations are underway in many areas, including supply chain management[2] (Waller and Fawcett 2013) and chemical engineering (Data Science 2018), only time and concrete results will tell the extent and value of the transformations. The companion chapter *On Developing Data Science* (Brodie 2019) discusses with the transformation myth.

While there is much science in many domain-specific data science activities, there is little fundamental science that is applicable across domains. To warrant the designation *data science*, this emerging paradigm requires fundamental principles and techniques applicable to all relevant domains, just as the scientific principles of the scientific method apply across many domains. Since most data science work is domain specific, often model- and method-specific, data science does not yet warrant the designation as a science.

This chapter explores the current nature of data science, its qualitative differences with its predecessor scientific discovery paradigms, its core value and components that, when mature, would warrant the designation *data science*. Descriptions of large-scale data science activities referenced in this chapter apply, scaled down, to data science activities of all sizes, including increasingly ubiquitous desktop data analytics in business.

2 What Is Data Science?

Due to its remarkable popularity, there is a plethora of descriptions of data science, for example:

> Data Science is concerned with analyzing data and extracting useful knowledge from it. Building predictive models is usually the most important activity for a Data Scientist.[3]

> Data Science is concerned with analyzing Big Data to extract correlations with estimates of likelihood and error. Brodie (2015a)

> Data science is an emerging discipline that draws upon knowledge in statistical methodology and computer science to create impactful predictions and insights for a wide range of traditional scholarly fields.[4]

Due to data science being in its infancy, these descriptions reflect some of the many contexts in which it is used. This is both natural and appropriate for an emerging discipline that involves many distinct disciplines and applications. A definition of data science requires the necessary and sufficient conditions that

[2]Selecting the best delivery route for 25 packages from 15 septillion alternatives, an ideal data science application, may explain the some of the $1.3–$2 trillion a year in economic value projected to be gained in the transformation of the supply chain industry due to AI-based data analytics (Economist 2018c).

[3]Gregory Piatetsky, KDnuggets, https://www.kdnuggets.com/tag/data-science

[4]Harvard Data Science Initiative, https://datascience.harvard.edu

distinguish it from all other activities. While such a definition is premature, a working definition can be useful for discussion. The following definition is intended to explore the nature of this remarkable new discovery paradigm. It is based on studying over 150 data science use cases and benefits from 3 years' research and experience over a previous version (Brodie 2015a). Like many data science definitions, it will be improved over the next decade in which data science will mature and gain the designation as a new science.

> Data Science is a body of principles and techniques for applying data analytic methods to data at scale, including volume, velocity, and variety, to accelerate the investigation of phenomena represented by the data, by acquiring data, preparing and integrating it, possibly integrated with existing data, to discover correlations in the data, with measures of likelihood and within error bounds. Results are interpreted with respect to some predefined (theoretical, deductive, top-down) or emergent (fact-based, inductive, bottom-up) specification of the properties of the phenomena being investigated.

A simple example of a data science analysis is the pothole detector developed at MIT (Eriksson et al. 2008) to identify potholes on the streets of Cambridge, MA. The data was from inexpensive GPS and accelerometer devices placed in a fleet of taxis that drive over Cambridge streets. The model was designed ad hoc for this application. A model consists of the features (i.e., variables) essential to the analysis and the relationships among the features. It was developed in this case ad hoc by the team iteratively refining the model through imagination, observation, and analysis. Ultimately, it consisted of a large number of movement signatures, that is, model features, each designed to detect specific movement types that may indicate potholes and non-potholes, for example, manholes, railroad tracks,[5] doors opening and closing, stopping, starting, accelerating, etc. Additionally, the size of the pothole was estimated by the size of the movement. The analytical method was the algorithmic detection and filtering of non-pothole signatures leaving as a result those movements that correlate with potholes with an estimated severity, likelihood, and error bound. The severity and likelihood estimates were developed ad hoc based on verifying some portion of the detected movements with the corresponding road surfaces thus contributing to estimating the likelihood that the non-potholes were

[5]The pothole models consist of a number of signature movements, that is, abstractions used to represent movements of the taxi, only some of which are related to the road surface. Each signature movement was created using the data (variables or features) available from a smartphone including the clock for time, the GPS for geographic location (latitude and longitude), and the accelerometer to measure changes in velocity along the *x*, *y*, and *z* axes. For example, the taxi crossing a railroad track would result in many signature "single tire crossing single rail line" movements, one for each of four tires crossing each of several rail lines. A "single tire crossing single rail line" involves a sudden, short vertical (*x*-axis) acceleration combined with a short lateral (*y*-axis) movement, forward or backward, with little or no lateral (*z*-axis) movement. Discounting the railroad crossing as a pothole involves recognizing a large number of movements as a taxi is crossing a rail line—all combinations of "single tire crossing single rail line" forward or backward, at any speed, and at any angle—to determine the corresponding staccato of the multiple single tire events over multiple lines. The pothole model is clearly ad hoc, in contrast to well-established models in physics and retail marketing.

excluded, and potholes were included. Error bounds were based on the precision of the equipment, for example, motion device readings, network communications, data errors, etc. The initial result was many thousands of locations with estimated severities, likelihoods, and error bounds. Conversion of likely pothole locations (correlations) to actual potholes severe enough to warrant repair (causal relationships between movements and potholes) were estimated by a manual inspection of some percentage of candidate potholes. The data from the inspection of the actual locations, called ground truth, was used to verify the likelihood estimates and establish a threshold above which confidence in the existence of a pothole warranted sending out a repair crew to repair the pothole. The customer, the City of Cambridge, MA, was given a list of these likely potholes.

The immediate value of the pothole detector was that it reduced the search for potholes from manually inspecting 125 miles of roads and relying on citizen reports that takes months, to discovering likely, sever potholes within days of their creation. Since 2008, pothole detectors have been installed on city vehicles in many US cities. The pothole detector team created Cambridge Mobile Telematics that develops applications for vehicle sensor data, for example, they annually produce reports on distracted driving across the USA based on data from over 100 million trips (Cambridge Mobile Telematics 2018). While these applications were used initially by insurance companies, they are part of the burgeoning domain of autonomous vehicles and are being used by the US National Academy of Sciences (Dingus et al. 2016) for driving safety.

3 Data Science Is a New Paradigm of Discovery

Data science emerged from, and has many commonalities with, its predecessor paradigm, the scientific method[6]; however, they differ enough for data science to be considered a distinct, new paradigm. Like the scientific method, data science is based on principles and techniques required to conduct discovery activities that are typically defined in terms of a sequence of steps, called a workflow or pipeline; results are specified probabilistically and with error bounds based on the data, the model, and the analytical method used; and the results are interpreted in terms of the hypothesis being evaluated, the model, the methods, and the probabilistic outcome relative to the accepted requirements of the domain of the study. In both paradigms, *models* are collections of features (represented by variables that determine the data to be collected) that characterize the essential properties of the phenomenon being analyzed. Data corresponding to the features (variables) in the model are collected

[6]The scientific method is a body of techniques for investigating phenomena, acquiring new knowledge, or correcting and integrating previous knowledge. To be termed scientific, a method of inquiry is commonly based on empirical or measurable evidence subject to specific principles of reasoning. https://en.wikipedia.org/wiki/Scientific_method

from real instances of the phenomena and analyzed using analytical methods developed for the type of analysis to be conducted and the nature of the data collected, for example, different methods are required for integers uniformly distributed in time versus real numbers skewed due to properties of the phenomenon. The outcomes of the analysis are interpreted in terms of the phenomena being analyzed within bounds of precision and errors that result from the data, model, and method compared with the precision required in the domain being analyzed, for example, particle physics requires precision of six standard deviations (six sigma). Data science differs paradigmatically from the scientific method in data, models, methods, and outcomes, as described below. Some differences may be due to data science being in its infancy, that is, models for real-time cyber-attacks may not yet have been developed and proven; however, some differences, discussed below, are inherent. We are in the process of learning which is which.

3.1 Data Science Data, Models, and Methods

Data science data is often obtained with limited knowledge of the conditions under which the data was generated, collected, and prepared for analysis, for example, data found on the web; hence, it cannot be evaluated as in a scientific experiment that requires precise controls on the data. Such data is called observational. Compared with empirical scientific data, data science data is typically, but not necessarily, at scale by orders of magnitude in one or more of volume, velocity, and variety. Scale requires management and analytic methods seldom required in empirical science.

Data science models used in most scientific domains have long histories of development, testing, and acceptance, for example, the standard model of particle physics[7] emerged in 1961 after decades of development and has matured over the subsequent decades. In contrast, currently data science models, for example, for real-time bidding for online advertising, are created on demand for each data science activity using many different, innovative, and ad hoc methods. Once a model is proven, they can be accepted and put into productive use with periodic tuning, for example, real-time ad placement products. It is likely that many proven data science models will emerge as data science modelling matures. StackAdapt.com has developed such a model for real-time bidding and programmatic ad purchasing (RTB) that is its core capability and intellectual property with which it has become a RTB world leader among 20 competitors worldwide. The StackAdapt model is used to scan 10 billion data points a day and manage up to 1,50,000 ad opportunity requests per second during peak times.

[7]The Standard Model of particle physics is the theory describing three of the four known fundamental forces (the electromagnetic, weak, and strong interactions, and not including the gravitational force) in the universe, as well as classifying all known elementary particles. It was developed in stages throughout the latter half of the twentieth century, through the work of many scientists around the world. https://en.wikipedia.org/wiki/Standard_Model

Data science analytical methods, like data science models, are often domain- and data-specific and are developed exclusively for a specific data science activity. There are generic methods, often named by a class name. For example, the primary classes of machine-learning algorithms[8] are: Linear Classifiers—Logistic Regression, Naive Bayes Classifier; Support Vector Machines; Decision Trees; Boosted Trees; Random Forest; Neural Networks; and Nearest Neighbor. There are generic algorithms for each class each of which can be applied in many domains. However, to be applied in a specific use case they must be refined or tuned often to the point of being applicable in that use case only. This is addressed in the next section that questions whether there are, as yet, underlying, thus generalizable, principles in data science.

Both models and methods require tuning or adjusting in time as more knowledge and data are obtained. Empirical scientific models tend to evolve slowly, for example, the standard model of particle physics is modified slowly[9]; in contrast, data science models typically evolve rapidly throughout their design and development, and even in deployment, using dynamic learning. Typically, models and methods are trained using semi-automatic methods by which specific data or outcomes, called ground truths, are confirmed by humans as real to the model or method. More automatic methods, for example, reinforcement learning and meta-learning,[10] are being developed by which models and methods are created automatically (Silver et al. 2017).

3.2 Data Science Fundamentals: Is Data Science a Science?

Currently, most data science results are domain-, method-, and even data-specific. This raises the question as to whether data science is yet a science, that is, with generalizable results, or merely a collection of sophisticated analytical methods, with, as yet, a few underlying principles emerging, such as Bayes' Theorem, Uncle Bernie's rule,[11] and Information Bottleneck theory. The scientific method is defined by principles that ensure scientific objectivity, such as empirical design and the related controls to govern experimental design and execution. These and other scientific principles make experiments "scientific," the minimum requirement for a result to be considered scientific. Scientific experiments vary across domains, such as the statistical significance required in a given domain, for example, two sigma has traditionally been adequate in many domains besides particle physics. A necessary, defining characteristic of data science is that the data is either at scale (Big Data) or observational (collected without knowing the provenance—what controls were

[8] https://medium.com/@sifium/machine-learning-types-of-classification-9497bd4f2e14

[9] Validating the Higgs-Boson took 49 years.

[10] http://bair.berkeley.edu/blog/2017/07/18/learning-to-learn/

[11] See Morgan, N., & Bourlard, H. (1990). Generalization and parameter estimation in feedforward nets: Some experiments. In Advances in neural information processing systems (pp. 630–637).

applied or with no controls uniformly applied) as is generally the case in economics and social sciences. Under those conditions, data science cannot be "scientific," hence accommodations must be made to draw conclusions from analysis over such data. As data science is just emerging in each domain, we have few principles or guidelines per domain, for example, statistical significance of results, or across all domains, for example, the extent to which statistical significance is required in any data science analysis. The above-mentioned pothole analysis was designed and executed by sheer intuition beyond the general ideas of identifying the hypothesis (find potholes using motion devices in taxis), experimental design (put devices in taxis and record their signals), modeling (what features are critical), and analysis (what motions indicate potholes, and which do not), and iteration of the model and analysis until acceptable precision was reached. The pothole data science activity did not draw on previous methods, nor did it offer, that is, was not cited, principles for modeling, methods, or process.

Another practical example is at Tamr.com that offers one of the leading solutions for curating or preparing data at scale, for example, data from 1,00,000 typically heterogeneous data sources. It launched initially with a comprehensive solution in the domain of information services. Tamr soon found that every new domain required a substantial revision of the machine-learning component. Initially, like most AI-based startups, their initial solution was not generalizable. As can be seen at Tamr.com, Tamr now has solutions in many domains for which they have substantial commonality in the underlying solutions.

Another fundamental difference between science and data science concerns the *scale and nature of the outcomes.* The scientific method is used to discover causal relationships between a small number of variables that represent the essential characteristics of the natural phenomena being analyzed. The experimental hypothesis defines the correlation to be evaluated for causality. The number of variables in a scientific experiment is kept small due to the cost of evaluating a potentially vast number of combinations of variables of interest. PhD theses, that is, an experiment conducted by one person, are awarded on experiments with two or three but certainly less than ten variables. Large-scale experiments, for example, Laser Interferometer Gravitational-Wave Observatory (LIGO),[12] Kepler,[13] and Higgs-Boson, may consider hundreds of variables and take years and thousands of scientists to evaluate. Determining whether a correlation between variables is causal tends to be an expensive and slow process.

Data science, on the other hand, is used to rapidly discover as many correlations between the data values as exist in the data set being analyzed, even with very large models (millions of variables) and vast data sets. Depending on the analytical method used, the number of variables in a data science analysis can be effectively unlimited, for example, millions, even billions, as can be the number of correlations between those variables, for example, billions or trillions. Data science analytics are

[12] http://www.ligo.org/

[13] https://keplerscience.arc.nasa.gov/

executed by powerful, efficient algorithms using equally powerful computing infrastructure (CPUs, networks, storage). The combined power of new algorithms and infrastructure in the 1990s led to the current efficacy of machine learning that in turn contributed to the emergence of data science.

3.3 *The Prime Benefit of Data Science Is Accelerating Discovery*

Data science and empirical science differ dramatically, hence paradigmatically, in the scale of the data analyzed. Scientific experiments tend to evaluate a small number, for example, 10s or 100s, of correlations to determine if they are causal, and do so over long periods of time, for example, months or years. In contrast, data science can identify effectively unlimited numbers of correlations, for example, millions, billions, or more, in short time periods, from minutes to days. It is in this sense that data science is said to *accelerate discovery.* Originally developed in the 1990s for scientific discovery, the remarkable results of data science have resulted in its being applied to all endeavors for which adequate data is available. *The prime benefit of data science is that it is a new paradigm for accelerating discovery*, in general.

Ideally, data science is used to accelerate discovery by rapidly reducing a vast search space to a small number of correlations that are likely to be casual, as indicated by their estimated probability. Depending on the resources available, some number of the probabilistic correlations are selected to be analyzed for causality by well-established (non-data science) means in the domain being analyzed. For example, data science has been used to accelerate cancer drug discovery. The Baylor-Watson study (Spangler et al. 2014) used data science methods to identify nine likely cancer drug candidates. It used a simple, novel method to further evaluate their likelihood. The original analysis was conducted over drug research results published up to 2003 and identified nine likely candidate drugs. The likelihood of those nine candidate drugs was raised significantly when the research published from 2003 to 2013 showed that seven of the nine candidates had been validated as genuine cancer drugs. This raised the likelihood that the remaining two candidate drugs were real. Standard EPA-approved drug development and clinical trial testing were then used to develop the two new drugs. In this case, data science accelerated drug discovery for a specific type of cancer. It started with a vast search space of cancer research results from 2,40,000 papers. In three months it discovered the two highly likely cancer drug candidates. Conventional cancer drug discovery typically discovers one drug every 2–3 years. These times do not include the drug development and clinical trial periods.

3.4 Causal Reasoning in Data Science Is Complex and Can Be Dangerous

Just as the scale is radically different so is the nature of the results. The scientific method discovers results that, if executed correctly, are definitive, that is, true or false, with a defined probability and error bound, that a hypothesized relationship is causal. Data science discovers a potentially large number of correlations each qualified by a probability and error bound that indicate the likelihood that the correlation may be true. *Data science is used to discover correlations; it is rarely used to determine causal relationships*. The previous sentence is often misunderstood not just by novices but also, unfortunately, by data scientists. Empirical science discovers causal relationships in one step. Data science is frequently used to discover causal relationships in two steps: First, discover correlations with a strong likelihood of being causal; then use non-data science methods to validate causality.

Causality is the Holy Grail of science, scientific discovery, and if feasible, of data science. Typically, the goal of analyzing a phenomenon is to understand ***why*** some aspects of the phenomenon occur, for example, why does it rain? Prior to a full understanding of the phenomenon, initial discovery is often used to discover ***what*** conditions prevail when the phenomenon manifests, for example, as rain starts and during rain many raised umbrellas can be observed. A more informed observer may also discover specific climatic conditions. All of the conditions observed to be present consistently before and during the rain could be said to be correlated with rain. However, correlation does not imply causation, for example, raised umbrellas may be correlated with rain but do not cause the rain (Brodie 2014a). A more realistic example comes from an online retailer who, observing that increased sales were correlated with customers purchasing with their mobile app, invested significantly to get their app onto many customers' smartphones. However, the investment was lost since sales did not increase. Increased purchases were correlated with mobile apps on customers' smartphones; however, the causal factor was customer loyalty and, due to their loyalty, most loyal customers already had the app on their smartphones.

Data science is used predominantly to discover ***what***. Empirical science and many other methods are used to discover ***why*** (Brodie 2018). Data science is often used to rapidly reduce the search space from a vast number of correlations or possible results to a much smaller number. The much smaller number of highly probable results are then analyzed with non-data science methods, such as scientific experiments or clinical trials, to verify or reject the result, that is, automatically generated hypotheses, as causal.

There are mathematics and methods claimed for deducing causal effects from observational data (i.e., data not from controlled experiments but from surveys, censuses, administrative records, and other typically uncontrolled sources such as in Big Data and data science). They are very sophisticated and require a deep understanding of the mathematics, statistics, and related modelling methods. Judea

Pearl has developed such methods based on statistics, Bayesian networks, and related modelling (see Pearl 2009a, b, c). For decades, statisticians and econometricians have developed such methods with which to estimate causal effects from observational data, since most social and economic data is purely observational (Winship and Morgan 1999).

Causal reasoning involves going beyond the mathematics and modelling for data science in which correlations are obtained. "One of Pearl's early lessons is that it's only possible to draw causal conclusions from observational (correlational) data if you are willing to make some assumptions about the way that the data were sampled and about the absence of certain confounding influences. Thus, my understanding is that one can draw causal conclusions, but it's important to remember that these are really conditional on the validity of those assumptions," says Peter Szolovits, Professor, CSAIL, MIT, with a decade of experience applying data science in medical contexts for which he provided an example.[14]

Finding correlations between variables in (Big) data together with probabilities or likelihoods of the correlation occurring in the past or future are relatively easy to understand and safe to report. Making a causal statement can be misleading or dangerous depending on the proposed actions to be taken as a consequence. Hence, I do not condone nor confirm causal reasoning; it is above my pay grade; hence, I quote experts on the topic rather than make my own assertions. I recommend that causal reasoning not be applied without the required depth of knowledge and experience, because making causal statements as a result of data science analysis could be dangerous. In lecturing on correlation versus causation for over 5 years, I

[14]The full quote from personal communication: "There are various sophisticated ways to do all this but let me give you a relatively simple example: Suppose that we observe that in some cohort of patients, some were treated with drug X and others with drug Y. Suppose further that we see that fewer of the X patients died than of the Y ones. It's certainly NOT acceptable to conclude that X is a better drug, because we can't exclude the possibility that the treating doctors' choice of X or Y depended on some characteristics of the patient that also influenced their likelihood of survival. E.g., maybe the people who got Y were much sicker to start with, because Y is a stronger and more dangerous drug, so it is only given to the sickest patients.

"One way to try to mitigate this is to build a model from all the data we have about the patients in the cohort that predicts whether they are likely to get X or Y. Then we stratify the cohort by the probability of getting X, say. This is called a *propensity score*. Among those people with a high score, most will probably actually get X (that's how we built the model), but some will nevertheless get Y, and vice versa. If we assume that the doctors choosing the drug have no more information than the propensity model, then we treat their choice to give X or Y as a random choice, and we analyze the resulting data as if, for each stratum, patients were randomized into getting either X or Y, as they might have been in a real clinical trial. Then we analyze the results under that assumption. For many of the strata where the propensity is not near .5, the drugs given will be unbalanced, which makes the statistical power of the analysis lower, but there are statistical methods for dealing with this. Of course, the conclusions one draws are still very much dependent of the assumption that, within each stratum, the doctors' choice of drug really is random, and not a function of some difference among the patients that was not captured in the data from which the propensity model was built.

"This is just one of numerous methods people have invented, but it is typical of the kinds of assumptions one has to make in order to draw causal conclusions from data."

have found that an inordinate amount of interest is given to this difficult and little understood topic, perhaps with a desire to be able to provide definitive answers, even when there are none. I have found no simple explanation. You either study, understand, and practice causal reasoning with the appropriate care or simply stay away until you are prepared. Experts are appropriately cautious. "I have not, so far, made causal claims based on my work, mainly because I have not felt strongly enough that I could defend the independence assumptions needed to make such claims. However, I think the kinds of associational results are still possibly helpful for decision makers when combined with intuition and understanding. Nevertheless, I think most clinicians today do not use predictive models other than for more administrative tasks such as staffing or predicting bed occupancy"—Peter Szolovits, MIT. "I firmly believe that [deriving] causal results from observational data is one of the grand challenges of the data science agenda!"—David Parkes, co-lead of the Harvard Data Science Initiative. "Pearl once explained those ideas to me personally at Santa Catalina workshop, but I still don't fully understand them either:)"—Gregory Piatetsky-Shapiro, President of KDnuggets, co-founder of KDD Conferences and ACM SIGKDD.

3.5 Data Science Flexibility: Data-Driven or Hypothesis-Driven

Empirical science and data science have another fundamental difference. The scientific method uses deductive reasoning, also called hypothesis-driven, theory-driven, and top-down. Deductive reasoning is used when specific hypotheses are to be evaluated against observations or data. A scientific experiment starts by formulating a hypothesis to be evaluated. An experiment is designed and executed, and the results interpreted to determine if the hypothesis is true or false under the conditions defined for the hypothesis. It is called theory-driven in that a theory is developed, expressed as a hypothesis, and an experiment designed to prove or invalidate the hypothesis. It is called top-down since the experiment starts at the top—with the idea—and goes down to the data to determine if the idea is true.

Data science can be hypothesis-driven. That is, as with empirical science, a data science activity can start with a hypothesis to be evaluated. Unlike empirical science, the hypothesis can be stated with less precision and the models, methods, and data can be much larger in scale, that is, more variables, data volume, velocity, and variety. In comparison, data science accelerates discovery by rapidly reducing a vastly larger search space than would have been considered for empirical methods to a small set of likely correlations; however, unlike empirical science, the results are correlations that require additional, non-data science methods to achieve definitive, causal results.

One of the greatest advantages of data science is that it can discover patterns or correlations in data at scale vastly beyond human intellectual, let alone temporal,

capacity; far beyond what humans could have conceived. Of course, a vast subset of those found may be entirely spurious. Data science can use inductive reasoning, also called bottom-up, data-driven, or fact-based analysis, not to evaluate specific hypotheses but using an analytical model and method to identify patterns or correlations that occur in the data with a specific frequency. If the frequency meets some predefined specification, for example, statistical significance in the domain being analyzed, it can be interpreted as a measure of likelihood of the pattern being real. As opposed to evaluating pre-defined hypotheses in the theory-driven approach, the data-driven approach is often said to "automatically" generate hypotheses, as in Nagarajan et al. (2015). The inductive capacity of data science is often touted as its magic as the machine or methods such as machine learning "automatically" and efficiently discover likely hypotheses from the data. While the acceleration and the scale of data being analyzed are major breakthroughs in discovery, the magic should be moderated by the fact that the discovered hypotheses are derived from the models and methods used to discover them. The appearance of magic may derive from the fact that we may not understand how some analytical methods, for example, some machine learning and deep learning methods, derive their results. This is a fundamental data science research challenge as we would like to understand the reasoning that led to a discovery, as is required in medicine, and in 2018 in the European Union, by law [the General Data Protection Regulation (GDPR[15])].

3.6 Data Science Is in Its Infancy

The excitement around data science and its many successes are wonderful, and the potential of data science is great, but these positive signs can be misleading. Not only is data science in its infancy as a science and a discipline, its current practice has a large learning curve related largely to the issues raised above. Gartner, Forrester, and other technology analysts report that most (80%) early (2010–2012) data science projects in most US enterprises failed. In late 2016, Gartner reported that while most enterprises declare data science as a core expertise, only 15% claim to have deployed big data projects in their organization (Gartner 2016). Analysts predict 80+% failure rate through 2017 (Demirkan and Dal 2014; Veeramachaneni 2016; Lohr and Singer 2016).

3.7 It's More Complicated Than That

Data science methods are more sophisticated than the above descriptions suggest, and data-driven analyses are not as pure. Data science analytical methods and

[15] https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

models do not discover any and all correlations that exist in the data since they are discovered using algorithms and models that incorporate some hypotheses that could be considered biases. That is, you discover what the models and methods are designed to discover. One must be objective in data science across the entire workflow—data selection, preparation, modelling, analysis, and interpretation; hence, a data scientist must always doubt and verify (Brodie 2015b).

It may be useful to experiment with the models and methods. When a data science analysis reduces a vast search space, it (or the observing human) may learn something about the discovered correlations and may warrant an adjustment and a re-run of the model, the method, or even adjusting the data set. Hence, iterative learning cycles may increase the efficacy of the analysis or simply provide a means of exploring the data science analysis search space.

Top-down and bottom-up analytical methods can be used in combination, as follows. Start with a bottom-up analysis that produces *N* candidate correlations. Select a subset of *K* of the correlations with an acceptable likelihood and treat them as hypotheses to be evaluated. Then use them to run hypothesis-driven data science analyses and determine, based on the results, which hypotheses are again the most likely or perhaps even more likely than the previous run and discard the rest. These results can be used in turn to redesign the data science analysis, for example, iteratively modify the data, model, and method, and repeat the cycle. This approach is used to explore data, models, and methods—the main components of a data science activity. This method of combining top-down and bottom-up analysis has been proposed by CancerCommons, as a method for accelerating the development of cancer cures as part of the emerging field of translational medicine.[16]

4 Data Science Components

Extending the analogy with science and the scientific method, data science, when mature, will be a systematic discipline with components that are applicable to most domains—to most human endeavors. There are four categories of data science components, all emergent in the data science context awaiting research and development: (1) principles, data, models, and methods; (2) data science pipelines; (3) data science infrastructure; and (4) data infrastructure. Below, we discuss these components in terms of their support of a specific data science activity.

Successful data science activities have developed and deployed these components specific to their domain and analysis. To be considered a science, these components must be generalized across multiple domains, just as the scientific method applies to most scientific domains, and in the last century has been applied to domains previously not considered scientific, for example, economics, humanities, literature, psychology, sociology, and history.

[16]The National Center for Advancing Translational Sciences, https://ncats.nih.gov

4.1 Data Science Principles, Data, Models, and Methods

A data science activity must be based on ***data science principles***, ***models***, and *analytical* ***methods***. Principles include those of science and of the scientific method applied to data science, for example, deductive and inductive reasoning, objectivity or lack of bias relative to a given factor, reproducibility, and provenance. Particularly important are collaborative and cross-disciplinary methods. How do scientific principles apply to discovery over data? What principles underlie evidence-based reasoning for planning, predicting, decision-making, and policy-making in a specific domain?

In May 2017, the *Economist* declared, on its front cover, that *data* was *The World's Most Valuable Resource* (Economist 2017b). Without data there would be no data science or any of its benefits. Data management has been a cornerstone of computer science technology, education, and research for over 50 years, yet Big Data that is fueling data science is typically defined as data at volumes, velocities, and variety that cannot be handled by data management technology. A simple example is that data management functions in preparing data for data analysis take 80% of the resources and time for most data science activities. Data management research is in the process of flipping that ratio so that 80% of resources can be devoted to analysis. Discovering data required for a data science activity whether inside or outside an organization is far worse. Fundamental data research is required in each step of the data science pipeline to realize the benefits of data science.

A data science activity uses one or more models. A model represents the parameters that are the critical properties of the phenomenon to be analyzed. It often takes multiple models to capture all relevant features. For example, the LIGO experiment, that won the 2017 Nobel Prize in Physics for empirically establishing the existence of Einstein's gravitational waves, had to distinguish movement from gravitational waves from seismic activity and 1,00,000 other types of movement. LIGO required a model for each movement type so as to recognize it in the data and discard it as gravitational wave activity. Models are typically domain specific, for example, seismic versus sonic, and are often already established in the domain. Increasingly, models are developed specifically for a data science activity, for example, feature extraction from a data set is common for many AI methods. Data science activities often require the continuous refinement of a model to meet the analytical requirements of the activity. This leads to the need for model management to capture the settings and results of the planned and evaluated model variations. It is increasingly common, as in biology, to use multiple, distinct models, called an ensemble of models, each of which provides insights from a particular perspective. Each model, like each person in Plato's Allegory of the Cave, represents a different perspective of the same phenomenon, what Plato called shadows. Each model—each person—observes what appears to be the same phenomenon, yet each sees it differently. No one model—person—sees the entire thing, yet collectively they capture the whole phenomenon from many perspectives. It may also be that a critical perspective is missed. It is rarely necessary, feasible, or of value to integrate different

perspectives into a single integrated model. After all, there is no ultimate or truthful model save the phenomenon itself. Ensemble or shadow modelling is a natural and nuanced form of data integration (Liu 2012) analogous to ensemble modelling in biology and ensemble learning (Dietterich 2000) and forecasting in other domains.

A data science activity can involve many analytical methods. A given method or algorithm is designed to analyze specific features of a data set. There are often variations of a method depending on the characteristics of the data set, for example, sparse or dense, uniform or skewed, data type, data volume, etc., hence methods must be selected, or created, and tuned for the data set and analytical requirements, and validated. In an analysis, there could be as many methods as there are specific features with corresponding specific data set types. Compared with analytical methods in science, their definition, selection, tuning, and validation in data science often involves scale in choice and computational requirements. Unless they are experts in the related methods, it is unlikely that a practicing data scientist understands the analytical method, for example, a specific machine-learning approach, that they are applying relative to the analysis and data characteristics, let alone the thousands of available alternatives. Anecdotally, I have found that many practicing data scientists use the algorithms that they were taught rather than selecting the one most applicable to the analysis at hand. There are significant challenges in applying sophisticated analytical models and methods in business (Forrester 2015). Having selected or created and refined the appropriate model, that is, collection of features that determine the data to be collected, then collected and prepared the data to comply with the requirements of the model, and then selected and refined the appropriate analytical method, the next challenge is interpreting the results and, based on the data, model, and method, evaluating the likelihood, within relevant error bounds, that the results are meaningful hypotheses worthy of validating by other means.

4.2 Data Science Workflows or Pipelines

The central organizing principle of a data science activity is its ***workflow*** or ***pipeline*** and its life cycle management (NSF 2016). A data science pipeline is an end-to-end sequence of steps from data discovery to the publication of the qualified, probabilistic interpretation of the result in the form of a data product. A generic data science pipeline, such as listed below, is comprehensive of all data science activities, and hence can be used to define the *scope of data science*.

1. Raw data discovery, acquisition, preparation, and storage as curated data in data repositories
2. Selection and acquisition of curated data from data repositories for data analysis
3. Data analysis
4. Results interpretation
5. Result publication and optionally operationalization of the pipeline for continuous analyses

The state of the art of data science is such that every data science activity has its own unique pipeline, as each data science activity is unique. Due to the emergence and broad applicability of data science, there is far more variation across data science pipelines than across conventional science pipelines. Data science will benefit, as it develops, from a better understanding of pipelines and guidance on their design and development.

Data science pipelines are often considered only in terms of the analytics, for example, the machine-learning algorithms used to derive the results in step 3. However, most of the resources required to design, tune, and execute a data science activity are required not for data analysis, steps 3 and 4 of a data science pipeline, but for the design and development of the pipeline and for steps 1 and 2.

The design, development, and tuning of an end-to-end pipeline for a data science activity typically poses significant data modelling, preparation, and management challenges often requiring significant resources and time required to develop and execute a data science activity. Two examples are astrophysical experiments, the Kepler Space Telescope launched in 2009 to find exoplanets and LIGO that was awarded the 2017 Nobel Prize in Physics. Initial versions of the experiments failed not because of analysis and astrophysical aspects and models, but due to the data pipelines. Due to unanticipated issues with the data, the Kepler Science Pipeline had to be rewritten (Jenkins et al. 2010) while Kepler was inflight retaining all data for subsequent corrected processing. Similarly, earth-based LIGO's pipeline was rewritten (Singh et al. 2007) and renamed Advanced LIGO. Tuning or replacing the faulty pipelines delayed both experiments by approximately one year.

Once the data has been acquired, the most time-consuming activity in developing a pipeline was data preparation. Early data science activities in 2003 reported 80–90% of resources devoted to data preparation (Dasu and Johnson 2003). By 2014 this was reduced to 50–80% (Lohr 2014). In specific cases, this cost negatively impacted some domains (Reimsbach-Kounatze 2015) due to the massive growth of acquired data. As data science blossomed so did data volumes, leading experts in 2015 to analyze the state of the art and estimating that data preparation typically consumed 80% of resources (Castanedo 2015). By then products to curate data at scale, such as Tamr.com, were maturing and being more widely adopted. Due to the visibility of data science, the popular press surveyed data scientists to confirm the 80% estimates (Press 2016; Thakur 2016). In 2017, technical evaluations of data preparation products and their use again identified the 2003 estimates of 80% (Mayo 2017; Gartner G00315888 2017).

4.3 Data Science and Data Infrastructures

The core technical component for a data science activity is a ***data science infrastructure*** that supports the steps of the data science pipeline throughout its life cycle. A data science infrastructure consists of a workflow platform that supports the definition, refinement, execution, and reporting of data science activities in the

pipeline. The workflow platform is supported by the infrastructure required to support workflow tasks such as data discovery, data mining, data preparation, data management, networking, libraries of analytical models and analytical methods, visualization, etc. To support user productivity, a user interface is required for each class of user, each with their own user experience. There are more than 60 such data science platforms—a new class of product—of which 16 meet analysts' requirements (Gartner G00301536 2017; Gartner G00326671 2017; Forrester 2017). These products are complex with over 15 component products such as database management, model management, machine learning, advanced analytics, data exploration, visualization, and data preparation. The large number of products reflects the desire to get into a potentially large, emerging market, regardless of their current ability to support data science.[17]

Data, the world's most valuable resource (Economist 2017b), is also the most valuable resource for the data science activities of an organization (e.g., commercial, educational, research, governmental) and for entire communities. While new data is always required for an existing or new data science activity, data science activities of an organization require a ***data infrastructure***—a sustainable, robust data infrastructure consisting of repositories of raw and curated data required to support the data requirements of the organization's data science activities with the associated support processes such as data stewardship. Many organizations are just developing data infrastructures for data science, aka data science platforms. The best known are those that support large research communities. The US National Research Foundation is developing the *Sustainable Digital Data Preservation and Access Network Partners* to support data science for national science and engineering research and education. The 1000 Genomes Project Consortium created the world's largest catalog of genomic differences among humans, providing researchers worldwide with powerful clues to help them establish why some people are susceptible to various diseases. There are more than ten additional genomics data infrastructures, including the Cancer Genome Atlas of the US National Institutes of Health, Intel's Collaborative Cancer Cloud, and the Seven Bridges Cancer Cloud. Amazon hosts[18] the 1000 Genome Project and 30 other public data infrastructures on topics such as geospatial and environmental datasets, genomics and life science datasets, and datasets for machine learning. The Swiss Data Science Center started developing the Renga platform[19] to support data scientists with their complete workflow.

[17]By 1983 in response to the then emerging technology of relational database management systems (DBMSs) there were over 100 Relational DBMSs of which five survived.

[18]https://aws.amazon.com/public-datasets/

[19]https://datascience.ch/renga-platform/

5 What Is the Method for Conducting Data Science?

A data science activity is developed based on data science principles, models, and analytical methods. The result of its design and development is a data science pipeline that will operate on a data science infrastructure, or platform, and will access data in a data infrastructure. There are a myriad of design and development methods to get from the principles to the pipeline. What follows is a description of a fairly generic data science method.

The ***data science method***, until better alternatives arise, is modelled on the scientific method. The following is one example of applying the empirical approach to data science analysis, analogous to experimental design for science experiments. Each step requires verification, for example, using experts, published literature, previous analysis, and continuous iterative improvement to reach results that meet a predefined specification. Each step may require revisiting a previous step, depending on its outcome. As with any scientific analysis, every attempt should be made to avoid bias, namely, attempting to prove preconceived ideas beyond the model, methods, and hypotheses. The method may run for hours to days for a small analysis; months, as for the Baylor-Watson drug discovery (Spangler et al. 2014); or years, as for the Kepler Space Telescope and LIGO. Design and development times can be similar to run times. Otto for example, a German e-commerce merchant, developed over months an AI-based system that predicts with 90% accuracy what products will be sold in the next 30 days and a companion system that automatically purchases over 2,00,000 products[20] a month from third-party brands without human intervention. Otto selected, modified, and tuned a deep-learning algorithm originally designed for particle-physics experiments at CERN (Economist 2017a). These systems run continuously.

5.1 A Generic Data Science Method[21]

1. Identify the phenomena or problem to be investigated. What is the desired outcome?
2. Using domain knowledge, define the problem in terms of features that represent the critical factors or parameters to be analyzed (the WHAT of your analysis, that collectively form the model), based on the data likely to be available for the analysis. Understanding the domain precedes defining hypotheses to avoid bias.
3. If the analysis is to be top-down, formulate the hypotheses to be evaluated over the parameters and models.

[20]Stock Keeping Units (SKUs).

[21]This set of steps was derived from analyzing over 150 data science activities. Its purpose is as a basis for guidance for those new to data science and as one alternative to data scientists looking for commonality across domains.

4. Design the analysis in terms of an end-to-end workflow or pipeline from the data discovery and acquisition, through analysis and results interpretation. The analysis should be designed to identify probabilistically significant correlations (*what*) and set requirements for acceptable likelihoods and error bounds.
5. Ensure the conceptual validity of the data analysis design.
6. Design, test, and evaluate each step in the pipeline, selecting the relevant methods, that is, class of relevant algorithms, in preparation for developing the following steps.
   (a) Discover, acquire, and prepare data required for the parameters and models ensuring that the results are consistent with previous steps.
   (b) For each analytical method, select and tune the relevant algorithm to meet the analytical requirements. This and the previous step are highly interrelated and often executed iteratively until the requirements are met with test or training data.
   (c) Ensure the validity of the data analysis implementation.
7. Execute the pipeline ensuring that requirements, for example, probabilities and error bounds, are met.
8. Ensure empirical (common sense) validation—the validity of the results with respect to the phenomena being investigated.
9. Interpret the results with respect to the models, methods, and data analytic requirements. Evaluate the results (patterns or correlations) that meet the requirements for causality to be validated by methods outside data science.
10. If the pipeline is to operate continuously, operationalize and monitor the pipeline and its results.

6 What Is Data Science in Practice?

Each data science activity develops its own unique data science method. Three very successful data science activities are described below in point form descriptions, using the above terminology to illustrate the components of data science in practice. They were conducted over 18, 20, and 2 years, respectively. Their data science pipelines operated for 4 years, 3 years (to date), and 3 months, respectively.

6.1 Kepler Space Telescope: Discovering Exoplanets

The Kepler Space Telescope, initiated in 1999, and its successor project K2, have catalogued thousands of exoplanets by means of data analytics over Big Data. A

detailed description of Kepler and access to its data is at NASA's Kepler & K2 Website.[22]

- **Objective and phenomenon**: Discover exoplanets in telescopic images
- **Project**: NASA-led collaboration of US government agencies, universities, and companies.
- **Critical parameters**: Over 100, for example, planet luminosity, temperature, planet location relative to its sun.
- **Models**: There are over 30 established astrophysical models. A key Kepler model is the relationship between luminosity, size, and temperature. This model was established a century ago by Ejnar Hertzsprung and Henry Russell. This illustrates the fact that data science involves many models and analytical methods that have nothing to do with AI.
- **Methods**: Over 100, for example, multi-scale Bayesian Maximum A Priori method used for systematic error removal from raw data. AI was not a principle method in this project.
- **Hypotheses** (stated in Kepler documents as a query): Five, including "Determine the percentage of terrestrial and larger planets that are in or near the habitable zone of a wide variety of stars."
- **Data**: 100s of data types described in the Data Characteristics Handbook[23] in the NASA Exoplanet Archive.[24]
- **Pipeline**: The *Kepler Science Pipeline*[25] failed almost immediately after launch due to temperature and other unanticipated issues. After being repaired from earth, it worked well for 4 years.
- **Data discovery and acquisition**: Required approximately 90% of the total effort and resources.
- **Algorithm selecting and tuning**: Models and methods were selected, developed, tuned, and tested for the decade from project inception in 1999 to satellite launch in 2009 and were refined continuously.
- **Verification:** Every model and method were verified, for example, exoplanet observations were verified using the Keck observatory in Hawaii.
- **Probabilistic outcomes**[26]

  Kepler:

  - Candidates (<95%): 4496
  - Confirmed (>99%): 2330
  - Confirmed: <2X Earth-size in habitable zone: 30
  - Probably (<99%): 1285
  - Probably not (~99%): 707

[22] https://keplerscience.arc.nasa.gov/

[23] https://archive.stsci.edu/kepler/manuals/Data_Characteristics.pdf

[24] https://exoplanetarchive.ipac.caltech.edu/docs/KeplerMission.html

[25] https://keplerscience.arc.nasa.gov/pipeline.html

[26] Kepler's data is available at http://exoplanetarchive.ipac.caltech.edu

K2:

- Candidate (<95%): 521
- Confirmed (>99%): 140

6.2 LIGO: Detecting Gravitational Waves

The LIGO project detected cosmic gravitational waves predicted by Einstein's 1916 Theory of General Relativity for which its originators were awarded the 2017 Nobel Prize. Project information and its data are available at the LIGO Scientific Collaboration website.[27]

- **Objective and phenomenon**: Observe cosmic gravitational waves.
- **Project**: Initiated in 1997 with 1000 scientists in 100 institutes across 18 countries.
- **Equipment**: Laser Interferometer Gravitational-Wave Observatory (world's most sensitive detector).
- **Go Live:** September 2015 (after a massive upgrade).
- **Data:** 1,00,000 channels of measurement of which one is for gravitational waves.
- **Models**: At least one model per channel.
- **Methods**: At least one data analysis method per data type being analyzed. Initially, AI was not used. In the past 2 years, machine learning has been found to be very effective in many areas, for example, detector malfunctions, earthquake detection.
- **Challenges:** Equipment and pipeline (as is typical in data science activities).
- **Results:**
  - In September 2015 (moments after reboot following the massive upgrade), a gravitational wave, ripples in the fabric of space-time, was detected and estimated to be the result of two black holes colliding 1.3 billion light years from Earth.
  - Since then, four more gravitational waves were detected, one as this chapter went to press.
- **Collaboration**: The project depended on continuous collaboration between experimentalists who developed the equipment and theorists who defined what a signal from two black holes colliding would look like, let alone collaboration scientists, institutes, and countries.

[27] http://www.ligo.org/

6.3 *Baylor-Watson: Cancer Drug Discovery*

The Baylor-Watson drug discovery project (Spangler et al. 2014) is a wonderful example of data-driven discovery and automatic hypothesis generation that discovered two novel kinases as potential sources for cancer drug development. These results that were determined to have a very high likelihood of success were developed in 3 months using IBM's Watson compared with the typical multi-year efforts that typically discover one candidate in 2 years.

- **Objective and phenomenon**: Discover kinases that regulate protein p53 to reduce or stem cancerous cell growth that have not yet been evaluated as a potential cancer drug.
- **Project**: Two years starting in 2012 between IBM Watson and the Baylor College of Medicine.
- **Equipment**: Watson as a data science platform; PubMed as data repository containing a corpus of 23M medical research articles.
- **Data:** 23M abstracts reduced to 240,00 papers on kinases reduced to 70,000 papers on kinases that regulate protein p53.
- **Hypothesis**: Some of 500 kinases in the corpus regulate p53 and have not yet been used for drugs.
- **AI models/methods:** Network analysis (Nagarajan et al. 2015) including textual analysis, graphical models of proteins and kinases, and similarity analysis.
- **Pipeline**: Explore, interpret, and analyze
  - **Explore:** Scan abstracts to select kinase papers using text signatures.
  - **Interpret:** Extract kinase entities from papers and build connected graph of similarity among kinases.
  - **Analyze**: Diffuse annotations over kinases to rank order the best candidates for further experimentation.
- **Data discovery and acquisition:** Textual analysis of PubMed.
- **Challenge:** Designing, developing, and tuning models and methods to scan abstracts for relevant papers; to construct a graphical model of the relevant relationships; to select kinases that regulate p53.
- **Execution:** 3 months.
- **Results**: Two potential cancer drugs in 3 months versus 1 every 2 years (acceleration).
- **Validation**: The methods discovered 9 kinases of interest analyzing the corpus up to 2003; 7 of 9 were empirically verified in the period 2003–2013. This raised the probability that the remaining two that had not yet been verified clinically were highly likely candidates.
- **Causality**: Work is underway to develop drugs that use the kinases to regulate p53 to stem or reduce cancerous cell growth.
- **Collaboration**: The project involved collaboration between genetic researchers, oncologists, experts in AI and natural language understanding, and computer scientists.

7 How Important Is Collaboration in Data Science?

Data science is an inherently multidisciplinary activity, just as most human endeavors require knowledge, expertise, methods, and tools from multiple disciplines. Analyzing real-world phenomena requires multidisciplinary approaches, for example, how can you analyze the politics of a significant event without considering the economic factors (Brodie 2015c)? Data science requires expertise from multiple disciplines, from the subject domain, statistics, AI, analytics, mathematics, computing, and many more. However, multidisciplinary collaboration is especially critical for success at this early time in the emergence of data science. Success and advancement in research and industry are typically based on competitive achievements of individual people or teams rather than on collaboration. While collaboration and multidisciplinary thinking are praised, they are seldom taught or practiced. Successful data science requires a behavior change from competition to collaboration.

For disciplines required by scientific activities, there are well-established principles, methods, and tools from each discipline as well as how they are applied across scientific workflows. Collaboration was built into these mature disciplines and workflows years ago. In contrast, the principles, methods, and tools for each relevant discipline are just emerging for data science, as are methods of collaboration across workflows.

Currently, data science requires a data scientist to know the sources, conditions, and nature of the data to ensure that the domain-specific model has the appropriate data. Rather than becoming a data expert, the data scientist collaborates with a data expert. Rather than becoming an AI expert, a data scientist may need to collaborate with an AI expert to ensure the appropriate analytical methods are used. There can be as many as ten[28] disciplines involved in such an activity. Two current challenges in this regard are: (1) the shortage of data science-savvy experts, and (2) moving from a world of individual work to one of collaboration. Both challenges are being addressed by universities and institutes worldwide; however, the knowledge, as discussed above, and the teachers are themselves new to this game.

The need for collaboration on basic research and engineering on the fundamental building blocks of data science and data science infrastructures can be seen in a recent report from the University of California, Berkeley, researchers (Stoica et al. 2017). The report is a collaborative effort from experts from many domains—statistics, AI, data management, systems, security, data centers, distributed computing, and more.

Data science activities have emerged in most research labs in most universities and national research labs. Until 2017, many Harvard University departments had

[28]Ten is a somewhat arbitrary number chosen because most pipelines involve 5–10 expert tasks. The actual number of required disciplines varies significantly from simple analyses (e.g., reordering products for an online retailer) to very sophisticated (e.g., LIGO required analysis of ~1000 sources of motion).

one or more groups conducting data science research and offered a myriad of data science degrees and certificates. In March 2017, the Harvard Data Science initiative[29] was established to coordinate the many activities. This pattern has repeated at over 120 major universities worldwide, resulting in over 150 Data Science Research Institutes (DSRIs)[30] being established since 2015—themselves just emerging. The creation of over 150 DSRIs in approximately 2 years, most heavily funded by governments and by partner industrial organizations, is an indication of the belief in the potential of data science not just as a new discovery paradigm but as a basis for business and economic growth.

Collaboration is an emerging challenge in data science not only at the scientific level but also at the strategic and organizational levels. Analysts report that most early industry big data deployments failed due to a lack of domain-business-analytics-IT collaboration (Forrester 2015). Most of the over 150 DSRIs involve a grouping of departments or groups with an interest in data science, each in their own domain, into a higher level DSRI. A large example is the Fraunhofer Big Data Alliance,[31] which in the above terminology would be a DSRI of DSRIs and describes itself as: "The Fraunhofer Big Data Alliance consists of 30 institutes bundling their cross-sector competencies. Their expertise ranges from market-oriented big data solutions for individual problems to the professional education of data scientists and big data specialists."

In principle, a DSRI would strive for higher-level, scientific, and strategic goals, such as contributing to data science (i.e., the science underlying data science) in contrast with the contributions made in a specific domain by each partner organization. But how does the DSRI operate? How should it be organized so as to encourage collaboration and achieving higher-level goals?

While data science is inherently multi-disciplinary, hence collaborative, in nature, scientists and practitioners lack training in collaboration and are motivated to focus on their objectives and domain. Why would a bioinformaticist (bioinformatician) attempt to establish a data science method that goes beyond her requirements, especially as it requires an understanding of domains such as deep learning? Collaboration is also a significant organizational challenge specifically for the over 150 DSRIs that were formed as a federation of organizational units each of which conducts data science activities in different domains. Like the bioinformaticist, each organization has its own objectives, budget, and investments in funding and intellectual property. In such an environment, how does a DSRI establish strategic directions and set research objectives? One proposal is through a DSRI Chief Scientific Officer (Brodie 2019).

[29] https://datascience.harvard.edu/

[30] The DSRI list that I maintain by searching the web grows continuously—an excellent exercise for the reader.

[31] https://www.bigdata.fraunhofer.de/en.html

8 What Is World-Class Data Science Research?

While many data science groups share a passion for data science, they do not share common data science components—principles, data, models, and methods; pipelines; data science infrastructures; and data infrastructures. This is understandable given the state of data science and the research needs of the individual groups; however, to what extent are these groups pursuing data science, per se? This raises our original questions: *What is data science? What is world-class data science research?* These questions are central to planning and directing data science research such as in DSRIs.

There are two types of data science research: domain-specific contributions and contributions to the discipline of data science itself. Domain-specific, world class data science research concerns applications of data science in specific domains resulting in domain-specific discoveries that are recognized in its domain as being world class. There are many compelling examples, as in Sect. 6. To be considered data science, the research should adhere to the definition of data science, be based on some version of the data science method, use a data science pipeline, and utilize the components of data science. The data science components or the data science method, including scale, accelerating discovering, finding solutions that might not have been discovered otherwise, should be critical to achieving the result in comparison with other methods.

Equally or even more important, world class data science research should establish data science as a science or as a discipline with robust principles, data, models, and methods; pipelines; a data science method supported by robust data science infrastructures; and data infrastructures applicable to multiple domains. Such a contribution must be proven with appropriate applications of the first type. A wonderful example of generalizing a domain-specific data science method is extending the network analysis method applied to some specific medical corpora used successfully in drug discovery (Spangler et al. 2014) to domain-independent scientific discovery applied to arbitrary scientific corpora (Nagarajan et al. 2015). The original method was implemented in three stages—exploration, interpretation, and analysis—using a tool called Knowledge Integration Toolkit (**KnIT**). Exploration involved lexical analysis and text mining of abstracts of the entire corpora up to 2003 (2,40,000) of medical literature mentioning kinases, a type of protein that governs cell growth, looking for proteins that govern p53, a tumor suppressor. This resulted in 70,000 papers to analyze further. Interpretation analyzed some of the papers to produce a model of each kinase and built a connected graph that represents the similarity relationship among kinases. The analysis phase identified and eliminated kinases that are not p53, ultimately resulting in discovering nine kinases with the desired properties. A retrospective search of the literature verified that seven of the nine were proven empirically to be tumor suppressors (candidates for cancer drugs) in papers published in 2003–2013. This significantly raised the probability that the two remaining kinases were as yet undiscovered candidates for cancer drugs. These were world-class data science results and a magnificent example of analysis

involving complexity beyond human cognition. First and foremost, the two kinases were accepted by the medical community as candidate tumor suppressors, that is, published in medical journals. Second, the discovery was due to data science methods. Data science accelerated discovery since typically one such cancer drug candidate is found every 2–3 years; once the KnIT model was built, the candidate kinases were discovered in approximately 3 months. The verification method, the retrospective analysis of cancer drug discovery 2003–2013 was brilliant. As with most data science analysis, the results were probabilistic, that is, the nine candidate kinases were determined to likely candidates by the network model of the kinases, however, verification, or further confirmation, was established by a method outside data science altogether, that is, discovered previously published results. The original analytical method that provided automated hypothesis generation (i.e., these kinases are similar) based on text mining of medical corpora concerning proteins was generalized to automated hypothesis generation based on text mining of any scientific corpora. While the first result was domain-specific, hence an application of data science, the extension of the domain-specific method to all scientific domains was *a contribution to the science of data science*. This is a higher level of world-class data science research.

The charter of every DSRI should include both domain-specific data science research and research to establish data science as a discipline. Since most DSRIs were formed from groups successfully practicing domain-specific data science, they are all striving for world class domain-specific data science. Without world class research in data science per se, it would be hard to argue that the DSRI contributes more than the sum of its parts. One might argue that lacking research into data science per se means that the DSRI has more of an organizational or marketing purpose than a research focus. The primary objective of a significant portion of the 150 DSRIs referenced above appears to be organizational, for example, to bring together the various organizations that conduct data science. In contrast, in 2012 the Irish Government established Insight Center for Data Analytics as a national DSRI to conduct data science research and apply it in domains relevant to Ireland's future. In doing so, it set objectives much higher than bringing together data science activities from its seven universities. The government of Ireland, through its funding agency, Science Foundation Ireland (SFI), continuously evaluates Insight on world class data science. This includes advancing data science principles, data, models, and methods and proving their value by achieving results in health and human performance, enterprises and services, smart communities and Internet of things, and sustainability. More challenging, however, SFI requires that Insight contributes more than the sum of the parts, the individual units working on their own. This contributes to the science of data science by developing principles, data models, methods, pipelines, and infrastructure that is applicable to multiple domains.

9 Conclusions

Data science is an emerging paradigm with the primary advantage of accelerating the discovery of correlations between variables at a scale and speed beyond human cognition and previous discovery paradigms. Data science differs paradigmatically from its predecessor scientific discovery paradigms that were designed to discover causality—*Why a phenomenon occurred*—in real contexts. Data science is designed to discover correlations—*What phenomena may have or may occur*—in data purported to represent some real or imagined phenomenon. Unlike previous scientific discovery paradigms that were designed for scientific discovery and are now applied in many non-scientific domains, data science is applicable to any domain for which adequate data is available. Hence, the potential of broad applicability and accelerating discovery in any domain to rapidly reduce the search space for solutions holds remarkable potential for all fields. While already applicable and applied successfully in many domains, there are many challenges that must be addressed over the next decade as data science matures.

My decade-long experience in data science suggests that there are no compelling answers to the questions posed in this chapter. This is due in part to its recent emergence, its almost unlimited breadth of applicability, and to its inherently multidisciplinary, collaborative nature.

To warrant the designation *data science*, this emerging paradigm, as a science, requires fundamental principles and techniques applicable to all relevant domains. Since most "data science" work is domain specific, often model- and method-specific, "data science" does not yet warrant the designation of a science. This is not a mere appeal for formalism. There are many challenges facing data science such as validating results thereby minimizing the risks of failures. The potential benefits of data science, for example, in accelerating the discovery of cancer cures and solutions to global warming, warrant establishing rigorous, efficient data science principles and methods that could change our world for the better.

References

Brodie, M. L. (2014a, June). The first law of data science: Do umbrellas cause rain? *KDnuggets*.

Brodie, M. L. (2014b, October). Piketty revisited: Improving economics through data science – How data curation can enable more faithful data science (in much less time). *KDnuggets*.

Brodie, M. L. (2015a, June). Understanding data science: An emerging discipline for data-intensive discovery. In S. Cutt (Ed.), *Getting data right: Tackling the challenges of big data volume and variety*. Sebastopol, CA: O'Reilly Media.

Brodie, M. L. (2015b, July). Doubt and verify: Data science power tools. *KDnuggets*. Republished on ODBMS.org.

Brodie, M. L. (2015c, November). On political economy and data science: When a discipline is not enough. *KDnuggets*. Republished ODBMS.org November 20, 2015.

Brodie, M. L. (2018, January 1). Why understanding truth is important in data science? *KDnuggets*. Republished Experfy.com, February 16, 2018.
Brodie, M. L. (2019). On developing data science, to appear. In M. Braschler, T. Stadelmann, & K. Stockinger (Eds.), *Applied data science – Lessons learned for the data-driven business*. Berlin: Springer.
Cambridge Mobile Telematics. (2018, April 2). *Distraction 2018: Data from over 65 million trips shows that distracted driving is increasing*.
Castanedo, F. (2015, August). *Data preparation in the big data era: Best practices for data integration*. Boston: O'Reilly.
Dasu, T., & Johnson, T. (2003). *Exploratory data mining and cleaning*. Hoboken, NJ: Wiley-IEEE.
Data Science. (2018). *Opportunities to transform chemical sciences and engineering*. A Chemical Sciences Roundtable Workshop, National Academies of Science, February 27–28, 2018.
Demirkan, H., & Dal, B. (2014, July/August). The data economy: Why do so many analytics projects fail? *Analytics Magazine*.
Dietterich, T. G. (2000, June). Ensemble methods in machine learning. In *International workshop on multiple classifier systems* (pp. 1–15). Berlin: Springer.
Dingus, T. A., et al. (2016). Driver crash risk factors and prevalence evaluation using naturalistic driving data. *Proceedings of the National Academy of Sciences, 113*(10), 2636–2641. https://doi.org/10.1073/pnas.1513271113.
Economist. (2017a, April 12). How Germany's Otto uses artificial intelligence. *The Economist*.
Economist. (2017b, May 4). The World's most valuable resource. *The Economist*.
Economist. (2018a, January 6). Many happy returns: New data reveal long-term investment trends. *The Economist*.
Economist. (2018b, February 24). Economists cannot avoid making value judgments: Lessons from the "repugnant" market for organs. *The Economist*.
Economist. (2018c, March 28). In algorithms we trust: How AI is spreading throughout the supply chain. *The Economist*.
Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., & Balakrishnan, H. (2008) The pothole patrol: Using a mobile sensor network for road surface monitoring. In *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services (MobiSys '08)*. ACM, New York, NY.
Forrester. (2015, November 9). Predictions 2016: The path from data to action for marketers: How marketers will elevate systems of insight. *Forrester Research*.
Forrester. (2017, March 7). *The Forrester wave: Predictive analytics and machine learning solutions*, Q1 2017.
Gartner G00301536. (2017, February 14). *2017 magic quadrant for data science platforms*.
Gartner G00310700. (2016, September 19). *Survey analysis: Big data investments begin tapering in 2016*. Gartner.
Gartner G00315888. (2017, December 14). *Market guide for data preparation*. Gartner.
Gartner G00326671. (2017, June 7). *Critical capabilities for data science platforms*. Gartner.
Hey, T., Tansley, S., & Tolle, K. (Eds.). (2009). *The fourth paradigm: Data-intensive scientific discovery* Edited by Microsoft Research.
Jenkins, J. M., Caldwell, D. A., Chandrasekaran, H., Twicken, J. D., Bryson, S. T., Quintana, E. V., et al. (2010). Overview of the Kepler science processing pipeline. *The Astrophysical Journal Letters, 713*(2), L87.
Liu, J. T. (2012). Shadow theory, data model design for data integration. *CoRR, 1209*, 2012. arXiv:1209.2647.
Lohr, S. (2014, August 17). For big-data scientists, 'Janitor Work' is key hurdle to insights. *New York Times*.
Lohr, S., & Singer, N. (2016). How data failed us in calling an election. *The New York Times, 10*, 2016.
Mayo, M. (2017, May 31) Data preparation tips, tricks, and tools: An interview with the insiders. *KDnuggets*.

Nagarajan, M. et al. (2015). Predicting future scientific discoveries based on a networked analysis of the past literature. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '15)*. ACM, New York, NY, pp. 2019–2028.

NSF. (2016, December). Realizing the potential of data science. *Final Report from the National Science Foundation Computer and Information Science and Engineering Advisory Committee Data Science Working Group*.

Pearl, J. (2009a). *Causality: Models, reasoning, and inference*. New York: Cambridge University Press.

Pearl, J. (2009b). Epilogue*: The art and science of cause and effect*. In J. Pearl (Ed.), *Causality: Models, reasoning, and inference* (pp. 401–428). New York: Cambridge University Press.

Pearl, J. (2009c). Causal inference in statistics: An overview. *Statistics Surveys, 3*, 96–146.

Piketty, T. (2014). *Capital in the 21st century*. Cambridge: The Belknap Press.

Press, G. (2016, May 23). Cleaning big data: Most time-consuming, least enjoyable data science task, survey says. *Forbes*.

Reimsbach-Kounatze, C. (2015). *The proliferation of "big data" and implications for official statistics and statistical agencies: A preliminary analysis*. OECD Digital Economy Papers, No. 245, OECD Publishing, Paris. https://doi.org/10.1787/5js7t9wqzvg8-en

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., et al. (2017). Mastering chess and Shogi by self-play with a general reinforcement learning algorithm. *ArXiv E-Prints, cs.AI*.

Singh, G., et al. (2007). *Optimizing workflow data footprint special issue of the scientific programming journal dedicated to dynamic computational workflows: Discovery, optimisation and scheduling*.

Spangler, S., et al. (2014). Automated hypothesis generation based on mining scientific literature. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, pp. 1877–1886.

Stoica, I., et al. (2017, October 16). *A Berkeley view of systems challenges for AI*. Technical Report No. UCB/EECS-2017-159.

Thakur, A. (2016, July 21). *Approaching (almost) any machine learning problem*. The Official Blog of Kaggle.com.

Veeramachaneni, K. (2016, December 7). Why you're not getting value from your data science. *Harvard Business Review*.

Waller, M. A., & Fawcett, S. E. (2013). Data science, predictive analytics, and big data: A revolution that will transform supply chain design and management. *Journal of Business Logistics, 34*, 77–84. https://doi.org/10.1111/jbl.12010.

Winship, C., & Morgan, S. L. (1999). The estimation of causal effects from observational data. *Annual Review of Sociology, 25*(1), 659–706. https://doi.org/10.1146/annurev.soc.25.1.659.

Chapter 9
On Developing Data Science

Michael L. Brodie

Abstract Understanding phenomena based on the facts—on the data—is a touchstone of data science. The power of evidence-based, inductive reasoning distinguishes data science from science. Hence, this chapter argues that, in its initial stages, data science applications and the data science discipline itself be developed inductively and deductively in a virtuous cycle.

The virtues of the *twentieth Century Virtuous Cycle* (aka *virtuous hardware-software cycle,* Intel-Microsoft virtuous cycle) that built the personal computer industry *(National Research Council, The new global ecosystem in advanced computing: Implications for U.S. competitiveness and national security. The National Academies Press, Washington, DC, 2012)* were being grounded in reality and being self-perpetuating—more powerful hardware enabled more powerful software that required more powerful hardware, enabling yet more powerful software, and so forth. Being grounded in reality—solving genuine problems at scale—was critical to its success, as it will be for data science. While it lasted, it was self-perpetuating, due to a constant flow of innovation, and to benefitting all participants—producers, consumers, the industry, the economy, and society. It is a wonderful success story for twentieth Century *applied science*. Given the success of virtuous cycles in developing modern technology, virtuous cycles grounded in reality should be used to develop data science, driven by the wisdom of the sixteenth Century proverb, *Necessity is the mother of invention.*

This chapter explores this hypothesis using the example of the evolution of database management systems over the last 40 years. For the application of data science to be successful and virtuous, it should be grounded in a cycle that encompasses industry (i.e., real problems), research, development, and delivery. This chapter proposes applying the principles and lessons of the virtuous cycle to the development of data science applications; to the development of the data science discipline itself, for example, a data science method; and to the development of data science education; all focusing on the critical role of collaboration in data science

M. L. Brodie (✉)
Computer Science and Artificial Intelligence Laboratory, Massacheusetts Institute of Technology, Cambridge, MA, USA
e-mail: michael@michaelbrodie.com

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_9

research and management, thereby addressing the development challenges faced by the more than 150 Data Science Research Institutes (DSRIs) worldwide. A companion chapter (Brodie, *What is Data Science, in Braschler et al (Eds.), Applied data science – Lessons learned for the data-driven business, Springer 2019*), addresses essential questions that DSRIs should answer in preparation for the developments proposed here: *What is data science*? *What is world-class data science research*?

1 Introduction

Data science is inherently *data-* or *evidence-based analysis*; hence, it is currently an applied science. Data science emerged at the end of the twentieth century as a new paradigm of discovery in science and engineering that used *ad hoc* analytical methods to find correlations in data at scale. While there was science in each analysis, there was little science underlying data science *per se*. Data science is in its infancy and will take a decade to mature as a discipline with underlying scientific principles, methods, and infrastructure (Brodie 2019a). This chapter describes a method by which data scientists and DSRIs might develop data science *as a science* (e.g., fundamentals—principles, models, and methods) and *as a discipline* or *an applied science* (e.g., practices in the development of data science products, as described throughout this book (Braschler et al. 2019). The method is based on the twenty-first century Virtuous Cycle—a cycle of collaboration among industry, research, development, and delivery, for example, to develop and use data science products.

The cycle and its virtues evolved from medieval roots to surface in industry including in the research and development of large-scale computer systems and applications, extended to include product development as a *research and development (R&D) cycle*; now extended to deployment in a *research, development, and delivery (RD&D) cycle*. The cycle is used extensively in academic and industrial computer science research and development, by most technology startups, and is integral to the open source ecosystem (Olson 2019; Palmer 2019). It is used extensively in applied science and education, and increasingly in medical and scientific research and practice. We look at the lessons learned in the development of large-scale computer systems, specifically relational database systems based on a recent analysis (Brodie 2019b), tracing how the virtuous cycle was extended to a larger virtuous cycle of demand, research, product development, deployment, practice, and back again.

Section 2 introduces the twentieth century Virtuous R&D Cycle made famous by Microsoft and Intel. Section 3 extends the cycle to the Twenty-First Century Virtuous RD&D Cycle, illustrated using the mutual development of database management system (DBMS) research and products and extends the cycle to education. Section 4 builds upon this blueprint and applies it to three aspects of data science: concrete data products, the discipline itself, and data science education, and concludes by looking forward. Section 5 illustrates previous themes with lessons learned

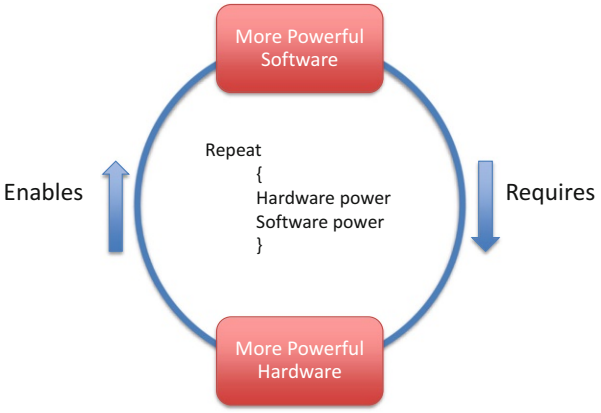

Fig. 9.1 The hardware-software cycle

in the development of data science and DSRIs, and exposing commonly reported data science facts as pure myths. Section 6 speculates on the impacts of data science, both benefits and threats; given the projected significance of data science, there may be more profound impacts. Section 7 concludes optimistically with challenges that lie ahead.

2 Twentieth Century Virtuous Cycles

The twentieth century Virtuous Cycle accelerated the growth of the personal computer industry with more powerful hardware (speed, capacity, miniaturization) that enabled more powerful software (functions, features, ease of use) that in turn required more powerful hardware (Fig. 9.1). Hardware vendors produced faster, cheaper, more powerful hardware (i.e., chips, memory) fueled by Moore's Law. This led software vendors to increase the features and functions of existing and new applications, in turn requiring more speed and memory. Increasing hardware and software power made personal computers more useful and applicable to more users, thus increasing demand and growing the market that in turn, through economies of scale, lowered costs in ever-shortening cycles. But what made the cycle virtuous?

The hardware-software cycle had two main virtues worth emulating. First, the cycle became self-perpetuating driven by a continuous stream of innovation—good hardware ideas, for example, next generation chips, and good software ideas, for example, next great applications (Fig. 9.2). It ended in 2010 (National Research Council 2012) when dramatic hardware gains were exhausted, the market approached saturation, and its fuel—good ideas—was redirected to other technologies. Second, all participants benefited: hardware and software vendors, customers, and more generally the economy and society through the growth of the personal

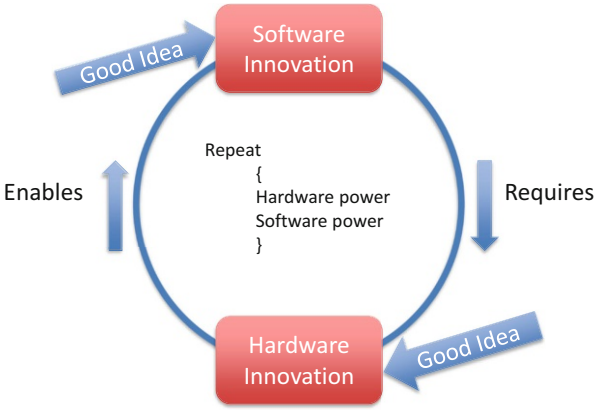

Fig. 9.2 Twentieth century virtuous hardware-software cycle

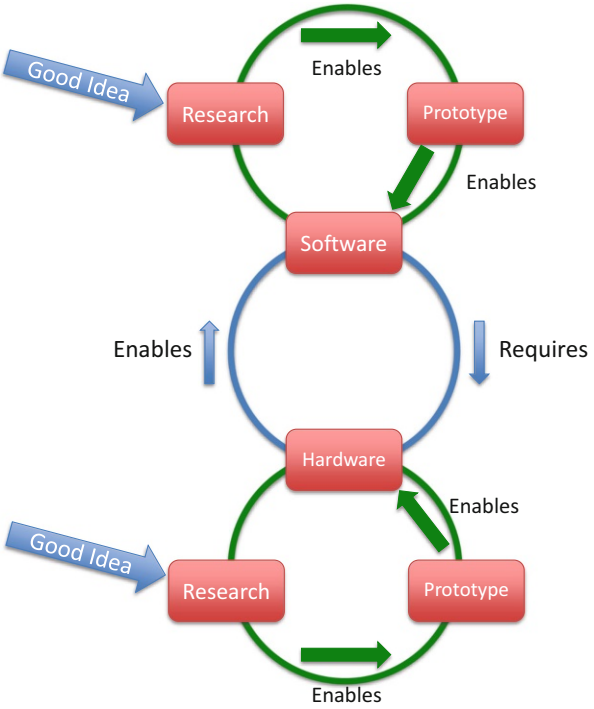

Fig. 9.3 Software and hardware R&D cycles

computer industry and the use of personal computers. The twentieth century Virtuous Cycle was simply *hardware innovation and software innovation in a cycle*.

The virtuous hardware-software cycle produced hardware and software each of which developed its own R&D cycle (Fig. 9.3). Hardware vendors and universities used the hardware (R&D) cycle to address hardware opportunities and challenges by conducting fundamental research into next generation hardware. As long as there was hardware innovation—good ideas—the hardware R&D cycle was virtuous.

Similarly, software vendors used the software R&D cycle to address software challenges and opportunities in their ever-shortening cycles. This also worked well for next generation applications. However, fundamental research into next generation systems, specifically database management systems, was conducted by vendors (e.g., IBM, Software AG, Honeywell Information Systems) and not by universities.

Addressing fundamental DBMS challenges and opportunities in a university requires access to industrial-scale systems, industrial applications, and use cases (i.e., data). Until the early 1970s, universities lacked industrial experience, case studies, and resources such as large-scale systems and programming teams. At that time, Michael Stonebraker at University of California, Berkeley, began to address this gap[1]. Stonebraker and Eugene Wong built Ingres (Stonebraker et al. 1976), a prototype industrial scale relational DBMS (RDBMS) for industrial scale geographic applications. They made the Ingres code line available as one of the first open source systems. The Ingres code line then enabled universities to conduct fundamental systems research. Ingres was the first example in a university of extending the twentieth century Virtuous Cycle to systems engineering, specifically to a DBMSs. The cycle was subsequently extended to large systems research in universities and industry. Due to the importance of the system developed in the process, it became known as the *twentieth century Virtuous R&D Cycle* which simply stated is *research innovation and engineering innovation, in a cycle* (Olson 2019).

3 Twenty-First Century Virtuous Research, Development, and Delivery Cycles

3.1 The Virtuous DBMS RD&D Cycle

Using Ingres for industry-scale geographic applications was a proof of concept of the feasibility of the relational model and RDBMSs. But were they of any value? How real were these solutions? Were relational systems applicable in other domains? These questions would be answered if there were a market for Ingres, that is, a demand. Stonebraker, Wong, and Larry Rowe formed Relational Technology, Inc., later named the Ingres Corporation, to develop and market Ingres. Many companies have used the open source Ingres and Postgres (Stonebraker and Kemnitz 1991) code lines to produce commercial RDBMSs (Naumann 2018) that together with IBM's DB2, Oracle, and Microsoft SQL Server now form a $55 bn per year market, thus demonstrating the value and impact of RDBMSs as a "good idea" (Stonebraker

[1] Stonebraker's DBMS developments coincided with the emergence of the open source movement. Together they created a virtuous cycle that benefited many constituencies—research, DBMS technology, products, applications, users, and the open source movement resulting in a multi-billion-dollar industry. Hence, this example warrants a detailed review as lessons for the development of data science.

2019a, b). This extended the twentieth century Virtuous R&D Cycle to DBMSs in which DBMS research innovation led to DBMS engineering innovation that led to DBMS product innovation. DBMS vendors and universities repeated the cycle resulting in expanding DBMS capabilities, power, and applicability that in turn contributed to building the DBMS market. Just as the hardware-software cycle became virtuous, so did the DBMS R&D cycle. First, research innovation—successive good ideas—led to engineering innovation that led to product innovation. This cycle continues to this day with the emergence of novel DBMS ideas especially with the new demands of Big Data. Second, all participants benefit: vendors, researchers, DBMS users, and more generally the economy using data management products and the growth of the data management industry. Big Data and data science follow directly in this line.

A wonderful example of *necessity being the mother of invention* is the use of abstract data types as the primary means of extending the type system of a DBMS and providing an interface between the type systems of a DBMS and its application systems—arguably Stonebraker's most significant technical contribution. To build an RDBMS based on Ted Codd's famous paper (Codd 1970), Stonebraker and Wong obtained funding for a DBMS to support Geographic Information Systems. They soon discovered that it required point, line, and polygon data types and operations that were not part of Codd's model. Driven by this necessity, Stonebraker chose the emerging idea of abstract data types to extend the built-in type system of a DBMS. This successful innovation has been a core feature of DBMSs ever since. Abstract data types is only one of many innovations that fed the 40-year-old virtuous necessity-innovation-development-product cycle around Ingres and Postgres.

In all such cycles, there is a natural feedback loop. Problems (e.g., recovery and failover), challenges, and opportunities that arose with relational DBMS products fed back to the vendors to improve and enhance the products while more fundamental challenges (e.g., lack of points, lines, and polygons) and opportunities went back to university and vendor research groups for the next cycle of innovation. Indeed, modern cycles use frequent iteration between research, engineering, and products to test or validate ideas, such as the release of beta versions to find "bugs".

Stonebraker, together with legions of open source contributors, extended the twentieth century Virtuous R&D Cycle in several important dimensions to become the *twenty-first century Virtuous Research, Development, and Delivery Cycle*. First, in creating a commercial product he provided a compelling method of demonstrating the value and impact of what was claimed as a "good idea" in terms of demand in a commercial market. This added the now critical delivery step to become the research-development-delivery (RD&D) cycle. Second, as an early proponent of open source software on commodity Unix platforms he created a means by which DBMS researchers and entrepreneurs have access to industrial scale systems for RD&D. Open source software is now a primary method for industry, universities, and entrepreneurs to research, develop, and deliver DBMSs and other systems. Third, by using industry-scale applications as use cases for proofs of concept, he provided a method by which research prototypes could be developed and demonstrated to address industrial-scale applications. Now benchmarks are used for

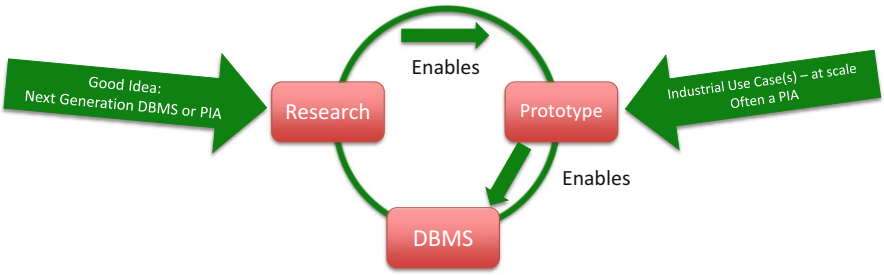

Fig. 9.4 Virtuous DBMS RD&D cycle

important industrial-scale problems as a means of evaluating and comparing systems in industrial-scale contexts. Fourth, and due to the above, his method provided means by which software researchers could engage in fundamental systems research, a means not previously available that is now a critical requirement for large-scale systems research.

The RD&D cycle is used to develop good research ideas into software products with a proven demand. Sometimes the good idea is a pure technical innovation, for example, a column store DBMS*: queries will be much faster if we read only the relevant columns!* This led to the Vertica DBMS (Stonebraker et al. 2005). More often it is a "pain in the ass" (PIA) problem, namely, a genuine problem in a real industrial context for which someone will pay for the development of a solution. Paying for a solution demonstrates the need for a solution and helps fund its development. Here is a real example: A major information service company creates services, for example, news reports, by discovering, curating, de-duplicating, and integrating hundreds of news wire reports from data items that are dirty, heterogeneous, and highly redundant, for example, over 500 reports of a US school shooting in 500 different formats. Due to the Internet, as the number of news data sources soared from hundreds to hundreds of thousands, the largely manual methods would not scale. This PIA problem led to Tamr,[2] a product for curating data at scale.

The RD&D cycle is the process underlying applied science. The RD&D cycle—an applied science method—becomes virtuous as long as there is a continuous flow of good ideas and PIA problems that perpetuate it (Fig. 9.4).

Stonebraker received the 2014 AM Turing Award—"the Nobel prize in computing"—"For fundamental contributions to the concepts and practices underlying modern database systems" (ACM 2015)[3]. Concepts mean good research ideas—DBMS innovations. Practice means taking DBMS innovations across the virtuous RD&D cycle to realize value and create impact. Following the cycle produced the open source Ingres DBMS that resulted in the Ingres DBMS product, and the Ingres

[2]Tamr.com provides tools and services to discover and prepare data at scale, for example, 1,00,000 data sources, for Big Data projects and data science.

[3]The RDBMS RD&D cycle was chosen to illustrate the theme of this chapter, as it is one of the major achievements in computing.

Corporation with a strong market, that is, users who valued the product. Stonebraker refined and applied his method in eight subsequent academic projects and their commercial counterparts: Ingres (Ingres) (Stonebraker et al. 1976), Postgres (Illustra) (Stonebraker and Kemnitz 1991), Mariposa (Cohera), Aurora (StreamBase), C-Store (Vertica) (Stonebraker et al. 2005), Morpheus (Goby), H-Store (VoltDB), SciDB (Paradigm4), and Data Tamer (Tamr) with BigDAWG Polystore and Data Civilizer currently in development. The concepts and practice of this RD&D cycle are a formula for applied science of which Stonebraker's systems are superb examples[4] (Stonebraker 2019a):

Repeat {

Find somebody who is in pain
Figure out how to solve their problem
Build a prototype
Commercialize it
}

The systems research community adopted open source methods and extended the cycle to all types of systems resulting in a *Twenty-First Century Virtuous RD&D Cycle* for systems that transformed academic systems research to deliver greater value for and higher impact in research, industry, and practice.

The *twenty-first century Virtuous Research, Development, and Delivery Cycle* is simply *research innovation, engineering innovation, and product innovation in a cycle*. As we will now see, its application and impacts go well beyond systems RD&D.

3.2 The Critical Role of Research-Industry Collaboration in Technology Innovation

Virtuous RD&D cycles require researchers-industry collaboration that mutually benefits research *and* industry. Industry often needs insight into challenges for which they may not have the research resources. More commonly, industry faces PIA problems for which there are no commercial solutions. As discussed in Sect. 4.2, this is precisely the case for data science today. Most US enterprises have launched data science efforts most of which fail as few in industry understand data science or can hire data scientists. But let us return to understanding the cycles before applying them to data science.

It is common that industry may not be aware of PIA problems that lurk below the surface. For example, all operational DBMSs, more than 5 m in the USA alone,

[4]Don't let the pragmatism of these examples hide the scientific merit. Computer science was significantly advanced by fundamental principles introduced in each of the systems mentioned.

decay due to their continuous evolution to meet changing business requirements. While database decay is a widely known pattern, it has not been accepted as a PIA problem since there is little insight into its causes, let alone technical or commercial solutions. Recent research (Stonebraker et al. 2016a, b, 2017) proposes both causes and solutions that will be realized only with industrial-scale systems and use cases with which to develop, evaluate, and demonstrate that the proposed "good ideas" actually work! Insights into causes and solutions came exclusively through a research-industry collaboration between MIT and B2W Digital, a large Brazilian retailer.

Industry gains in RD&D cycles in several ways. First, industry gains insight into good ideas or challenges being researched. Second, industry gets access to research prototypes to investigate the problem in their environment. Third, if successful, the prototype may become open source[5] available to industry to apply and develop, potentially becoming a commercial product. Fourth, industry can gain ongoing benefits from collaborating with research such as facilitating technology transfer and indicating to customers, management, and investors its pursuit of advanced technology to improve its products and services. Finally, a PIA industry problem may be resolved or a hypothesized opportunity may be realized.

Industry collaboration is even more critical for research, especially for research involving industrial-scale use cases. Researchers need access to genuine, industrial-scale opportunities or, more often, challenges that require research that is beyond the capability or means of industry to address, and to real use cases with which to develop, evaluate, and demonstrate prototype solutions. Scale is important as "the devil is in the details" that arise in industrial-scale challenges and seldom in toy use cases. Through collaboration, research can understand and verify the existence and extent of a problem or the likelihood and potential impact of a good idea by analyzing them in a genuine industrial context. Is the problem real? Is a solution feasible? What might be the impact of the solution? This is precisely what is needed in data science for both researchers and industry.

Ideally, collaboration occurs in a continuous RD&D cycle in which research and industry interact to identify and understand problems, opportunities, and solutions. It is virtuous if all participants benefit and as long as problems and opportunities arise. Such research–industry collaborations are better for technology transfer than conventional marketing and sales (Stonebraker 2019a, b).

By the mid-2000s, startups worldwide used a version of the Twenty-First Century Virtuous RD&D Cycle (Fig. 9.5) as their development method as a natural extension of the open source ecosystem. An obvious example is the World Wide Web that spawned an enormous number of apparently odd innovations. Who knew that a weird application idea like Twitter, a 140-character message service, would become

[5]Open source is not required for research-industry collaborations; however, open source can significantly enhance development, for example, Apache Spark's 42 m contributions from 1567 contributors, and impact, for example, used by over 1 m organizations, due in part to free downloads.

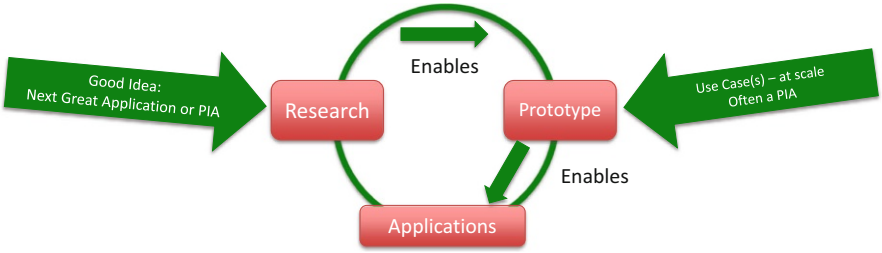

Fig. 9.5 Twenty-first century virtuous RD&D cycle

a thing (weaponized by a US president)? Or Snapchat, an image service where images self-destruct? The virtuous RD&D cycle was used on a much grander scale in the World Wide Web and in Steve Jobs' iPhone both of which went from self-perpetuating to viral and in so doing changed our world. These projects were developed, and continue to be developed, with extensive industry collaboration driven by good—sometimes weird—ideas, novel applications, and PIA problems to be proven at scale. One might argue that the Twenty-First Century Virtuous RD&D Cycle is one of the most effective development methods.

3.3 The Role of Innovation in RD&D Cycles

The virtues of the RD&D cycle apply to data science. First, data science should be grounded in reality by using industrial-scale challenges, opportunities, and use cases to drive the cycle to develop and validate solutions and products to prove value and impact. Second, it should be made self-perpetuating by ensuring a constant flow of innovation, especially in its emerging state—good ideas, challenges, PIA problems, and opportunities—with the result that the methods and results improve, thus benefiting all participants: producers, consumers, the industry, the economy, and society. Innovative ideas perpetuate the cycle, the best innovations accelerate the cycle.

As illustrated in Table 9.1, innovation is required in each stage, for the cycle to be virtuous—to self-perpetuate. There is a two-way flow between cycle stages. Technology, for example, a data science platform, transfers down (→) the cycle in the form of research results, prototypes, and products, while requirements transfer up (←) the cycle in the form of use cases, PIA problems, opportunities, challenges, and user requirements. Innovation—good ideas—can enter anywhere in the cycle, but must continuously enter for the cycle to self-perpetuate.

The cycle also applies to education—understanding *How* each stage works and educating participants in its successful application. For data science education, understanding *How* stages work leads to data science theories in research, to data science architectures and mechanisms in engineering, to data science products in

Table 9.1 The flow of good ideas in virtuous cycles

Activity	**Research**		**Engineering**		**Development**		**Delivery**
Result	**Publication**		**Prototype**		**Product**		**Application/ Use case**
Applied to technology							
Twentieth century hardware-software R&D cycle	Innovation	←→			Innovation		
Twentieth century infrastructure/Systems RD&D cycle	Innovation	←→	Innovation	←→	Innovation		
Twenty-first century RD&D cycle	Innovation	←→	Innovation	←→	Innovation	←→	Innovation
Applied to research, and education, and technology transfer							
Education	How	←→	How	←→	How	←→	How
Research and technology transfer	Innovation	←→	Innovation	←→	Innovation	←→	Innovation

development, and to data science applications in practice. Education also benefits from a two-way flow between theories in research, architectures in engineering, products in development, and use cases in practice. Innovation—good ideas—can enter anywhere in the cycle.

Education in an established domain such as DBMSs involves understanding the principles and techniques and *How* they work. Innovation for education across the cycle concerns innovation not only in data science *per se* but also in education—how data science is taught and understood. Research and technology transfer across the cycle requires innovation in each stage. The cycle is more dynamic and powerful in an emerging domain such as data science. Each stage in data science is in its infancy; hence each stage in research could involve developing, generalizing, and integrating the current results in that stage—principles, platforms, products, and practice. Applying virtuous cycle principles to data science means grounding the work in a real challenge, for example, drug discovery in cancer research (Spangler et al. 2014), with industrial-scale challenges and opportunities to drive the cycle, real use cases to develop and validate solutions, and products to determine value and impact. In the cancer case just cited, innovation occurred, that is, Spangler et al. developed a domain-specific data science method that was subsequently generalized to be more domain independent (Nagarajan et al. 2015), and the mechanisms used to further verify the results are now more widely applied in data science.

3.4 Establishing Causality: A Critical Challenge

Due to the critical problems to which data science is being applied, for example, IBM Watson is in the business of recommending medical treatments, it is critical that accurate likelihoods of outcomes be established. One of the greatest challenges of data science is doing just that—establishing accurate estimates of probabilistic outcomes and error bounds for those results, to which we now turn our attention.

The objective of the Twenty-First Century Virtuous RD&D Cycle is to continuously produce technology and applications that are grounded in reality, namely, that produce products that create value, or even a market of such products that have positive practical, economic, and social impacts. For example, there is a market for data science-based systems that automate aspects of online retailers supply chain, for example, automatically buying hundreds of thousands of products to meet future sales while not overstocking. In 2015, the cost of overstocking was approximately $470 bn and of understocking $630 bn worldwide (Economist 2018d). Normal economics and the marketplace are the mechanisms for demonstrating value and measuring impact. Determining value and impact is far from simple. Most technology such as DBMSs and products such as Microsoft Office have immense value and impact with continuously growing, multi-billion dollar markets. Data science-based products have the potential for great contributions to individuals, organizations, society, and the economy. Like most technology, data science holds equal potential for positive and negative impacts. Disliking a Netflix data-science-driven movie recommendation may waste half an hour of your time. Unfortunately, substantial negative consequences are being discovered in data science applications, such as ethical problems in parole sentencing used extensively in the USA (O'Neil 2016). What might be the impact of data-driven personalized medicine treatment recommendations currently being pursued by governments around the world?

Consider that question given that *Why Most Published Research Findings Are False* (Ioannidis 2005) has been the most referenced paper in medical research since 2005. Data science currently lacks robust methods of determining the likelihood of and error bounds for predicted outcomes, let alone how to move from such correlations to causality. While mathematical and statistical research may be used to address probabilistic causality and error bounds, consider the research required to address ethical and societal issues such a sentencing.

The scientific principles that underlie most research also underlie data science. Empirical studies report causal results while data science cannot. Data science can accelerate the discovery of correlations (Brodie 2019a). A significant challenge is to assign likelihoods and error bounds to these correlations. While the current mechanisms of the Twenty-First Century Virtuous RD&D Cycle to measure value and impact of products worked well for simple technology products, they may not work as well for technology that is increasingly applied to every human endeavor, thus directly influencing our lives. This is a significant issue for the development and operation of data science in many domains. This is yet another class of issues that illustrate the immaturity of data science and the need for multi-disciplinary

collaboration. The complex issue of causal reasoning in data science is addressed in greater detail in the companion chapter (Brodie 2019a).

4 Applying Twenty-First Century Virtuous RD&D Cycles to Data Science

A primary benefit of the Twenty-First Century Virtuous RD&D Cycle is to connect research, engineering, and products in a research-development-delivery cycle with the objective of being virtuous through a continuous flow of innovative, good ideas and challenging problems. The cycle has many applications. It is used extensively in computer science research in academia and industry, in startups that are building our digital world, and increasingly in medicine and science. It has been and is being used to transform education. I propose that it be used to guide and develop data science research, practice, and education.

4.1 A Data Science RD&D Cycle Example

In the mid-2000s, legions of software startups applied the Twenty-First Century Virtuous RD&D Cycle to customer facing applications. As an example, Stonebraker applied the RD&D cycle to Goby—an application that searches the web for leisure activities to provide users, for example, tourists, with a list of distinct local, leisure activities. The "good idea" was to find all activities on the web that might be of interest to tourists. The PIA problem is that there are thousands of leisure activities with many listings that are highly redundant (i.e., replicas), dirty, often inaccurate and contradictory, and in heterogeneous formats. As is typically the case in data science analyses, more than 80% of the resources were required to discover, deduplicate, and prepare the data, leaving less than 20% for analysis, in this case determining relevant activities. This real, industrial-scale use case led to research, Morpheus (Dohzen et al. 2006), that developed machine-driven, user-guided solutions to discover, clean, curate, deduplicate, integrate (a better term is unify), and present data from potentially hundreds of thousands of data sources. The "good idea" led to a PIA[6] problem that resulted in a prototype that led to a product with a commercial market that demonstrated its value and impact. Meanwhile, unanticipated challenges cycled back to Goby for product improvements and enhancements while more fundamental, research challenges went back to Morpheus. The good

[6]Good ideas hopefully arise in answer to a PIA challenge. In this example, the good idea, finding events on the web, led to a PIA problem that was resolved with the now conventional machine driven (ML) and human guided method. The trick is a combination of good ideas and PIA challenges, leading to valuable results.

idea—find events on the web—was generalized from events to the data discovery and preparation of any type of information leading to further innovation that led to a new research project—Data Tamer—that in turn led to a new product—Tamr.com—and a burgeoning market in data discovery and preparation for data science (Forrester 2017; Gartner G00315888 2017). Tamr and similar products are part of the budding infrastructures for data science, called data science platforms (Gartner G00301536 2017; Gartner G00326671 2017; Gartner G00335261 2018).

The Twenty-First Century Virtuous RD&D Cycle is being used to design, develop, and deliver data science tools and platforms. Data discovery and preparation, and data science platforms are concrete examples of this cycle in practice. Over 30 data preparation tools and 60 data science platforms are emerging (Gartner G00301536 2017; Gartner G00326671 2017; Gartner G00326456 2018; Gartner G00326555 2018). This cycle is virtuous as long as there are continuous innovation and broad benefits. Currently, aspects of most human endeavors are being automated by means of digital tools developed to study, manage, and automate those endeavors. Data preparation tools are being developed by being applied to an increasing number of new domains, each presenting new challenges. The continuous flow of practical applications, use cases, PIA problems and other challenges contribute to the cycle being virtuous. The cycle becomes virtuous when all participants benefit. Data science tools and platforms are beginning to flip the ratio of the data-preparation to analysis resources from 80:20 to 20:80, so that data scientists can devote the majority of their time to analysis and not to plumbing. Data science practiced in a virtuous cycle is applied science at its best—producing broad value and contributing to accelerating data science practice and the development of data science *per se*.

4.2 Developing Data Science in Practice and as a Discipline

Data science is an emerging phenomenon worldwide that will take a decade to mature as a robust discipline (Brodie 2015, 2019a). Its growth and diversity can be seen in the number (over 150) and nature of DSRIs, most of which were established after 2015. The emerging state of data science can be seen in the fact that each DSRI provides different answers to key data science questions that all DSRIs should answer (Brodie 2019a): *What is data science*? *What is the practice of data science*? *What is world class data science research*?

The Twenty-First Century Virtuous RD&D Cycle can guide the development and practice of data science. First, the domain is just emerging characterized by a constant flow of new ideas entering the cycle. Data science is being attempted in every human endeavor for which there is adequate data (Brodie 2019a). Second, due to its immaturity (Brodie 2015) data science must be grounded in reality, for example, real data in real use cases at the appropriate scale. The cycle can be used to guide the development and work of individual data scientists and, at a greater scale, of DSRIs. Major features of the cycle are already present in most DSRIs, specifically research-industry collaboration in their research and education. Most

have industry partners and collaborations for education, RD&D, for case studies, and for technology transfer. In most cases, significant funding has come from industry partners. The charter of the Center of Excellence at Goergen Institute for Data Science[7] includes collaborating with industry "to apply data science methods and tools to solve some of the world's greatest challenges in sectors including: Medicine and Health, Imaging and Optics, Energy and the Environment, Food and Agriculture, Defense and National Security, and Economics and Finance." The mission statement of the recently launched Harvard Data Science Initiative[8] states "Applications are by no means limited to academia. Data scientists are currently key contributors in seemingly every enterprise. They grow our economy, make our cities smarter, improve healthcare, and promote civic engagement. All these activities—and more—are catalyzed by the partnership between new methodologies in research and the expertise and vision to develop real-world applications."

Applying the Twenty-First Century Virtuous RD&D Cycle to DSRIs must recognize three factors that distinguish data science from conventional academic research that often lacks research-industry engagement. First, while core or theoretical research is equally important in both cases, DSRI resources must be allocated to applied research, technology transfer, and supporting research-industry collaboration[9]. Unlike a computer science research institute and in support of this objective, a DSRI might have a *Chief Scientific Officer* to establish DSRI-wide data science objectives, such as contributing more than the sum of its parts, and coordinating research across the many organizational units into the components of data science, for example, principles, models, and analytical methods; pipelines and infrastructure; and a data science method, to support data science in all domains. Second, special skills, often not present in research staff, are required for research-industry engagement, the research-development-delivery cycle, and technology transfer. For example, emerging data science platforms are increasingly important for developing and conducting data science. A data science platform includes workflow engines, extensive libraries of models and analytical methods, platforms for data curation and management, large-scale computation, and visualization; that is, a technology infrastructure to support end-to-end data science workflows or pipelines. Hence, research into the development of data science platforms should be a DSRI research objective. Again, unlike a computer science research institute, a DSRI might also establish a *Chief Technology Officer* responsible for those functions including the development and maintenance of a shared data science technology infrastructure.

The third distinguishing factor is the relative immaturity of data science versus most academic disciplines; excitement and hype cloud the real state of data science. A common claim is that data science is successful, ready for technology transfer and application in most human endeavors. While there are successful data science

[7] http://www.sas.rochester.edu/dsc/

[8] http://datascience.harvard.edu/

[9] In its emerging state, data science lacks a scientific or theoretical base. Establishing data science as a science should be a fundamental objective of data science researchers and DSRIs (Brodie 2019a).

technologies and domain-specific results, in general this impression, often espoused by vendors and enthusiasts,[10] is false. While there are major successes and expert data scientists, data science is an immature, emerging domain that will take a decade to mature (Brodie 2015, 2019a). Analysts report that most early (2010–2012) data science projects in US enterprises failed (Forrester 2015a, b; Demirkan and Dal 2014; Veeramachaneni 2016; Ramanathan 2016). In late 2016, Gartner reported that while most (73%) enterprises declare data science as a core objective, only 15% have deployed Big Data projects in their organization (Gartner G00310700 2016) with well-known failures (Lohr and Singer 2016). This reflects confusion concerning data science and that technology analysts are not reliable judges of scientific progress.

Slow progress makes perfect sense as data science is far more complex than vendors and enthusiasts report. For example, data science platforms provide libraries of sophisticated algorithms [visualization (Matplotlib, Matlab, Mathematica); data manipulation, aggregation, and visualization (Pandas); linear algebra, optimization, integration, and statistics (SciPy); image processing and machine learning (SciKit-Learn); Deep Learning (Keras, TensorFlow, Theano); Natural Language Processing (NLTK)] that business users have significant difficulty fitting to business problems (Forrester 2015b). There is a significant learning curve—few people understand deep learning, let alone statistics at scale—and substantial differences with conventional data analytics. *What do you mean these aren't just spreadsheets*?

Over the next decade, research will establish data science principles, methods, practices, and infrastructure and will address these key questions. This research should be grounded in practical problems, opportunities, and use cases. DSRIs should use the Twenty-First Century Virtuous RD&D Cycle to direct and conduct research, practice, education, and technology transfer. Initially, they might use the R&D cycle to explore good ideas. Research-industry collaborations should be used to identify and evaluate novel data science ideas. When collaborations can identify plausible use cases or PIA problems, the research-development-delivery cycle should be used. That is, to identify research domains and directions, DSRIs should identify industrial partners with whom to collaborate to establish virtuous cycles that equally benefit researchers and industry partners. As with applied university research funding, a significant portion of data science research funding should come from industry to increase industry-research engagement and quickly identify valuable research with impact potential.

[10]Michael Dell, Dell CEO, predicted at the 2015 Dublin Web Summit that big data analytics is the next trillion-dollar market. IDC predicts 23.1% compound annual growth rate, reaching $48.6 billion in 2019. Forrester Research declared that "all companies are in the data business now." Gartner predicts "More than 40% of data science tasks will be automated by 2020" (Gartner G00316349 2016).

4.3 Developing Data Science Education

Data science is one of the fastest growing subjects in education due to the demand for data scientists. Data science courses, programs, degrees, and certificates are offered by most universities and professional training institutes and are part of the mission of most DSRIs. Given the decade to maturity of data science, how should data science education programs be developed?

Just as the Twenty-First Century Virtuous RD&D Cycle is used to transform the research, development, delivery, and use of computer systems and applications, it can also be used to transform education. The intention of the recently launched *Twenty-First Century Applied PhD Program in Computer Science*[11] at Texas State University is for PhD level research ideas, innovations, and challenges to be developed in prototype solutions and refined and tested in industrial-scale problems of industrial partners. The cycle is to be driven by industrial partners that investigate or face challenges collaboratively with the university. PhD candidates work equally in research and in industry to identify and research challenges and opportunities that are grounded in real industrial contexts and to develop prototype solutions that are refined using industrial use cases. This educational cycle requires technology transfer from research to advanced prototypes to industry with opportunities and problems transferring, in the opposite direction, from practice to advanced development and to research. It becomes virtuous with a constant stream of "good ideas"—challenges and opportunities—and of PhD candidates in one direction, and industry PIA problems, challenges, and opportunities in the other. The primary benefits of this program are that research, teaching, and products are grounded in reality.

These ideas are not new. The Fachhochschule system (universities of applied sciences) applied virtuous cycle principles in Germany since the late 1960s, and in Austria and Switzerland since the 1990s as a graduate extension of the vocational training and apprenticeship (Berufslehre und Ausbildung) programs that have roots in mentorships and apprenticeships from the middle ages.

While the quality and intent of the European and US educational systems are the same, the systems differ. Academic universities focus on theory and applied universities focus on the application of science and engineering. Fachhochschulen usually do not grant PhDs. In addition, research in applied universities is funded differently from research in academic universities. Usually, over 80% of applied research funding comes from third parties to ensure research-industry engagement[12] and as a test of the PIA principle. Unsuccessful research is quickly identified and terminated. Dedicated government agencies provide partial funding and promote innovation and technology transfer through collaboration between industry and the applied universities. Enrollments in Fachhochschulen are soaring, indicating the demand for education grounded in reality—closely mirroring successful startup behavior. Due to

[11] https://cs.txstate.edu/academics/phd/

[12] A similar principle applied by the funding agency in Sect. 5.1 story was initially considered a death knell by the DSRI and by me. It took a year for me to see the value.

the significance of, demand for, and perceived value of data science, education programs should be revisited considering adding more applied aspects to conventional research and education for data science. A good example of this vision is the *Twenty-First Century Applied PhD Program in Data Science* at Texas State University, based on a collaborative research-industry-development-delivery model.

5 Lessons Learned

5.1 Data Science and DSRI Stages of Development

In 2013, I was invited to join the Scientific Advisory Committee (SAC) of Ireland's Insight Center for Data Analytics, at the time one of the first and largest DSRIs, composed of four partner institutes. Since then I have actively participated on the SAC as well as on Insight's Governance Committee. Over the following years, I observed the development of Insight as a DSRI as well as the establishment of over 150 DSRIs at major institutions worldwide. Insight's development as a DSRI was not without challenges. In 2017, Science Foundation Ireland (SFI) reviewed Insight for a potential 5-year funding renewal. Insight needed to tell SFI what data science was, what world class data science research was, and to measure its progress accordingly. This led me to the observation, stated to the review board, that Insight's development as a DSRI reflected the development of data science as a discipline. The most thoughtful contributors to data science fully understood that while the potential benefits for Ireland and the world were enormous, data science as a discipline was in its infancy and faced considerable scientific and organizational developmental challenges. Further, that Insight in operating for 5 years and in aspiring to world class data science contributions as a world class DSRI had faced and overcome significant challenges that I had witnessed first-hand at Insight and indirectly in eight other DSRIs.

Over 5 years, Insight had gone through the four stages of development that younger DSRIs are just encountering. Insight is currently at stage five—a critical stage. Successful progress through the stages revolved around three fundamental issues:

- Just as the science and the scientific method are far more than experiments in a single domain, so too is data science more than data science activities in a single domain.
- Changing centuries of research behavior to enable collaboration across disciplines in data science pipelines, as well as across academic and organizational boundaries.
- Producing, for Ireland and for data science, more than the sum of the parts, that is, the results of individual member institutes.

The five stages are simple.

1. ***Act of creation***: An organizational decision was made to form a DSRI from independent, one might say competing, institutes with a new focus, the emerging discipline of data science. The institutes—researchers and administrators alike—in a behavioral and legal tradition of individual progress and reward were not happy campers. Awkwardness arose.
2. ***Initial participation***: Participants continued business as usual, but expressed a willingness to participate and cooperate followed by little actual collaboration and some ingrained competitiveness. The DSRI administration soldiered on toward understanding the bigger picture that had not been defined by anyone—funders, researchers, or advisors.
3. ***Data science objectives understood—conceptually***: After a few years of successful execution of individual research efforts and attempts to understand data science, modest progress was made, especially once it was clear that funding would depend on the DSRI being more than the sum of the parts and would be measured on world-class data science, interpreted then as contributing to data science, *per se*. But what is that exactly?
4. ***Data science objectives understood—emotionally***: Goals provide focus. Five years of funding of the now seven institutes depended on the DSRI being "more than the sum of the parts". This was not an abstract concept but required providing benefits such as accelerating discovery in specific parts of the Irish economy, educating data scientists, and economic growth in Ireland, involving not just researchers but major industrial partners. Individual researchers rose to the challenge to propose a collaborative DSRI. By the time of the review, they had become a band of data science brothers and sisters, together with industrial partners.
5. ***Stand and deliver***: While the DSRI will continue to produce specific data science results that are world class in specific domains, for example, physiology, it is defining and planning contributions to data science, including data science principles, models, methods, and infrastructure (Brodie 2019a).

Many DSRIs around the world have been created, like Insight, by a higher-level organization, typically a university, to coordinate the myriad data science activities in that organization. The critical factor missing in many DSRIs, at least as viewed through their websites, is an imperative to understand and contribute to data science *per se*, to contribute more than the sum of the contributions of the partner organizations. SFI's funding of Insight depends on contributing to data science *per se*, worded as "contributing more than the sum of the parts." This imperative is not present in many DSRIs.

5.2 Myths of Applying Data Science in Business

As often happens with new technology trends, their significance, impact, value, and adoption are exaggerated by the analysts and promoters as well as by optimists and

the doomsayers. Technology analysts see their roles as reporting on new technology trends, for example, Gartner's Hype Cycles. If a technology trend is seen as significant, investment analysts join the prediction party. Technology and investment analysts are frequently wrong as they are now with data science. Many technology trends reported by Gartner die before reaching adoption, for example, the 1980s service-oriented architectures. Some trends that are predicted as dying become widely adopted, for example, the .com boom was reported as a failure, largely due to the .com stock market bubble, but the technology has been adopted globally and has led to transforming many industries. Data science is one of the most visible technology trends of the twenty-first century with data scientists called "the sexiest job of the twenty-first century" (Davenport and Patil 2012) and "engineers of the future" (van der Aalst 2014). To illustrate the extent to which data science is blown out of proportion to reality, let us consider several data science myths. A reasonable person might ask, given the scale, scope, and nature of the change of data science as a new discovery paradigm, how could anyone predict with any accuracy how valuable it will be and how it will be adopted, especially when few people, including some "experts", currently understand it (that, by the way, was myth #1).

Everyone Is Successfully Applying Data Science As reported above most (80%) early (2010–2012) data science projects in most US enterprises failed. By early 2017, while 73% of enterprises declare data science as a core objective, only 15% have deployed it. In reality, AI/data science is a hot area, with considerable, perceived benefit. Hence many companies are exploring it. However, such projects are not easy and require ramping up of rare skills, methods, and technologies. It is also difficult to know when and how to apply the technology and to appropriately interpret the results. Hence, most initial projects are highly unlikely to succeed but are critical to gain the expertise. Applying AI/data science in business will have major successes (10%) and moderate successes (40%) (Gartner G00310700 2016). Most companies are and should explore AI/data science but be prepared for a significant learning curve. Not pursuing AI/data science will likely be an advantage to your competitors.

Reality: organizations perceiving advantages should explore data science and prepare for a learning curve.

Data Science Applications Are Massive While scale is a definitive characteristic of Big Data and data science, successful applications can be small and inexpensive. The pothole example (Brodie 2019a) was a very successful launch of now flourishing startups in the emerging domain of autonomous vehicles. It was based on building and placing small, inexpensive (~$100) motion detectors in seven taxis. It started with the question shared by many businesses, *What is this data science stuff*? It was a pure exploration of data science and not to find a solution to a PIA problem. As data science matures, we see that the critical characteristics of a data analysis are determined by the domain and the analytical methods applied. Volume is one characteristic that must meet statistical requirements but even GB or TB may be adequate and can be handled readily by laptops.

Reality: data science can be applied on modest data sets to solve interesting, small problems.

Data Science Is Expensive Famous, successful data analytics (Higgs Boson, Baylor-Watson cancer study, LIGO, Google, Amazon, Facebook) often require budgets at scale (e.g., massive processing centers, 1,00,000 cores, 1000s of analysts); however, data analytics even over relatively large data volumes can be run on desktops using inexpensive or free open source tools and the cloud. Businesses can and should conduct initial explorations like the pothole analyses at negligible cost.

Reality: small players with small budgets can profit from data science.

Data Science Predicts What Will Happen Otto, a German retailer orders 2,00,000 SKUs fully automated. Above we cited predictions of trillions of dollars in related savings worldwide. However, the results of good data analytics are at best probabilistic with error bounds. This is somewhat similar to science (scientific method) but is typically less precise with lower probabilities and greater error bounds due to the inability of applying the controls that are applied in science. Businesses should explore the predictive power of data science but with the full understanding of its probabilistic and error-prone nature. Otto and the supply chain industry constantly monitors and verifies results and adjusts as needed or, like H&M, you might end up with a $4.3 bn overstock (New York Times 2018).

Reality: predictions are probabilistic and come with error bounds.

Data Science Is Running Machine Learning Over Data Machine learning is highly visible in popular press accounts of data science. In reality, one must select from thousands of AI and non-AI methods and algorithms depending on the phenomenon being analyzed and the characteristics of the data. What's more, as reported above, while algorithm selection is critical, 80% of the resources including time for a data analysis is required just to find and prepare the data for analysis.

Reality: as there is no free lunch (Wolpert and Macready 1997), there is no single methodology, algorithm, or tool to master to do successful data science, just as it is in science.

AI/Data Science Is Disrupting and Transforming Conventional Industries and Our Lives This widely reported myth (Marr 2017; Chipman 2016) makes eye catching press but is false. There is ample evidence that AI/data science is being applied in every human endeavor for which adequate data is available such as reported throughout this book. The list of impacted industries is long: mechanical engineering and production of industrial goods (shop floor planning, robotics, predictive maintenance); medicine (personalized health); commerce/trade (e-commerce, online business, recommenders); hospitality (demand planning and marketing via analytics, pricing based on customer analytics); transportation (ride-sharing/hailing); automotive (self-driving cars, e-mobility); services (new business models based on data); and many more. In reality, five industries have been massively disrupted by digital innovation—music, video-rental, publishing (books, newspapers), taxicabs, and retailing (predominantly clothing). They are in the process of being transformed, for example, the Spotify business model is an example

of transformation in music; Uber's is in taxicabs, but the process takes years or decades. However, the vast majority of industries are currently unaffected. If an industry is being transformed, it is reflected in the stock market, for example, a price-earnings ratio of less than 12 is generally forecast imminent collapse. According to that rule of thumb, Ford and GM's price-earnings ratio of 7 suggest disruption and transformation if not collapse possibly due to electric vehicles (EVs) such as Tesla and ride-sharing/hailing. There are no such indications for the other "conventional industries" (Economist 2017).

Reality: almost all conventional industries are impacted, but only few are disrupted.

It's All About AI Current popular and even scientific press suggests that AI is one of the hottest and potentially most significant technologies of the twenty-first century. AI is sometimes referred to as an object as in "an AI is used to … ". Without doubt AI and specifically machine learning (ML) and deep learning (DL) have been applied to a wide range of problems with significant success and impact as described above. It is very probable that ML will be applied much more extensively with even greater success and impact. However, like most "hot" technical trends, the press characterization is a wild exaggeration—a myth. First of all, AI is a very broad field of research and technology that pursues all forms of intelligence exhibited by a machine (Russell and Norvig 2010). ML is one of perhaps 1000 AI technologies. Second, until we understand ML, its application will be limited. The current, very successful ML technologies arose in the early 2000s from a previously unsuccessful technology, neural networks. Amazingly ML, augmented by massive data sets and high-performance computing, has been applied to images, sentences, and data to appear to identify entities that are meaningful to humans, for example, pizzas, cats, and trains on tracks, and cluster those meaningful entities based on similarities meaningful to humans. We have no idea why there is a correlation between the results of an ML analysis and meaning understood by humans. Considerable research is being invested in understanding such reasoning, but it is far from mature. As a result, the use of ML in the European Community is restricted by the GDPR law. Finally, the successful application of ML is proportional to the data to which it is applied, typically ML works most effectively on massive data sets. Massive data analysis requires high performance computing, one of the critical components that moved neural networks from failure to success. Hence, most naïve misuses of the term AI should be replaced with the specific AI technology, for example, ML, plus data plus high-performance computing. This sounds remarkably like data science.

Reality: AI is a key component, among many others, that is necessary to conduct data science; AI does not perform miracles; in many cases "AI" should be replaced by the technologies used to support analytical workflows.

6 Potential Impacts of Data Science

The development of data science involves not just the science, technology, and applications, it also involves the opportunities, challenges, and risks posed by the applications of data science. Hence, I now briefly review some potential benefits and threats of applying data science, many of which have been reported in the popular press. However, popular press descriptions of hot technical topics and their impacts are usually to be taken with a grain of salt, especially concerning AI and data science that are not well understood by some experts.

In the early 2010s, Big Data was the hot technology topic. In 2018, AI and its power was the hot topic, not unreasonably as Sundar Pichai, Google CEO, said that AI will have a "more profound" impact than electricity or fire (Economist 2018a). Consider it a matter of terminology. Big Data on its own is of no value. Similarly, without data AI is useless. The hot technical topics that surface in the media are equally attributable to AI, massive data, and powerful computation. In what follows, as above, I refer to applications of that combination as AI/data science. Yet even those three terms are not adequate to achieve the hot results since data science depends also on domain knowledge and more, but this will suffice for the following discussion.

According to Jeff Dean, director of Google's AI research unit, Google Brain, more than 10 m organizations "have a problem that would be amenable to a machine-learning solution. They have the data but don't have the experts on staff" (Economist 2018b). That is, the potential impacts of AI/data science will have broad applicability.

As with a new, powerful technology, society, for example, legislation, is seldom able to keep up with its impacts. In the case of AI/data science, let's consider its impacts on our daily lives, both the benefits and the threats, of a multi-billion-dollar industry that currently has almost no regulations to restrain its application.

6.1 Benefits

Google's youthful founders, Sergey Brin and Larry page, defined its original vision "to provide access to the world's information in one click" and mission statement "to organize the world's information and make it universally accessible and useful" with the famous motto "Don't be evil". Indeed, they *almost* succeeded beyond their wildest dreams. The entire world benefits from instant access to much of the world's information. We went from this utopian view of the potential of AI/data science to one in which Google, in 2015, dropped its famous motto from its code of conduct. I address some shortcomings, such as use and protection of personal information, in the next section.

It is infeasible to list here the many benefits of AI/data science, so let's consider two examples, medicine and autonomous vehicles. A data-science-based medical

analysis can compare a patient's mammogram with 1 m similar mammograms in seconds to find potential causes and treatments that were most effective for the conditions present in the subject mammogram. Similar analyses and achievements are being made with our genetic code to identify the onset of a disease and effective treatment plans based on millions of similar cases, something no human doctor could possibly do on their own.

Autonomous vehicles depend on AI/data science. It is commonly projected that autonomous vehicles will radically reduce the 1 m annual traffic deaths per year worldwide, pollution and traffic congestion while shortening travel times, freeing us up for a better quality of life. The impacts could be far greater than those of the automobile. But how will autonomous vehicles change the world? One factor to consider is that currently the average car sits parked 95% of the time. What might be the impacts of autonomous vehicles on real estate, roads, automobile manufacturing, and employment?

Most benefits of technology harbor unanticipated threats—autonomous vehicle results can be applied in many domains, for example, autonomous weapons are used by 80 countries including the USA that has over 10,000[13]. Let us consider a few threats posed by AI/data science.

6.2 Threats

On May 6, 2010, the US stock market crashed. In the 2010 Flash Crash, over a trillion dollars in value was lost and the indexes (Dow Jones Industrial average, S&P 500, Nasdaq Composite) collapsed (Dow Jones down ~1000 points, 9% in value). Within 36 min the indexes and value largely but not entirely rebounded. This was due in part to algorithmic trading that operates 60% of trading in US exchanges, and in part to the actions of Navinder Singh Sarao, a trader who the US Department of Justice convicted for fraud and market manipulation.[14] Algorithmic trading is a data science-based method of making trades based on complex economic and market analysis based on potentially all trades ever transacted. This was not a threat. It was a reality and a harbinger of similar threats.

How might AI/data science threaten employment? Consider the potential impact of autonomous vehicles on America's 4 m professional drivers (as of 2016, US Bureau of Labor Statistics). Robots will impact a vastly larger number of jobs. McKinsey Global Institute estimated that by 2030 up to 375 m people could have their jobs "automated away" (Economist 2018c). These examples are the tip of the AI/data science unemployment iceberg. The Economist (2018f) and the

[13]Do you trust this computer? http://doyoutrustthiscomputer.org

[14]https://www.justice.gov/opa/pr/futures-trader-pleads-guilty-illegally-manipulating-futures-market-connection-2010-flash

Organisation for Economic Co-operation and Development (OECD) (Nedelkoska and Quintini 2018), estimate that over 50% of all jobs are vulnerable to automation.

An insidious threat is bias in decision making. Our lives are increasingly determined by algorithms. Increasing machine learning and other sophisticated algorithms are used to make decisions in our lives, in our companies, in our careers, in our education, and in our economy. These algorithms are developed with models that represent the significant features of the problem being addressed. No one but the developers see the code, fewer people actually understand the code. So, what is in the code? Are race, sex, or a history of past behavior significant and *acceptable* features in parole sentencing? Are these algorithms biased against certain types of individuals? ProPublica proved that parole sentencing is indeed biased against blacks (Angwin et al. 2016). The 12 vendors of the systems that the US government uses for sentencing refuse to release their code for inspection. Ironically, ProPublica proved their case using data science. They collected and analyzed sentencing data to prove with high confidence that the systems were inherently biased. This has led to the algorithmic accountability movement in the legal community.

In many countries, tech companies, for example, Apple, Alphabet (Google parent), Microsoft, Amazon, Facebook, Alibaba, and Tencent, know more about us and can predict our behavior better than we can. In some countries, the government takes this role (e.g., China's social credit system). Over the past decade, there has been increasing concern for personal information. Legislation to govern the use and privacy of personal information [General Data Protection Regulation[15] (GDPR)] was enacted in Europe only in May 25, 2018. US congressional hearings only began in early 2018 prompted by the alleged illegal acquisition of 87 m Facebook profiles by Cambridge Analytica (CA), described below.

The power and growth of the seven companies mentioned above, the largest companies in the world by market capitalization, is directly attributable to AI/data science. Their average age is less than 10 years in contrast to average age of 141 years of the legacy companies that they are supplanting from the top 10 largest companies. These tech leaders vastly outspend the largest legacy companies in research and development, for example, Apple's 2017 $22.6 bn R&D investment was twice that of non-tech Johnson & Johnson, established in 1886. It is frequently argued (Lee 2017; Economist 2018e) that the power of AI/data science is such that the country that dominates the field will wield disproportionate economic and ultimately political power worldwide, that is, will monopolize not just AI/data science but areas of the economy for which it is a critical success factor. Currently, China and the USA are the leaders by far. However, the playing field is beginning to favor China. The power and development of AI solutions is heavily dependent on vast amounts of data. Increasing restrictions on data, such as privacy legislation mentioned above, will significantly inhibit US AI companies while there is little or no such limitations by the Chinese government that itself collects massive data on its citizens.

[15] https://en.wikipedia.org/wiki/General_Data_Protection_Regulation

It may seem dramatic, but data science has allegedly been used to threaten democracy (Piatetsky 2016). Alexander Nix, the now-suspended CEO of now-insolvent CA, claimed to have applied a data science-based methodology, psychometrics, to influence political opinion. Nix reported that it was used to influence the outcomes of political elections around the world including the 2016 British EU referendum, aka Brexit referendum, in favor of leaving, and the 2016 US election in favor of Donald Trump. Psychometrics is based on a physiological profiling model from Cambridge and Stanford Universities. For the US election, CA illegally and legally acquired up to 5000 data points each of 230 m Americans to develop a detailed profile of every American voter. Profiles were used to send "persuasion" messages (e.g., on gun rights) targeted to and nuanced for the weaknesses and preferences of individual voters. CA activities were first reported in 2015 and resurfaced in January 2017 when Trump took office. It wasn't until April 2018 that CA's actions in the 2016 US election were considered for prosecution. Notwithstanding CA's illegal actions and potentially violating American democratic principles, CA's data-science method appears to have been very effective and broadly applicable, for example, being applied in targeted, 1-on-1 marketing. Such methods are allegedly being used by governments, for example, in the Chinese social credit system and in Russian interference with the 2016 US election. This genie is out of the bottle.

6.3 More Profound Questions

A more profound question is: Will these advanced technologies enhance or replace man? In *Homo Deus* (Harari 2016), the author Yuval Noah Harari hypothesizes that the human race augmented by advanced technologies, specifically AI/data science, will transform homo sapiens into a new species. Just as homo sapiens surpassed and replaced Neanderthals, so will humans augmented with machines surpass homo sapiens without automation. Could you compete or survive without automation? This is well beyond considering the impacts of data science. Or is it? In 2018 there were multiple attacks on the very foundations of democracy (see above). At the TED 2018 conference, Jaron Lanier, virtual reality creator, suggested that, using data, social networks had become behavior modification networks. Harari speculated that just as corporations use data now, so too could dictatorships use data to control populations.

Technological progress is never solely positive, for example, automation that eliminates waste due to optimized supply chains. Progress is relative to our expectations, for example, computers will eliminate most human drivers thereby reducing road accidents by 95%. In this case, the cost of saving lives is a loss of jobs. The greatest impacts of technology are seldom foreseen, for example, the redistribution

of populations from cities to suburbs due to the mobility offered by automobiles. What might be the impact of a machine beating humans playing Jeopardy?

The Future of Life Institute[16] was established "To catalyze and support research and initiatives for safeguarding life and developing optimistic visions of the future, including positive ways for humanity to steer its own course considering new technologies and challenges." Its motto is: "Technology is giving life the potential to flourish like never before . . . or to self-destruct. Let's make a difference."

7 Conclusions

Data science is potentially one of the most significant new disciplines of the twenty-first century, yet it is just emerging, poses substantial challenges, and will take a decade to mature. The potential benefits and risks warrant developing data science as a discipline and as a method for accelerated discovery in any domain for which adequate data is available. That development should be grounded in reality following the proverb: *Necessity is the mother of invention*. This chapter proposes a long-standing, proven development model.

Innovation in computing technology has flourished through three successive versions of the virtuous cycle. The twentieth century virtuous cycle was hardware innovation and software innovation in a cycle. The twentieth century Virtuous R&D Cycle was research innovation and engineering innovation in a cycle. The emerging Twenty-First Century Virtuous RD&D Cycle is research innovation, engineering innovation, and product innovation in a cycle. While innovation perpetuates the cycle, it is not the goal. Innovation is constantly and falsely heralded as *the* objective of modern research. Of far greater value are the solutions. Craig Vintner—a leading innovator in genetics—said, "Good ideas are a dime a dozen. What makes the difference is the execution of the idea." The ultimate goal is successful, efficient solutions that fully address PIA problems or major challenges, or that realize significant, beneficial opportunities. Data science does not provide such results. Data science accelerates the discovery of probabilistic results within certain error bounds. It usually does not produce definitive results. Having rapidly reduced a vast search space, to a smaller number of likely results, non-data science methods, typically conventional methods in the domain of interest, are used to produce the definitive results. Once definitive results are achieved, the data science analysis can be converted to a product, for example, a report, inventory replenishment, etc.; however, the results of such a product must be monitored as conditions and data can change constantly. For more on this see Meierhofer et al. (2019).

The principles and objectives of the Twenty-First Century Virtuous RD&D Cycle are being applied in many domains beyond computer science, startups, education, and data science. In medicine it is called translational medicine (STM 2018) in which healthcare innovation and challenges go across the

[16] https://futureoflife.org

benchside/research-bedside-community[17] cycle, delivering medical innovations to patients and communities more rapidly than conventional medical practice and taking experience and issues back for research and refinement. The US National Institutes of Health (NIH) established The National Center for Advancing Translational Sciences in 2012 for this purpose and is increasingly requiring its practice in NIH-funded research programs. In the broader scientific community, such activities are called translational science and translational research, for example (AJTR 2018; Fang and Casadevall 2010). The RD&D cycle is now incorporated in all natural science and engineering research funded in Canada.[18]

Data science researchers and DSRI leaders might consider the Twenty-First Century Virtuous RD&D Cycle to develop and contribute to data science theory, practice, and education.

Acknowledgments Thanks to Dr. Thilo Stadelmann, Zurich University of Applied Sciences, Institute for Applied Information Technology in the Swiss Fachhochschule system, for insights into these ideas; and to Dr. He H. (Anne) Ngu, Texas State University, for insights into applying these principles and pragmatics to the development of Texas State University's Twenty-First Century Applied PhD Program in Computer Science.

References

ACM. (2015). *Michael Stonebraker, 2014 Turing Award Citation*, Association of Computing Machinery, April 2015. http://amturing.acm.org/award_winners/stonebraker_1172121.cfm

AJTR. (2018). *American Journal of Translational Research*, e-Century Publishing Corporation. http://www.ajtr.org

Angwin, J., Larson, J., Mattu, S., Kirchner, L., Machine Bias: There's software used across the country to predict future criminals. And it's biased against blacks, ProPublica, May 23, 2016. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

Braschler, M., Stadelmann, T., & Stockinger, K. (Eds.). (2019). *Applied data science – Lessons learned for the data-driven business*. Berlin: Springer.

Brodie, M. L. (2015). Understanding data science: An emerging discipline for data-intensive discovery. In S. Cutt (Ed.), *Getting data right: Tackling the challenges of big data volume and variety*. Sebastopol, CA: O'Reilly Media.

Brodie, M. L. (2019a). What is data science? In M. Braschler, T. Stadelmann, & K. Stockinger (Eds.), *Applied data science – Lessons learned for the data-driven business*. Berlin: Springer.

Brodie, M. L. (Ed.). (2019b, January). *Making databases work: The pragmatic wisdom of Michael Stonebraker*. ACM Books series (Vol. 22). San Rafael, CA: Morgan & Claypool.

Chipman, I., (2016). How data analytics is going to transform all industries. *Stanford Engineering Magazine*, February 13, 2016.

[17]The US National Institutes of Health support of translational medicine in which the research process includes testing research (benchside) results in practice (bedside) to speed conventional clinical trial methods.

[18]Dr. Mario Pinto, President of the Natural Sciences and Engineering Research Council of Canada, in 2017 announced that a research-development-delivery method was to be used in all NSERC-funded projects.

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM, 13*(6), 377–387.
Davenport, T. H., & Patil, D. J. (2012). Data scientist: The sexiest job of the 21st century. *Harvard Business Review, 90*(10), 70–76.
Demirkan, H. & Dal, B. (2014). The data economy: Why do so many analytics projects fail? *Analytics Magazine*, July/August 2014
Dohzen, T., Pamuk, M., Seong, S. W., Hammer, J., & Stonebraker, M. (2006). Data integration through transform reuse in the Morpheus project (pp. 736–738). *ACM SIGMOD International Conference on Management of Data*, Chicago, IL, June 27–29, 2006.
Economist. (2017). Who's afraid of disruption? The business world is obsessed with digital disruption, but it has had little impact on profits, *The Economist*, September 30, 2017.
Economist. (March 2018a). GrAIt expectations, Special Report AI in Business, The *Economist*, March 31, 2018.
Economist. (March 2018b). External providers: Leave it to the experts, Special report AI in business, The *Economist*, March 31, 2018.
Economist. (March 2018c). The future: Two-faced, Special report AI in business, The *Economist,* March 31, 2018.
Economist. (March 2018d). Supply chains: In algorithms we trust, Special report AI in business, The *Economist,* March 31, 2018.
Economist. (March 2018e). America v China: The battle for digital supremacy: America's technological hegemony is under threat from China, The *Economist,* March 15, 2018.
Economist. (2018f). A study finds nearly half of jobs are vulnerable to automation, *The Economist*, April 24, 2018.
Fang, F. C., & Casadevall, A. (2010). Lost in translation-basic science in the era of translational research. *Infection and Immunity, 78*(2), 563–566.
Forrester. (2015a). *Brief: Why data-driven aspirations fail*. Forrester Research, Inc., October 7, 2015.
Forrester. (2015b). *Predictions 2016: The path from data to action for marketers: How marketers will elevate systems of insight*. Forrester Research, November 9, 2015.
Forrester. (2017). *The Forrester WaveTM: Data preparation tools, Q1 2017*, Forrester, March 13, 2017.
Gartner G00310700. (2016). *Survey analysis: Big data investments begin tapering in 2016*, Gartner, September 19, 2016.
Gartner G00316349. (2016). *Predicts 2017: Analytics strategy and technology, Gartner, report G00316349*, November 30, 2016.
Gartner G00301536. (2017). *2017 Magic quadrant for data science platforms*, 14 February 2017.
Gartner G00315888. (2017) *Market guide for data preparation*, Gartner, 14 December 2017.
Gartner G00326671. (2017). Critical capabilities for data science platforms, Gartner, June 7, 2017.
Gartner G00326456. (2018). *Magic quadrant for data science and machine-learning platforms*, 22 February 2018.
Gartner G00326555. (2018). *Magic quadrant for analytics and business intelligence platforms*, 26 February 2018.
Gartner G00335261. (2018) *Critical capabilities for data science and machine learning platforms*, 4 April 2018.
Harari, Y. N. (2016). *Homo Deus: A brief history of tomorrow*, Random House, 2016.
Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLOS Medicine, 2*(8), e124.
Lee, K-F., The real threat of artificial intelligence. *New York Times*, June 24, 2017.
Lohr, S. & Singer, N. (2016) How data failed us in calling an election. *New York Times*, November 10, 2016.
Marr, B., (2017). *How big data is transforming every business*. In Every Industry, Forbes.com, November 21, 2017.
Meierhofer, J., Stadelmann, T., & Cieliebak, M. (2019). Data products. In M. Braschler, T. Stadelmann, & K. Stockinger (Eds.), *Applied data science – Lessons learned for the data-driven business*. Berlin: Springer.

Nagarajan, M., et al. (2015). Predicting future scientific discoveries based on a networked analysis of the past literature. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '15)* (pp. 2019–2028). New York, NY: ACM.

National Research Council. (2012). *The new global ecosystem in advanced computing: Implications for U.S. competitiveness and national security*. Washington, DC: The National Academies Press.

Naumann, F. (2018). *Genealogy of relational database management systems*. Hasso-Plattner Institüt, Universität, Potsdam. https://hpi.de/naumann/projects/rdbms-genealogy.html

Nedelkoska, L., & Quintini, G. (2018) Automation, skills use and training. *OECD Social, Employment and Migration Working Papers*, No. 202, OECD Publishing, Paris, doi:https://doi.org/10.1787/2e2f4eea-en.

New York Times. (2018). H&M, a Fashion Giant, has a problem: $4.3 Billion in unsold clothes. *New York Times*, March 27, 2018.

O'Neil, C. (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. New York, NY: Crown Publishing Group.

Olson, M. (2019). Stonebraker and open source, to appear in (Brodie 2019b)

Palmer, A. (2019) How to create & run a Stonebraker Startup – The Real Story, to appear in (Brodie 2019b).

Piatetsky, G. (2016). *Trump, failure of prediction, and lessons for data scientists*, KDnuggets, November 2016.

Ramanathan, A. (2016). *The data science delusion*, Medium.com, November 18, 2016.

Russel, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Boston, MA: Pearson Education.

Spangler, S., et al. (2014). Automated hypothesis generation based on mining scientific literature. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)* (pp. 1877–1886). New York, NY: ACM.

STM. (2018). Science Translational Medicine, a journal of the American Association for the Advancement of Science.

Stonebraker, M. (2019a). How to start a company in 5 (not so) easy steps, to appear in (Brodie 2019b).

Stonebraker, M. (2019b). Where do good ideas come from and how to exploit them? to appear in (Brodie 2019b).

Stonebraker, M., & Kemnitz, G. (1991). The postgres next generation database management system. *Communications of the ACM, 34*(10), 78–92.

Stonebraker, M., Wong, E., Kreps, P., & Held, G. (1976). The design and implementation of INGRES. *ACM Transactions on Database Systems, 1*(3), 189–222.

Stonebraker, M., Abadi, D. J., Batkin, A., Chen, X., Cherniack, M., Ferreira, M., et al. (2005). C-store: A column-oriented DBMS. In *Proceedings of the 31st International Conference on Very Large Data Bases*, 2005.

Stonebraker, M., Castro Fernandez, R., Deng, D., & Brodie, M. L. (2016a). Database decay and what to do about it. *Communications of the ACM, 60*(1), 10–11.

Stonebraker, M., Deng, D., & Brodie, M. L. (2016b). Database decay and how to avoid it. In *Proceedings of the IEEE International Conference on Big Data* (pp. 1–10), Washington, DC.

Stonebraker, M., Deng, D., & Brodie, M. L. (2017). Application-database co-evolution: A new design and development paradigm. In *New England Database Day* (pp. 1–3).

van der Aalst, W. M. P. (2014). Data scientist: The engineer of the future. In K. Mertins, F. Bénaben, R. Poler, & J.-P. Bourrières (Eds.) *Presented at the Enterprise Interoperability VI* (pp. 13–26). Cham: Springer International Publishing.

Veeramachaneni, K. (2016). Why you're not getting value from your data science. *Harvard Business Review*, December 7, 2016.

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation, 1*(1), 67–82.

Chapter 10
The Ethics of Big Data Applications in the Consumer Sector

Markus Christen, Helene Blumer, Christian Hauser, and Markus Huppenbauer

Abstract Business applications relying on processing of large amounts of heterogeneous data (Big Data) are considered to be key drivers of innovation in the digital economy. However, these applications also pose ethical issues that may undermine the credibility of data-driven businesses. In our contribution, we discuss ethical problems that are associated with Big Data such as: How are core values like autonomy, privacy, and solidarity affected in a Big Data world? Are some data a public good? Or: Are we obliged to divulge personal data to a certain degree in order to make the society more secure or more efficient? We answer those questions by first outlining the ethical topics that are discussed in the scientific literature and the lay media using a bibliometric approach. Second, referring to the results of expert interviews and workshops with practitioners, we identify core norms and values affected by Big Data applications—autonomy, equality, fairness, freedom, privacy, property-rights, solidarity, and transparency—and outline how they are exemplified in examples of Big Data consumer applications, for example, in terms of informational self-determination, non-discrimination, or free opinion formation. Based on use cases such as personalized advertising, individual pricing, or credit risk management we discuss the process of balancing such values in order to identify legitimate, questionable, and unacceptable Big Data applications from an ethics point of view. We close with recommendations on how practitioners working in applied data science can deal with ethical issues of Big Data.

M. Christen (✉)
Center for Ethics, University of Zurich, Zurich, Switzerland
e-mail: christen@ethik.uzh.ch

H. Blumer · C. Hauser
Department of Entrepreneurial Management, University of Applied Sciences HTW Chur, Chur, Switzerland

M. Huppenbauer
Center for Religion, Economy and Politics, University of Zurich, Zurich, Switzerland

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_10

1 Introduction

Terms like "Big Data," "digitalization," or "Industry 4.0" have become keywords for indicating the radical changes implied by the pervasive use of digital technology. Big Data basically stands for the fact that today we are not only able to create, record, store, and analyze large amounts of heterogeneous data but that data about almost any fact in the world is available for such purposes as a commodity (Mayer-Schönberger and Cukier 2013). Computers, smartphones, and wearables, as well as the emerging "Internet of things" routinely generate digital data about where we are, what we do, and with whom we communicate. Actually, it is an inherent property of digital technology to generate data in order to function properly. For example, a telecommunication provider needs to "know" the geographic location of a smartphone for providing even its most basic functions (communication). The use of this technology generates data on processes that were mostly obscure in the "pre-digital age"—for example, if one compares rummaging in an old-fashioned bookstore with searching for books on Amazon, where each click leaves a digital trace. But digitalization not only makes it easier to create data, it also has become increasingly cheaper and convenient to store and analyze it. Production and consumption processes thus become ascertainable in a way that was almost unthinkable a few decades ago.

Such radical changes spark both hopes and fears. Some believe that Big Data will be the "oil of the twenty-first century,"[1] that is, an enormous resource for innovation, progress, and wealth. Others consider Big Data to be a fundamental threat for freedom and privacy—a demonic instrument of an Orwellian surveillance state (Helbing et al. 2015). Both scenarios are probably overstated, but they point to difficult ethical problems that are associated with Big Data: How are core values like autonomy, privacy, and solidarity affected in a Big Data world? Are some data a public good? Are we obliged to divulge personal data to a certain degree in order to make society more secure or more efficient?

In this chapter, we will discuss these questions from the perspective of Data Science applications in the consumer sector. This concerns, for example, personalized advertising, tailored product offers, or individualized pricing. What are the ethical questions raised by such applications? Which values have to be weighed against each other in such cases? What are realistic chances and risks of Big Data for consumers? The answers to these questions given in this chapter rely on a study executed by the authors for the Swiss Academy of Engineering Sciences (Hauser et al. 2017). In the following Sect. 2, we provide some background information on ethical thinking in the field of Big Data and on methodological aspects. In Sect. 3, we outline the results of a bibliometric study and we describe five use cases of Big Data applications. In Sect. 4, we will evaluate those case studies from an ethical

[1]The notion of "data as the oil of the twenty-first century" first appeared in 2006 and has become a widespread quote for outlining the economic potential of Big Data; see https://www.quora.com/Who-should-get-credit-for-the-quote-data-is-the-new-oil (last accessed August 10, 2016).

perspective. This analysis will result in some "lessons learned" in Sect. 5, that is, in suggestions how practitioners working in applied data science can deal with ethical issues raised by Big Data. We close by conclusions regarding the possible role of the state in empowering companies and customers for dealing with the ethics of Big Data.

2 Background Information

2.1 Big Data Ethics

The ethical debate concerning Big Data is embedded in a broader discourse on the ethics of data protection that has been developed over several decades (Davis and Patterson 2012). As our bibliometric study indicates (Sect. 3.1), terminologies related to privacy and surveillance still dominate the ethics discourse with respect to Big Data. Certainly, depending on the domain of application, other values will be of importance as well. For example, in the insurance sector, we can expect that the value of solidarity will be of particular importance, since some Big Data applications in this industry may involve a significant potential for unjustified discrimination, and this could lead to a destruction of common grounds for society. Another relevant value affected by Big Data is fairness, as the current ecosystem around Big Data may create a new kind of digital divide: The Big Data rich and the Big Data poor—and, for example, large insurance companies may be representatives of former, putting them into a position to have privileged access to knowledge on societal processes (Boyd and Crawford 2012).

The large majority of papers published so far on the ethical debate on Big Data concern either the health sector or research ethics in the context of Big Data (to illustrate: 13 out of the 20 most cited papers in Scopus searched with the keywords "big data" and "ethic*" refer to healthcare issues; search dated 2016). This indicates a need for further developing the ethical discourse with respect to Big Data in other fields—including consumer applications.

Within the ethical discourse, some authors suggest to consider the fundamental societal changes implied by Big Data: A profound effect of the digitalization of information is that the boundaries around which human beings conceptualize and organize their social, institutional, legal, and moral world are compromised or relativized (van den Hoven et al. 2012). The traditional offline distinctions and demarcations of separate social spheres (family, work, politics, education, healthcare, etc.) are threatened by the enhanced reproducibility and transmissibility of digitalized data and the use of multidimensional Big Data analytics. Thus, a first line or research with respect to Big Data ethics concerns contextual integrity of social spheres as proposed by Helen Nissenbaum (2004), whereas spheres are defined through the expectations and behaviors of actors.

In this framework, what is often seen as a violation of privacy is oftentimes more adequately construed as the morally inappropriate transfer of personal data across the

boundaries of what we intuitively think of as separate "spheres of justice" or "spheres of access" (van den Hoven 1999). This complex moral reason also accounts for the moral wrongness of discrimination. Discrimination of a person P in a particular context implies the use of information about P to her disadvantage while the information is morally irrelevant to that context (e.g., using the information about the gender of a person for determining his or her salary). The art of separation of spheres (or contexts of use) and the blocked exchanges would prevent discrimination. It secures that information is used in contexts where it is relevant and morally appropriate (van den Hoven et al. 2012).

We argue that spheres also differ with respect to the emphasis of certain values. For example, equality (the right of different groups of people to receive the same treatment with respect to the same interests irrespective of their social position) plays a particularly important role in the health sphere, fairness (treatment in accordance with accepted rules or procedures of justice) is an overarching value in the business domain and freedom (the power or right to act, speak, or think as one wants) is a guiding value in the political sphere. Related to this, Alan Fiske's "social relational theory" proposes that there are various but universal types of social interactions or relationships, each of them describing qualitatively distinct structures with their own norms and rules of interactions (Fiske and Tetlock 1997). Within each type, people can usually make trade-offs without great difficulty, but between the domains, comparisons and, for example, applying market-price rules are problematic. We propose that a deeper understanding of the ethical issues raised by Big Data requires an analysis of which values are affected by Big Data applications and how the understanding and weight of these values depends on different social spheres or types of social relationships (Lane et al. 2014). Some of these values may have the status of "protected values" (Tanner and Medin 2004) for the involved persons, which further complicates the picture. Previous studies have shown that when people expect protected values to be under threat, they are likely to trigger reactions of outrage and objection to alleged violations (Tetlock et al. 2000).

A second line of research relevant for Big Data ethics concerns the question how ethics can be integrated into the design process of information technology. Creating Big Data applications is an issue of data product design—and making such a process compliant with ethical requirements puts engineers and managers in the focus. A frame of reference is the methodology of value sensitive design (VSD) that has been put forward by Batya Friedman et al. (2006). In her words, employing value-sensitive design means to account "[. . .] for human values in a principled and comprehensive manner throughout the design process." Through a combination of conceptual, empirical, and technical investigations, one investigates how people are affected through the technology to be designed. Case studies demonstrating how this idea can be implemented in practice can be found in Friedman et al. (2006).

Today, a number of researchers have used the methodology of VSD and it also found its way into textbooks used in engineering education (Van de Poel and Royakkers 2011). Through a combination of conceptual, empirical, and technical investigations, VSD investigates how direct and indirect stakeholders are affected through the technology to be designed. VSD means choosing, among available

technological solutions, those meeting normative requirements and desiderata. "Normative requirements" is a broader concept than requirements sanctioned by law. When a software architecture is designed, there is normally more than one way for the software to solve the problems that it is intended to solve; in making specific engineering choices at different levels of software design, developers implicitly or explicitly express their commitment to grounding principle and values, thereby attributing (a different) importance to them. In this approach, it is assumed that maximizing user satisfaction is not the only goal of good software design, because user satisfaction should not be achieved by sacrificing more important normative constraints.

2.2 Methodology of the Study

The study on which this chapter relies was based on a qualitative and quantitative literature analysis, on expert interviews, and on two workshops with practitioners (company representatives as well as data protection officers).

The literature analysis was performed in two scientific databases (Web of Science and Scopus)[2] as well as in the media database Factiva[3]; the timeframe was restricted to 2006–2015. The papers identified in this way served for a differentiation of the various thematic strains discussed in Big Data. Based on these papers, we identified keywords for characterizing Big Data publications that discuss ethical aspects along six categories: privacy, security, surveillance, harm, self-related topics, and ethics in general.[4] We also analyzed the disciplinary diversity of highly cited Big Data papers (those who received at least more than 10 citations until March 2016) by referring to the subject categories of the WoS papers (those categories refer to the discipline(s) to which the journal, in which a paper has been published, is attributed).

[2]Web of Science (WoS): https://apps.webofknowledge.com; Scopus: http://www.scopus.com. The search term was in both databases "big data" (WoS: search under "topics"; Scopus: search in the category "title, abstract, keywords").

[3]This database is hosted by Bloomberg and includes contributions from the most important international print media (such as New York Times, etc.), and contributions from a multitude of information sources mostly from the business domain; see: https://global.factiva.com. The search term was "big data" as well.

[4]Each ethics category was characterized by a set of 2–5 keywords as follows; the specificity of each keyword was checked individually: Privacy (privacy OR anonym*), security (security OR protection), surveillance (surveillance OR profiling), harm (discrimination OR harm), self-related (identity OR reputation OR ownership) ethics in general (ethic* OR moral OR fairness OR justice OR autonomy). For the quantitative analysis, these keyword sets were combined with "big data" using the Boolean operator AND. Those keywords had to be present either in the title, the abstract, or the keywords of the scientific papers. Those categories do not match the eight values identified further below in the paper, because some of them such as contextual integrity are hard to quantify using a bibliometric approach.

In the two workshops, 22 experts from Swiss institutions and companies were present in total. In the workshops, current and likely future Big Data applications were identified and discussed with respect to the associated ethical questions. The experts emerged from the following industries: banking, consultancy, insurances, marketing, retail business, soft- and hardware producers, telecommunication, and transport. In addition, cantonal (state-level) and federal data protection officers and scientists active in the field complemented the workshop participants. The experts identified five paradigmatic use cases that are discussed in Sect. 3:

- Prevent debt loss
- Improve risk management
- Tailor offer conditions
- Increase the efficiency of advertising
- Create business innovations

They also pointed to eight groups of ethical values that are affected by Big Data applications:

- Privacy protection
- Equality and non-discrimination
- Informational self-determination
- Controlling the own identity
- Transparency
- Solidarity
- Contextual integrity
- Property and copyright

Those eight value groups will serve as a framework for the ethical assessment in Sect. 4.

3 Big Data in the Scientific Literature and in Business

3.1 Bibliometric Study

The bibliometric analysis serves to provide a first overview on the topic of Big Data by showing the frequency of published papers that contain the keyword “big data”, the relative weight of the six ethics categories used in the bibliometric study, and the disciplinary spectrum of highly cited papers. The original study has been performed in March 2016 but has been updated in February 2018, leading to some changes as discussed in the text. Figure 10.1 shows the frequency of Big Data articles both in the scientific literature as well as in the media, that is, the number of papers published in a single year compared to all papers found in the reference timeframe. It is striking that almost no paper has been published before 2011—the first paper that uses the term “big data” in the current understanding was published in 1998. Since 2011, an enormous growth in the literature can be observed, whereas the growth rate was

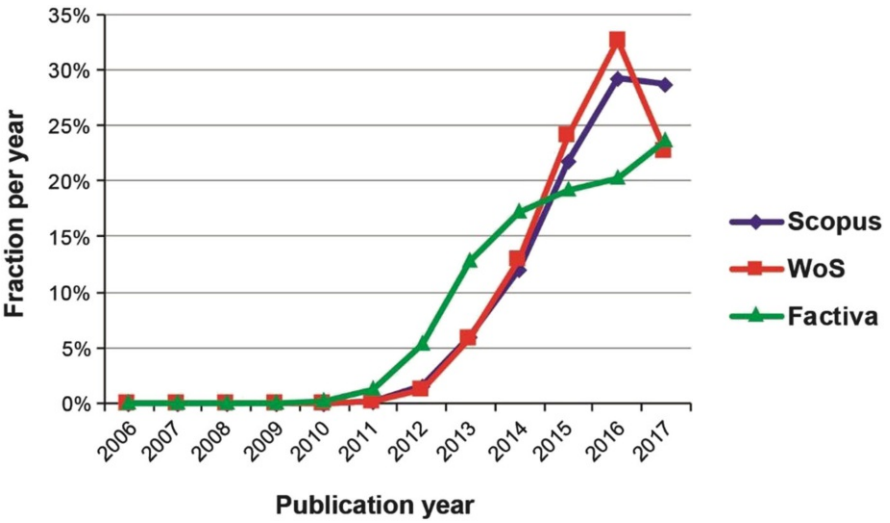

Fig. 10.1 Per-year-fraction of Big Data papers published between 2006 and 2017 in each database; the data of 2017 are incomplete, that is, the diminishment in this year is likely an artefact

higher in the general media compared to the scientific literature: in Factiva, the trend seems to weaken to some degree, but in the scientific literature, more than half (58.1% WoS and 55.4% Scopus) of the papers have been published in the last 2 years. The diminishment in the last year is mainly an effect of database curation, as both databases do not yet contain all papers published in 2017 at the time of updating (February 21, 2018).[5]

The enormous dynamics in the field is also discernible when looking at the number of highly cited papers (>10 citations per paper): In 2016, we identified 164 highly cited papers (2.60% of all papers identified), in 2018, 1333 papers (5.79%) fulfilled this criterion.

We remind that this search only focuses on the use of the term "big data" in the literature, not on more generic topics within computer science that are today attributed to the field of Big Data (such as data mining or data visualization techniques). Thus, the data does not allow to make inferences on how those scientific fields have developed over time.

Figure 10.2 shows the relative weight of Big Data papers that contain keywords of one of the six ethical categories (see Footnote 4). An interesting observation is that in the lay media (Factiva), papers referring to security and "self" (reputation and ownership issues) have a much higher weight compared to the two scientific

[5]Database curation also affects the comparison of the results from 2016 with those from 2018: We found that the number of entries already in the year 2015 more than doubled in WoS, but was comparable for the earlier years.

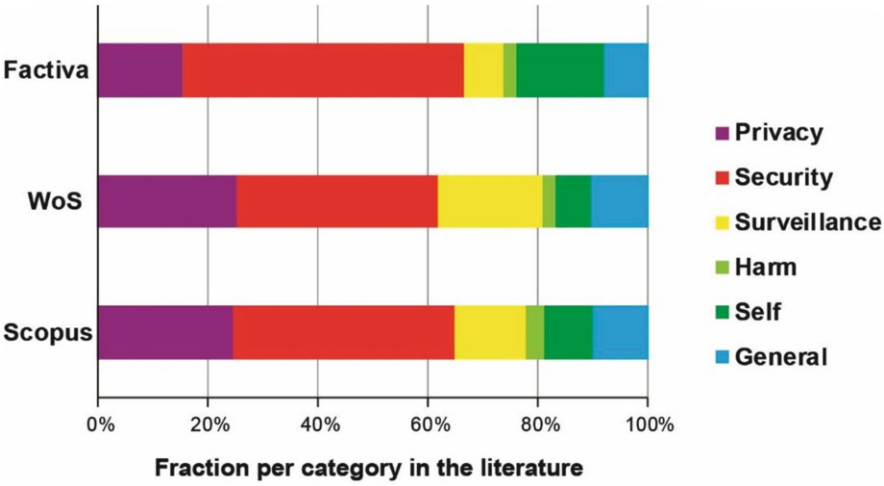

Fig. 10.2 Fraction of papers referring to one of the six ethics categories published between 2006 and 2017 in each database

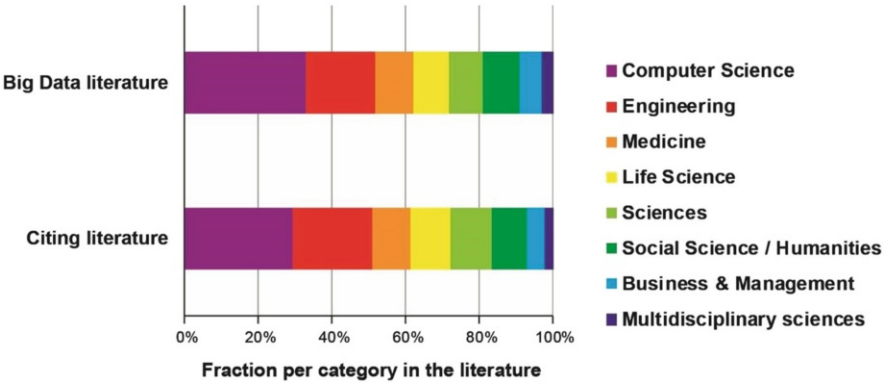

Fig. 10.3 Disciplinary profile of highly cited Big Data papers compared to the papers that cite them; 2018 data

databases, where the "classic" topics of computer ethics, namely, privacy and surveillance, are more relevant. This pattern did not change in the 2018 update.

Finally, Fig. 10.3 outlines the disciplinary profile of the highly cited papers (1333 papers) and compares it with the profile of the papers citing them. This analysis gives an estimation on the "knowledge flow," that is, which disciplines tend to cite papers more (or less) often. Here, interesting differences between the 2016 and 2018 data is discernible. In 2016, in particular humanities and social sciences (the fraction increased from 6.6% to 13.2%), life sciences (increase from 12.2% to 16.5%) cite Big Data papers more often, indicating that the debate is more pronounced in these disciplines. In the 2018 data, such a difference when comparing publications and

citations is not discernible any more. Humanities and social sciences, for example, now account for 10.0% of all publications compared to 6.6% in 2016. Both in the 2016 and 2018 data, a decrease in the domain "economy/management" is visible (from 6.9% to 4.4% in 2016 and from 6.2% to 4.8% in 2018), which could indicate a less-developed discussion of Big Data in this domain compared to other disciplines.

In summary, the bibliometric analysis shows that the topic "Big Data" is very young and displays indications of a "hype." Furthermore, there are different weights with respect to the ethical debate when comparing the scientific literature with the lay media. Finally, there are indications that Big Data is particularly discussed in the life sciences, humanities, and social sciences.

3.2 Use Cases

In the following, we briefly describe five use cases for outlining Big Data applications in the consumer sector. In Sect. 4, we discuss ethical issues of Big Data based on these cases.

Case 1: Prevent Debt Loss Big Data allows new ways for companies to assess the payment moral of their customers based on the customers' digital behavior. Traditionally, companies relied on registries such as the Swiss "Zentralstelle für Kreditinformation" or the German "Schutzgemeinschaft für allgemeine Kreditsicherung" for credit rating. As an alternative, social scoring based on Big Data is increasingly used. For this, algorithms analyze entries and behavior of customers on social networks (friends, likes, leisure activities) as well as information that—on the first sight—seems to be unrelated to credit rating such as search behavior, fonts used when writing, speed of filling out forms, or technical data of the computer used when surfing the Internet. Social scoring is quite common in online shopping, for example, for determining whether a customer is allowed to pay the bill using an invoice. Also in banking, social scoring gains importance. For example, some banks provide credit under the condition that the customer downloads an App that collects personal information of various kinds (geographic location, duration of phone calls, address book information, etc.)—the more data the customer reveals, the higher are the chances for getting a credit (an example of such a social scoring system for improving the access to credits is provided by the Australian loans and deposits application platform Lodex; Graham 2017).

Case 2: Improve Risk Management For many industries such as insurances, risk management is key for business success. Big Data provides the opportunity to better evaluate risks, for example, the probability and magnitude of damages. In particular, risks can be assessed more individually. An example is the use of telematics solutions in car insurances. Information on how the customer is driving (speed, acceleration, braking behavior, duration of driving, distances, etc.) allows to calculate the probability of an accident or of car theft, leading to "pay as you drive" models. Wearables such as smartwatches or fitness trackers provide information that

is relevant for health insurances. For example, unhealthy behavior of customers can be tracked more easily, allowing for prevention to decrease the insurance rate. Also, the genetic information of people can be more easily determined and shared using online services. Big Data also allows for better identification of fraud by customers using profiling and predictive modelling.

Case 3: Tailor Offer Conditions Traditionally, companies calculate prices for large customer groups based on costs, prices of competitors, and aggregated customer behavior. Big Data now allows in principle to determine the individual "best price" for each customer. The first step for this is dynamic pricing, which has become a standard in several industries. Airlines, e-businesses, or gas station providers use algorithms to dynamically change prices based on various types of information (demand, availability, weather, time in the day, behavior of competitors). Individualized prices are the next step, allowing to best skim the consumer surplus. For this, not only environmental factors used in dynamic pricing but also information on the individual customer is used (e.g., gender, age, geographic origin, friends, personal preferences) based on cookies, customer cards, smartphone ID, GPS position, IP address or other technical means. For example, an online tour operator displayed higher prices to Apple users, because they tend to be less price-sensitive when booking (Mattioli 2012). An online shop gave away discount tickets based on the individual shopping behavior of the customer for nudging the customer to products with higher prices (Metzler 2016).

Case 4: Increase the Efficiency of Advertising Advertising is effective when it reaches the customer who is potentially interested in the product—but traditional advertising campaigns mostly work based on the "shotgun approach" (e.g., billboard advertising). The accuracy of such campaigns is increased using Big Data, based on search history, social network data, GPS data, etc., of customers. A common technique is re-targeting: a customer that searched for a specific product finds advertising of this product later on many other sites he/she is visiting. Pre-targeting aims to show potential products to the customer based on his/her online behavior. Geo-targeting aims to show advertising related to the geographic localization of the customer (e.g., a nearby restaurant). Future applications that are currently investigated experimentally in computer games include emotional targeting: based on visual (face expression) and auditory information (voice recognition), the emotional state of the customer is assessed in order to display advertising adapted to his/her emotional state.

Case 5: Create Business Innovations Big Data is also used for generating new revenue sources or enlarging the product or service portfolio of a company. For example, companies could sell the information of their customers to other companies, which includes the possibility to supplement existing products (e.g., sportswear) with sensors (wearables) that generate this data. Telematics systems in cars can be used to better identify bugs in new models or provide new maintenance services. Data emerging from social networks, etc., can be used to identify trends and adapt product developments to such trends (e.g., in car manufacturing or streaming

services that develop own TV shows). Data generated from voice recognition software can be used to increase voice control technology or related services such as translation. Big Data also allows for innovations in infrastructure or traffic planning. For example, traffic jams could be prevented by such systems.

4 Ethical Evaluation

In the context of economy, ethics is usually construed to be an antagonist of business in the sense that ethics defines which business activities are legitimate and which are not. Although this is one of the functions of ethics, such a perspective is missing the following aspects:

1. Market economy itself has a moral foundation by assuming that a regulated market economy with informed actors serves to pursue ethical goals such as individual freedom and public welfare.
2. Ethical values and norms are usually abstract and need to be applied to concrete problems, which is in most cases not a straightforward process.
3. The claims associated with ethical values and norms can be in conflict to each other, which requires some degree of balancing.

The following analysis structured along eight ethical values is based on these preconditions.

4.1 Protection of Privacy

Article 12 of the Universal Declaration of Human Rights[6] declares that "no one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honor and reputation. Everyone has the right to the protection of the law against such interference or attacks." The goal of this norm is to protect spheres of life of the individual where he/she can move, develop, and behave freely. In the current Swiss data protection law[7] (which is currently under revision), this value of privacy is protected by the two principles of "purpose limitation" ("Zweckbindung") and "data minimization" ("Datensparsamkeit"): data should be collected only for specific purposes and only as much as is needed for this purpose. All five use cases above involve the potential to infringe these principles. For example, smartphone apps may violate these principles by collecting data that is

[6]Available at: http://www.un.org/en/universal-declaration-human-rights/ (last access: February 28, 2018).

[7]Available at: https://www.admin.ch/opc/de/classified-compilation/19920153/index.html (last access: February 28, 2018).

not necessary for their primary purpose. These risks increase when data emerging from different sources are combined in order to gain new insights, for example, in the case of pre-targeting (Case 4). However, the potential of Big Data lies in this combination of various data sets. Obviously, secondary use and recombination is in conflict with the principle of purpose limitation—and anonymization is not sufficient given the potential of re-identification when integrating data from different sources.

However, in the case of preventing debt loss (Case 1) and increase of risk management (Case 2), the violation of privacy has to be balanced with legitimate claims of the companies, which have the right to know about the solvency of their debtors and other business-relevant risks. Thus, not every violation of privacy is equally ethically problematic—and in cases 1 and 2 one would have to look at the individual circumstances. However, when in the case of tailoring offer conditions (Case 3) data of very different type are integrated (financial transactions, credit rating, medical treatments, social relationships, etc.), an unjustified violation of privacy is likely. Furthermore, missing transparency, the potential of discrimination and the violation of contextual integrity of the data are additional risks that often go hand in hand with such applications; they are discussed below.

Increasing the efficiency of advertising (Case 4) and creating business innovations (Case 5) are legitimate goals of companies. In those cases, one has to balance the gains for the customers (e.g., not being disturbed by advertising that is not interesting at all for the customer) with the drawbacks (e.g., surveillance by the company). From an ethical perspective, violation of the privacy of the customer seems to be justified, if he/she is informed on the magnitude of data collection, if he/she has consented to data collection, and if there is a realistic alternative when not consenting. The problem that the general terms and conditions of companies are often hard to understand is discussed further below in Sect. 4.5.

4.2 Equality and Non-discrimination

Discrimination means the unequal treatment of persons that is not justified by ethically sound reasons. Non-discrimination is founded by fairness intuitions that are undisputed and that are mirrored by legal principles such as equality before the law. However, non-equal treatment of people is not necessarily ethically problematic or may even be requested (e.g., differences in wages when the work performance of people differs). But when non-equal treatment is based on criteria (such as gender, race, or political opinions) that are not relevant for accessing certain goods or positions, it becomes ethically problematic discrimination.

This can happen when tailoring offer conditions (Case 3): algorithms may classify persons based on properties they themselves can hardly influence. This may be the reason that businesses currently tend not to use individualized pricing directly but rely on mechanisms such as discount tickets—although this is not fundamentally different from the former. Using Big Data for tailoring offer conditions involves a systematic non-equal treatment of customers. However, it is in

principle not ethically wrong that the willingness to pay of a customer is part of the pricing mechanism, as long as one does not exploit a state of emergency and as long as there is no monopoly. Economic institutions such as the bazar or auctions are accepted mechanisms of using the willingness to pay for the pricing mechanism. The problem of Big Data, however, is that information asymmetry is involved between the vendor and the customer, for example, when the company has access to information about the psychological profile of the customer based on his or her behavior on social networks and the customer is unaware that this data (and the associated model) is available to the vendor. This may undermine the mechanisms of efficient allocation in markets. Of particular ethical relevance is that customers may not know based on which mechanisms they may be treated.

Individualized prices are problematic when they are used ubiquitously within an industry and customers have no possibility for evasion. However, this argument is only valid when there is only one mechanism to determine the individual price. This would require that all providers use the same algorithm and the same data—the situation of a classic monopoly or syndicate, that is, a market without competition.

The example of social scoring (Case 1) involves both benefits and risks: On the one hand, this mechanism offers access to credits for people who usually would never be able to enter the classical banking system. On the other hand, those digital credit providers require disclosing sensitive personal data that is not the case in classical banking. Is this unjustified discrimination? Again, this example requires balancing of the violation of privacy with the right of the company to prevent credit losses. Ethically problematic is when the disclosed data allows for inferences for third persons or is later used in a way that violates the contextual integrity of the data.

4.3 Informational Self-Determination

Usually, informational self-determination is defined as the right of an individual to decide upon collecting, storing, using, and transferring of personal data. This right is founded in the value of autonomy, that is, the ability to shape and control the personal life. A practical expression of informational self-determination is informed consent—a concept that originally emerges from the medical sphere. In the case of data this means that a person should consent explicitly and in an informed way to the use of his/her personal data.

Informational self-determination is likely to be violated when targeted advertising aims to manipulate the person, for example, in the case of emotional targeting (Case 4). Problematic is that the persons did not explicitly consent to the use of their data (e.g., facial expressions). Furthermore, the manipulative nature of the intervention may undermine the process of free opinion forming. That advertising has some manipulative character is not new and willful deception is surely wrong. But the use of Big Data has the potential to strongly increase the efficiency of such mechanisms. A general statement is, however, not possible and requires a case-by-case evaluation. Another problem is that customers who insist on their right of informational self-

determination may not be able to get certain services when denying access to personal information or would have to pay substantially more. This may result in a de facto discrimination that can be considered unfair.

4.4 Controlling the (Digital) Identity

Closely related to informational self-determination is the claim of being able to control the own digital identity. Digital identities can be the result of Big Data applications, when features of customers are aggregated and correlated. At first glance, single data points such as how people use a keyboard or when they usually do phone calls seem to be unrelated to, for example, the credit rating of a person. However, when multidimensional data is aggregated and analyzed, categories can be created in order to match the digital identity of persons with these categories—as exemplified in Case 1. This is particularly ethically problematic, when the person does not know that his/her data is used in this way and when the person has no possibility to change his/her attribution to one of these categories, in particular when the attribution is obviously mistaken or was based on a spurious correlation.

An additional problem is that such digital profiles may include outdated data—so there is no forgetting or prescription. Data related to personal situations are context-dependent with respect to the age of the person: youthful folly leave digital traces that then may fall back to the adult person. Companies that use automatized techniques of Big Data analysis without being aware of this time- and context-dependency of personal data treat their customers unfairly, because they cannot contribute to the interpretation of their own data.

However, the fact that digital identities are always incomplete and selective is not per se ethically problematic, as it is in the nature of things. Persons themselves often construe digital identities adapted to contexts (e.g., a profile on a dating network compared to a profile on a business network), which is actually an expression of informational self-determination, as long as there is no intention of deception. Rather, the incomplete digital identities that companies may have from their customers result from the principle of data minimization.

Furthermore, if people change their digital behavior in anticipating that the data generated through this behavior helps to construct digital identities is not a new problem. Customers, for example, may change their online behavior in order to profit from discounts. This kind of heteronomy of the own digital identity is part of the way people tend to interact socially in order to increase personal advantages. An ethical problem, however, arises when no room for "non-strategic behavior" is left due to ubiquitous commercial surveillance of the digital behavior of people.

4.5 Transparency

Transparency in the context of business means that customers, business-partners, or investors are informed properly of the state of a company, its business processes, services, and products such that they can form an informed decision (e.g., whether to invest or not). Thus, transparency is also a precondition for any informed consent for using personal data. Without transparency, markets cannot function properly. In the case of Big Data, transparency means that every person has the right to know which data are collected about him/her and how they are used.

An obvious problem of transparency is that companies tend to provide extensive and incomprehensible terms and condition forms toward their customers. Although one can expect that customers should read these forms, the way they are presented practically prohibits an informed decision unless one is a legal expert. Furthermore, there is a lack of transparency which data are collected on which online platforms, who is analyzing this data, and to whom the results of these analyses are given. Often, app providers sell the collected data to third parties without informing their customer properly—they probably fear that customers will not use their services any more if they would know.

When tailoring offer conditions (Case 3), the algorithms used for generating the prices are usually confidential. Furthermore, the data cannot be checked with respect to their quality, reliability, and completeness. Whereas online trading made markets more transparent, because comparing prices became easier, Big Data now undermines this gained transparency. Thus, tensions appear between the claims of companies to protect their algorithms (i.e., the intellectual property associated with them) and the claim of customers for transparency. In liberal societies, solving this challenge is a task of the company primarily. Legal regulation should be considered when they fail doing so.

Using learning systems aggravates this problem: Those algorithms may increasingly be used for preventing debt losses. For example, so-called deep learning algorithms may learn classifications of risk ratings that are even intransparent for the software developers—they are "black boxes" (Pasquale 2015). This is a fundamental problem of many currently used machine-learning systems relying on neural networks, as there are currently few underlying theories that explain how or why the models are effective for a particular type of problem and no baseline exists to predict their eventual performance. Machine-learning models are equations that have no obvious underlying physical or logical basis. Reading these models provides no insight into the underlying phenomena, where they originated, or how they will behave in a given situation. Furthermore, a model may produce radically different results for two scenarios that seem quite similar to humans. This poses significant problems related to testing (and trusting) such algorithms (Informatics Europe & EUACM 2018).

This lack in transparency endangers the value of informational self-determination, because people may agree to reveal information that is processed in a way that leads to a new type of classification scheme where no person reasonably

expects that this scheme emerges. The decision to reveal this information is thus based on the wrong assumption that one understands what one can reasonably infer from the information one discloses. For example, people may accept to disclose their favorite color, preferred food, and most-liked movies on a convenient website by considering this information as unproblematic—and neither the person nor the provider of this website initially had the idea that a complex evaluation algorithm could infer out of this information the risk for insurance fraud. Furthermore, it could involve legal risk with respect to the new EU data protection legislation that restricts what the EU calls "automated individual decision-making"—the task of supervised machine learning like deep learning neural networks (Goodman and Flaxman 2016).

4.6 Solidarity

Solidarity concerns duties of mutual support in a community. In today's social state models, solidarity involves financial support in case of illness or poverty, based on the intuition that every human being could end up in a situation of need independent of negligence. In this way, solidarity provides the moral foundation of any type of insurance, whereas the range of solidarity is limited to some degree by the costs-by-cause principle. Persons who intentionally cause harm to themselves are less likely to benefit from the solidarity of others.

The challenge of Big Data is that the aggregation of multidimensional data for increasing risk management (Case 2) could lead to an extension of the costs-by-cause principle. A certain online behavior could be coupled to a higher risk of, for example, liability—making this behavior object of potentially "causing" liability, as the behavior results from a free choice. For example, playing certain video games may be correlated with a higher incidence of being absent from work due to illness—and the "choice" of playing these games may finally become a reason to deny solidarity in that case. These kinds of analyses could also provide a conceptual basis for prevention programs, for example, insurance companies could demand for certain diets or fitness programs to decrease certain health risks. This type of behavior control that is economically attractive for insurance companies is in conflict with the right of self-determination. To prevent this undermining of solidarity, there are legal barriers. In Switzerland, for example, health insurance companies are not allowed to exclude anyone from basic health insurance based on their behavior.

4.7 Contextual Integrity

The human environment is structured in social spheres that provide important reference points for human beings. They expect to be treated differently in a family context compared to, for example, in a governmental organization. They accept inequality in treatment in the economic sphere that they would not accept in the

health, legal, or education spheres. The interpretation of moral values such as justice[8] or autonomy, and the rules related to these values differ along these social spheres. Accordingly, also the way information is produced and the social meaning people attach to information differs in these spheres. This is what is called the contextual integrity of information (see also Sect. 2.1). For example, if a person discloses personal information in the health sphere for research purposes, the moral foundation of this choice is to help other people. But if this information is used in a different sphere such as the economic sphere, to tailor offer conditions (Case 2) or to maximize profit (Case 5), the original intention to disclose this information and thus its contextual integrity is violated.

Based on these considerations, Big Data relying on multidimensional sources inherently entails the danger to violate contextual integrity of data. As data are increasingly traded by data brokers and are used in complex algorithmic or statistical models to make inferences on person groups, a violation of contextual integrity of data is hard to detect even for the commercial user of such data. This also undermines the value of transparency.

It is hard to evaluate which violations of contextual integrity pose an ethical problem, as the borders between social spheres as well as the rules within those spheres are fluid to some degree. The interpretation of values can change, for example, when individuals generally tend to disclose more personal information in social networks and also have this expectation toward their fellow humans—and in this way change the normative weight of privacy. Nevertheless, social spheres remain central points of reference for understanding the world, which explains why most people are filled with indignation when information emerging from their personal friends is used for individualized prices (Case 3).

4.8 Property and Copyrights

The functioning of the economic sphere depends on certain moral foundations, among which are the property right and the copyright. Both values are protected, for example, by the Swiss constitution. In the case of Big Data the question emerges, whether data also falls under these legal norms.

Using online services often entails the generation or disclosure of personal data, which is the basis for new revenue sources of companies (Case 5). The economic value of some companies is even measured based on the number of their customers and the amount of data they generate. This poses the question: who owns this data? When customers generate data on location or device usage when using smartphones, tablets, or computers: are these data streams creations in the sense of copyright law? Or is this the case not until companies use technologies to analyze this data?

[8]In the case of justice, different allocation rules exist. Examples include “an equal share for everyone” or “sharing according to needs”.

Depending on how these open questions are answered, the foundations of business cases of many companies active in Big Data may be shattered. For example, customers could have a share in the profits made by these data or they should have the right that the company deletes all personal data of this person, as foreseen by the new EU General Data Protection Regulation (GDPR).

From an ethical perspective, both the data provider and the companies that invest in the analysis of this data should have a fair share of the profit. For the latter, the current economic system is concerned, as companies only invest when they reasonably expect revenues—and they are free in investing or not. However, this is not the case for the data providers. The current model is that the customers are compensated by freely using certain services. In the future, this may not be sufficient and companies should start considering alternative compensation models. As the GDPR[9] foresees mechanisms such as data access (Art 15) and data portability (Art. 20; data subject have the right to receive the personal data concerning him or her), companies have incentives to increase trust and fairness toward their customers providing data. This may provide an incentive for new compensation models in order to avoid costly executions of those rights.

5 Lessons Learned

As our use cases demonstrate, Big Data is transforming the way companies develop, produce, offer, and advertise products and services. This may lead to added values both for businesses and their customers—but these applications entail also ethical risks, as they affect core values such as informational self-determination, transparency, and contextual integrity. Companies are well advised to be sensible to those risks, as customers and other stakeholders are likely to become more critical with respect to violation of those values. Also, Big Data applications need a "license to operate" and companies have to demonstrate a responsible use of data. We therefore suggest the following:

- **Take the "ethics case" into account:** When evaluating novel Big Data applications, companies should not only focus on the business case, but they should from the beginning also evaluate which of the core values described in this study may be affected in what way. This systematic involvement of ethics should be mapped on the appropriate corporate responsibility structures. Companies may also consider creating industry guidelines for a responsible use of Big Data.
- **Take the customer-point-of-view:** A simple test case for assessing the ethical risks of a Big Data application is the following: Would the customer still agree on disclosing his/her data when he/she knows exactly what is done with the data? What is the actual state of knowledge of the customers with respect to Big Data

[9] Available at: https://gdpr-info.eu/ (last accessed February 28, 2018).

and how likely is it that this will change? What are likely benefits customers would accept for trading in their data?

– **Create transparency and freedom to choose:** Trust and acceptance of consumers is a mandatory requirement for the successful application of Big Data. This requires that companies inform transparently and comprehensibly on how data is collected and used. Depending on the service, opt-in solutions and the provision of acceptable alternatives are successful strategies in that respect.

6 Conclusions

This chapter provided a summary of a study that intends to outline the ethics of Big Data applications in the consumer sector. It made clear that an ethical evaluation always involves a balancing in the single case in order to evaluate whether the violation of some core values can be justified. Individual companies may be overburdened in performing such an evaluation, making a public-private partnership advisable. The state should support the relevant industries in creating industry standards. In particular, some kind of standardization of general terms and conditions forms may be advisable in order to increase the informed consent capacity of customers. The goal should be that customers, and citizens, are empowered to better understand the way and magnitude of data collection and analysis in the age of Big Data.

References

Boyd, D., & Crawford, K. (2012). Critical questions for big data. Provocations for a cultural, technological, and scholarly phenomenon. *Information, Communication and Society, 15*(5), 662–679.

Davis, K., & Patterson, D. (2012). *Ethics of big data*. Sebastopol, CA: O'Reilly Media.

Fiske, A. P., & Tetlock, P. E. (1997). Taboo trade-offs: Reactions to transactions that transgress the spheres of justice. *Political Psychology, 18*, 255–297.

Friedman, B., Kahn, P. H., Jr., & Borning, A. (2006). Value sensitive design and information systems. In P. Zhang & D. Galletta (Eds.), *Human-computer interaction in management information systems: Foundations* (pp. 348–372). New York: M.E. Sharpe.

Goodman, B., & Flaxman, S. (2016). EU regulations on algorithmic decision-making and a "right to explanation". *2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*, New York, NY. Available at http://arxiv.org/pdf/1606.08813v1.pdf

Graham, B. (2017). *How banks could use an online 'score' to judge you*. news.com.au, November 6 2017. Available at http://www.news.com.au/finance/business/banking/how-banks-could-use-an-online-score-to-judge-you/news-story/009ca6df681c5fc69f583c4feac718c2

Hauser, C., Blumer, H., Christen, M., Huppenbauer, M., Hilty, L., & Kaiser, T. (2017). Ethical challenges of big data. *SATW Expertenbericht*. Available at https://www.satw.ch/digitalisierung/detail/publication/ethische-herausforderung-von-big-data/

Helbing, D., Frey, B.S., Gigerenzer, G., Hafen, E., Hagner, M., Hofstetter, Y., van den Hoven, J., Zicari, R.V., & Zwitter, A. (2015). Das Digitale Manifest. Digitale Demokratie statt

Datendiktatur. *Spektrum der Wissenschaft*, 17.12.2015. Last accessed August 10, 2016, from http://www.spektrum.de/news/wie-algorithmen-und-big-data-unsere-zukunft-bestimmen/1375933

Informatics Europe & EUACM. (2018): *When computers decide: European recommendations on machine-learned automated decision making*. Last accessed February 28, 2018, from http://www.informatics-europe.org/component/phocadownload/category/10-reports.html?download=74:automated-decision-making-report

Lane, J., Stodden, V., Bender, S., & Nissenbaum, H. (Eds.). (2014). *Privacy, big data, and the Public good: Frameworks for engagement*. Cambridge: Cambridge University Press.

Mattioli, D. (2012, August 23). On Orbitz, Mac users steered to pricier hotels. *The Wall Street Journal*. Available at http://www.wsj.com/articles/SB10001424052702304458604577488822667325882

Mayer-Schönberger, V., & Cukier, K. (2013). *Big data: Die revolution, die unser Leben verändern wird*. München: Redline.

Metzler, M. (2016, October 23). *Reiche bezahlen mehr.* NZZ am Sonntag. Available at http://www.nzz.ch/nzzas/nzz-am-sonntag/personalisierte-preise-reiche-bezahlen-mehr-ld.123606

Nissenbaum, H. (2004). Privacy as contextual integrity. *Washington Law Review, 79*, 119–157.

Pasquale, F. (2015). *The black box society: The secret algorithms that control money and information*. Cambridge: Harvard University Press.

Tanner, C., & Medin, D. L. (2004). Protected values: No omission bias and no framing effects. *Psychonomic Bulletin and Review, 11*(1), 185–191.

Tetlock, P. E., Kristel, O. V., Elson, S. B., Green, M., & Lerner, J. S. (2000). The psychology of the unthinkable. Taboo trade-offs, forbidden base rates, and heretical counterfactuals. *Journal of Personality and Social Psychology*, (5), 853–870.

Van de Poel, I., & Royakkers, L. (2011). *Ethics, technology, and engineering: An introduction*. Hoboken: Wiley.

Van den Hoven, J. (1999). Privacy and the varieties of informational wrongdoing. *Australian Journal of Professional and Applied Ethics, 1*, 30–44.

Van den Hoven, J., Helbing, D., Pedreschi, D., Domingo-Ferrer, J., Gianotti, F., & Christen, M. (2012). FuturICT – The road towards ethical ICT. *European Physical Journal – Special Topics, 214*, 153–181.

Chapter 11
Statistical Modelling

Marcel Dettling and Andreas Ruckstuhl

Abstract In this chapter, we present statistical modelling approaches for predictive tasks in business and science. Most prominent is the ubiquitous multiple linear regression approach where coefficients are estimated using the ordinary least squares algorithm. There are many derivations and generalizations of that technique. In the form of logistic regression, it has been adapted to cope with binary classification problems. Various statistical survival models allow for modelling of time-to-event data. We will detail the many benefits and a few pitfalls of these techniques based on real-world examples. A primary focus will be on pointing out the added value that these statistical modelling tools yield over more black box-type machine-learning algorithms. In our opinion, the added value predominantly stems from the often much easier interpretation of the model, the availability of tools that pin down the influence of the predictor variables in concise form, and finally from the options they provide for variable selection and residual analysis, allowing for user-friendly model development, refinement, and improvement.

1 Introduction

Statistical modelling refers to the technique of finding a systematic relation between a response variable y and a number of predictors while accounting for random deviation. Mathematically, this may be expressed as

$$y \approx f\left(x_1, x_2, \ldots, x_p\right)$$

In what sense random deviation makes the relation approximate remains to be clarified. Since function space is infinite-dimensional, it is impossible to learn such relations from observed data that always come in finite quantities only. The problem

M. Dettling · A. Ruckstuhl (✉)
Institute for Data Analysis and Process Design, ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: andreas.ruckstuhl@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_11

of learning $f()$ from data is only feasible if previous knowledge is available and/or (strong) assumptions on the structure of $f()$ are made. A simple albeit very popular way out is the restriction[1] to linear models, that is,

$$f(x_1, \ldots, x_p) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p.$$

This reduces the problem to learning the parameters $\beta_0, \beta_1, \ldots, \beta_p$ from data, while considering an appropriate form of random deviations. In many cases, this is a challenge that can be tackled easily. In this chapter, we detail how such linear models can be adapted to cope with various types of response variables y; how they can be made more flexible by incorporating variable transformations and further generalizations; in which form they are used in practice; and what the benefits over other, more complex and modern approaches are.

Multiple linear regression models and the *ordinary least square algorithm* (OLS) for estimating their coefficients date back as far as to the beginning of the nineteenth century, when these techniques had first been used to solve applied problems in astronomy (Plackett 1972; Stigler 1981). They are applied for dealing with quantitative response variables y that are on a continuous scale, that is, $y \in (-\infty, +\infty)$ and an additive Gaussian error term E to model the random deviations:

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + E.$$

The main advantage of the OLS algorithm lies in the existence of a unique solution that can be written in explicit form; hence, it was possible to fit such statistical regression[2] models without the use of modern computers. Even today, it remains one of the most frequently applied techniques in data analysis. This popularity may be rooted in the fact that while the solution of a linear regression model is easy to interpret, the use of variable transformations allows for great flexibility and accuracy for many predictor/response relations. Finally, but importantly, there are also mathematical optimality results for the OLS estimator (see, e.g., Sen and Srivastava 1990). Simply put, it can be shown that there are no other, equally simple approaches that are more efficient. In this chapter, we will introduce the basic theory and explain the practical application with a special focus on the benefits of multiple linear regression over more modern but less transparent methods. All this will be

[1] In practice, the restriction is not as severe as it may seem. We point out what is encompassed by "linear models" in Sect. 2.1.

[2] This nomenclature is somewhat unfortunate since it has little to do with "going backward", as the word regression implies. The name has been coined in an early (1885) application by Sir Francis Galton, which dealt with the relationship of heights of parents and heights of offspring. He showed that the heights of children of tall parents tend to regress down toward an average height and vice versa. He called this biological phenomenon "regression toward mediocrity". His analysis approach was later extended and applied in more general studies involving relationships but the biologically motivated nomenclature has been preserved.

based on an example that is taken from the authors' research and deals with predicting the daily cost of inpatients in neurological rehabilitation.

The *logistic regression* model is used in cases where the response variable y is a binary 0/1 variable and has been introduced by David Cox in 1958. Although it can be used for classification, it is more suited to determine class membership probabilities and to recognize which of the used predictors drive the variation in the response. While the logistic regression problem has no explicit solution and (basic) numerical optimization is required, the approach is still relatively simple and theoretically well founded. As with multiple linear regression, the identified prediction model is easy to interpret, enhance, and verify. Moreover, it often provides a concise and intuitive output. We will explain theory and practice for logistic regression models using an example rooted in the authors' research. It deals with estimating churn (contract cancellation) probabilities for a Swiss telecommunication provider.

Another sub-field of regression modelling is the analysis of *time-to-event data*. For these kinds of problems, the durations until a specified event happens are analyzed in order to estimate how they are affected by prognostic factors, treatments, or interventions. The considered event may be death, failure of a technical component, or any other endpoint such as healing, divorce, promotion of employees, termination of a task, arrival of a request, and so on. One of the distinctive features of such data is that not all events necessarily occur within the observed period or that earlier events have already happened that preclude the target event (e.g., a failure of a system cannot be observed anymore after the system has been replaced). Hence, for some observations, we know a priori that the time to event is larger than the observed duration. Such data are called *censored*. A crucial point is dealing with censoring correctly in a statistical analysis of such data.

First techniques might be attributed to the early work on mortality tables in the seventeenth century (John Graunt published a mortality table of London in 1662). The modern era started during World War II with applications to the reliability of military equipment. Most of the statistical research for engineering applications was concentrated on parametric distribution models, such as the Weibull (regression) model. Applications in life sciences in the second half of the twentieth century shifted the statistical focus to semi-parametric approaches like the proportional hazard approach in *Cox regression* (Cox 1972), where the distribution family of the duration is unknown. We will give some insight into analyzing time-to-event data in cases of discretized failure times of water pipe data. Obviously, the reliability of pipes is affected by age and many more factors. The big advantage in such a setting is that it allows us to use the binary regression framework.

The remainder of this chapter is organized as follows: First, we give some further background information on the statistical models used in the examples. Then we discuss three application cases mainly with respect to our understanding of the statistical modelling task. Finally, we will conclude with some general thoughts.

2 Background Information

2.1 Multiple Linear Regression

When people talk about statistical modelling and regression techniques, they usually have the basic multiple linear model in mind. This corresponds to the formula

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + E, \tag{11.1}$$

where y is the response variable, $x_1, \ldots, x_p$ are some predictors, and E is an error term, accounting for the remaining (random) variation in the response. Usually, a Gaussian distribution is assumed for E and the regression coefficients $\beta_0, \beta_1, \ldots, \beta_p$ are estimated using the ordinary least square (OLS) algorithm. Its idea is that the coefficients are determined such that the sum of squared residuals, that is, the differences between observed and predicted values, are minimized. Under some mild conditions, this problem has a unique and explicit solution, so that numerical optimization is not required. In case of independent and identically distributed Gaussian errors, OLS corresponds to the *maximum likelihood estimator* (MLE), assuring that this is an efficient technique. Furthermore, exact confidence intervals and tests for the regression coefficients and the predicted values can be derived, allowing for theoretically sound inference and precision intervals. There are many textbooks discussing the theoretical foundations of statistical regression analysis. We recommend the work of Montgomery et al. (2006).

One may fear that the relatively simple linear combination of predictor variables is not powerful enough to meet the needs of the data scientist. However, by using transformations and interaction terms, the model can be greatly enhanced without leaving the theoretical and methodical envelope of OLS regression. For any predictor variable, arbitrary transformations such as the log ($x_1' = f_1(x_1) = \log(x_1)$) or powers ($x_2' = f_2(x_2) = x_2^2$) can be carried out. In addition, interaction terms such as $x_3' = f_3(x_1, x_2) = x_1 x_2$ may be added to the model. The only technical restriction is that the model has to remain *linear in the regression coefficients* $\beta_0, \ldots, \beta_p$, excluding terms such as $x_4^{\beta_4}$. Furthermore, the data analyst has to decide on the right transformations and model formulations mostly by himself, so statistical regression modelling remains a creative act, guided by intuition and field knowledge. Even further flexibility is gained if the response variable $y' = \log(y)$ is used, that is, a log-transformed response is linked to a multiple linear regression model with Gaussian error. If we express that relation on the original scale, we obtain

$$y = e^{\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p + E} = e^{\beta_0} \cdot e^{\beta_1 x_1} \cdot \ldots \cdot e^{\beta_p x_p} \cdot e^{E}.$$

This is now a multiplicative relation with a multiplicative error term that follows a *lognormal distribution*. Since all terms on the right hand side are positive, all predicted values will be positive too. Because many response variables in data analysis are strictly positive, this enhanced multiple linear regression model very

often is more realistic than the original one. Please note that it is still possible to use variable transformations of predictor variables or adding interaction terms.

2.2 Logistic Regression

From a basic view, logistic regression can be seen as a multiple linear model (11.1) for a *transformed* target variable. If $p(x) = P(y = 1|x_1, \ldots, x_p) = E[y|x_1, \ldots, x_p]$, we can work with the *logit transformation*:

$$\log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \beta_1 x_1 + \ldots. + \beta_p x_p \tag{11.2}$$

Again, transformations of the predictor variables and/or interaction terms may also be included. Hence, the essence again lies in using a linear combination on the right-hand side of the model equation. The logit transformation ensures that the predicted values for $p(x)$ are restricted to the interval $[0, 1]$. In fact, we can also re-express the relation for $p(x)$:

$$p(x) = \frac{\exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right)}{1 + \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right)}$$

From this formulation, it is obvious that the problem is no longer linear in the unknown regression coefficients $\beta_0, \ldots, \beta_p$, making their estimation more complicated than in the OLS model. Moreover, the absence of the error term E is notable. As we are estimating probabilities for a (conditionally) Bernoulli distributed response variable, there is no room for a Gaussian error, but the variation of the response can be accommodated conceptually with the estimation of probabilities for a 0/1 response variable.[3] Finding the parameters is again based on MLE principles, requiring the assumption of independent and identically distributed cases. Under these circumstances, the *Bernoulli log-likelihood* $l(\beta)$ for $p(x)$ can be optimized:

$$l(\beta) = \sum_{i=1}^{n} (y_i \log(p_i) + (1 - y_i)\log(1 - p_i))$$

Please note that this is a sensible *goodness-of-fit measure* for 0/1-response. For all observations with $y_i = 1$ we aspire for high p_i to keep the contribution to $l(\beta)$ low and vice versa for the observations with $y_i = 0$. Log-likelihood maximization needs to be

[3] More precisely, the target quantity is instead a suitably transformed conditional expectation of the original response variable itself.

done numerically here; usually an iteratively reweighted sequence of OLS regressions is used, minimizing issues with convergence to the global optimum.

Under the formulation we used, multiple linear regression and logistic regression are two statistical modelling techniques sharing some common background, namely, the linear combination of predictor variables on the right-hand side of the model equation. However, there are also very important differences (i.e., error term, optimization method). As it turns out, there is a common framework where both methods as well as related techniques for other types of response variables neatly fit. The framework is known as *generalized linear models* (GLMs, see McCullagh and Nelder 1989). The fundamental idea of GLMs and hence of all statistical regression models is to explain the suitably transformed conditional expectation of the response variable by a linear combination (of potentially transformed) predictor variables:

$$g\big(E\big[y|x_1, x_2, \ldots, x_p\big]\big) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$$

Rather than modelling the response or the probability for positive response directly, the approach consists of describing the (suitably transformed) conditional expectation of the response with a linear combination of the predictors. GLM theory furthermore lends itself to estimating the unknown coefficients using MLE and provides a battery of tests that can be used for assessing the output. For the multiple linear regression model, the conditional response distribution is Gaussian and MLE corresponds to the OLS algorithm. With logistic regression, the conditional distribution is Bernoulli (or Binomial for grouped data) and coefficient estimation is based on optimizing the respective likelihood.

2.3 Time-to-Event Models

Time-to-event models are technically not GLMs. However, with discretized duration times, *time-to-event models* can be expressed as binary regression models for which logistic regression is a special case. Before we go into details of such binary regression models, we introduce the *hazard rate* $h(t)$, which is the most often targeted quantity in time-to-event analysis. It quantifies the instantaneous risk that an event will occur at duration time t. The height of the risk may depend on the features of the object or subject at risk. The higher the hazard risk is, the earlier the event occurs and, hence, the distribution of the duration time is concentrated more toward the left on the positive time axis. Cox (1972) suggested modelling such a hazard rate as a product[4] of a baseline hazard rate $\lambda_0(t)$, which is left unspecified, and a linear function of a set of p fixed predictors that is then exponentiated:

[4]That is why it is also called proportional hazard model.

$$h_i(t) = \lambda_0(t) \cdot \exp\left(\beta_1 \cdot x_{i1} + \ldots + \beta_p \cdot x_{ip}\right).$$

The baseline hazard rate $\lambda_0(t)$ would specify the distribution[5] of the duration time, but it is unnecessary to explicitly do so. The second factor acts as a proportionality factor to the baseline hazard that depends on the predictor variables only. If there are many ties in duration times, as we have when they are discretized, for example, in years, the standard Cox estimating procedure is biased toward 0. However, one can show that a binary regression model is equivalent to the Cox model [see, e.g., Fahrmeir and Tutz (2001), or Kalbfleisch and Prentice (2002)]. In such cases, we consider the conditional probability of failing in time period k given by the discretization:

$$p(k|x) = E[\text{event happens at duration time period } k|X = x]$$

This can be interpreted as the hazard probability that the event occurs in duration period k, given the event has not occurred before. Note that $p(k|x)$ is a probability, whereas the hazard function $h(t)$ is a rate. For estimating the parameters, we could use the logistic regression model introduced in (11.2). In order to obtain a discretized version of Cox's model, we must however use the *complementary log-log transformation* of the probability $p(k|x)$:

$$\log(-\log(1 - p(t = k|x))) = \gamma_k + \beta_1 x_1 + \ldots \beta_p x_p, \tag{11.3}$$

where γ_k is the log-transformed baseline hazard probability at $t = k$ (see Sect. 9.2 in Fahrmeir and Tutz 2001, or Sects. 2.4 and 4.6 in Kalbfleisch and Prentice 2002). In addition, this transformation allows us to apply the methods of GLM for fitting the model. However, the data have to be prepared such that at every duration time period k all observational units are considered which are at risk that the event could occur. Thus, observational units whose event occurred before duration time period k or observational units which are censored will not be considered as being at risk. This setup allows dealing smartly with censored data while still applying standard estimation theory. There are many textbooks discussing theoretical concepts and applications of the analysis of time-to-event data, and we recommend the work of Allison (2010) for a practical introduction.

[5]There is an explicit relation between the density *f(t)* of a time-to-event distribution and the corresponding hazard rate *h(t)*, see Sect. 1.2.1 in Kalbfleisch and Prentice (2002).

3 Statistical Regression Models

In the following, we will explain the modelling process, as well as the benefits of multiple linear regression, logistic regression, and time-to-event regression models based on examples. All these examples are taken from the authors' research.

3.1 Multiple Linear Regression for Continuous Response

The goal in this first example is to study the daily cost in inpatient neurological rehabilitation. From seven hospitals, a random sample of 407 patients was obtained, most of whom were originally suffering from craniocerebral injuries or apoplectic strokes. The total (time) effort for care, therapy, and medical examinations was measured over the course of one week of stay, expressed in CHF/day and serves as the response variable, subsequently denoted with cost. Obviously, this average daily cost cannot take negative values, which needs to be taken into account when we set up our regression model. There are a number of predictors available for each patient: these include

- The hospital in which the patient was treated (*factor variable with 6 levels A-G, anonymized*).
- His insurance plan (*factor variable with 3 levels basic/medium/comfort*).
- Whether the patient felt pain at the start of the survey period (*factor variable with 3 levels no/mild/severe*).
- A numerical score (14–45) reporting about each patient's multimorbidity
- The ADL (*activities of daily life*) assessment was taken at the start of the survey period (the ADL assessment is based on about 20 items that quantify the autonomy of a patient, e.g. personal hygiene, feeding, etc. and results in a numerical score between 0 (maximum autonomy) and 56).

Please note that the original study was more complex, with some simplification being performed here to give a concise exposition. In Fig. 11.1, we first display a histogram showing the distribution of the response variable "cost" and a scatterplot, showing the univariate dependence of cost on ADL. We start out with a simple regression model and estimate the regression coefficients β_0, β_1 using OLS. This does not yield a sensible and correct solution. As the plots in Fig. 11.1 indicate, the scatter around the red regression line is imbalanced, with much bigger deviations toward higher than lower costs. This leads to a regression line that in general suggests too high expected costs. Both phenomena are visible in the scatterplot (top right panel in Fig. 11.1). However, some diagnostic plots as shown in the bottom two panels allow for clearer identification of the model's deficits and in more complex, multiple regression models these are the only options for assessing the quality of a model. The plot of residuals vs. fitted values clearly indicates a systematic error in the regression relation, while the normal plot very clearly

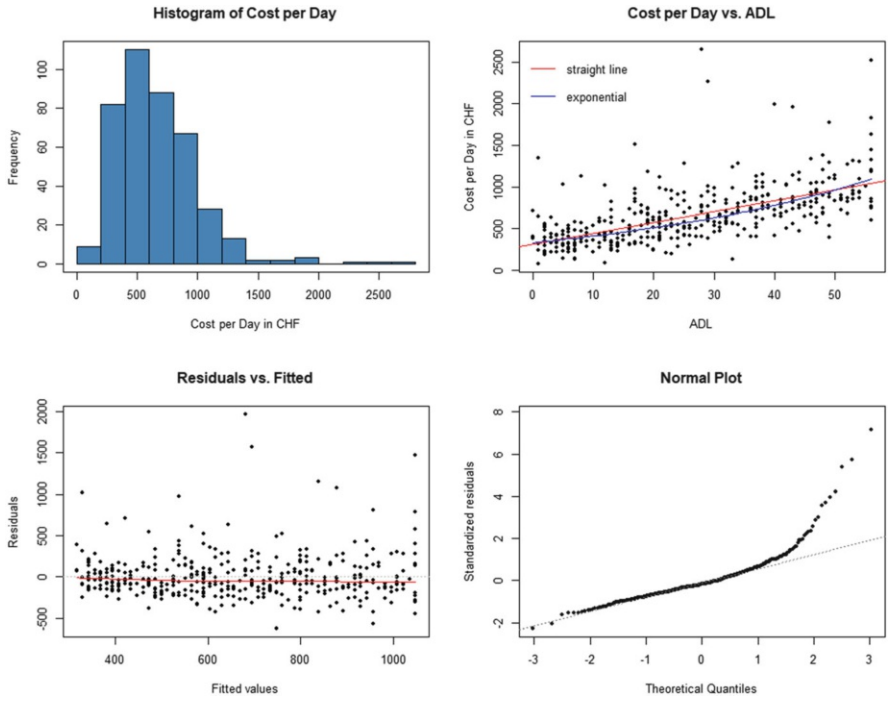

Fig. 11.1 In the top left panel, a histogram of cost indicates that this variable is positive and right-skewed. The scatterplot in the top right panel shows the regression situation of cost vs. ADL. Two regression fits were added, in red the faulty straight-line model and in blue a more appropriate exponential fit obtained after variable transformation. The bottom panels show the popular diagnostic visualizations of residuals vs. fitted and the normal plot for the faulty straight line model. They indicate that there is a systematic misfit with skewed residual distribution

indicates that the residuals follow a strongly skewed distribution. Finally, but importantly, please note that this simple regression model without variable transformations will also result in prediction intervals with negative cost values that cannot exist in practice. The lesson we learn from this simple exposition is that there are diagnostic tools for assessing the quality of a regression fit and that it is relatively easy to recognize faulty models.

Improving these aforementioned drawbacks is surprisingly simple; we just have to log-transform the response variable, that is, use the model of $\log(\text{cost}) = \beta_0 + \beta_1 \cdot \text{ADL} + E$. If expressed for cost, this is an exponential relation with a relative, lognormal error term E':

$$\text{cost} = e^{\beta_0 + \beta_1 \cdot \text{ADL} + E} = \beta_0' \cdot e^{\beta_1 \cdot \text{ADL}} \cdot E'$$

This is much closer to reality from a theoretical and practical viewpoint (see Sect. 2.1) and the diagnostic plots (not shown here) also indicate a better fit for this model.[6] Simply put, the expected cost for a patient increases exponentially with his ADL value with multiplicative errors that follow a lognormal (and hence right-skewed) distribution. Due to the log-transformation, negative predictions for the cost variable will not appear.

Our initial goal, however, was to relate the daily cost in inpatient neurological rehabilitation not only to ADL but also to multiple variables at the same time. The main goals in this extended analysis are to perform variable selection and designing a practically suitable system that allows for precise yet simple prediction of daily patient costs on a weekly basis. Another requirement lies in correcting the prediction scheme for the confounding effect of hospital and insurance plan. As we will explain in the course of our analysis, these aims can ideally be targeted with multiple regression models and would be much more difficult to achieve with modern methods which do not explicitly report the contribution of the predictors. For the reasons mentioned above, we will stick to the log-transformation for the response variable and start out with the following model:

```
> fit <- lm(log(cost) ~ clinic + insurance + pain +
                        multimorbidity + adl, data=dat)
```

We here use some typical annotation for regression models that is similarly used in many statistical software tools. On the left hand side of "~", we have the response variable; while on the right there is a list of the predictors with which a linear combination will be formed, as explained in Sect. 2. The same residual plots as in Fig. 11.1 (not shown here) confirm that this model is more appropriate: there is no systematic error so that we can believe that the cost deviations for the various individuals are completely random with an expectation of zero. If anything, the distribution of the residuals is slightly long-tailed and violating the Gaussian assumption. As it is symmetric though, this will not negatively influence our fit in a worrying manner. Next, we inspect the typical regression summary that all statistical software provides in a similar fashion (Fig. 11.2).

From this output, there is a lot to extract which is very relevant for practical application. As it turns out, the spread between the most and least cost-effective hospital is at $\exp(0.063 - (-0.096)) = 1.172$. This means that the expected cost in clinic F is 17.2% higher than in clinic E. Such a difference is within a realistic range, since there are differences in quality and number of staff, the salaries, infrastructure, and treatment. Due to the positive regression coefficient estimates, we also recognize that there is an increase in expected cost with better insurance plans (e.g., by exp

[6]This approach is not the only way for improving the first, inadequate model. An alternative model is the gamma regression model, part of GLM, which we will however not pursue any further in this chapter.

```
> summary(fit)
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       5.496242   0.102768  53.482  < 2e-16 ***
clinicA           0.000000         NA      NA       NA
clinicB          -0.010156   0.060201  -0.169  0.86611
clinicC          -0.002814   0.068818  -0.041  0.96740
clinicD          -0.018907   0.071527  -0.264  0.79166
clinicE          -0.095570   0.066671  -1.433  0.15252
clinicF           0.062975   0.066246   0.951  0.34238
insurancebasic    0.000000         NA      NA       NA
insurancemedium   0.137015   0.061097   2.243  0.02548 *
insurancecomfort  0.195722   0.059157   3.309  0.00102 **
painno            0.000000         NA      NA       NA
painmild          0.011865   0.047709   0.249  0.80373
painsevere        0.040569   0.050398   0.805  0.42132
multimorbidity    0.013452   0.004075   3.301  0.00105 **
adl               0.020436   0.001264  16.164  < 2e-16 ***
---
Residual standard error: 0.3785 on 395 degrees of freedom
Multiple R-squared:  0.4797,  Adjusted R-squared:  0.4652
F-statistic:  33.1 on 11 and 395 DF,   p-value: < 2.2e-16
```

Fig. 11.2 Summary output of the multiple linear regression model that was fitted to the neurological rehabilitation data. It reports the estimated coefficients, that is, the contribution of the predictor variables in explaining the observed, logged cost. Additionally, the standard errors (i.e., precision of estimation) and information about the statistical significance of these predictors (columns entitled `t value` and `Pr(>|t|)` is given. The lower three lines report information about the model as a whole; they all target the ratio between the amount of the logged cost explained by the predictors and the remaining variation which remains unaccounted for

(0.137) = 1.147 or 14.7% for the medium over the basic insurance plan or by exp (0.196) = 1.216 or 21.6% for the comfort over basic insurance plan), more pain, more multimorbidity, and increasing ADL (i.e., less autonomy). The residual standard error allows to exactly characterize the scatter of the costs: the expected cost has a multiplicative lognormal error with $\mu = 0$ and $\sigma = 0.38$. Our model explains around 48% (multiple *R*-squared) of the differences in the (logged) costs of the patients, while the rest needs to be attributed to individual differences not accounted for by our predictors. As the *F*-statistic shows our model is also highly significant.

We now put some emphasis on the effect of pain. The coefficients and hence the influence on expected costs are relatively small, but formal inference is not possible from the above summary output. However, regression theory provides a formal test (named *hierarchical model comparison* or *F-test*) that infers whether the presence of said predictor and its two associated regression coefficients is beneficial for explaining the response variable. The test is based on comparing the residual sum of squares of models with and without the pain variable. As it turns out, the effect of pain is non-significant with a *p*-value of 0.72. Other options for cleaning the model

from unnecessary predictors that are popular among practitioners include stepwise procedures that are based on the *Akaike Information Criterion* (AIC). These also suggest removing the pain variable, while all others are retained. From a clinical perspective, this model reduction was well received, because pain is notoriously difficult to evaluate, has great individual scatter with low reliability, and requires cooperation of the patient, which is not the case for the other attributes that can be assessed by medical staff.

Hence, we have now identified our final prediction model for daily patient cost. Many alternative methods are conceivable as well. However, in our opinion, the advantage of our approach lies in the clear underlying theoretical concepts, which allow for unbiased prediction with a sound confidence interval for the expected value, as well as a prediction interval for potential future observations. Many modern regression methods do not offer such precise uncertainty metrics with clear, theoretically founded concepts. For our project, the goal was to determine four different *patient cost groups* (PCGs). For reasons rooted in the practical implementation of the system, both clinic and insurance plans were not considered in this grouping process. The grouping was obtained from the predicted cost value for every patient from the remaining regression coefficients for multimorbidity and ADL. From these values, clustering into four equally sized patient cost groups (PCGs) was performed. With these, a second multiple linear regression for the logged cost was fitted, using only the three predictor variables clinic, insurance, and PCG:

```
> fit <- lm(log(cost) ~ clinic + insurance + pcg, data=dat)
```

While the first-stage regression model resulted in the grouping logic to separate all present and future patients into four entities, the regression coefficients for the PCG variable from the second-stage regression model show the expected cost difference among the groups. According to federal law, both the grouping logic and the cost weights will build the mandatory tariff system to be used in all rehabilitation clinics in Switzerland after January 1, 2020. Still, a base rate specific for each hospital and insurance class can be negotiated between the involved parties. The entire process associated to this multiple-regression-driven patient classification system is displayed in Fig. 11.3. It shows that the fourth and most expensive PCG on average has around 2.4× higher cost per day than the first and least expensive one. This difference has been found as being economically very relevant by both the hospital's financial officers and the politicians responsible for the approval of the system. From a more mathematical viewpoint, the classification into four equally sized PCGs seems arbitrary, with better options that offer stronger separation of cost weights nearby. Such alternatives were considered, but finally rejected in the political process.

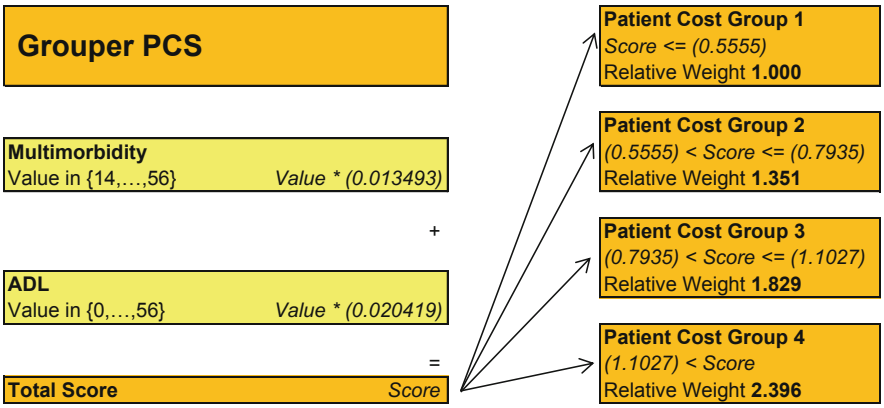

Fig. 11.3 Schematic display of a classification system for inpatient neurological rehabilitation in Switzerland. From a random sample consisting of 407 patients, their score was determined based on a weighting scheme that originated from a multiple regression model of logged cost vs. several other variables. This allowed for classification into four patient cost groups, for which relative weights were then determined in a second-stage multiple regression model. Please note that the values displayed here differ from the original results and are for exposition only

3.2 Logistic Regression for Binary Response

This section focuses on binary response problems and the regression methods that are suited for tackling these. The example that will be used originates from a Swiss business, where contract cancellation (aka *churn* for *change and turn*) is studied. We are using data from 983 customers, of which 144 (14.65%) churned during the observation period. Furthermore, we have information on which of three regions in Switzerland the customer resided in, about his/her sex, age, and duration of the contract, as well as which of seven different products the customer was subscribed to. Again, the original study was more complex with a much bigger sample size that was reduced here to improve clarity of exposition. In this section, we will point out that the big advantage of logistic regression models is not necessarily in the most precise prediction of churn probabilities or outcome, but that they provide a theoretically well-founded, concise solution that is (relatively) easy to interpret and tells us which customers (segments) are at high risk for (e.g.) churn.

We start out with the simpler problem of studying the dependence of the response variable churn on customer age and contract duration first. This has the advantage of having two predictors only, so that data and fitted values can be displayed graphically, see Fig. 11.4. Later on, we will discuss the solution for the full problem with all predictor variables. The poor man's solution to logistic regression consists of fitting a multiple linear regression model with OLS, which we strongly discourage for the reasons to follow:

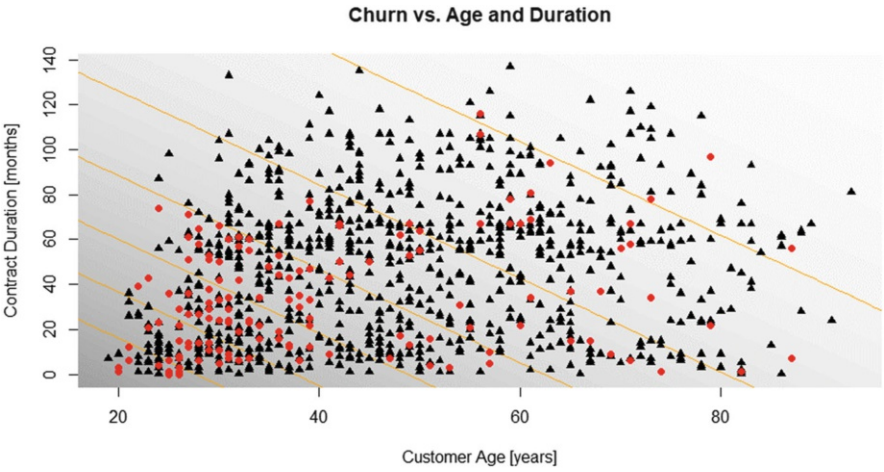

Fig. 11.4 Scatterplot showing customer age in years and contract duration in months for 983 individuals. Red dots correspond to customers who cancelled their contract in the observation period (i.e., *churned*), while the ones marked with black triangles remained. It is easy to discern that younger age and shorter contract duration leads to higher churn risk. This is picked up by our logistic regression model, whose fitted values have been plotted as gray background color. The orange contours correspond to probabilities of 0.05, 0.10, 0.15, 0.20, and 0.25 (from top right to bottom left) for contract cancellation

$$\text{churn} = \beta_0 + \beta_1 \cdot \text{age} + \beta_2 \cdot \text{duration} + E$$

This model will ultimately lead to fitted values (which are interpreted as something like churn "probabilities") that are beyond the interval $[0, 1]$ and thus lack practical relevance. Furthermore, since churn is a 0/1-variable, the error term E has a binary distribution and definitely is non-Gaussian, violating the assumptions to make OLS a good estimator for the regression coefficients. In summary, the above model is miss-specified and should not be used. As pointed out in the introduction, the key lies in modelling the appropriately transformed conditional expectation of the response variable. This is the core idea of generalized linear modelling. If we denote with $p(x) = P[y = 1|X = x]$ the conditional probability (and expectation!) for churn, the logistic regression model is:

$$\log\left(\frac{p(x)}{1 - p(x)}\right) = \beta_0 + \beta_1 \cdot \text{age} + \beta_p \cdot \text{duration}$$

Using MLE, the coefficients can be estimated numerically, by solving iteratively reweighted least squares problems. Such routines are implemented in many statistical software packages. The *R* command and output are as follows:

```
> fit <- glm(churn ~ age + duration, data=dat, family=binomial)
> fit
```

```
Coefficients:
(Intercept)          age     duration
   -0.13701     -0.02560     -0.01227
```

The fit from this model is displayed in Fig. 11.4, see above. We now turn our attention to the interpretation of the regression coefficients. Obviously, the term on the left-hand side of the model equation, the so-called log-odds for churn $\log(p/(1-p))$, is a linear function of the predictors. Thus, if a particular predictor x_j is increased by 1 unit, then the log-odds in favor of $y = 1$ increase by β_j if all other predictors remain unchanged. Hence, when the customer's age increases by one year, the log-odds decrease by -0.02560. This can also be expressed as the odds being multiplied with factor $\exp(-0.02560) = 0.9747$. While these figures are constant and independent of duration, the effect on churn probability is not, as Fig. 11.4 instructively shows.

Naturally, the model provides a churn probability for each of the individuals that were part of fitting. Applying the model to future cases is quick and easy. Usually, all instances with a predicted churn probability of 50% (or any arbitrary other value in case of unequal misclassification costs) or more are classified as churners. Finally, it is also possible to provide standard errors from which confidence intervals for the fitted probabilities can be given. We now turn our attention to the inference and model selection tools that logistic regression offers. We enhance our current two-predictor-model with the additional variables *region*, *sex*, and *product*. After the model was fitted, we can display an output with hierarchical model comparisons (see Fig. 11.5).

Asymptotic theory (cf. McCullagh and Nelder 1989) suggests that hierarchical model comparison can be performed based on differences in the residual deviance of the models. The p-value of 0.187 in Fig. 11.5 suggests that there arises no difference in the churn mechanics from the region the customer resides. Due to p-values (clearly) below 0.05, all other variables have a significant contribution. How these

```
> drop1(fit, test="Chisq")
Single term deletions
Model: churn ~ region + sex + alter + dauer + produkt
         Df  Deviance     AIC      LRT   Pr(>Chi)
<none>         736.61  762.61
region    2    739.96  761.96   3.3508  0.1872346
sex       2    748.98  770.98  12.3770  0.0020529 **
alter     1    757.20  781.20  20.5952  5.674e-06 ***
dauer     1    748.04  772.04  11.4314  0.0007221 ***
produkt   6    760.80  774.80  24.1917  0.0004815 ***
---
    Null deviance: 818.98  on 982  degrees of freedom
Residual deviance: 736.61  on 970  degrees of freedom
```

Fig. 11.5 Model inference output of the logistic regression model that was fitted to the churn data. It reports the statistical significance of the contribution of the predictor variables in explaining the churn probability. The primary interest here lies in the column `Pr(>Chi)` which is the p-value derived from hierarchical model comparisons

variables exactly affect the churn probability can be derived from their coefficients (not pursued here).

Moreover, a global test for the logistic regression model can be carried out by comparing the deviance of the null model with only the intercept versus the one from the model that was fitted here. In our current case, this yields a highly significant result. We conclude this section by stating that logistic regression models are based on a clear theoretical concept, which allows pinning down the influence and significance of each term in a prediction model. Moreover, all predicted values have a sound interpretation as probabilities, can be used for classification, and can be attributed with confidence intervals.

Dealing with Imbalanced Response

From Fig. 11.4, we can conjecture that the highest churn probabilities that are fitted by the logistic regression model using only the two predictors age and duration must be in the region of 0.3. To be exact, the maximum is at 0.34 and hence when using the 0.5 threshold for classification, none of the customers from our database would be predicted as being a churner. First and foremost, this logistic regression result is sensible: even for young customers with short contract duration, the majority do not churn, hence predicting each of them as being non-churners is technically the correct decision. However, the widespread perception that "logistic regression does not work with imbalanced data" that may arise from this fact is not true. As King and Zeng (2001) show in their article, the logistic regression model is perfectly fine in a setup with even very imbalanced classes. However, if the number of events is very low, some issues with a bias in MLE may turn up. They state that throwing away data in order to balance the response does not make sense, but recommend a minimum of at least 10–20 events per degree of freedom used in the logistic regression model. A way out in a situation where that rule of the thumb is clearly violated consists of using *exact logistic regression*, *bias correction approaches*, or *penalized likelihood methods* (see Leitgöb (2013) and references therein).

In our case, we have 144 churners out of 983 customers, so we can safely afford to fit models that have up to about 10–15 regression coefficients, hence the larger model from Fig. 11.5 with 5 predictors and 13 coefficients should be fine. While this larger model leads to six customers that reach probabilities exceeding 0.5 and hence are classified as churners, it is more of a matter of applying the right optic on the result. What we obtain is a customer ranking according to their churn probability, so that we can identify the subjects that are most at risk for cancelling their contract. Whether that probability is below or above 0.5 is usually of lesser interest. We can display the results in a so-called lift chart, which graphically represents the improvement of the logistic regression model compared to a random guess (Tufféry 2011). For constructing the lift chart, we evaluate the proportion of true churners in each percentile of the total sample, ordered after their churn probability, and compute the ratio with the a-priori-churn-rate (i.e., the overall churn rate when dividing the number of churners by the overall number of samples).

In Fig. 11.6, we display the output for our example dataset. As we can see, the lift value for the 1% of customers with the highest predicted churn probabilities amounts

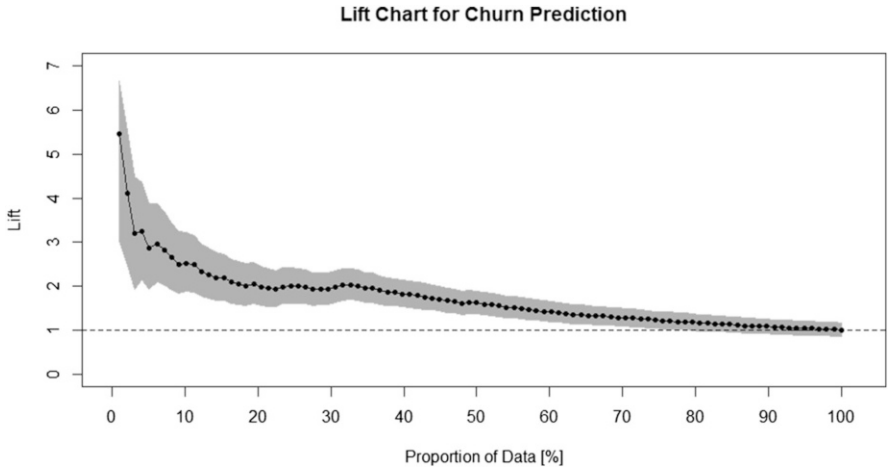

Fig. 11.6 Lift chart for churn prediction with our logistic regression model. It is constructed by ordering the customers according to their predicted churn probability. Then, for each percentile (1%, 2%, 3%, . . ., 100%) of customers, the proportion of churners among them is compared to the a priori churn rate. The corresponding ratio is called the lift and displayed with black dots. Furthermore, 95% confidence intervals for the lift are determined from the binomial distribution and highlighted in gray color

to 5.46. This value is obtained since among the 10 customers with the highest predicted churn probabilities (with probabilities ranging from 0.71 to 0.47), 8 turned out to be true churners. Hence, had we based customer selection for retention activities on our model, we would have identified 80% true churners. Compared to the a priori rate of 14.65%, this seems like a good success, which can be expressed as a lift of $80/14.65 = 5.46$. With the same approach as for the lift value for the top 1% of customers, the values are then computed for the top 2%, 3%, and so on.

We can also determine 95% confidence intervals for the lift values by using properties of the binomial distribution. In particular, for the top 1% of customers, (where 8 out of 10 customers churned), we obtain for a 95% confidence interval values ranging from 0.444 to 0.975 for the churn probability. These values can then be transformed into lift values of $[3.03, 6.65]$ by dividing with the a priori rate. In summary, the lift chart together with the confidence intervals confirms that our logistic regression model is clearly beneficial for understanding the churn mechanics and predicting which customers end the contract.

3.3 Regression Models for Time-to-Event Data Considering Censoring

The goal of this section is to give some insight into modelling time-to-event data considering censoring. The most common type of censoring is right censoring which

occurs when a subject leaves the study before an event occurs, or the study ends before the event has occurred. In many situations, the reported duration time is discretized. In our example we know how many years it took until a water pipe (i.e., a piece of pipe between two branching points) broke in a water supply network. Data are available for pipes that are actively used, but may have gone through one or more repairs. If a pipe had to be replaced, all its corresponding information has been removed from the database. This kind of data "cleaning" will introduce bias into our final results. How severe the bias will be can unfortunately not be assessed from the data only. In what follows, we disregard this issue, but it has to be considered when deploying the results of the modelling. In this section, we study the failure time of 2402 pipes up to the first breakdown. We consider two types of pipes, each made of different materials and having different pipe diameters. Additionally, we use information about the length of the pipe. Many other potentially important attributes of the pipes were unknown.

Because of the discretized durations [in years], we do, as explained above, estimate the hazard probability of failing in year t if the pipe is at risk to break down at the beginning of year t. If we observe a failure of pipe i in year t, we set the new variable failure to the value of 1. Thus for pipe i, we have an observation vector with elements age t (is equal to failure time), length l, type m, and failure $= 1$. Since the pipe has been at risk a year before as well but did not fail, we construct another observation vector for pipe i with elements age $(t - 1)$, length l, type m, and failure $= 0$. This construction of additional "observations" is repeated until age is equal to 1. Hence, a failure of a pipe in year t generates a set of t observation vectors. These sets of observation vectors, which are generated from the 2402 pipes, are merged to a dataset of 99,039 rows, which can be analyzed by binary regression models as presented in formula (11.3).

The response variable is failure and its expectation is the probability $p(t = k|x)$. A very simple model, which could describe the available data adequately, is a *Weibull regression model* with different hazard rates for the two different pipe types. Additionally, the hazard rate should be proportional to the length of the pipe, that is, the longer the pipe the higher the hazard rate that the pipe will break. To express these ideas in terms of the model in formula (11.3), the predictor variables must be set up suitably. As explained in Eq. (11.3), γ_k specifies the distribution function. If a Weibull distribution is demanded, γ_k must be approximately proportional to the logarithm of age evaluated at the discrete time grid k. In the following, it is called log$K$. To express the proportionality of the hazard rate to the length of the pipe in model (11.3), the log-transformed length (i.e., *lLength*) must be added as predictor variable but with a fixed parameter being equal to 1. Finally, to include different hazard rates for the two pipe types, the log-transformed age of pipes from the second type (called logKG) must be included as a predictor variable. To summarize, the model can be specified as follows:

$$\log(-\log(1 - p(t = k|x))) = \beta_0 + \beta_1 \cdot \log K + \beta_2 \cdot \log KG + lLength \qquad (11.4)$$

Hence, $\beta_0 + \beta_1 \cdot \log K$ yields the baseline hazard probabilities of pipes consisting of type 1 and $\beta_0 + \beta_1 \cdot \log K + \beta_2 \cdot \log KG$ yields baseline hazard probabilities of pipes consisting of type 2 with respect to one unit of pipe length. To explore whether this simple model fits the data adequately, we set up a more general model which the simple model is part of. Thus, we start with

$$\log(-\log(1 - p(t = k|x))) = s_1(\log K) + s_2(lLength) + s_3(\log KG) + s_4(lLengthG) \tag{11.5}$$

where s_1, s_2, s_3, and s_4 are smooth functions which must be estimated. This is a *generalized additive model* (GAM) and can be fitted by adequate algorithms (cf., e.g., Hastie and Tibshirani 1990 or Wood 2006). The smooth functions add flexibility; if s_1, s_2, s_3 are straight lines and s_4 is a horizontal line, we obtain the simpler model described by formula (11.4). The smooth functions are estimated by the local regression approach called *lowess* or *loess* (cf., e.g., Hastie and Tibshirani 1990). Based on a GAM fit of our data we obtain the following slightly shorted summary output (Fig. 11.7).

```
Call: gam(formula = Failure ~ lo(logK) + lo(logKG) + lo(lLength) +
          (lLengthG), family = binomial(link=cloglog), data=DCGI)
(Dispersion Parameter for binomial family taken to be 1)
Null Deviance: 2656.597 on 99038 degrees of freedom
Residual Deviance: 2146.296 on 99022.41 degrees of freedom
AIC: 2179.474
Number of Local Scoring Iterations: 17

Anova for Parametric Effects
                Df Sum Sq Mean Sq F value          Pr(>F)
lo(logK)         1     41  41.500  69.584  < 2.2e-16 ***
lo(logKG)        1     13  12.642  21.196  4.151e-06 ***
lo(lLength)      1    216 215.846 361.916  < 2.2e-16 ***
lo(lLengthG)     1     85  84.673 141.973  < 2.2e-16 ***
Residuals    99022  59057   0.596
---
Anova for Nonparametric Effects
Npar         Df Npar   Chisq     P(Chi)
(Intercept)
lo(logK)         3.0  5.0359  0.166635
lo(logKG)        3.2 15.1415  0.002095 **
lo(lLength)      2.5  5.9731  0.079497 .
lo(lLengthG)     2.9 14.3497  0.002202 **
```

Fig. 11.7 Model summary output of the generalized additive model that was fitted to the discretized time-to-event data. It reports the statistical significance of the contribution of the predictor variables by separating the contributions of the parametric (linear) and non-parametric effects. The primary interest here lies in the column `Pr(>F)` or `Pr(>Chi)`, which is the *p*-value derived from hierarchical model comparisons

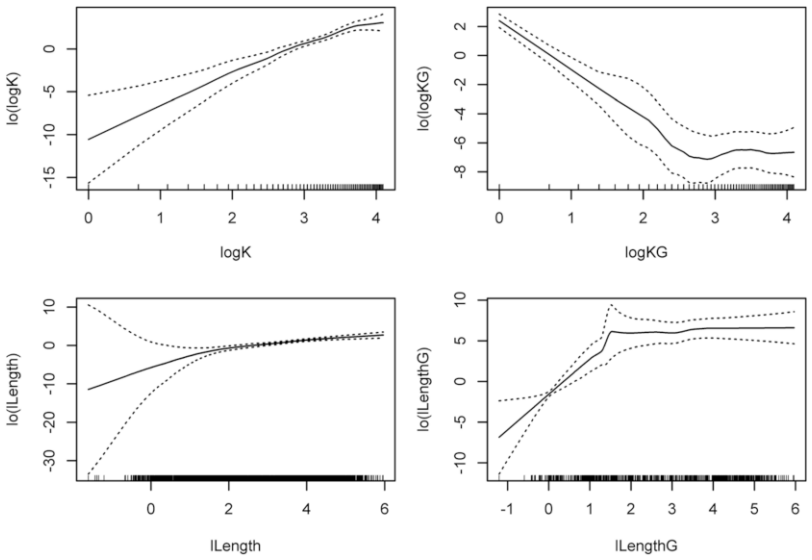

Fig. 11.8 Partial residual plots showing the pure effect of each predictor variable on the response variable "failure." The effect is visualized by the smooth full line. The dotted lines indicate pointwise 95% confidence intervals. The vertical ticks displayed above the *x*-axis are showing the occurrence of each predictor variable. For the predictors log*K* and *lLength*, the nonlinear extension is not significant, which can be derived from the fact that a straight line fits within the confidence bounds

The partial residual plots in Fig. 11.8 illustrate that the two variables log*KG* and *lLengthG* do not go linearly into the predictor function of the complementary log-log transformed hazard probability. As a rule of thumb, we can identify a simple linear relationship if a straight line fits between the two dashed lines which indicate pointwise 95% confidence intervals. Consulting these partial residual plots is a form of multiple testing where some hypotheses are accepted and some not. To be more formal, we run a hypothesis test comparing the more complex model expressed by formula (11.5) with the simple model given in (11.4). The simple model from formula (11.4) is fitted by a GLM algorithm. To include the variable *lLength* into the model but without estimating a corresponding coefficient, we apply the *R* modelling option `offset()`, see Fig. 11.9.

We can now compare the two models based on the difference of their residual deviances. The difference is 17.904 (=2164.2–2146.296) and the difference of the degrees of freedom is −13.59 (=99,036–99,022.41). As before, asymptotic theory tells us that the difference of the residual deviance is asymptotically following a chi-squared distribution with 13.59 degrees of freedom. Hence, the corresponding *p*-value is 0.19. This tells us that there is no statistical evidence that the simple model does not describe the data as well as the complex one. Hence, we can conclude that a Weibull Regression model with different hazard rates for the two different pipe types and with hazard rates that are directly proportional to the length of the pipe does

```
Call:
glm(formula = Failure ~ offset(lLength) + logK + logKG,
    family = binomial(link = cloglog), data = DCGI)
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -18.83934    0.90503 -20.816  < 2e-16 ***
logK          2.80780    0.26317  10.669  < 2e-16 ***
logKG        -0.30121    0.04736  -6.359 2.03e-10 ***
---
(Dispersion parameter for binomial family taken to be 1)
    Null deviance: 2390.4  on 99038  degrees of freedom
Residual deviance: 2164.2  on 99036  degrees of freedom
AIC: 2170.2
Number of Fisher Scoring iterations: 12
```

Fig. 11.9 Model summary output of the generalized linear model that was fitted to the discretized time-to-event data. It reports the estimated coefficients, that is, the contribution of the predictor variables. This is along with their standard errors (i.e., precision of estimation) and information about the statistical significance of these predictors (columns entitled `z value` and `Pr(>|z|)`. The lower four lines report information about the model as a whole

describe this data set adequately. Figure 11.8 may, however, indicate in which directions potential improvements of the model could be made. For example, the variable log*KG* may be transformed such that it is a straight line up to 2.5 and then horizontal.

4 Conclusions

Three important aspects underlie all our endeavors to develop a statistical model based on available data: the goal of the modelling, the subject matter knowledge, and the data-generating process. For the first aspect, we may distinguish technically between prediction and description. In prediction, we may not require an interpretable relationship, that is, model, between input and output, but rather an accurately predicted output. Definitely, the prediction must be based on predictors that are known at the time of applying the prediction model.

A descriptive model, however, is aimed at capturing the data structure in a statistically sound way (i.e., not contradicting the model assumptions), and in what is useful (i.e., not contradicting the subject matter background of the data) and parsimonious. We may also have the option to choose between different equivalent predictor variables, but the practitioners need to keep the effort for collecting these data into account and may want to exchange one variable for another one. Another challenge in developing a statistical model is to find suitable transformations of the predictor variables. We find that generalized additive models (GAM) are very helpful to explore useful transformations, which of course must agree with the subject matter knowledge. Finally, the aspect of the data-generating mechanism is

crucial when assessing the range of use of the resulting statistical model. In data mining, for example, data are often not collected for the actual question of interest in modelling but for other intentions.

Statistical modelling is not just about knowing individual statistical methods (linear regression, generalized linear or additive models, proportional hazard models, etc.) but rather about choosing multiple methods, about how to apply and combine them, and finding a useful balance between inferential and graphical methods. To learn this handcraft there is no other choice than to practice and to collect the necessary experience, ideally supported by a senior expert. While there is a wealth of literature on technical aspects, only relatively few books discuss the virtuosity of statistical modelling. Three of them are Diggle and Chetwynd (2011), Harrell (2015), and Kuhn and Johnson (2013). The first two feature regression modelling strategies with applications to linear models, logistic regression, survival analysis, etc. Diggle and Chetwynd (2011) should be accessible by any scientist, whereas Harrell (2015) goes into much deeper detail. Kuhn and Johnson (2013) focus on strategies for predictive modelling including methods like classification and regression trees, random forest, boosting, neural networks, support vector machines, etc.

We finish our contribution by emphasizing that from our practical experience as data scientists, using the relatively simple, traditional statistical modelling methods is often a superior choice or at least a serious contender when it comes to providing added value and insight to our customers and collaborators. Modern black box approaches certainly have their merits and place in data science, but statistical modelling is of greater use if either insight into the model mechanics or the option of human intervention in the modelling process is desired. We summarize our contribution into the following lessons learned:

- In contrast to pure prediction tasks, descriptive analysis requires explicit statistical models. This includes concrete knowledge of the model formulation, variable transformations, and the error structure.
- Statistical models are verifiable: It is possible to explore if the fit is in line with the model requirements and the subject matter knowledge. In case of important discrepancies, action needs to be taken!
- To obtain sound results and reliable interpretations, we need to consider the data-generating mechanism within the model developing process and during model assessment.

Acknowledgments The authors thank the editors for their constructive comments, which have led to significant improvements of this article.

References

Allison, P. D. (2010). *Survival analysis using SAS: A practical guide* (2nd ed.). Cary, NC: SAS Institute.

Cox, D. R. (1958). The regression analysis of binary sequences (with discussion). *Journal of the Royal Statistical Society B, 20*, 215–242.

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society, Series B., 34*(2), 187–220.

Diggle, P. J., & Chetwynd, A. G. (2011). *Statistics and scientific method: An introduction for students and researcher*. New York: Oxford University Press.

Fahrmeir, L., & Tutz, G. (2001). *Multivariate statistical modelling based on generalized linear models* (Springer series in statistics). New York: Springer.

Harrell, F. (2015). *Regression modeling strategies: With applications to linear models, logistic and ordinal regression, and survival analysis* (Springer series in statistics). Heidelberg: Springer.

Hastie, T., & Tibshirani, R. (1990). *Generalized additive models*. London: Chapman and Hall.

Kalbfleisch, J. D., & Prentice, R. L. (2002). *The statistical analysis of failure time data* (Wiley series in probability and statistics) (2nd ed.). Hoboken, NJ: Wiley.

King, G., & Zeng, L. (2001). Logistic regression in rare events data. *Political Analysis, 9*, 137–163.

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. New York: Springer.

Leitgöb, H. (2013). *The problem of modelling rare events in ML-based logistic regression – Assessing potential remedies via MC simulations*. Conference Paper at European Survey Research Association, Ljubliana.

McCullagh, P., & Nelder, J. (1989). *Generalized linear models* (Monographs on statistics & applied probability) (2nd ed.). Boca Raton, FL: Chapman & Hall/CRC.

Montgomery, D., Peck, E., & Vining, G. (2006). *Introduction to linear regression analysis*. New York: Wiley Interscience.

Plackett, R. L. (1972). The discovery of the method of least squares. *Biometrika, 59*(2), 239–251.

Sen, A., & Srivastava, M. (1990). *Regression analysis: Theory, methods, and applications*. New York: Springer.

Stigler, S. M. (1981). Gauss and the invention of least squares. *Annals of Statistics, 9*(3), 465–474.

Tufféry, S. (2011). *Data mining and statistics for decision making*. Chichester: Wiley.

Wood, S. (2006). *Generalized additive models: An introduction with R* (Texts in statistical science). Boca Raton, FL: Chapman & Hall/CRC.

Chapter 12
Beyond ImageNet: Deep Learning in Industrial Practice

Thilo Stadelmann, Vasily Tolkachev, Beate Sick, Jan Stampfli, and Oliver Dürr

Abstract Deep learning (DL) methods have gained considerable attention since 2014. In this chapter we briefly review the state of the art in DL and then give several examples of applications from diverse areas of application. We will focus on convolutional neural networks (CNNs), which have since the seminal work of Krizhevsky et al. (ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems 25, pp. 1097–1105, 2012) revolutionized image classification and even started surpassing human performance on some benchmark data sets (Ciresan et al., Multi-column deep neural network for traffic sign classification, 2012a; He et al., Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *CoRR*, Vol. 1502.01852, 2015a). While deep neural networks have become popular primarily for image classification tasks, they can also be successfully applied to other areas and problems with some local structure in the data. We will first present a classical application of CNNs on image-like data, in particular, phenotype classification of cells based on their morphology, and then extend the task to clustering voices based on their spectrograms. Next, we will describe DL applications to semantic segmentation of newspaper pages into their corresponding articles based on clues in the pixels, and outlier detection in a predictive maintenance setting. We conclude by giving advice on how to work with DL having limited resources (e.g., training data).

T. Stadelmann (✉) · B. Sick · J. Stampfli
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stdm@zhaw.ch

V. Tolkachev
University of Bern, Bern, Switzerland

O. Dürr
HTWG Konstanz - University of Applied Sciences, Konstanz, Germany

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_12

1 Introduction to Deep Learning

Deep neural networks have been greatly influencing the world of pattern recognition for several decades (Schmidhuber 2014). The disruptive nature of the approach became obvious to a wider audience since Krizhevsky et al. (2012)'s exploit on the ImageNet task. Since then the corresponding gain in perceptual performance has often been such that error rates could be halved or even improved by an order of magnitude with respect to the previous state of the art on open benchmark datasets (LeCun et al. 2015). In this chapter, we show how deep learning (DL) methods can be applied not only to classical computer visions tasks from research but also to a wide variety of tasks in industry beyond classification.

While it is easy for humans to recognize someone's speech or classify objects in an image, problems like these had previously posed a serious challenge for computers for a long time. In the traditional pattern recognition paradigm, researchers tended to manually design informative features, on which classification algorithms were applied. In computer vision, these were, among others, Haar features or Gabor filters (Szeliski 2010). In speech recognition, one used, for example, Mel frequency cepstrum coefficients (Zheng et al. 2001), while in Natural Language Processing (NLP), there were *n*-gram features or mutual information between the words (Bouma 2009). These features were burdensome to engineer manually and it was unclear which ones were the most informative for the task at hand.

DL revolutionized the field by offering end-to-end learning, starting at almost raw data input without the need for kernel or feature engineering and allowing a hierarchy of neural network layers to learn the necessary features on its own. In the following paragraphs we provide a brief overview of these developments.

1.1 Fully Connected Neural Networks for Classification

The simplest architecture, from which the development in the field of neural networks started, is a fully connected feed-forward neural network (Rosenblatt 1957). It can be considered as a directed acyclic graph where the information flows from left to right (see Fig. 12.1). A neuron is made up of a circle (summing up the inputs), followed by a square (depicting a nonlinear activation function that serves as a threshold). Inspired by a biological brain, each neuron $z_j^l(x)$ in a layer of neurons (vector[1] $z^l(x)$) receives an input from all the neurons from the previous layer z^{l-1} with a weight matrix W. The weighted sum of inputs for the neuron is then passed through a nonlinear activation function f inside the neuron that acts as a trainable threshold: if f receives a high value, the neuron is activated and passes the transformed signal to the neurons in the next layer on the right. In general, the output

[1]Vector arrows are usually not drawn in the DL literature.

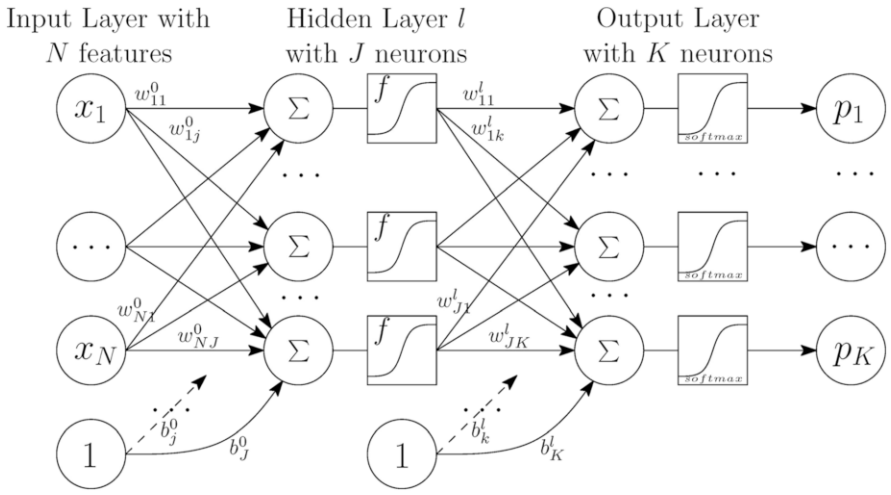

Fig. 12.1 A feed-forward neural network with features $x_1, \ldots, x_N$ and label probabilities $p_1, \ldots, p_K$

of all neurons of a layer l can be recursively described with the weight matrices $W^{l-1}, W^{l-2}, \ldots$ and bias vectors $b^{l-1}, b^{l-2}, \ldots$ as:

$$z^l(x) = f\left(w_0{}^{l-1} + W^{l-1} z^{l-1}(x)\right) = f\left(w_0{}^{l-1} + W^{l-1} f\left(w_0{}^{l-2} + W^{l-2} z^{l-2}(x)\right)\right) = \ldots$$

The network can possess many hidden, interconnected layers. For classification into classes $1, 2, \ldots, K$, its last (output) layer will have as many nodes as there are classes to distinguish in the data (K in this case). To obtain probabilities $P(y_k|x)$ for each class k, the raw aggregated inputs (scores) must be standardized by their sum over all classes to produce values between 0 and 1, which is usually done with the softmax function (Bishop 2006, p. 115):

$$P(y_k|x) = \frac{\exp\left(-\sum_{j=1}^{J} w_{jk}{}^{l} z_i{}^{l}(x)\right)}{\sum_k \exp\left(-\sum_{j=1}^{J} w_{jk}{}^{l} z_i{}^{l}(x)\right)}$$

where $w_{jk}{}^{l}$ are the learned weights of layer l which are elements of the matrix W^l. Historically, sigmoid $f(x) = (1 + e^{-x})^{-1}$ and hyperbolic tangent $f(x) = \tan h(x)$ were used as activation functions; now it is recommended to use a rectified linear unit (ReLU) $f(x) = \max(0, x)$ which significantly speeds up training because of improved gradient flow (Krizhevsky et al. 2012).

To train the neural network (i.e., find the optimal weights), a loss (discrepancy between the true and predicted classes) is computed once the signal is propagated to the last layer and the class probabilities are calculated. Then the weights are adjusted to minimize the loss function, which is usually done with a maximum likelihood approach: The weights are optimized by stochastic gradient descent (SGD) (Goodfellow et al. 2016) and the required gradients are efficiently calculated making

use of the chain rule. For example, to calculate the gradient of the loss at the layer $l-2$ one can use the gradient at the layer $l-1$:

$$\frac{\partial \text{Loss}}{\partial z^{l-2}} = \frac{\partial \text{Loss}}{\partial f} \frac{\partial f}{\partial z^{l-1}} \frac{\partial z^{l-1}}{\partial z^{l-2}}$$

The gradient thus propagates "back" from the loss to previous layers. Therefore, this procedure is also called backpropagation in the context of neural networks (Rumelhart et al. 1988); a gentle introduction is provided by Nielsen (2015). Training usually runs on GPUs for computational reasons (speed-up of an order of magnitude as compared to CPUs), and the training data is split into so-called mini-batches, which fit into the GPU memory and on which SGD is run. Nowadays, more advanced variants of SGD like ADAM (Kingma and Ba 2014) and ADADELTA (Zeiler 2012) are usually employed instead of the standard SGD and should be preferred.

1.2 Convolutional Neural Networks (CNNs)

While fully connected networks possess significant flexibility, they have many parameters and tend to significantly overfit the data while not capturing any local structures such as the 2D correlations of pixels in images. CNNs were introduced in order to resolve these issues.

The first neural networks with convolutional filters date back to the work of Fukushima (1980). They were made trainable end-to-end via backpropagation by LeCun et al. (1998a), yet their applicability was limited at that time due to scarce computing capacities and the shortage of large labeled datasets. The revival came in 2012 with the works of Ciresan et al. (2012a) and Krizhevsky et al. (2012), who independently presented significantly deeper nets trained on modern graphics cards (GPUs). This GPU training enabled increased depth by exploiting the cheap hardware originally developed for 3D games, using its massively parallel matrix computation capabilities. It was thus possible to solve the problems of the traditional approach and completely outperform it in numerous pattern recognition challenges. Currently, deep CNNs have error rates as low as humans [or sometimes even better (Nielsen 2017)] in many tasks, including image classification (He et al. 2015a), geolocation (Weyand et al. 2016), speech recognition (Xiong et al. 2016), lip reading (Chung et al. 2016), as well as the games of GO (Silver et al. 2016) and poker (Moravcik et al. 2017).

The intuition behind CNNs goes back to the physiological experiments of Hubel and Wiesel (1959) on the response of visual neurons in a cat's brain to various oriented stimuli. The main idea of CNNs is to design a neural network that can easily exploit local structure in its input in hierarchically organized layers to extract subsequently more abstract features: convolutional kernels (resembling the filters

of classical image processing[2]) are slid over the complete input and the dot product of the input with the kernel at each location is computed. Thereby, each possible kernel location shares the same weights, which massively saves parameters in this layer, which in turn can be "invested" back into additional layers. Convolutional layers usually alternate with some sort of sub sampling (originally: pooling) layers and thus allow the CNN to abstract local structure to global insights.

The convolutional operation makes sense because it processes information locally and converts it to a feature map (which is the output of a specific filter, evaluated at every pixel of the current layer, resembling the filtered image) that indicates the presence or absence of the very feature the convolutional filter describes using its learned coefficients. A learned feature could be, for example, an edge, a simple shape, or a combination of these in the later layers. The feature map can then be compressed by means of a down-sampling operation (e.g., max pooling[3]) to create a global big picture of the input contents out of the local features (see Fig. 12.2). In CNNs, several blocks of convolution and down-sampling are thus stacked in the network with various input sizes to achieve sufficient generality and to capture enough detail, so that every block is responsible for some image property. As a result, a hierarchical representation of object properties is built by the convolutional layers. Finally, a fully connected output layer produces class probabilities.

To cope with varying input image sizes and to produce richer outputs than just class labels (e.g., full images again), the fully convolutional network (FCN) has been proposed (Long et al. 2014), which implements all layers (also down-sampling and fully connected ones) using convolutions only (see Sect. 4).

[2]In digital image processing, to apply a filter (or kernel) to a specific region of an image, centered around a specific pixel, means to take the weighted sum of pixels in the center pixel's neighborhood. The size of the neighborhood is determined by the filter size (e.g., 3×3 pixels), whereas the weights are determined by the filter designer. Numerous classical filters for all kinds of image processing tasks are known. For example, to smoothen an image, one applies a filter with each of the N weights equaling $1/N$, so the filter response is an average over the filter's spatial area. The filters "filter 1" and "filter 2" in Fig. 12.2. show vertical and horizontal edge detectors, respectively (when white pixels stand for a weight of -1 and blue pixels for a weight of 1, or vice versa). In CNNs, the filter weights are learned, while the size and number of filters are chosen hyperparameters. This means that each convolutional layer in a CNN can learn any classical image transformation (see https://en.wikipedia.org/wiki/Digital_image_processing), one per filter (you see the number of filters by counting the number of feature maps in the next layer, cp. Fig. 12.2).

[3]Max-pooling describes the process of moving a kernel of, for example, 2×2 pixels over an image-like representation (a layer in the neural network); for each location, only the maximum pixel value is carried over to the next layer, thus resulting in down-sampling the original 2×2 pixels (to keep the example from above) information to just 1×1. The size of the kernel as well as its step size (stride) typically are hyperparameters of the neural network architecture. However, in some modern designs, the architecture offers down-sampling at various degrees after each convolutional step, with the possibility to learn during training for the task at hand which of several alternative paths through the network should be followed at which layer. Thus, it offers to "learn" the degree of down-sampling to a certain extent (Szegedy et al. 2014; He et al. 2015b).

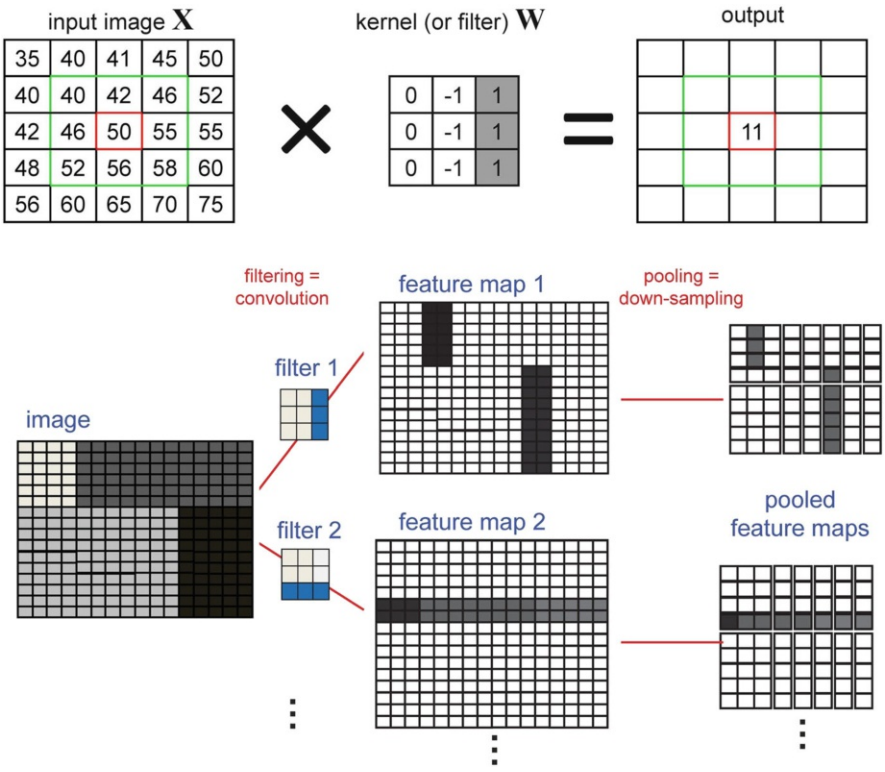

Fig. 12.2 Convolutional filters (top) slide over the images, creating feature maps, which are then down-sampled to aggregate the most important information about the image structure

Overall, a CNN still contains a lot more free/trainable parameters (usually in the order of hundreds of millions) than observations used for training, so that with a "wrong" training procedure it is easy to overfit the data. There are a number of possible solutions which are now application standards. First, traditional CNNs are not intrinsically invariant to transformations like object rotations, flips, lighting, etc. In order to enrich the training data, it is common to do image augmentation prior to training that reflects the input's nature (i.e., apply transformations like rotation, translation, and random scaling to the data and add natural "noise," using the transformed images for training as well). Second, a regularization technique called dropout (Srivastava et al. 2014) was introduced to significantly reduce overfitting, which consists of randomly deactivating each neuron in a layer usually with a probability of 0.5 at training.

Wrong weight initialization in a network can pose a serious problem as well. With an inappropriate initialization, some of the neurons may soon come into an over- or under-saturated regime, and, depending on the activation function, the gradients will be close to zero. This in turn means that there would be almost no update of the weights during backpropagation, and parts of the network will die out. To avoid this

and other problems, including overfitting, the batch normalization technique (batchnorm) has been proposed (Ioffe and Szegedy 2015). It consists of standardizing a mini-batch *at each layer* of the network with its mean and standard deviation after each training iteration in order to keep a stable input distribution to each neuron, thus facilitating gradient flow. Moreover, batchnorm also allows to learn the shift and scale normalization parameters to undo the standardization when needed. Batchnorm alleviates the dependence on initialization, allows faster learning rates and training times, and acts as a regularizer due to a more even sampling in the parameter space.

To summarize, the use of GPUs in conjunction with the abundance of large (annotated) datasets made CNNs applicable to a wide range of problems. This was only possible in combination with the algorithmic improvements outlined earlier (i.e., ReLU activation, batchnorm initialization of the weights, ADAM or ADADELTA optimizer, dropout regularization, and data augmentation)—compare (Szegedy et al. 2014). All these improvements are now implemented in modern software frameworks used for production-ready deep learning, such as TensorFlow[4] or Torch,[5] or included in high-level libraries on top of these frameworks like Keras[6] or TFLearn.[7] These frameworks also offer a collection of pre-trained networks available for many image recognition tasks. They can be adapted to similar tasks using transfer learning (Pan and Yang 2010), eliminating the need for time-consuming training from scratch, which could still take 1–2 weeks for any real-world task even on modern hardware.

1.3 Non-obvious Use Cases

In the following sections, we describe various applications of deep neural networks. We focus on non-classical tasks, given that the performance of CNNs on image classification tasks is well known. Table 12.1 gives an overview of the selected tasks with a focus on the properties of every use case. Moreover, the table describes in which section of this chapter the use case is described in more detail. Table 12.2 summarizes the special challenge of each task and the main deviation from the classical image classification approach.

We start with an application of CNNs to fluorescence microscopy images, an application which up to now requires much tedious and time-consuming work from highly trained experts in biology and image analysis. We then continue to speaker clustering, where pre-trained CNNs are used to extract learned feature vectors per speech utterance for subsequent hierarchical clustering. This is followed by an

[4] https://www.tensorflow.org/

[5] http://torch.ch/

[6] https://keras.io/

[7] http://tflearn.org/

Table 12.1 Overview of task properties for each of the following use cases

Sec.	Application	Type of final task	Training data	Results
2	Cell phenotype classification for drug discovery	Classification (supervised)	Ca. 40 k images having 5 color channels	Outperforms state of the art (better than linear discriminant analysis (LDA) and support vector machine (SVM)
3	Media segmentation according to voice	Clustering (unsupervised)	Spectrograms of raw audio (ca. 25 s on average for each of 100 speakers)	Outperforms state of the art (better than hand-coded features and statistical models)
4	Newspaper segmentation into articles	Semantic segmentation (supervised)	Ca. 430 scans of newspaper pages (+ additional input from OCR) + 5 k partially labeled pages (+OCR)	Outperforms state of the art (better than classification CNN)
5	Predictive maintenance of rotating machinery	Anomaly/outlier detection (unsupervised)	Spectrograms of ca. 1 k raw vibration signal measurements	On par with state of the art (SVM, Principal Component Analysis (PCA), statistical models)

Table 12.2 What makes the following tasks special and how can this be handled using deep learning?

Sec.	Non-obvious because?	Solved by?
2	Introductory case, but 5 color channels instead of the usual 1–3	Straightforward extension of standard model using data augmentation on training data
3	Audio instead of image as input; final goal is a clustering	Input is converted to a spectrogram to be treated as an image; output is a learned representation to be clustered offline by another method
4	Output is a cutting mask (outline of the text columns and images that make up an article on a page)	Output is an image of the same size as the input: pixels of same color indicate areas belonging to the same article
5	Training data has only one class, model shall indicate if new data deviates from it (instead of segregating it from a well-specified second class)	Using an autoencoder architecture for the network and interpreting the reconstruction error as the degree of novelty in the test signal

application in which a fully convolutional network segments the pixels of a scanned newspaper page into sets of semantically belonging articles. Finally, the use of DL for predictive maintenance is illustrated before we conclude by giving an outlook on how to generally apply deep nets in contexts with usually very limited training data and computational resources.

2 Learning to Classify: Single Cell Phenotype Classification Using CNNs

High content screening (HCS) is an essential part of the drug discovery pipeline used in the pharmaceutical industry. Screening involves the application of many thousands of drug candidates (compounds) to living cells with the aim to investigate the cell response. This response manifests itself in the change of the phenotype. Examples for those phenotypes are: dead cells, dying cells (apoptosis), dividing cells (mitosis), and cells expressing certain proteins.

In simple settings, an applicable approach for classification is to extract predefined features for each cell (e.g., diameter, area, circumference of the nucleus or the cell, the intensity of different fluorescent stains in the nucleus or the cell, or other organelles) and use them as an input for classification (see Fig. 12.3, upper right panel). Such pre-defined features can be extracted by a specialized software such as CellProfiler.[8] However, more challenging cases require a tailored image analysis procedure to extract appropriate features, which needs to be done from scratch for each experiment, requiring both in-depth knowledge of cell biology and advanced knowledge of image processing.

Deep learning, on the other hand, does not rely on those predefined or handcrafted features, and employs only labeled data, a task which is feasible for a biologist without a profound competence in image processing. Hence, deep learning has the potential to radically change the workflow in HCS. The envisioned CNN approach allows to learn the features and the classification model in one training procedure (see Fig. 12.3, lower right panel).

2.1 Baseline Approach

We use part of the image set BBBC022v1 (Gustafsdottir et al. 2013) (the "Cell Painting" assay), available from the Broad Bioimage Benchmark Collection (Ljosa et al. 2009). We analyze the images of human cells treated with 75 compounds—each compound resulting in one of three phenotypes (named A, B, C). In addition, we add the phenotype D of the cell without treatment (mock class). In total, we have the following number of detected cells, which were imaged in 21 different wells on 18 different plates: 40,783 (mock class), 1988 (cluster A), 9765 (cluster B), and 414 (cluster C).

Approximately 20% of the data is put aside for testing, containing the following number of examples per class: 8217 (mock class), 403 (cluster A), 1888 (cluster B), and 82 (cluster C) cells from 10 different wells on 5 different plates. The remaining 80% of the data is used to train and tune different classifiers comprising a CNN based

[8] http://cellprofiler.org/

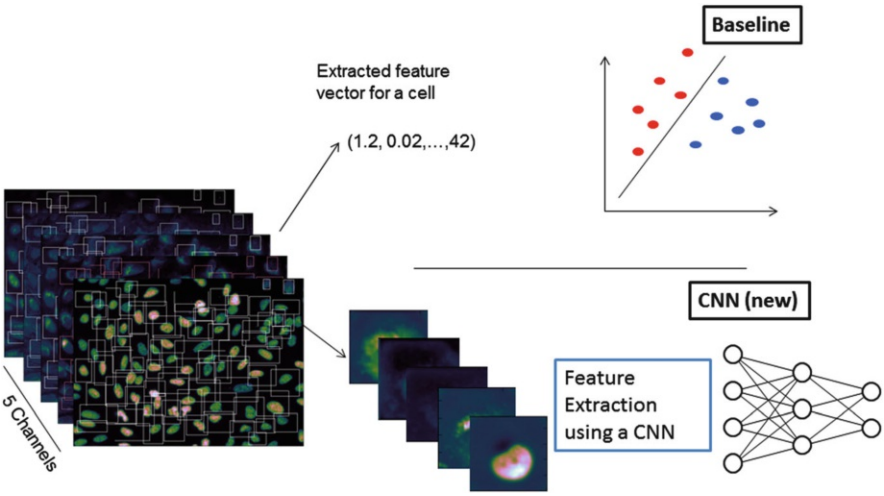

Fig. 12.3 Overview of the used analysis scheme. The baseline approach (upper part) needs handcrafted features, which are extracted using CellProfiler prior to classification using, for example, the SVM. In the CNN approach (lower part), the features are learned automatically

on the raw image data as well as the three baseline approaches often used in HCS (Dürr et al. 2007): Fisher linear discriminant analysis (LDA), Random Forest (RF), and support vector machine (SVM), based on CellProfiler features. Before the input into CNNs, 5 images of size 72×72 are cropped for each cell from the original images. The bounding box is constructed to be quadratic so that the entire cell is within the box.

For the baseline workflows, each of the extracted features is normalized to have zero mean by a z-transformation. We then use the following implementations and parameterizations for classification: an SVM with a linear kernel (the penalty parameter C of the SVM is optimized using a 10-fold cross-validation on the training set); an RF with the default value of 500 trees; and LDA. All algorithms have been implemented in Python using the scikit-learn library.[9]

2.2 CNN Analysis

As the only preprocessing step for CNN, we normalize the values per pixel. The architecture of the CNN is inspired by the second-best entry of the 2014 ImageNet competition (Simonyan and Zisserman 2014). All convolutional filters (C) have the size of (3,3) and a stride of 1 pixel and use ReLU activations; no padding is applied at the boundaries. Two convolutional layers are followed by a (2,2) max-pooling

[9] http://scikit-learn.org/stable/

layer, forming a stack. Our network consists of 3 such stacks, which have 32, 64, and 128 kernels each. These stacks are followed by 3 fully connected layers with 200, 200, and 50 nodes, respectively, and a final softmax layer for the 4 classes. The network has about 1.2 million learnable weights. For learning the weights of the network, we split the data available for training into two parts: one part is used for fitting the weights (training set), the other 20% are used for validation (validation set). Note that the test set described above is only used for the evaluation of the trained CNN.

To prevent overfitting, we use dropout for the hidden layers, setting a fraction of $p = 0.3$ of all nodes randomly to zero in the training phase. We further used data augmentation to artificially enlarge the training set by applying the following random transformations on each image after each epoch (one epoch comprises a full pass through the training set): a random rotation uniformly chosen in the range of 0° to 360°; a random translation up to 5 pixels in each direction (uniformly chosen); and a scaling with a scaling factor uniformly chosen in the range 0.9 to 1.1.

The network is implemented using the nolearn extension of the Lasagne python library.[10] All runs have been done on an off-the-shelf PC with a NVIDIA GeForce GPU.

2.3 Results and Discussion

The training of the CNN took on average 135 s per epoch when using augmentation of the training data; without augmentation an epoch took just 70 s. The network was trained for 512 epochs (18 h). Without augmentation, we were overfitting already after about 20 epochs, meaning the training loss continued to decrease, but the validation loss on the validation set (which was not used for parameter optimization) began to deteriorate. When using the data augmentation strategy as described above we avoided overfitting even after 512 epochs. Averaged over the last 100 epochs, the validation accuracy is (0.9313 mean, 0.0079 std).

We applied the learned network to the test set consisting of 10,590 cell images. In contrast to the long training phase, the prediction of the probabilities for the 4 classes only takes approximately 6.9 s for all images. The overall accuracy on the test set is 93.4%. The confusion matrix is shown in Table 12.3 together with the best baseline approach (LDA).

In this HCS study, the CNN trained with raw images yields the best classification accuracy when compared to three state-of-the-art image analysis approaches with the traditional pipeline of image feature extraction followed by training a classifier based on those features. Besides the better performance of the CNN-based approach, it has additional benefits such as saving time and costs during the image analysis step and providing high robustness and broad application range.

[10] http://lasagne.readthedocs.io

Table 12.3 Results of baseline and CNN approach on the test set

	DMSO (True)	Cluster A (True)	Cluster B (True)	Cluster C (True)
CNN				
DMSO	7775	13	208	0
Cluster A	28	382	23	1
Cluster B	414	8	1657	0
Cluster C	0	0	0	81
LDA				
DMSO	7949	20	542	0
Cluster A	15	323	35	12
Cluster B	251	60	1310	1
Cluster C	2	0	1	69

3 Learning to Cluster: Extracting Relevant Features for Speaker Diarization

Speaker diarization is the task of segmenting an audio recording of a meeting, a lecture, a political debate, or some broadcast media by speaker identity to answer the question "*who spoke when*" (Beigi 2011). No prior knowledge about the number or specific identities of participating speakers is assumed. If we assume a pre-segmentation into speaker-specific segments by some other process, but still need to answer the question of which segments belong to the same speaker and how many speakers exist, the task is called *speaker clustering*. Typical business use cases arise as a preprocessing step to general media indexing (in order to make it searchable), specifically in media monitoring (e.g., who has been covered on radio), meeting summarization (e.g., to search by panelist), or the evaluation of qualitative interviews in psychological research.

Speaker clustering is typically approached by first extracting base audio features like Mel-frequency cepstrum coefficients (MFCC) for the whole audio stream (Ganchev et al. 2005), followed by a segment-wise modeling [e.g., using adapted Gaussian mixture models (Reynolds et al. 2000)] to create higher-level speaker-specific features per segment [e.g., i-vectors (Dehak et al. 2011)]. These higher-level features of each segment are then subject to a clustering process. Typically, agglomerative hierarchical clustering is used (Kotti et al. 2008).

In general, clustering is viewed as the prototypical example of an unsupervised learning task, using algorithms like k-means (MacQueen 1967) or density-based spatial clustering of applications with noise (DBSCAN) (Ester et al. 1996) as alternatives to hierarchical clustering. As with supervised learning schemes, these algorithms have their inductive biases (Mitchell 1980). They will find structure in the data if and only if (a) that structure is reflected in the extracted features, and (b) the structure fits what the algorithm is biased to look for. K-means, for example, will find structure expressed in the mutual distances between data points and hypothesized

cluster centers, while DBSCAN finds clusters only if they are reflected in the density structure of the data set.

In general, clustering is also close in spirit to the task of classification: while a classifier groups the test data into any of a pre-defined number of classes, a clustering algorithm basically has the same goal of grouping test data together—just that the number and identity of classes/clusters is not predefined. Given the success of deep neural networks in classification on the one hand, and their general ability to extract meaningful and task-specific features from almost raw data on the other hand (Razavian et al. 2014), it seems compelling to bring these properties to bear on the task of speaker clustering.

3.1 Supervised Learning for Improved Unsupervised Speaker Clustering

The typical deep learning approach to clustering uses the neural network as a data-driven feature extractor to transform the input into so-called embeddings (Mikolov et al. 2013; Romanov and Rumshisky 2017). Each embedding is then used as the new representation of the input vector and fed into a subsequent clustering process using one of the abovementioned classic algorithms. The embedding is found for a respective input by extracting the activations of one of the upper layers of the neural network, which has previously been trained for a related or "surrogate" task.

For this setup to be successful for speaker clustering, it is important that the learned embeddings (or high-level features) incorporate the following ideas:

- *Contain prosodic information:* Stadelmann and Freisleben (2009) highlighted the importance of the evolution of a sound using short segments of ca. 120 ms in length for human-level recognition performance [i.e., temporal information matters instead of a pure bag-of-frames approach (Aucouturier et al. 2007)].
- *Be voice-specific:* When the surrogate task to train the feature-extracting network is speaker identification using a discriminative model, chances are that the extracted features are better suited to distinguish the specific set of speakers used during training from each other (rather than modeling what makes any voice unique, which is what is needed for clustering).

We use spectrograms[11] as input and built up a CNN architecture inspired by Dieleman and Schrauwen (2014) to extract embeddings based on these two principles, and evaluated it on the well-known TIMIT speech corpus. The architecture is shown in Fig. 12.4; Lukic et al. (2016, 2017) give all details. The rationale behind this setup is twofold: First, we address the temporal aspect mentioned above

[11] A spectrogram is a 2D image representing a time-frequency decomposition of an audio signal: the *x*-axis represents time, the *y*-axis represents frequency, and the color encodes energy (compare the leftmost part of Fig. 12.4, showing 3 s of voiced speech).

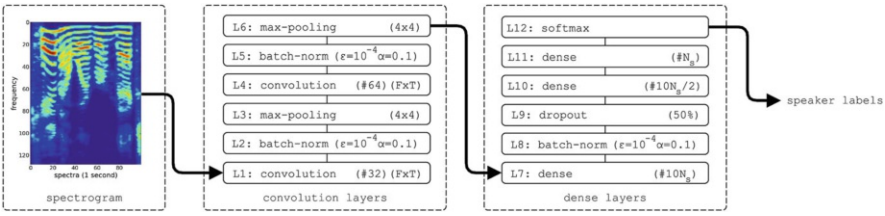

Fig. 12.4 Architecture of the CNN used to extract speaker embeddings. © 2017 IEEE. Reprinted, with permission, from Lukic et al. (2017)

("*prosodic information*") by using convolutional networks: the convolutional layers are able to extract time-dependent aspects of a voice through 2D convolutional kernels that operate on the spectrograms and thus operate on the time axis. Second, the loss function of Hsu and Kira (2015) ensures that the embeddings explicitly focus on being similar for identical speakers ("be voice-specific"), and dissimilar for different speakers (irrespective of the concrete speaker identity). This ensures a proper closeness of the surrogate supervised training task to the final task of clustering (i.e., grouping voices by closeness of their embeddings).

3.2 Results

We took the first n speakers in lexicographic ordering from the TIMIT test set for the clustering experiment. We divided the 10 sentences per speaker into two utterances by taking the first 8 sentences (lexicographically ordered by filename) for utterance one, and the last two for the second utterance. Utterance one is approximately 20 s long on average, while utterance two is ca. 5 s long. Using the architecture and experimental setup described in greater detail in Lukic et al. (2017), we have been able to cluster up to $n = 80$ speakers with a reasonable misclassification (MR) rate of 13.75%.[12] The best reported previous results worked only for up to 40 speakers with an MR of 5%, which is on par with our approach. Ca. 14% MR are a starting point for unsupervised media indexing tasks, but should be improved in the future. The main message in this result is that now automatic indexing becomes feasible because it can cope with practically relevant speaker set sizes.

Figure 12.5 allows for a qualitative assessment of the embeddings of $n = 5$ speakers. The used t-SNE visualization method performs nonlinear dimension reduction from the dimensionality of the embedding vectors into 2D (van der Maaten and Hinton 2008) while preserving the original similarity structure as much as

[12]MR counts the share of utterances that are grouped into the wrong cluster. Wrong can mean two things: utterances of different speakers are either joined into the same cluster, or utterances of one speaker are distributed over several (pure) clusters instead of combined to a single one.

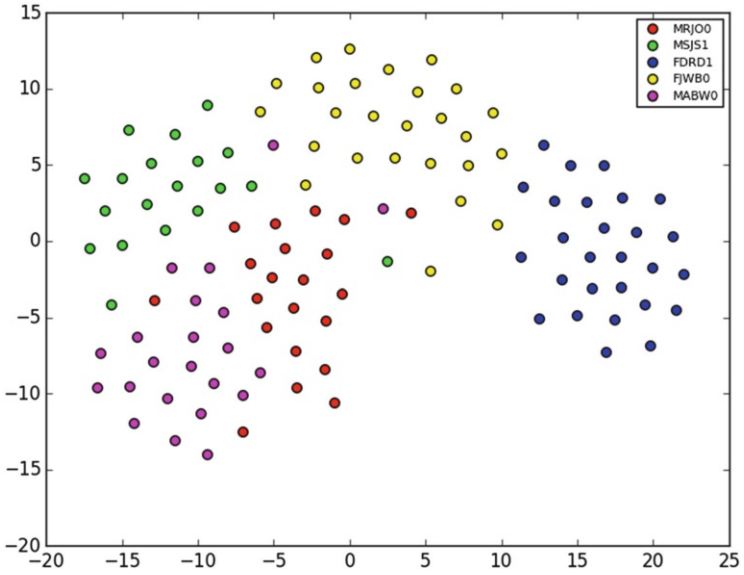

Fig. 12.5 A t-SNE visualization of the embeddings of several speech segments from 5 TIMIT speakers. © 2017 IEEE. Reprinted, with permission, from Lukic et al. (2017)

possible. We observe that overall the embeddings of any speaker group together nicely.

We conclude that for the task of speaker clustering, the sequence-learning capabilities of the CNN architecture together with the Kullback-Leibler divergence-related loss function enable the extraction of voice-specific features for subsequent clustering. The involved learning task seems to be quite non-trivial: only by using batchnorm and 30,000 epochs of training using ADADELTA (Zeiler 2012) we were able to produce useful results. A next step would be to embed the clustering in a truly end-to-end optimizable process that includes the actual clustering.

4 Learning to Segment: FCNs for Semantic Segmentation of Newspaper Pages

Newspapers are provided and consumed to a large extent in printed form. Large archives of such papers do exist, containing a historically important cultural heritage. In order to analyze or search them, they need to be available in a suitable digital form. Since newspapers consist of articles that can be considered as independent units, one usually wants to access these semantically meaningful units directly instead of whole pages. Therefore, digitization of newspapers not only needs optical character recognition (OCR) (Mori et al. 1999) but also semantic segmentation

(Long et al. 2014). The term semantic segmentation means to "cut" a page into connected components (headers, text, images) that together constitute a semantic unit we call an article. In the use case of media monitoring, today's products and services are very costly because this segmentation work has to be done manually. This also means that no real-time monitoring is possible, and neither is the processing of larger archives feasible using manual work.

In this case study, we improve a straightforward application of a classification CNN by a much better suited network architecture to achieve practically useful segmentation results of newspaper pages into sets of semantically connected articles. Both approaches are based on CNN architectures and provide segmentation masks which can be used to extract the articles from the corresponding newspaper pages. A segmentation mask is a binary image with black pixels standing for articles and white pixels for borders. In order to extract articles using the segmentation mask, we apply a post-processing step to get the coordinates of the (black) article areas matching the original scans.

Our dataset consists of 507 high-resolution scans (i.e., images) of newspaper pages from the papers with highest circulation among Swiss newspapers, ranging from classical weekly newspapers to boulevard media. It is accompanied by manually created segmentation masks as ground truth. We transform the original scans of the newspaper pages to simplified representations to be used as input for our CNNs. This is done by replacing illustrations with gray areas and by blackening lines of texts (after OCR). In the end, our dataset contains 507 pictures with two channels each (plus ground truth segmentation mask, see Fig. 12.6): the original scan, and the abovementioned transformation. We use approximately 85% of the dataset for training and hold out the remaining 15% for testing. In addition to this fully labeled dataset, we have ca. 5500 partially labeled pages (i.e., each page contains also unsegmented articles).

4.1 CNN-Based Pixel Classification vs. One-Pass FCNs

A straightforward first approach is based on the work of Ciresan et al. (2012b): we use a CNN-based pixel classification network (PCN) to classify a newspaper page pixel by pixel using subsequent applications of the CNN to every pixel. The class of each pixel (article or border) is predicted from pixel values in a 25×25 pixel-square window centered on it. The network is trained by using only the transformed images, for which we adjust the resolution so that all of them have a height of 100 pixels without changing the aspect ratio. Classifying such an image with, for example, 100×75 pixels results in 7500 windows. We therefore used ca. 3.5 million windows in training. Figure 12.7 shows the architecture of the PCN with 7 layers and approximately 2,50,000 weights to be learned.

The fully convolutional neural network used in our second approach is built with three logical parts (cp. Meier et al. (2017) and Fig. 12.8). Initially, feature extraction is done the same way as with a standard CNN. This is followed by a network

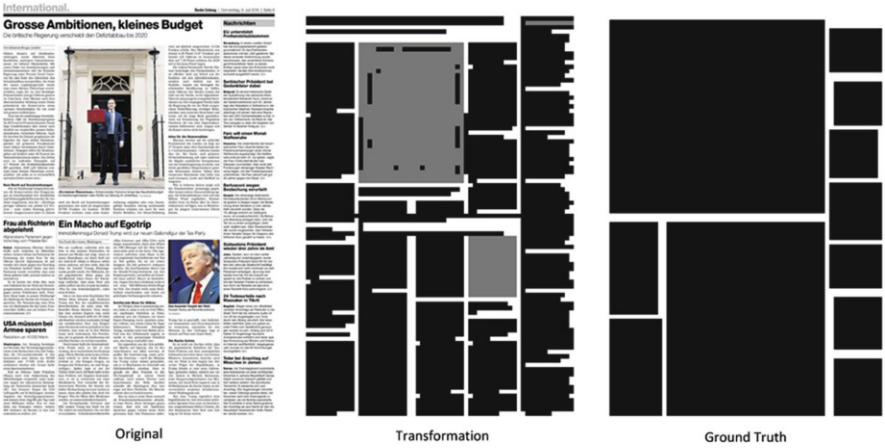

Fig. 12.6 Example of our dataset showing an original scan of a newspaper page (left), the transformed representation (middle), and the manually created segmentation mask (right)

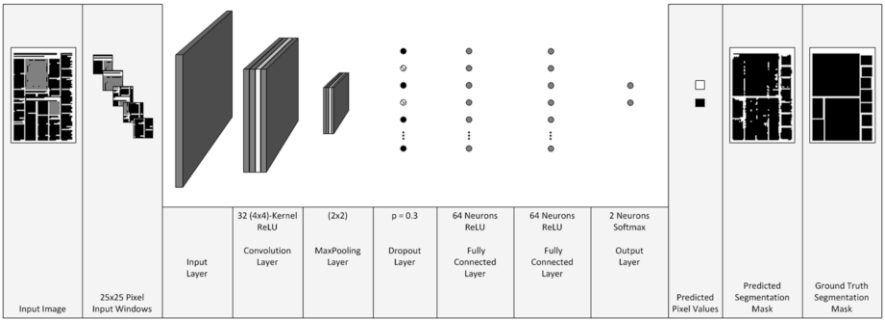

Fig. 12.7 Architecture of the PCN that segments newspaper pages by classifying each pixel of an input image (article or border) using 25 × 25 pixel windows centered on each pixel to be classified

performing an upscaling, resulting in a segmentation mask as output. Finally, a very small refinement network adjusts the edges of the article regions (black) to be rectangular, since this is one of the typical characteristics of newspaper pages. We train this network architecture in two steps: first, we run a pre-training with the bigger partially labeled dataset. For this case, the unlabeled parts are replaced by white areas. Second, we use the fully labeled data to finalize the model. For both training steps we insert the original scans together with the transformations as separate channels. Both channels are scaled down to a resolution of 256 × 256 pixels (keeping the aspect ratio by adding white background where necessary).

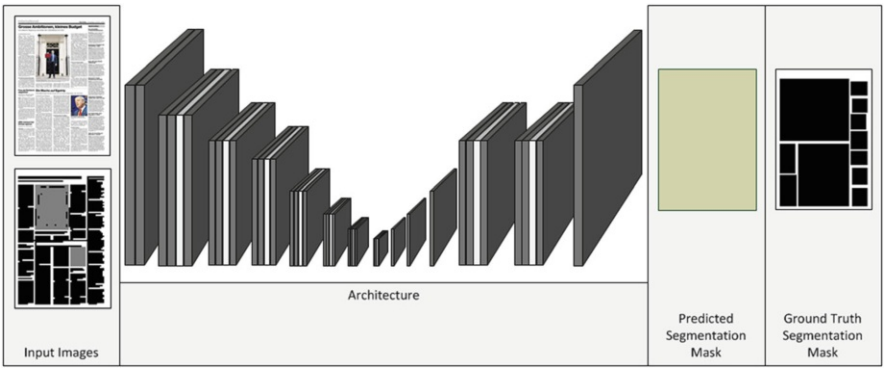

Fig. 12.8 Architecture of the FCN, consisting of three logical parts. First the feature extraction with a standard CNN (up to the center of the figure), second the segmentation (done by upscaling convolutions), and third a refinement network to ensure typical properties of newspaper articles (last block in "Architecture")

4.2 Results

For the evaluation of the PCN and FCN, we chose the diarization error rate (DER) known from speaker diarization (Kotti et al. 2008). The DER is a combination of the three error types possible when grouping elements into an unknown number of clusters: confusion error (CE) measures parts of predicted articles that are wrongly assigned; miss error (ME) measures parts of articles that are not included in the predicted segmentations; false alarm error (FE) counts parts of predicted articles that do not overlap with any labeled article from the ground truth.

The FCN has a DER score of 0.1378, thereby outperforming the still respectable PCN (0.2976 DER) by more than 50%. This result shows the impact of a suitable network architecture for the task at hand. While both architectures have comparable runtimes during prediction (roughly 3.7 s per page, largely impacted by similar post-processing), the FCN can process images that are approximately 18 times bigger considering that two images are inserted at the same time. On the other hand, while we used around 6000 pages to train the FCN, we trained the PCN with only 507 pages. We conclude that both approaches can be useful depending on the amount of labeled data that is available. For the given use case, the industrial partner provided the additional 5 k partially labeled training images in order to use the FCN approach in practice.

5 Learning to Detect Outliers: Predictive Maintenance with Unsupervised Deep Learning

The condition of critical and costly mechanical equipment is increasingly monitored by observing the vibrations of the machinery under surveillance. In order to detect faults before they damage the whole machinery, traditional methods such as envelope analysis have been used for decades (Randall and Antoni 2011). However, these methods require knowledge of the machinery's exact geometry and a skilled human operator. An alternative data-driven approach is to automatically detect changes in the signal. This is known as novelty detection and there are plenty of classical methods. For a review, see Pimentel et al. (2014).

While almost every possible combination of features and classifiers has been tried previously for condition monitoring, the respective literature lacks comparability in terms of data and metrics used as well as given details for reproducibility (Stadelmann et al. 2016). In this case study, we compare several classical novelty detection methods against DL-based approaches on a standard bearing data set (Lee et al. 2007), which consists of $n_{\text{train}} + n_{\text{test}} = 984$ measurements of vibration signals in run-to-failure tests.

As a first step, we use a Fourier transformation to extract $p = 100$ features[13] per measurement to obtain a data matrix $X \in R^{(n_{\text{train}}+n_{\text{test}})\times p}$. The details of the feature extraction and the data set can be found in Fernández-Francos et al. (2013). In the following discussion, we assume that the fault starts to be detectable at frame number 532. This is in line with findings from other researchers (Fernández-Francos et al. 2013) and is also observable from Fig. 12.9, which shows the data matrix (spectrogram).

The output of all methods is a real valued vector of size n_{test}, reflecting the deviation from the normal state learned during the training: the so-called novelty signal. All methods are trained on the first $n_{\text{train}} = 200$ rows, where we assume that no fault has occurred. Before training and testing, we apply a robust z-transformation for each feature using the median and the median absolute deviation (MAD) calculated on the training data.

5.1 Classical Approaches

We use the following classical methods as baseline:

- One-class SVM (Schölkopf and Smola 2002) with the variable parameter η, which can be understood as an upper bound of the fraction of outliers.
- Gaussian mixture model (GMM) with a number of $n_{\text{components}}$ mixtures (Reynolds and Rose 1995).

[13] Fast fourier transformation (FFT) features: energies of 100 equally spaced frequency sub-bands, computed over the whole length of the signal (10 s).

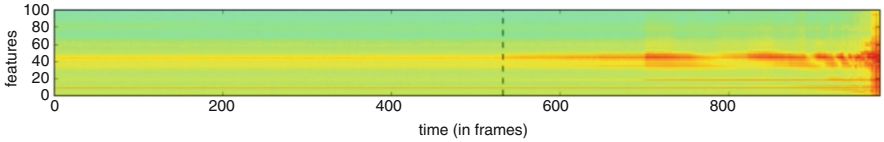

Fig. 12.9 The 100 extracted features (columns) for the 984 time points (rows) for the example data set. The vertical dashed line indicates the first change of the data (frame number 532) as visible by eye

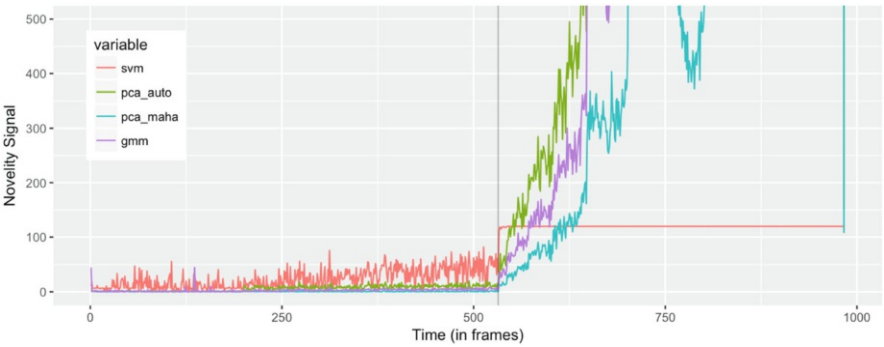

Fig. 12.10 Novelty signal for the classical methods. The used parameters are $\eta = 0.1$ for the SVM, $n_{\text{components}} = 16$ for GMM and $n_{\text{comp}} = 50$ for the PCA-based methods. To focus on the sudden change of the novelty signal at 532, the maximum of the y-axis has been limited to 500 (maximum signal is in the order of 1E10)

- A simple but robust baseline approach is done in the spirit of the Mahalanobis distance. To reduce the noise, we first transform our data into a n_{comp}-dimensional subspace using PCA with whitening. In that subspace, we calculate the squared Mahalanobis-distance (Bersimis et al. 2007) to determine the outliers.
- The last classical method in our evaluation uses a PCA learned on the training data, to transform the test data X_{test} into a n_{comp}-dimensional subspace. After that, the data is transformed back into the original space yielding $\widehat{X_{\text{test}}}$. We use the L^2-based reconstruction error as the novelty signal. This corresponds to an autoencoder without nonlinearities and with tied weights.

Figure 12.10 shows the results of the classical methods described above.

All methods show an amplitude increase in the novelty signal at frame number 532, where we assume that the fault is detectable. The `pca_auto` method shows the strongest increase in the signal and is used later for the comparison with the deep learning-based methods.

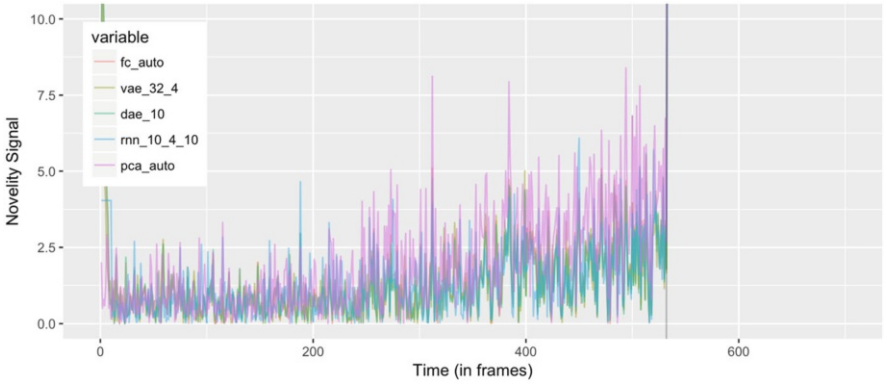

Fig. 12.11 Novelty signal for the deep learning-based methods and the best classical approach (`pca_auto`). All methods show a steep ascent after the fault is detectable at frame number 532. To better illustrate the signal before the fault, we limited the range of the normalized novelty signal to [0, 10]

5.2 Deep Learning-Based Methods

After using the classical methods to establish a baseline, we now consider deep autoencoders. All methods are trained for 50 epochs with a batch size of 20. We start with a simple fully connected autoencoder using sigmoids as activations. A detailed description of a fully connected autoencoder can be found, for example, in Goodfellow et al. (2016). We investigated different numbers of hidden layers and determined that for 5 hidden layers there is the best compromise between steepness of the novelty signal after the fault and the noise before it (see `fc_auto` in Fig. 12.11).

In addition to the `fc_autoencoder`, we also include a recurrent version, in which the neurons are replaced by Long Short-Term Memory cells (LSTMs) (Hochreiter and Schmidhuber 1997). We found that an architecture with three hidden layers consisting of 10, 4, and 10 nodes, respectively, performed best. These results are displayed in Fig. 12.11. The behavior is similar to the standard autoencoder and suggests that the temporal ordering of frames is unimportant here.

The final two autoencoders introduce means for additional regularization. The first one, the denoising autoencoder, does this by injecting additional noise, see Vincent et al. (2010) for details. The best performance was observed with 10 nodes in the hidden layer. The second one is the variational autoencoder (VAE) (Kingma and Welling 2013). Its optimal architecture turned out empirically to have 32 nodes in the hidden layer and a 4-dimensional latent space, which is shown in Fig. 12.11, labeled as `vae_32_4`. Note that, in principle, the VAE can also be extended to generate novel data.

In conclusion, all methods (classical and DL) show a similar novelty signal and detect the fault at time frame 532. However, the DL-based methods give a weaker novelty signal in a region where there is no fault. Here, the best classical method

(`pca_auto`) shows a stronger signal at times before the fault occurred. We conclude that the given task is too simple to profit from the more capable models—DL is not needed on this specific data set.

6 Lessons Learned

Deep learning approaches have proven useful not only in academic computer vision settings but also in various scenarios inspired by real business use cases. We have improved the state of the art in high content screening, speaker clustering, and automatic article segmentation, while showing at least comparable results for condition monitoring. Overall, the authors have verified the practicability of DL applications on at least 10 substantial research projects in collaboration with industry during the last 4 years. Contrary to public opinion, Szegedy et al. (2014) note that "*most of this progress is not just the result of more powerful hardware, larger datasets and bigger models, but mainly a consequence of new ideas, algorithms and improved network architectures.*" This is according to our experience worth considering.

6.1 Working with Limited Resources

Our biggest take-home message is the importance of working well with limited resources. Having a good set of data for training and evaluation (i.e., available at the start of the project, ideally large,[14] in a good shape for further processing, resembling the true distribution of the problem to be solved) is the starting point: it does not pay off to "negotiate" minimum numbers of needed data with business owners. Rather, "the more the better" is key. If the most one can get is still little, the following tricks may apply:

- Using available pre-trained networks that have been trained for a "close enough" task (e.g., the VGG-16 network[15] for any image classification task) to do transfer learning.
- Use trainable architectures like Inception (Szegedy et al. 2014) or Resnet (He et al. 2015b) that adapt their complexity to the available data and may even be compressible (Han et al. 2015).

[14]Personal rule of thumb of one of the authors (T.S.): I feel comfortable with a small to medium four-digit number of instances per class in a classification setting.

[15]http://www.robots.ox.ac.uk/~vgg/research/very_deep/

- Do sensible data augmentation (see Sect. 2): provide the training procedure with variants of your original data that (a) you can create randomly on the fly and that (b) resemble distortions/alterations relevant and realistic in practice.
- Often there is enough unlabeled data, but labeling is costly. In that case one can try to employ semi-supervised learning methods, which are currently being actively developed (Kingma et al. 2014). Another possibility is to use high-level features created by a first network to do a clustering or t-SNE embedding similar to Fig. 12.5 (see Sect. 3). This allows to label lots of data after a short inspection.

Sometimes, data is not the limiting factor, but hardware is (at least for applying the trained model later). While compressed networks help to speed up network application considerably, it should be noted that while neural network training of practically relevant size may take weeks on dedicated hardware (i.e., latest generation of GPU workstations), the application might be doable in real time even on embedded devices like a raspberry pi[16] (see also Sect. 4). And as Sect. 5 has shown, DL approaches might not always outperform simple baseline approaches; so it always pays off to compare against classical methods (at least to establish a benchmark, see Sect. 2).

6.2 *Other Advice*

Additional advice can be summarized as follows:

- Having a good start on a new use case often depends on (a) starting from an easy, well-understood baseline model closely resembling a published architecture and task,[17] and (b) to slowly increase the complexity of the architecture. As a rule of thumb, if a human can see/hear/... the solution to a pattern recognition problem in the training data, it can be extracted using machine-learning algorithms (Domingos 2012).
- If it is not a standard problem, ensure to provide a loss function which really describes the problem that is going to be solved (see Sect. 3).
- Latest algorithmic developments in neural nets like dropout or batchnorm, ADAM/ADADELTA and ReLU are "always on" in our projects if applicable[18] as they considerably ease training to the point that makes applications possible that just do not work without them (see Sect. 3).

[16] https://www.martinloeser.eu/deutsch/forschung/pivision/

[17] Find a collection of models per task, for example here: https://github.com/sbrugman/deep-learning-papers

[18] For example, dropout does not work with ResNets and has largely been replaced by batchnorm in these architectures (Zagoruyko and Komodakis 2016).

- It is common that a first instance of a DL model does not work on a completely new task and data set. Then, debugging is key, ranging in methodology from checking for the application of best practices,[19] hand-calculating the training equations for toy examples (to find implementation problems, e.g., in the loss function[20]), visualizing the pre-processed data (to see if data loading might be buggy) or learned weights,[21] and inspecting loss values (does it learn at all[22]?) as well as misclassified training examples [to get intuition into what goes wrong (Ng 2019)].
- The speed of new advances in DL is breathtaking at the moment. While new developments are published daily on arXiv,[23] news aggregators like reddit[24] or Data Machina[25] and explanation-focused journals like Distill[26] help to stay up-to-date. For a real project, it is important to check the current state of the art at least back to the latest major DL conferences neural information processing (NIPS),[27] international conference on machine learning (ICML),[28] and international conference on learning representations (ICLR)[29] and the trends discussed there in tutorials and keynotes: paradigms are still evolving, and new applications are shown daily.

General best practices for DL applications are also summarized by Ng (2016), Hinton et al. (2012), and LeCun et al. (1998b).

Acknowledgments The authors are grateful for the support by CTI grants 17719.1 PFES-ES, 17729.1 PFES-ES, and 19139.1 PFES-ES.

References

Aucouturier, J.-J., Defreville, B., & Pachet, F. (2007). The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music. *The Journal of the Acoustical Society of America, 122*(2), 881–891.

Beigi, H. (2011). *Fundamentals of speaker recognition*. Springer Science & Business Media.

[19] See also https://engineering.semantics3.com/2016/10/09/debugging-neural-networks-a-checklist/

[20] See also https://gab41.lab41.org/some-tips-for-debugging-deep-learning-3f69e56ea134

[21] See, for example, https://distill.pub/2017/feature-visualization/ and https://github.com/bruckner/deepViz

[22] See also http://russellsstewart.com/notes/0.html

[23] https://arxiv.org/

[24] https://www.reddit.com/r/MachineLearning/

[25] https://www.getrevue.co/profile/datamachina

[26] http://distill.pub/

[27] https://nips.cc/

[28] http://icml.cc

[29] http://iclr.cc

Bersimis, S., Psarakis, S., & Panaretos, J. (2007). Multivariate statistical process control charts: An overview. *Quality and Reliability Engineering International, 23*, 517–543.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. In: *From form to meaning: Processing texts automatically, Proceedings of the Biennial GSCL Conference 2009* (pp. 31–40). https://svn.spraakdata.gu.se/repos/gerlof/pub/www/Docs/npmi-pfd.pdf

Chung, J. S., Senior, A. W., Vinyals, O., & Zisserman, A. (2016). Lip reading sentences in the wild. *CoRR*, Vol. 1611.05358. http://arxiv.org/abs/1611.05358

Ciresan, D., Meier, U., Masci, J., & Schmidhuber, J. (2012a). *Multi-column deep neural network for traffic sign classification*. http://people.idsia.ch/~juergen/nn2012traffic.pdf

Ciresan, D., Giusti, A., Gambardella, L. M., & Schmidhuber, J. (2012b). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in Neural Information Processing Systems, 25*, 2843–2851.

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., & Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing, 19*(4), 788–798.

Dieleman, S., & Schrauwen, B. (2014). End-to-end learning for music audio. In *Proceedings of ICASSP* (pp. 6964–6968).

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM, 55*(10), 78–87.

Dürr, O., Duval, F., Nichols, A., Lang, P., Brodte, A., Heyse, S., & Besson, D. (2007). Robust hit identification by quality assurance and multivariate data analysis of a high-content, cell-based assay. *Journal of Biomolecular Screening, 12*(8), 1042–1049.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (KDD-96) (pp. 226–231) AAAI Press.

Fernández-Francos, D., Martínez-Rego, D., Fontenla-Romero, O., & Alonso-Betanzos, A. (2013). Automatic bearing fault diagnosis based on one-class ν-SVM. *Computers & Industrial Engineering, 64*(1), 357–365.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics, 36*(4), 193–202.

Ganchev, T., Fakotakis, N., & Kokkinakis, G. (2005). Comparative evaluation of various MFCC implementations on the speaker verification task. In *Proceedings of SPECOM 2005* (Vol. 1, pp. 191–194).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. http://www.deeplearningbook.org.

Gustafsdottir, S. M., Ljosa, V., Sokolnicki, K. L., Wilson, J. A., Walpita, D., Kemp, M. M., Petri Seiler, K., Carrel, H. A., Golub, T. R., Schreiber, S. L., Clemons, P. A., Carpenter, A. E., & Shamji, A. F. (2013). Multiplex cytological profiling assay to measure diverse cellular states. *PLoS One, 12*, e80999.

Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. *CoRR*, Vol. 1510.00149. https://arxiv.org/abs/1510.00149

He, K., Zhang, X., Ren, S., & Sun, J. (2015a). Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *CoRR*, Vol. 1502.01852. http://arxiv.org/abs/1502.01852

He, K., Zhang, X., Ren, S., & Sun, J. (2015b). Deep residual learning for image recognition. *CoRR*, Vol. 1512.03385. https://arxiv.org/abs/1512.03385

Hinton, G. E., Srivastava, N., & Swersky, K. (2012). Lecture 6a: Overview of mini-batch gradient descent. In *Neural Networks for Machine Learning*, University of Toronto. https://www.coursera.org/learn/neural-networks

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Hsu, Y.-C., & Kira, Z. (2015). Neural network-based clustering using pairwise constraints. *CoRR*, Vol. 1511.06321. https://arxiv.org/abs/1501.03084

Hubel, D. H., & Wiesel, T. N. (1959). Receptive fields of single neurons in the cat's striate cortex. *Journal of Physiology, 148*, 574–591.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning* (Vol. 37, pp. 448–456). https://arxiv.org/pdf/1502.03167.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, Vol. 1412.6980. http://arxiv.org/abs/1412.6980

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational Bayes. *CoRR*, Vol. 1312.6114. https://arxiv.org/abs/1312.6114

Kingma, D. P., Mohamed, S., Rezende, D. J., & Welling, M. (2014). Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems* (pp. 3581–3589). https://papers.nips.cc/paper/5352-semi-supervised-learning-with-deep-generative-models

Kotti, M., Moschou, V., & Kotropoulos, C. (2008). Speaker segmentation and clustering. *Signal Processing, 88*(5), 1091–1124.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems, 25*, 1097–1105.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998a). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

LeCun, Y., Bottou, L., Orr, G. B., & Mueller, K.-R. (1998b). Efficient BackProp. In G. B. Orr, & K.-R. Mueller (Eds.), *Neural networks: Tricks of the trade*, Lecture Notes in Computer Science (Vol. 1524, pp. 9–50).

LeCun, Y., Bengio, Y., & Hinton, G. E. (2015). Deep learning. *Nature, 521*, 436–444.

Lee, J., Qiu, H., Yu, G., & Lin, J. (2007). *Bearing data set*. IMS, University of Cincinnati, NASA Ames Prognostics Data Repository, Rexnord Technical Services. https://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/

Ljosa, V., Sokolnicki, K. L., & Carpenter, A. E. (2009). Annotated high-throughput microscopy image sets for validation. *Nature Methods, 9*, 637.

Long, J., Shelhamer, E., & Darrell, T. (2014). *Fully convolutional networks for semantic segmentation*. https://people.eecs.berkeley.edu/~jonlong/long_shelhamer_fcn.pdf

Lukic, Y. X., Vogt, C., Dürr, O., & Stadelmann, T. (2016). Speaker identification and clustering using convolutional neural networks. In *Proceedings of IEEE MLSP 2016*.

Lukic, Y. X., Vogt, C., Dürr, O., & Stadelmann, T. (2017). Learning embeddings for speaker clustering based on voice quality. In *Proceedings of IEEE MLSP 2017*.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability* (Vol. 1, pp. 281–297). Berkeley: University of California Press.

Meier, B., Stadelmann, T., Stampfli, J., Arnold, M., & Cieliebak, M. (2017). Fully convolutional neural networks for newspaper article segmentation. In *Proceedings of ICDAR 2017*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems* (pp. 3111–3119). https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

Mitchell, T. M. (1980). *The need for biases in learning generalizations*. Technical Report, Rutgers University, New Brunswick, NJ. http://www.cs.nott.ac.uk/~pszbsl/G52HPA/articles/Mitchell:80a.pdf

Moravcik, M., Schmid, M., Burch, N., Lisy, V., Morrill, D., Bard, N., Davis, T., Waugh, K., Johanson, M., & Bowling, M. H. (2017). DeepStack: Expert-level artificial intelligence in no-limit poker, *CoRR*, Vol. 1701.01724. http://arxiv.org/abs/1701.01724

Mori, S., Nishida, H., & Yamada, H. (1999). *Optical character recognition*. New York, NY: Wiley. ISBN 0471308196.

Ng, A. (2016). *Nuts and bolts of building AI applications using deep learning*. NIPS Tutorial.

Ng, A. (2019, in press). *Machine learning yearning*. http://www.mlyearning.org/

Nielsen, M. A. (2015). *Neural networks and deep learning*. Determination Press. http://neuralnetworksanddeeplearning.com.

Nielsen, F. A. (2017). *Status on human vs. machines, post on "Finn Årup Nielsen's blog"*. https://finnaarupnielsen.wordpress.com/2015/03/15/status-on-human-vs-machines/

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering, 22*(10), 1345–1359. http://ieeexplore.ieee.org/abstract/document/5288526/.

Pimentel, M. A. F., Clifton, D. A., Clifton, L., & Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, 215–249.

Randall, R. B., & Antoni, J. (2011). Rolling element bearing diagnostics—A tutorial. *Mechanical Systems and Signal Processing, 25*(2), 485–520.

Razavian, A. S., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: An astounding baseline for recognition. *CVPR 2014* (pp. 806–813). https://arxiv.org/abs/1403.6382

Reynolds, D. A., & Rose, R. C. (1995). Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing, 3*(1), 72–83.

Reynolds, D. A., Quatieri, T. F., & Dunn, R. B. (2000). Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing, 10*(1), 19–41.

Romanov, A., & Rumshisky, A. (2017). Forced to learn: Discovering disentangled representations without exhaustive labels. *ICRL 2017*. https://openreview.net/pdf?id=SkCmfeSFg

Rosenblatt, F. (1957). *The perceptron – A perceiving and recognizing automaton*. Technical report 85-460-1, Cornell Aeronautical Laboratory.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Learning representations by back-propagating errors. In *Neurocomputing: Foundations of Research* (pp. 696–699). MIT Press. http://www.cs.toronto.edu/~hinton/absps/naturebp.pdf

Schmidhuber, J. (2014). *Deep learning in neural networks: An overview*. https://arxiv.org/abs/1404.7828

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. Cambridge, MA: MIT Press.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature, 529*, 484–503.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, vol. 1409.1556. https://arxiv.org/abs/1409.1556

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15*(1), 1929–1958.

Stadelmann, T., & Freisleben, B. (2009). Unfolding speaker clustering potential: A biomimetic approach. In *Proceedings of the 17th ACM International Conference on Multimedia* (pp. 185–194). ACM.

Stadelmann, T., Musy, T., Duerr, O., & Eyyi, G. (2016). Machine learning-style experimental evaluation of classic condition monitoring approaches on CWRU data. Technical report, *ZHAW Datalab* (unpublished).

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2014). Going deeper with convolutions. *CoRR*, Vol. 1409.4842. https://arxiv.org/abs/1409.4842

Szeliski, R. (2010). *Computer vision: Algorithms and applications. Texts in Computer Science*. New York: Springer. http://szeliski.org/Book/.

van der Maaten, L., & Hinton, G. E. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research, 9*(Nov), 2579–2605.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research, 11*, 3371–3408.

Weyand, T., Kostrikov I., & Philbin, J. (2016). PlaNet – Photo geolocation with convolutional neural networks. *CoRR*, Vol. 1602.05314. http://arxiv.org/abs/1602.05314

Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., Yu, D., & Zweig, G. (2016). Achieving human parity in conversational speech recognition. *CoRR*, Vol. 1610.05256. http://arxiv.org/abs/1610.05256

Zagoruyko, S., & Komodakis, N. (2016). Wide residual networks. In Wilson, R. C., Hancock, E. R., & Smith, W. A. P. (Eds.), *Proceedings of the British Machine Vision Conference (BMVC)* (pp. 87.1–87. 12. BMVA Press.

Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. *CoRR*, Vol. 1212.5701. http://arxiv.org/abs/1212.5701

Zheng, F., Zhang, G., & Song, Z. (2001). Comparison of different implementations of MFCC. *Journal of Computer Science and Technology, 16*(6), 582–589.

Chapter 13
The Beauty of Small Data: An Information Retrieval Perspective

Martin Braschler

Abstract This chapter focuses on Data Science problems, which we will refer to as "Small Data" problems. We have over the past 20 years accumulated considerable experience with working on Information Retrieval applications that allow effective search on collections that do not exceed in size the order of tens or hundreds of thousands of documents. In this chapter we want to highlight a number of lessons learned in dealing with such document collections.

The better-known term "Big Data" has in recent years created a lot of buzz, but also frequent misunderstandings. To use a provocative simplification, the magic of Big Data often lies in the fact that sheer volume of data will necessarily bring redundancy, which can be detected in the form of patterns. Algorithms can then be trained to recognize and process these repeated patterns in the data streams.

Conversely, "Small Data" approaches do not operate on volumes of data big enough to exploit repetitive patterns to a successful degree. While there have been spectacular applications of Big Data technology, we are convinced that there are and will remain countless, equally exciting, "Small Data" tasks, across all industrial and public sectors, and also for private applications. They have to be approached in a very different manner to Big Data problems. In this chapter, we will first argue that the task of retrieving documents from large text collections (often termed "full text search") can become easier as the document collection grows. We then present two exemplary "Small Data" retrieval applications and discuss the best practices that can be derived from such applications.

1 Introduction

It may seem counterintuitive at first that the present chapter focuses on "Small Data." There is a strong association of "difficult" (and by extension, "exciting" or "interesting") that goes with the "big" in "Big Data"—the sheer volume of data that is

M. Braschler (✉)
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: bram@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_13

often tackled in Big Data scenarios supposedly renders providing a solution for the problems especially hard. We examine this perception in the case of "full text search." Search—of data and information—is a prominent Data Science use case. The academic discipline that considers search on unstructured (textual) information is usually termed "Information Retrieval" (IR).[1] We will argue that neither data volume (nor data variety or velocity) in itself makes Information Retrieval inherently more difficult—rather, it may be more appropriate to say that it changes the characteristics of the problem.

Definitions of what "Small Data" entails vary. Many definitions are based on the human capacity for comprehension or action based on data—data that is "small enough" to be comprehensible or actionable.[2] We think it is necessary to accept that "small" and "big" is a shifting qualification—what is big in today's data science terms may well seem small in the future. In the field of Information Retrieval, the famous TREC challenge in 1992 (Harman 1993) was framed originally as a challenge in terms of tackling a heretofore unattained data volume[3]—participants had to index and search one gigabyte of text! One might retrospectively call this a "Big Data" challenge for that time.

How to identify the boundaries between small and big may thus well be an exercise in detecting those "tipping points" where the problems we solve shift in characteristic—where new phenomena arise based on the fact that the data volume (or velocity, or variety) reaches a certain threshold. However, aiming to identify these boundaries in a generalized way goes beyond the intent and scope of this chapter.

Instead, we work with document collections and their associated retrieval applications that are—considering the state-of-the-art in the field—safely on the "small" side (by virtue of containing significantly less than a million items).

Independently from the question of specific boundaries, accepting the viewpoint that "Small Data" and "Big Data" are distinct disciplines, however, immediately implies that problems that are truly "Big Data" should be approached in a fundamentally different manner than Small Data problems. Perhaps more intriguingly, and more subtle, is the question of whether such "Big Data" problems are truly more difficult to solve than "Small Data," or whether the shift of challenges is more in terms of the nature of the problem than its difficulty. While we want to keep the focus of the chapter on giving advice as to how "Small Data Information Retrieval problems" are solved, we still will attempt to show that indeed there are multiple hints at the latter viewpoint—some operations may actually become decidedly easier when the data volume increases.

[1] Good introductions to the field can be found in, for example, Baeza-Yates and Riebeiro-Neto (2011) and (Manning et al. 2008).

[2] See for example, http://jwork.org/main/node/18 and http://whatis.techtarget.com/definition/small-data

[3] Harman speaks of "providing a very large test collection" and notes that "TREC is designed to encourage research in Information Retrieval using large data collections" (Harman 1993).

The remainder of the chapter is structured as follows: In Sect. 2, the academic field of Information Retrieval is introduced, and related work is discussed. In Sect. 3, the "matching task," that is, the task of matching the user's description of information need to documents that carry relevant information is analyzed with respect to its interaction with collection size. First, the changing nature of the retrieval problem is analyzed (Sect. 3.1), then the distribution of word occurrences (Sect. 3.2) and its effect on term weighting (Sect. 3.3) are discussed. Whether collection size has an impact on the number of word forms used in documents is investigated in Sect. 3.4. Having established how collection size changes the nature of the matching task, Sect. 4 then presents two operational applications that serve as examples of how to implement retrieval applications for Small Data problems. In Sect. 5, we provide conclusions and list best practices for retrieval on Small Data. These practices stand in a line with some previous work on best practices for Information Retrieval applications such as published in Braschler and Gonzalo (2009) and Braschler et al. (2012).

2 The Academic Field of Information Retrieval and Related Work

For the present chapter, we will mostly cover search on unstructured document collections as our main object of study. This problem is at the core of the academic field of Information Retrieval. A document collection can be of arbitrary size, from small to (very) large, and indeed there are unstructured document collections that are processed with Information Retrieval technology that number billions or even trillions of retrievable items. A famous example is the "Google Web Search" service that foremost provides access to the textual pages of the World Wide Web.[4] There are, thus, Information Retrieval problems that are clearly all over the spectrum of data sizes that can be processed by today's technology, including up to truly "Big Data" size.

Much Information Retrieval literature covers text retrieval, that is, the search on unstructured full text (the Google Web Search example falls under this category as well). Relevant research spans more than 50 years (Spärck Jones and Willett 1997), and text retrieval remains an active and very important problem. Generally speaking,

[4]See www.google.com. Official statements as to the number of web pages indexed by the Google Web Search service are hard to interpret. In 2008, Google reported (Google 2008) that its web crawlers have discovered more than 1 trillion unique URLs pointing to web pages, though it remains unclear what subset of these is accessible through the search index. By 2016, Google reported (Google 2016) that it estimates the web to contain approximately 130 trillion pages, but again, it is unclear how many are searchable.

text retrieval approaches are very mature, and corresponding systems are available for ready use in many different guises, including free open-source systems.[5]

A modern Information Retrieval system allows the user to query the document collection with either a sequence of keywords or fully fledged natural language expressions. Typically, in contrast to database systems, no complex, formal query language is used. Classically, it is assumed that the user has an information need that is satisfied by finding as much relevant information as possible on a topic of interest. A key aspect of the "text retrieval problem" solved by the IR system is the reconciliation of the description of the information need that a user gives to the system with the formulations of an arbitrary author that are contained in a retrievable document. Put more simply, the user formulates a question, but the task of the IR system is to provide a document that contains an answer.

There are two main obstacles here:

- Firstly, a user's formulation of information need ("the question") is often inadequate: it would be paradoxical to expect a user with an *information deficit* to come up with keywords that match the formulation of what is essentially the answer.
- Secondly, natural language gives us great freedom in how we express ourselves, choosing between synonyms and paraphrasings, which precludes solutions that rely on exact string matches.[6]

We argue that while the first point is essentially independent of the actual document collection to be searched (one way to look at this is that the user's query formulation is usually not really informed by the contents of the collection), the second point often is very much influenced by the size of that collection. This is due to an increasing probability in many nonartificial large collections that they contain redundant information, that is, the same facts are included repeatedly in the document collection as the size of the collection grows. Consider the World Wide Web as an example: due to the very large number of independently contributing authors on the web, all but the most obscure facts and news stories are reported numerous times.

With increasing redundancy frequently comes a much better coverage of the many different possibilities of formulation that natural language affords; simply put, every fact is conveyed in every possible formulation as the collection size grows toward infinity. In converse, should a fact only be present in one item of the document collection, the burden is on the user to "guess" the exact formulation in order to successfully retrieve the item (although, as we will demonstrate, there are of course measures that can be taken in building an Information Retrieval system that supports the user in this burden). In a very large collection, the user can match at

[5]Two particularly notable open-source Information Retrieval projects are Lucene (lucene.apache.org) and Terrier (terrier.org).

[6]For more in-depth discussion of these issues, including a look at the larger "information acquisition cycle" that is supported by an Information Retrieval system, see Peters et al. (2012).

least *some* items[7] using an essentially arbitrary formulation. Formally speaking, we can argue that smaller collections tend to be more "dense" in terms of information content, in the sense that many facts are only represented once or a few times (less redundancy), whereas with growth a collection tends to become "thinner" (more redundancy).

To progress from the above considerations, we want to first spotlight the question on how the matching task changes as a collection becomes larger and more redundant ("thinner"). This specific question has received relatively little research in Information Retrieval so far, although one major thread of exploration is especially noteworthy. The associated work has come in the context of the TREC IR evaluation forum in the "Very Large Track" (Hawking and Thistlewaite 1997). It is best summarized in the comprehensive paper by Hawking and Robertson (2003) that goes much farther in analyzing the impact of IR-specific considerations such as term weighting and characteristics of evaluation measures with respect to collection size than we can by necessity do here in this chapter, which is more application-oriented. Their paper is very much recommended reading when more insight into consequences of using specific measures or sampling from document collections is welcome. We will also qualify some of our own findings using the more thorough conclusions in Hawking's and Robertson's work.

The belief that there is value in the search for potentially shifting characteristics of the matching task is also motivated by earlier work in web search, where a multiyear struggle of substantial scale was necessary by the IR research community to properly assess the influence of the use of page link information in document ranking. In summary, early efforts in web search evaluation were unable to demonstrate any benefit in using link information over classical, pure keyword-based retrieval methods. This held true as test collections were scaled up to 100 gigabytes of size (Hawking et al. 1998, 1999a, b). This result was, as observed by the authors of these studies, counterintuitive at the time, since the use of such link information was (and still is) prevalent by commercial web search engines. Indeed, first successful verifications of the benefit of using link information came only as both the collection size was further increased and the nature of the task was changed to better reflect the real-world use of such an enlarged collection (Singhal and Kaszkiel 2001).

[7]Note the distinction between matching *some* relevant items versus matching *all* relevant items. Users that are interested in the former are usually termed "precision-oriented," while those interested in the latter are referred to as "recall-oriented." Information needs of both these user groups are equally valid, but need to be addressed differently.

3 The Changing Matching Task

3.1 The Retrieval Problem and Its Interaction with Collection Size

In our pursuit to understand the influence of collection size on the effectiveness of Information Retrieval systems, let us first formalize the main task of such a system. In the literature, this task is termed the "retrieval problem." Robertson et al. (1982) define the problem thus: "... the function of a document retrieval system is to retrieve all and only those documents that the inquiring patron wants (or would want)." Some variants of this definition call the "documents the (...] patron wants" the "relevant documents." Our aim is therefore to understand how this task changes fundamentally in nature as a document collection grows.

As we already stated, it goes beyond the intent of the chapter to deliver a definitive exploration of all effects of collection size on retrieval effectiveness and its associated aspects (such as term weighting, term distribution, etc.).[8] Instead, our intention is twofold: firstly, we want to experimentally gather indications of a changing nature of the retrieval problem by working with a suitable test collection, and secondly, most importantly, we want to discuss the best practices we derive from designing retrieval applications operating on small data for the past two decades.

The discussion benefits from a systematic look at the functioning of an information retrieval system. From a high-level perspective, an Information Retrieval system has to solve the following two subtasks (Peters et al. 2012):

1. *Indexing*: Transform/process the query entered by the user, as well as the items (typically documents) that are made accessible by the system, into such a form that they become comparable.
2. *Matching*: Rank the items through comparison with the processed query in a way that is most efficient and effective to satisfy the user's original information need.

Both the "indexing" and "matching" phases are influenced by difficulties ensuing from the general setup in which we operate: the query may be a less than optimal expression of the original information need, and the retrieved documents still need to be digested by the user after the retrieval phase to gather the actual information needed.

We are now interested in investigating both these subtasks for the influence that document collection growth has on them. We will base our investigation on our basic assumption that *with an increase in collection size there normally follows an increase in redundancy*. On the basis of this assumption, the difficulties encountered in solving the indexing subtask, which revolve around the task of bridging between the language and understanding of the querier and the document author, are partly alleviated: we can expect to find many different formulations of the same fact as the

[8] For the most in-depth treatment of the theoretical aspects, we again refer the reader to Hawking and Robertson (2003).

collection grows. The problem of picking the right terms or the right word forms should thus be much less pressing.

When considering the "matching" subtask, common practice is to rank the matched items by decreasing probability of their (estimated) relevance to the user's query.[9] This ranking is guided by what in the Information Retrieval literature is termed a "weighting scheme," which assigns a score ("retrieval status value" RSV) to each retrievable item (document). Most popular weighting schemes [tf.idf and successors (Singhal et al. 1996), Okapi/BM.25 (Walker et al. 1998), Divergence from Randomness (Amati and van Rijsbergen 2002), and language modeling (Hiemstra and de Jong 1999)] can be represented as a combination of the same building blocks. Essentially, these blocks are: a count of the local occurrences of terms in documents (*term frequency*, tf), a count of the global occurrences of a term in the collection (*document frequency*, df), and some notion of a "*document length*." The underlying assumption always being that the occurrence of the (query) terms is indicative for the relevance of the document.

It is thus interesting to analyze the influence that term occurrence has on the "hardness" of the matching subtask. To approach this question, we consider Zipf's law (Schäuble 1999). The law states that when compiling a list of all terms in a text corpus sorted by their descending order of the number of occurrences, in that list it holds that

$$\mathrm{cf}(\varphi_i) * r_i = \mathrm{const.}$$

where $\mathrm{cf}(\varphi_i)$ (collection frequency) is the total frequency of the term over all documents in the corpus, and r_i is the rank of the term in the list.

Put another way, there are very few terms that occur very frequently and very many terms that occur very infrequently. In fact, when analyzing text corpora it is not uncommon to find that more than half of the surface word forms occur only once or twice. Words that occur only once in a body of text are usually referred to as "hapax legomenon" in linguistics. Evidently, it is hard to estimate a true weight for a term based on very few occurrences: in such cases, it is difficult to distinguish whether the occurrence of a term is meaningful or just "noise." Increasing the collection size should have the effect of "shifting" the Zipf curve: terms will have an increased number of occurrences overall, and weighting thus becomes more stable. Note, however, that inevitably a number of new word forms or terms will be introduced by the increase, which in turn will have very few occurrences. That is, the fundamental problem with the Zipf curve and the behavior of language, in that many terms are used only infrequently, does not change—but in absolute terms, we have more terms that reach a "useful" level of term occurrences.

[9] This modus operandi is supported by the so-called Probability Ranking Principle. See for example, Robertson (1977).

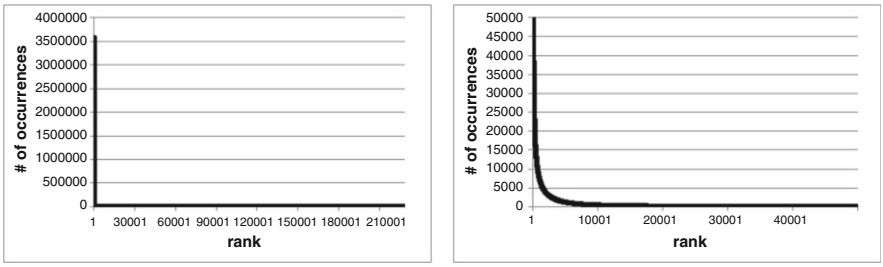

Fig. 13.1 Plot of word occurrence frequencies in the *LA Times* 1994 news article collection. For the purpose of this plot, we look at unique character strings. On the left, all such words (~228 k) are plotted: it is nearly impossible to discern more than an "L" shape: very few words are extremely frequent, while the vast majority of words are very rare. The right graph shows a magnified crop of the full graph on the left: a bend becomes apparent

3.2 Hapax Legomena

We have studied this behavior with the help of a document collection used in the CLEF evaluation campaigns for Information Retrieval systems (Peters and Braschler 2001), the "*LA Times* 1994 collection." This collection contains 1,13,005 news articles in English language covering the whole of the year 1994. Total size is approximately 420 MByte of text (including minimal structural formatting). Since the news articles have a timestamp, we can analyze the "stream" of news articles with regard to term occurrence. To this end, we start with 1 month of articles (January) ("small" collection), expand to 6 months (first half of 1994) ("half" collection) and finally process the entire year ("full" collection).

As expected, word occurrence in the *LA Times* articles follows Zipf's law. Over the entire year, ~228 k different "words" or rather "word forms" are used (we are strictly looking at "unique character strings" here, not at words in a linguistic sense).[10] A plot of the word occurrence frequencies is shown in Fig. 13.1.

How does the number of words develop as we follow the "stream" of articles? If we start with the month of January, we find more than 79 k unique character strings for that 1 month. As we progress to include the whole first half of 1994, that number grows to ~167 k; a substantial increase, but clearly below linear growth: approximately equal to 27 k new word forms per month. Expansion to the whole year further slows this growth to ~19 k new word forms per month. This slowdown in the number of new word forms cannot be attributed to the overall number of articles per month or their length; we observe no meaningful change as the year progresses in

[10]Terminology is tricky in these cases. A number of concepts we use here have multiple or vague definitions depending on the viewpoint of the sources. For this chapter, by "word" or "word form" we denote a unique character string, that is, for our discussion, "house" and "houses" are two "words." The occurrence of such a word is often denoted a "token" in IR literature. A "term" then is a (potentially) normalized representation of a token, as produced during the indexing phase of an IR system.

either of these measures (as should be expected). The slowdown is rather due to the fact that with adding new articles, previously seen words are repeated.

This potential shift for words from having few observations to having a larger number of occurrences is interesting, as it may provide us with more evidence to weigh the corresponding terms for retrieval. In our "small" collection, 24 k out of 79 k words appear only once in the entire body of text (collection frequency cf. = 1; again, this is conformant with what we expect from Zipf's law—a lot of words are used very infrequently). Another way to look at this slanted distribution of word occurrences is that 29 k words appear in one article only (document frequency df = 1). The numbers increase to 51 k words with cf. = 1 and 64 k words with df = 1, respectively, for the "half" collection. In the "full" collection, we find 72 k cases of words with cf. = 1 and 91 k cases of words with df = 1. While the number of hapax legomena therefore grows with increasing collection size, of course the number of non-hapax "words" grows as well. The proportion of hapax legomena compared to all words hovers around 30–31% and changes little as we vary collection size.

Given the above focus on term weighting, most interesting is how many "words" that occur only once in the "small" collection gain additional observations. We found that this is the case for ~18 k words, thus around three quarters of all hapax legomena, which is an indication that term weighting could benefit from the collection size increase. The main problem with this line of argument is that influence of term weighting is difficult to measure directly, as term weighting is not an end in itself.

3.3 Term Weighting Depending on Collection Size

Judging from our observations on term occurrence, we would expect that term weighting should improve when collection size increases. All other parameters remaining equal, we would therefore hope that retrieval effectiveness, that is, the ability of the system to retrieve relevant items, will increase.

Retrieval effectiveness is usually measured through the two measures "precision" and "recall." Precision is defined as the number of relevant documents retrieved in relation to the total number of documents retrieved, while recall is defined as the number of relevant documents retrieved in relation to the total number of known relevant documents. A system with high precision values will retrieve few irrelevant items, while a system with high recall values will miss few relevant items. For details on how to calculate the measures on ranked lists, see, for example, Schäuble (1999).

The two measures frequently stand in conflict: a high precision implies a system that avoids retrieval of irrelevant documents, and thus likely uses a conservative matching strategy: only the best matching items are retrieved. As a consequence, relevant items may be missed as well, and recall suffers. Conversely, high recall implies an aggressive matching strategy that may introduce more noise into the

Table 13.1 Runs on both the "small" and "full" collection, no special indexing. P@1 increases as the collection size grows, MAP decreases with growing collection size

	Small collection		Full collection	
	P@5	MAP	P@5	MAP
Short queries	0.2231	0.3195	0.4154	0.2178
Long queries	0.2846	0.4278	0.5154	0.3098

result. Precision and recall values are thus often reported as a precision–recall curve, tracking the changes of precision as recall increases.

It is oftentimes assumed that users on large collections care mainly about precision. Test collections for IR system evaluation such as TREC VLC (Hawking and Thistlewaite 1997) use mainly precision-related measures for their comparisons. Please note that determining recall values becomes prohibitively difficult as collection size increases; conceptually, the whole document collection (every single document) needs to be scanned to obtain the true number of relevant documents for each query.

For similar considerations, Hawking and Robertson (2003) use Precision@10 (P@10; precision after ten retrieved documents) as the measure for their discussion. They find that Precision@10 should go up as collection size increases, that is, the number of irrelevant items in the top ranks of the result list decreases. We would—considering our discussion in this chapter so far—expect to see a similar effect when toggling between our "small" and "full" collections. It is, however, difficult to make a direct comparison, as our "small" collection has only approximately six relevant items on average per query—which would directly influence the P@10 values.[11] Trec_Eval,[12] the tool commonly used for reporting effectiveness measures for ranked retrieval, offers Precision@5 (P@5) as the best alternative, which we use for our following discussion. Indeed, using the queries from the CLEF campaign, we measure an 86% increase when using short, keyword-style queries[13] and an 81% increase when using longer, sentence-style queries as we go from the small to the full collection (see Table 13.1). These differences are statistically significant at significance level $\alpha = 0.01$.

This result needs to be interpreted with care: as Hawking and Robertson (2003) point out, such results may well be an artifact of the measure and the behavior of the system: if a system has a typical precision of 40% at a recall level of 10%, the precision at a fixed cutoff point (such as, in their case, 10 documents) is highly likely to increase with collection size, since the absolute number of relevant items can be expected to rise. Thus, it remains unclear if the better retrieval effectiveness we observe can really be attributed to better term weighting. In any case, there is a

[11] The system could never attain a perfect score of 1.0.

[12] Obtainable at https://github.com/usnistgov/trec_eval

[13] The short queries have an average length of 2.8 (key-)words, whereas the long queries have an average length of 18.5 words. These numbers will vary slightly depending on indexing measures taken, such as stemming or stopword elimination (for more on this, see below).

measurable improvement in performance with respect to Precision@5 with increased collection size.

To underscore the caveat with respect to retrieval effectiveness, we can alternatively look at the "mean average precision" measure (MAP) (again reported in Table 13.1). This measure averages precision values over all recall levels (Schäuble 1999). Like Precision@ x, MAP is not without problems: precision and recall are effectively merged into one measure, which makes interpretation difficult when a user expresses a clear preference for one of the two criteria (as is often the case for very large collections, see above). Still, the measure is popular, if only for essentially providing a way to score the retrieval effectiveness with a single number. When looking at MAP numbers, we see that performance *falls* by 27–32% depending on query length when we increase collection size. Significance testing of these results paints a mixed picture: while the result on long queries is statistically significant at significance level $\alpha = 0.05$, the result on short queries falls just shy of this threshold.[14]

We can thus refine our observations with regard to the difficulty of the retrieval problem with respect to increasing collection size:

- As collection size increases, it becomes easier to match at least *some* relevant items.
- As collection size increases, it becomes harder to match *all* relevant items.

3.4 Impact of Collection Size on Vocabulary

Our investigation on the influence of collection size with regard to term weighting has addressed mainly the matching phase of retrieval, that is, how the similarity between query and retrievable item (document) should be determined. As we have pointed out earlier, an indexing phase precedes the matching phase. In essence, we have to make things comparable first before we can calculate similarities. The indexing phase thus extracts the terms, and normalizes them to an appropriate degree, to alleviate problems originating from linguistic phenomena such as morphology (word forms, such as singular/plural, past/present, etc.) and synonymy/homonymy. Authors have considerable freedom on how to express their thoughts in natural language: while a user may be looking for a "good bank," the author may have written about "reputable financial institutes." We claim that these differences are particularly problematic when collections are small. Additional redundancy in large collections may alleviate the problem, as many different formulations of the same information may be present—the user will at least match *some* relevant items, regardless of actual formulation.

[14]For all significance tests, we used a tool based on the findings of Smucker et al. (2007).

To explore this assumption, we have measured how many different word forms per "base form"[15] are used in the "small" and "full" collections. To carry out this analysis, we need a mechanism to link the different word forms to their base form, that is, we conflate word forms "with the same meaning" to a set.[16] We have a slightly different focus than in linguistics, as we are less interested whether word forms are related in a linguistic sense and more whether a conflation of two word forms increases the number of matches in a way that leads to better retrieval effectiveness. Consequently, in Information Retrieval, a "stemmer" is often employed to deal with issues of morphology. A stemmer is a rather crude, rule-based component that tries to strip suffixes and prefixes from word forms in order to reduce them to "stems" [for a good overview, see Hull (1996)]. In a stemmer, we accept output that is potentially incorrect from a linguistic perspective.[17] By running a stemmer on the vocabularies derived from our different sized collections, we can determine how large the groups of word forms are that get conflated. This, in turn, indicates how much leeway a querier has to match documents containing any one of these forms in the absence of a stemmer.

We find that while there are on average 1.36 different word forms per stem in the "small" collection, 1.68 word forms (+24%) can be found in the "full" collection. This indicates that it is indeed easier to match at least *some* relevant items as the collection grows, whereas we need additional measures, such as the use of a stemmer, when a collection is smaller.[18]

The latter conclusion can be put to a simple test by running our set of CLEF queries on the "small" and "full" collections with stemming toggled on and off. The results we obtained from this test were mixed. We did, however, find by far the biggest benefit of stemming when queries are short and the collection small, which indeed is the configuration where we expect that finding the exact right form becomes hardest for the querier. In this situation, we measured an increase of 22% when switching stemming on. In all other cases (longer queries and/or larger collection), the difference between stemming and not stemming was negligible. Note, however, that the effectiveness of stemming in English has been debated before [e.g., in Harman (1991)]; it is usually claimed that stemming has much more impact for other languages, which have a richer morphology, such as German (Braschler and Ripplinger 2004).

[15]This could be the "lemma" as used in linguistics, but can also be a different representative, as output by a stemming process – see below.

[16]The analog in linguistics is called "lexeme."

[17]Consider, for example, the word "footer," which denotes a repeated block of text that appears at the bottom of all pages of a document. Clearly, the word is derived from "foot." A conflation of the two words is however very likely to introduce mainly noise into the search results, and should therefore be avoided. On the other hand, being rule-based, stemmers typically cannot handle all the irregularities of natural languages. As a result, word forms that are related in meaning may not conflate.

[18]We would expect similar effects for linguistics phenomena such as synonyms.

4 Example Retrieval Applications Operating on Small Data

We have so far demonstrated that indications are such that both the indexing and matching phases benefit from an increasing size in the collection: it becomes easier to find at least *some* relevant items, due to more variety in the vocabulary used to represent information, and due to more detailed term occurrence statistics. How can we address the difficulties of retrieving documents from small collections?

For the remainder of the chapter we want to briefly outline two example retrieval applications that operate on Small Data per our definition. The two applications serve to illustrate best practices we have derived from nearly two decades of work on similarly sized document collections.

The first application, "Stiftung Schweiz," allows access to descriptions of all (charitable) foundations in Switzerland. The service addresses both applicants that are looking for the right place to submit their proposals, and the foundations themselves, which can publicize their strategies and programmes in the hope of attracting the right audience (Buss and Braschler 2015). In all, descriptions of approximately 13,000 foundations can be accessed—a very comprehensive, helpful index for applicants, but hardly a document collection of "Big Data" proportions. We describe a development version of the application that was built during a joint innovation project with the company running the service.[19] The operational service has since slightly deviated from this form.

The consequences of the "Small Data" nature of such a search application can be illustrated with a simple search: consider a fictional applicant that proposes to preserve traditional beer brewing handicraft. Such an applicant may start the search for funding opportunities very much in "web-search style" by inputting "bier" (German for "beer") as initial query.[20] Let us stick with the web search comparison for a minute. Suppose the identical query is sent to the Google Web Search service.[21] At the time of the composition of this article, this query returns nearly 80 million hits. Clearly, there is not much urgency to generate even more hits through measures such as stemming. We also certainly have ample occurrences of the word to base term weighting on. How does this compare to what we experience on the data underlying the "Stiftung Schweiz" application? In fact, *no* description of any foundation contains the word "bier." If we do an "exact" matching we will either get no matches at all, or will at best return some irrelevant matches if we extend the search to additional fields such as persons involved in the foundations, since there actually is a person with last name "Bier" in the collection. Both approaches are not helpful to the information need of the applicant.

---

[19]The application can be publicly accessed at www.stiftungschweiz.ch. The development of the search mechanisms has been partially funded by the Swiss agency CTI under grant no. 15666.1.

[20]It has been reported numerous times that users tend to input very short queries in web search. Spink et al. (2001) report an average query length of 2.1, with approximately 27% of all queries consisting of a single keyword.

[21]www.google.ch for Switzerland.

How do we address this more complex matching problem? As demonstrated in our analysis of the nature of retrieval on small versus big document collections above, there are essentially two ways to approach the problem, both of which were explored during the work on the application. We have to help the applicant to find the right terms (vocabulary problem) and use the right word forms (morphology problem). We have demonstrated that vocabulary grows with collection size, making the task of picking terms that match easier. We cannot easily grow the collection in this case,[22] but we can extend the query—increasing the number of query terms should consequently increase the chances of some of them retrieving matches. Such query expansion can be implementing by allowing users to up-vote and down-vote items from the result list. Based on these votes, the application automatically derives new query keywords and calculates a refined result list. The implementation is based on a process called "relevance feedback" [see, e.g., Peters et al. (2012)]. As a consequence, the application is built to handle (very) long queries—it benefits from users sharing as much information as possible about their information needs. Applicants can even upload entire proposals in place of entering keywords for a query. The proposal in its entirety is then matched to the descriptions of the foundations.

The morphology problem can be addressed by using a suitable stemming component, as discussed above. The German language allows the formation of very complex compound nouns—for example, "phone number" in German becomes "Telefonnummer," and "soccer world cup" becomes "Fussballweltmeisterschaft." The same information can nearly always be paraphrased by using the individual components of the compound: "Nummer eines Telefons," "Weltmeisterschaft im Fussball," etc. A component that splits the compounds ("decompounder") is thus combined with the German stemmer.

When we use all these measures, the applicant actually gets a more helpful result: it turns out that there is in fact a foundation associated with the Swiss brewery association—see Fig. 13.2. The description of this foundation does not contain the word form "Bier," but the application is able to derive a match from "Bierbrauervereins"[23] (literally "of the brewery association"). Would we need to go to these lengths in the Google example? Clearly not. Here, however, the availability to generate these matches is crucial.

To further illustrate how query expansion can alleviate the vocabulary problem, we consider a second retrieval application, named "Expert Match."[24] This application was built for in-house usage by a recruiting company to find candidates for highly specialized positions where only a small number of suitable candidates can be

[22] Although preliminary work on enriching the descriptions was actually carried out.

[23] Although it may appear so on surface for this specific example, substring matching, for example, simply detecting the character string "bier" in the word, is no alternative here, since it would add a lot of noise, such as place names starting with "Bier. . . ."

[24] The development of this retrieval application has been partially funded by the Swiss agency CTI under grant no. 13235.1.

Fig. 13.2 The "Stiftung Schweiz" application matches an item that does not contain the keyword of the query, based on stemming and decompounding components

expected to exist. The specificity of the positions implies that the CVs of potential candidates rarely contain the exact terms to match information such as job titles—rather, information about degrees and past employers is listed. While pertinent, this information can only provide a tenuous link to the position that is to be filled. The application contains the profiles of several tens of thousands of professionals, so there should be suitable candidates for many needs, but we are very far from Big Data territory where we could expect to obtain matches for arbitrary formulations.

Again, we address the problem by introducing "relevance feedback" to the application: the recruiter starts by providing as much information as possible about the position: job advertisement, company description, potentially even information about predecessors. This leads to a very comprehensive, but potentially fairly noisy list of candidates. The recruiter can now assess as many of the candidates with respect to their suitability for the position as is desired, and then start a new, refined search. The application automatically constructs a *long* query that tries to encapsulate both explicit criteria derived from the original description of the position as well as implicit criteria as extracted from the information on the candidates that were assessed. Crucially, this way candidates can be found that share few or none of the obvious keywords in the job description—we observed that often connections were established based on the companies with which other interesting candidates have been previously involved. Figure 13.3 shows a screenshot of the simple interface the application uses: the recruiter can traverse a list of candidates that have been ranked by their estimated probability of being suited to the position, and then decide by a drag-and-drop operation on whether or not the candidate is interesting. Based on this, multiple new, refined search cycles can be started.

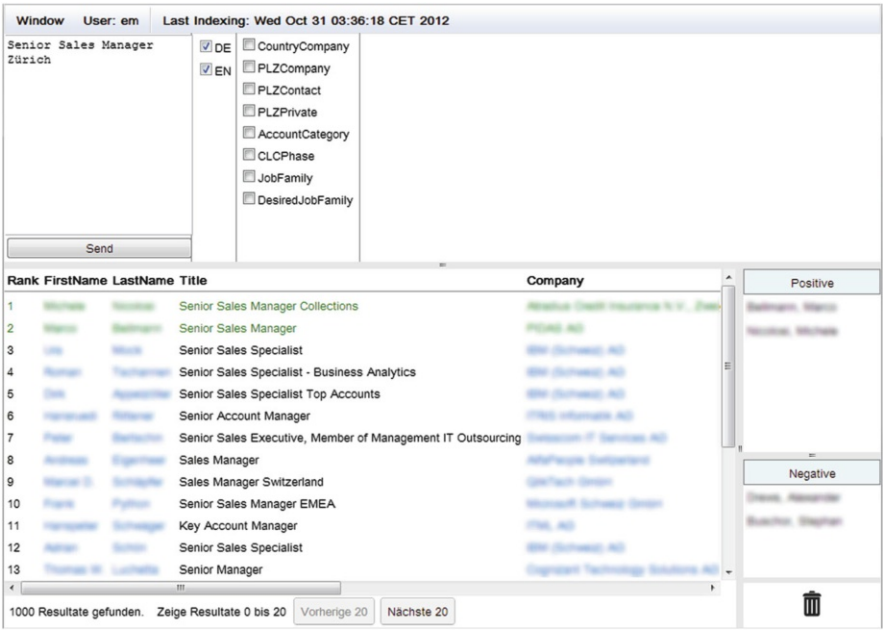

Fig. 13.3 The recruiter assesses candidate profiles by dragging the respective entries to the positive or negative "bins." Names have been blurred for privacy reasons

5 Conclusions: Best Practices for Retrieval on Small Document Collections (Small Data)

We have argued in this chapter that while there is ample justification to be excited about the ability to process huge amounts of data to gather potentially new insight ("Big Data"), there are many interesting (search) problems that operate on document collections that are far smaller. Counterintuitively, difficulty does not necessarily follow size: for example, while more hardware may be needed to process the sheer amount of data as the size of a document collection grows (making processing more difficult on that count), it is often demonstrably easier to match *some* relevant items during retrieval on that bigger collection.

Consequently, much like Information Retrieval on Big Data has its unique challenges, specific measures should be taken when a document collection is small. We have concentrated on the observation that with less factual redundancy, it becomes both harder to establish matches between keywords by the searcher and word (forms) used by the original authors, as well as weigh them accurately for calculating similarity scores. Our work on two retrieval applications, "Stiftung Schweiz" and "Expert Match" has served as examples of search on small document collections, and forms the basis for conclusions that we can derive from having built many more such applications.

Specifically, we venture the following best practices:

- Increase the occurrences per term for better term weighting by using stemming (and, in languages such as German that have a rich compound formation process, use decompounding).
- Synthesize new, additional keywords for the query from known, relevant items ("relevance feedback"). This helps to match items that do not share any of the original keywords, possibly due to synonymy issues.
- Increase the number of matches between keywords of the query and documents by (again) using stemming.
- Enrich the target items/documents with additional information, for example, from external sources such as the web.

Additional measures may include the provision of facilities to browse the information (e.g., through categorization), as well as measures that allow the crossing of language and/or media boundaries, thus potentially opening up more items for retrieval.

Most of our recommendations do not readily translate to retrieval applications operating on very large document collections, where the searcher's focus is on high precision (i.e., retrieving a few, highly ranked, relevant items). Both relevance feedback and stemming/decompounding are computationally relatively costly, which hinders adoption in such cases. As we have discussed, there is also usually less urgency due to higher likeliness of variety in how information is represented.

Acknowledgments The retrieval applications "Stiftung Schweiz" and "Expert Match" were partially funded by Swiss funding agency CTI under grants no. 15666.1 and no. 13235.1.

References

Amati, G., & van Rijsbergen, C. J. (2002). Probabilistic models of Information Retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS), 20*(4), 357–389.

Baeza-Yates, R., & Riebeiro-Neto, B. (2011). *Modern information retrieval*, 2nd edn. New York: ACM Press.

Braschler, M., & Gonzalo, J. (2009). *Best practices in system and user oriented multilingual information access*. TrebleCLEF Consortium, ISBN 9788888506890.

Braschler, M., & Ripplinger, B. (2004). How effective is stemming and decompounding for German text retrieval? *Information Retrieval, 7*(3–4), 291–316.

Braschler, M., Rietberger, S., Imhof, M., Järvelin, A., Hansen, P., Lupu, M., Gäde, M., Berendsen, R., Garcia Seco de Herrera, A. (2012). *Deliverable 2.3. best practices report, PROMISE participative laboratory for multimedia and multilingual information systems evaluation*.

Buss, P., & Braschler, M. (2015). *Stiftungsschweiz.ch Effizienzsteigerung für das Stiftungsfundraising*. In Stiftung & Sponsoring, Ausgabe 5|2015.

Google. (2008). https://googleblog.blogspot.ch/2008/07/we-knew-web-was-big.html

Google. (2016). http://www.google.com/insidesearch/howsearchworks/thestory/

Harman, D. K. (1991). How effective is suffixing? *Journal of the American Society for Information Science, 42*(1), 7.

Harman, D. (1993). *Overview of the first TREC conference, Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

Hawking, D., & Robertson, S. (2003). On collection size and retrieval effectiveness. *Information Retrieval, 6*(1), 99–105.

Hawking, D., & Thistlewaite, P. B. (1997). Overview of TREC-6 very large collection track. In *Proceedings of The Sixth Text REtrieval Conference, TREC 1997* (pp. 93–105), NIST Special Publication 500-240.

Hawking, D., Craswell, N., & Thistlewaite, P. (1998). Overview of TREC-7 very large collection track. In E. M. Voorhees, & D. K. Harman (Eds.), *Proceedings of the Seventh Text REtrieval Conference (TREC-7)* (pp. 91–103). NIST Special Publication 500-242.

Hawking, D., Voorhees, E., Craswell, N., & Bailey, P. (1999a). Overview of the TREC-8 web track. In E. M. Voorhees, & D. K. Harman (Eds.), *Proceedings of the Eighth Text REtrieval Conference (TREC-8)* (pp. 131–150). NIST Special Publication 500-246.

Hawking, D., Craswell, N., Thistlewaite, P., & Harman, D. (1999b). Results and challenges in Web search evaluation. *Computer Networks, 31*(11–16), 1321–1330.

Hiemstra, D., & de Jong, F. (1999). Disambiguation strategies for cross-language information retrieval. In Abiteboul, S., Vercoustre, A. M. (eds) *Research and advanced technology for digital libraries*. ECDL 1999. Lecture Notes in Computer Science (Vol. 1696, pp. 274–293). Berlin: Springer.

Hull, D. A. (1996). Stemming algorithms: A case study for detailed evaluation. *JASIS, 47*(1), 70–84.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Peters, C., & Braschler, M. (2001). European research letter: Cross-language system evaluation: The CLEF campaigns. *Journal of the Association for Information Science and Technology, 52* (12), 1067–1072.

Peters, C., Braschler, M., & Clough, P. (2012). *Multilingual information retrieval*. Berlin: Springer.

Robertson, S. E. (1977). The probability ranking principle in IR. *Journal of Documentation, 33*(4), 294–304.

Robertson, S. E., Maron, M. E., & Cooper, W. S. (1982). Probability of relevance: A unification of two competing models for document retrieval. *Information Technology – Research and Development, 1*, 1–21.

Schäuble, P. (1999). *Multimedia information retrieval*. Kluwer Academic.

Singhal, A., & Kaszkiel, M. (2001). A case study in web search using TREC algorithms. In *Proceedings of the 10th International Conference On World Wide Web* (pp. 708–716). ACM.

Singhal, A., Buckley, C., & Mitra, M. (1996). Pivoted document length normalization. In: *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '96)* (pp. 21–29). New York: ACM.

Smucker, M. D., Allan, J., & Carterette, B. (2007) *A comparison of statistical significance tests for information retrieval evaluation, CIKM '07*. Portugal: Lisboa

Spärck Jones, K., & Willett, P. (1997). *Readings in information retrieval*. Morgan Kaufmann.

Spink, A., Wolfram, D., Jansen, M. B., & Saracevic, T. (2001). Searching the web: The public and their queries. *Journal of the Association for Information Science and Technology, 52*(3), 226–234.

Walker, S., Robertson, S. E., Boughanem, M., Jones, G. J. F., & Spärck Jones, K. (1998) Okapi at TREC-6, automatic ad hoc, VLC, routing, filtering and QSDR. In E. M. Voorhees, & D. K. Harman (Eds.), *The Sixth Text REtrieval Conference (TREC-6)* (pp. 125–136). NIST Special Publication 500-240.

Chapter 14
Narrative Visualization of Open Data

Philipp Ackermann and Kurt Stockinger

Abstract Several governments around the globe have recently released significant amounts of open data to the public. The main motivation is that citizens or companies use these datasets and develop new data products and applications by either enriching their existing data stores or by smartly combining datasets from various open data portals.

In this chapter, we first describe the development of open data over the last few years and briefly introduce the open data portals of the USA, the EU, and Switzerland. Next we will explain various methods for information visualization. Finally, we describe how we combined methods from open data and information visualization. In particular, we show how we developed visualization applications on top of the Swiss open data portal that enable web-based, interactive information visualization as well as a novel paradigm—narrative visualization.

1 Introduction to Open Data

The idea of freely sharing open data has been around for several decades. For instance, the World Data Center[1] developed a concept for open access to scientific data in 1957–58. However, the open data movement for access to public data has only recently gained worldwide traction (Bauer and Kaltenböck 2011). One of the main drivers of the movement[2] is Tim Berners-Lee, who is often considered as the father of the World Wide Web. The main goals are to make local, regional, and national data electronically available to the public and to lay the foundations for different actors to build rich software applications upon them.

[1] www.icsu-wds.org

[2] Open Data Handbook: www.opendatahandbook.org

P. Ackermann · K. Stockinger (✉)
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stog@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_14

Fig. 14.1 List of data catalogs curated by experts around the world (http://dataportals.org/)

Table 14.1 Open data portals of the USA, the EU, and Switzerland as of April 2017

Open data portal	Provider	Number of datasets	Number of applications
data.gov	US Government	1,92,322	76
	European Union	10,702	70
https://opendata.swiss	Swiss Government	2169	30

Another important driver in the open government data movement is "The Memorandum on Transparency and Open Government" signed by US President Barack Obama shortly after his inauguration in January 2009 (Orszag 2009). The aim was to establish a modern cooperation among politicians, public administration, industry, and private citizens by enabling more transparency, democracy, participation, and collaboration. In European countries, Open Government is often viewed as a natural companion to e-government (Bauer and Kaltenböck 2011).

Figure 14.1 shows an aggregated number of open data catalogs curated by experts around the world. We can see that the major activities are in Europe and the East Coast of the USA.

Table 14.1 shows some facts about the open data portals provided by the USA, the European Union, and Switzerland. We can see that the US portal contains 1, 92, 322 datasets, while the portals of the European Union and Switzerland contain 10,702 and 2169, respectively. More interesting from our viewpoint, however, is the utilization of these datasets and thus the applications that are built upon them. The web portals currently list 76, 70, and 30 applications for the USA, the EU, and Switzerland, respectively. The applications are very diverse and are in the areas of health, finance, environment, government, etc. These exemplary applications demonstrate the great potential in harvesting open data and thus generating either new business models or new services for citizens.

Different countries pursue different strategies with Open Government Data (Huijboom and Van den Broek 2011). Whereas the emphasis of the USA is on transparency to increase public engagement, Denmark, for example, underscores the

Table 14.2 Top 10 ranking open government data by country as of 2017 (http://index.okfn.org/)

Rank	Country	Score (%)
1	Taiwan	90
2	Australia	79
2	Great Britain	79
4	France	70
5	Finland	69
5	Canada	69
5	Norway	69
8	New Zealand	68
8	Brazil	68
10	Northern Ireland	67

opportunities that open data offers for the development of new products and services. The UK explicitly mentions the use of open data to strengthen law enforcement.

The Global Open Data Index[3] annually measures the state of open government data around the world. The index measures the level of conversion to open data based on datasets provided by different areas such as national statistics, government and budget, etc. (see Table 14.2). According to the 2015 ranking, Taiwan is leading ahead of the UK and Denmark.

Open data is stored in portals under various formats such as comma separated values (CSV), PDF, or text files. However, the main storage technologies and APIs are based on two major frameworks:

- RDF (Resource Description Framework) and its query language SPARQL
- CKAN (Comprehensive Knowledge Archive Network)

RDF and SPARQL are the main technologies used for the semantic web as well as the Linked Data[4] movement. In RDF, every data item is stored as a triple of subject, predicate, and object, and enables linking objects on the web via URIs (uniform resource identifiers).

SPARQL is the query language for accessing data stored in RDF. It can be considered as the equivalent of SQL for relational databases.

RDF and SPARQL are used as the underlying technology for the open data portals of the European Union.

CKAN is an open source data catalog for storing and distributing open data developed by the Open Knowledge Foundation.[5] CKAN is used by the open data portals of the USA, the UK, and Switzerland.

In principle, the above-mentioned storage catalogs are not compatible. However, there exist SPARQL extensions for CKAN, which enable querying data stored in CKAN via SPARQL. An advantage of RDF/SPARQL over CKAN is that it is used by a much larger community, in particular by Linked Data, as discussed previously.

[3] http://index.okfn.org/

[4] http://linkeddata.org/

[5] https://okfn.org/

Moreover, CKAN is merely a storage archive solution, while RDF/SPARQL provides standards for storing and querying data embedded in a rich semantic framework.

2 Visualization Techniques

The primary goal of visual presentations of data is to enable the discovery and mediation of insights. *Data visualization* supports users in intuitively exploring the content of data, identifying interesting patterns, and fosters sense-making interpretations. Starting with a table of numbers is not an efficient way of interpreting data—an experience that is also commonly expressed in the idiom "a picture is worth a thousand words." Complex information can only be slowly digested from rows of tabular data. By using graphical representations, we leverage the fact that our visual perception and our human cognition processes are more effective (Ware 2012).

In data-driven projects, analysts often map table-based datasets to *data graphics*. Embedded data visualization tools in MS Excel or R provide standard chart types such as bar charts and scatter plots to easily generate data graphics. Visual discovery with data graphics takes advantage of the human's capacity to grasp a lot of information and identify patterns from visual representations of abstract data. Data graphics is already helpful with small amounts of data and becomes essential when datasets get bigger. Visually investigating data supports therefore discoveries driven by observation during exploratory analysis.

If the explored data sources reflect a certain complexity, analysts would like to interactively browse through the underlying databases. This demands for interactive *information visualization* (Shneiderman 1996) that enables users to ask additional questions and to dive into the underlying data within an elaborated information architecture. Because database schemas are not per se capable to evoke a comprehensible mental image, an information architecture transforms data sources toward mental models. We will provide a concrete example in Sect. 4.

After several iterations, the analyst may discover some findings worth presenting. Additional efforts are needed to make insights available to a wider audience. By providing an elaborated content structure within a semantic vocabulary and specific interaction patterns to filter, drill-down, and navigate through the information space, users better understand the presented content within an interactive information visualization.

If the underlying data sources provide the basis for several insights, analysts would like to communicate findings in a guided manner while maintaining the interactive exploration to users. Documenting the discovery of insights by guiding users through data landscapes is supported by *narrative visualization* techniques (Segel and Heer 2010). By presenting curated data exploration, users can step through a story, but also interactively explore the underlying datasets to gain their own insights. Data-driven journalism applies narrative visualization to present

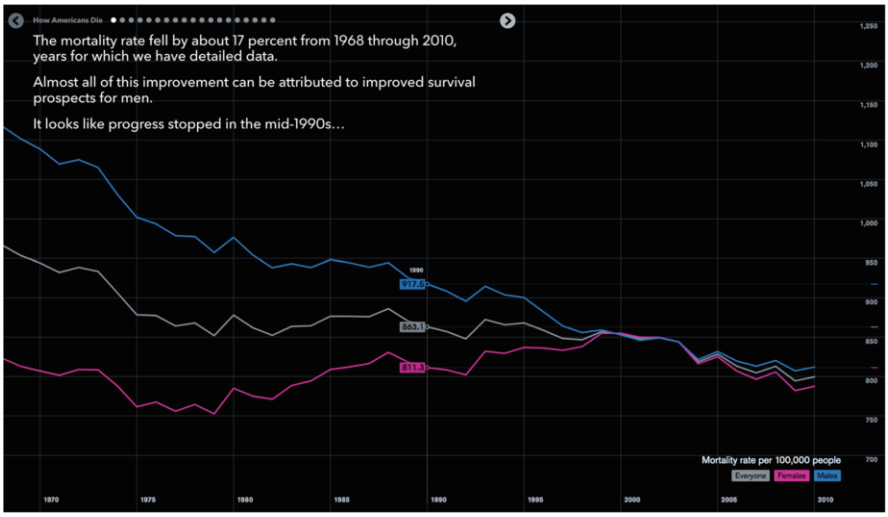

Fig. 14.2 Excerpts from a sequence in a narrative visualization (www.bloomberg.com/dataview/2014-04-17/how-americans-die.html). The figure shows the declining mortality from 1968 to 2010. For men the improvement was most significant

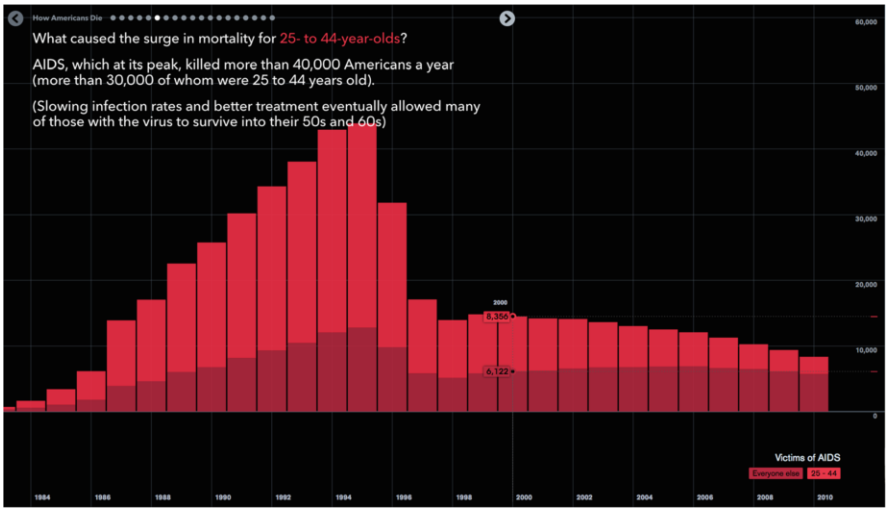

Fig. 14.3 Excerpts from a sequence in a narrative visualization (continued). The figure attributes the increased mortality rates in the 1980s and 1990s for 25–44-year-olds to AIDS

interactive visual stories. A good, illustrative example that was produced by Bloomberg is shown in Figs. 14.2 and 14.3.

Narrative visualization depicts key insights using storytelling with animations and therefore enhances data graphics and information visualization. Applications of

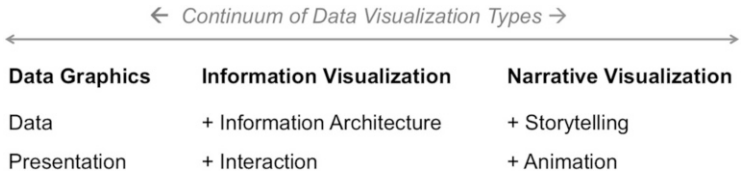

Fig. 14.4 Types of data visualizations

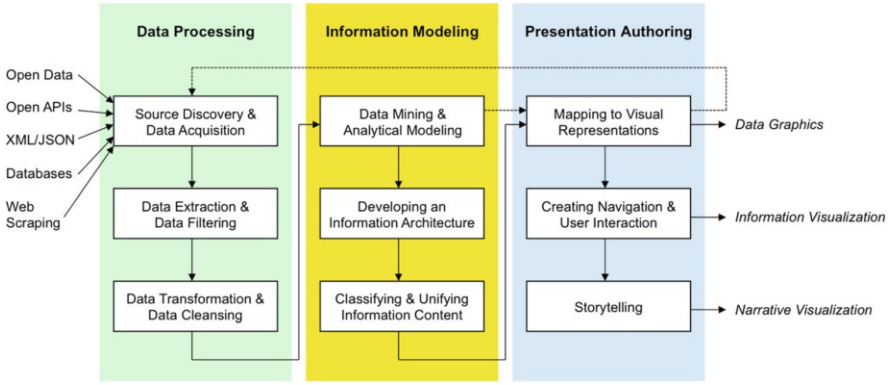

Fig. 14.5 Data visualization workflow

data visualization often use a mix of data graphics, interactive information visualization, and narrative visualization (Fig. 14.4). Applications using interactive and narrative visualizations need custom development of specific software to present findings to end users. To reach many users, such visual presentations are often deployed via the Internet or via an organization-specific intranet. Development libraries for web-based interactive data visualization such as Shiny[6] (Beeley 2016) and d3.js[7] (Bostock et al. 2011) provide core visualization technologies and many examples in the continuum of the three data visualization types.

3 Data Visualization Workflow

The workflow for developing data visualizations comprises several steps for data processing, information modeling, and presentation authoring (see Fig. 14.5). Although the workflow process is mainly sequential, typically many iterations for refining the deliverables are needed to create convincing data visualization results.

[6] shiny.rstudio.com
[7] www.d3js.org

Data Processing Phase A first step in data processing includes the discovery, identification, and acquisition of available data sources. With a certain interest and goal in mind, analysts make decisions about extracting and filtering relevant datasets. By combining different data sources their diverse content may be transformed to equal number formats and unit measures or may be converted by resampling to common time intervals. During the transformation and cleansing phase, a strategy needs to be developed to deal with wrong and missing values.

Information Modeling Phase Without knowing the content of data sources upfront, analysts explore and analyze the content with an open mindset oblivious to what exactly they are searching for. Exploring unknown datasets means interactively performing a sequence of queries to gain insight and understanding. During such data mining iterations an analytical model is elaborated by calculating statistical results that enhance the raw data. Additionally to the analytical model focused on numerical calculations, textual data needs to be brought into an information architecture (Rosenfeld and Morville 2015) by applying a unified content structure and using an explicit, typically domain-specific ontology that standardizes on used vocabulary and classifications.

Presentation Authoring Phase Once a unified information model is established, decisions have to be made how to map data to visual representations. The generation of data graphics compromises visual mappings to shapes, positions, dimensions, rotations, colors as well as the overall chart layout. If interactive information visualization should be provided, user interaction for navigation, filtering, and drill-downs need to be developed to communicate complex insights. The visual information seeking mantra by Ben Shneiderman "Overview first, zoom and filter, then details on demand" (Shneiderman 1996) requires some preprocessing in order to structure and aggregate the data to create overviews and drill-down navigation paths. If narrative visualization is aimed to additionally achieve storytelling, animations become part of the data visualization workflow.

4 Visualization for Exploring Open Data

By working through the data visualization workflow, data visualization itself is used to iteratively improve the results. Visual representations of data are helpful in gaining insights and in making evidence-based decisions to reach a convincing information presentation. Data exploration and analysis tools such as Tableau[8] and QlikView[9] provide interactive environments to inexpensively run visual analytics on a variety of data sources.

[8] tableau.com

[9] qlik.com

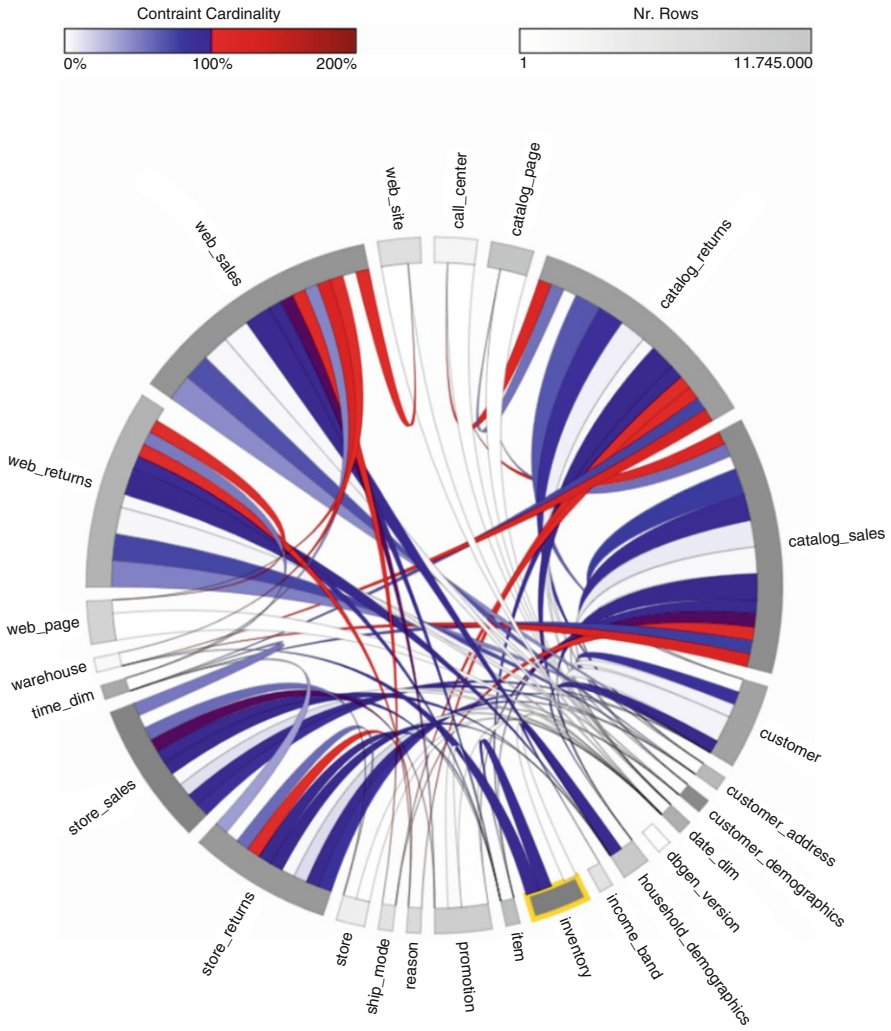

Fig. 14.6 Visual exploration of data structure and data content (Waser 2016). © 2016 Zurich University of Applied Sciences, reprinted with permission

The content of open data sources is often represented as relational tables in a database. In order to get familiar with an unknown database, Waser (2016) developed a visualization tool for SQL databases (see Fig. 14.6) to visually explore the structure and the content in parallel. The tables of the database are visually arranged in a circular layout showing the relations between tables that are color-encoded as arcs. Selecting tables and/or relations in the visual diagram automatically generates queries with corresponding joins. These visual queries are executed on the fly and the query results are shown as tables that can be sorted and filtered.

Providing such easy to use visual queries, users can investigate the data and gain insights in the structure and content of the database. On demand, histograms provide statistical evaluation of the content of table fields. Such visual data mining and exploratory data analysis tools help data analysts to make discoveries of patterns and trends driven by observations. These insights will then build the basis for interesting features worth presenting to a wider audience by developing specific information visualization applications.

5 Narrative Visualization for Presenting Open Data

We now describe how we used the Swiss open data portal to present the situation of health care services in Switzerland. Before the data was visualized, several data preprocessing steps were required.

In our case, we used the CKAN API provided by the Swiss open data platform to access the data. This API gives access to the metadata about the datasets, such as data owner, time of upload, description about the datasets, etc. However, it turns out that in most cases the metadata could not be directly used for visualization since the attribute description or units were missing. In this case, we needed to add the metadata manually.

Most of the data provided at the Swiss open data platform are in ODS-format, which cannot be directly read by the visualization framework D3.js. Hence, we needed to transform all datasets into tab-separated values (TSV) to enable visualization. Note that each additional data transformation step might introduce potential data quality problems that need to be handled explicitly by the application developers.

The next step was to integrate various datasets that contain information about heart attack, high blood pressure, body mass index (BMI), as well as public health data. An example of the entity relationship diagram is given in Fig. 14.7 (Montana and Zahner 2015). The main challenge here was that the datasets and the time ranges are of different granularity. Hence, we needed to standardize and harmonize the data before we could link it with other datasets. These are typical data warehousing tasks that all companies need to perform when they integrate data from various, heterogeneous datasets.

Once these data preprocessing steps were finished, we proceeded to tackle the actual task, namely, building interesting visualization stories. In particular, we were interested in the occurrence of high blood pressure, heart diseases, and overweightness of people in various age groups. We also wanted to know the differences in females and males. The goal was to build a sequence of interactive visualizations with intuitive narratives that guide a user from one visualization page to the next one. This approach gives the users a similar experience to reading a book or an article. The added value of interactive narrative visualization is that users can follow different lines of thought in a story by interacting with the visualized content.

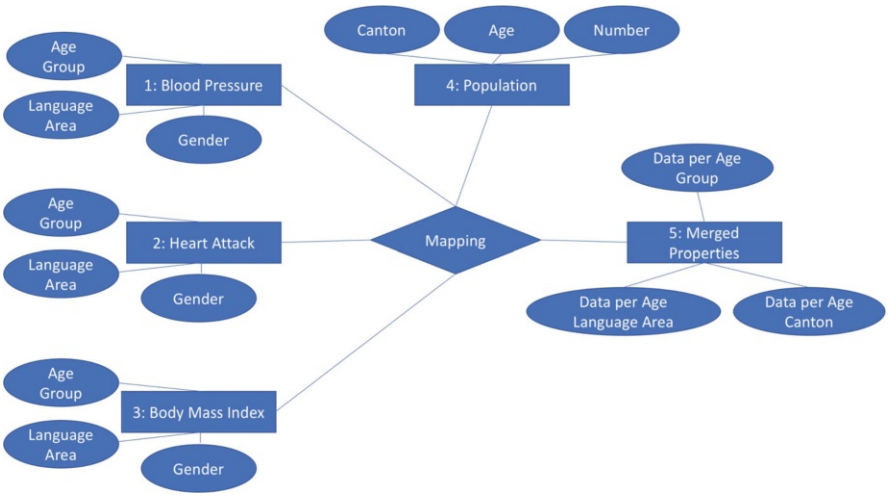

Fig. 14.7 Entity Relationship Diagram for heart attack, high blood pressure, body mass index, and people

In order to develop the visualization stories, we applied an iterative development process of using data visualization for exploring and finding relevant correlations. Additionally, we enriched the interactivity with storytelling aspects resulting in a guided user experience.[10]

In order to meaningfully interpret visualized data, it is helpful to present chosen information in a way that comparison between some grouping and/or within a historical, time-dependent context is supported. Figure 14.8 presents the differences in body mass indices between age groups. The users can hover over the visualization and get more detailed information. For instance, users might be interested in the body mass index of people in the age range of 15–24. By clicking on the respective charts, the details about the color coding of the charts are presented.

This kind of visualization makes sure that the charts are not overloaded with information and gives the user the flexibility of zooming into details interactively.

Once the user clicks on the next page of the visualization, the story continues and informs the user about mortality rates. Figure 14.9 shows the historical development of mortality reasons of Swiss citizens over the last years based on different causes such as lung cancer or car accidents. The charts show that lung cancer was the main cause of death followed by suicide. For both causes, however, we see that the mortality rates were highest around 1980 and dropped afterward. Moreover, the users can use time slides to analyze concrete time periods in more detail.

The next part of the visualization story provides more information on geographical differences about treatments in hospitals. The rationale of the narrative is that besides temporal comparison, regional differences are often of interest and may be

[10] http://www.visualcomputinglab.ch/healthvis/

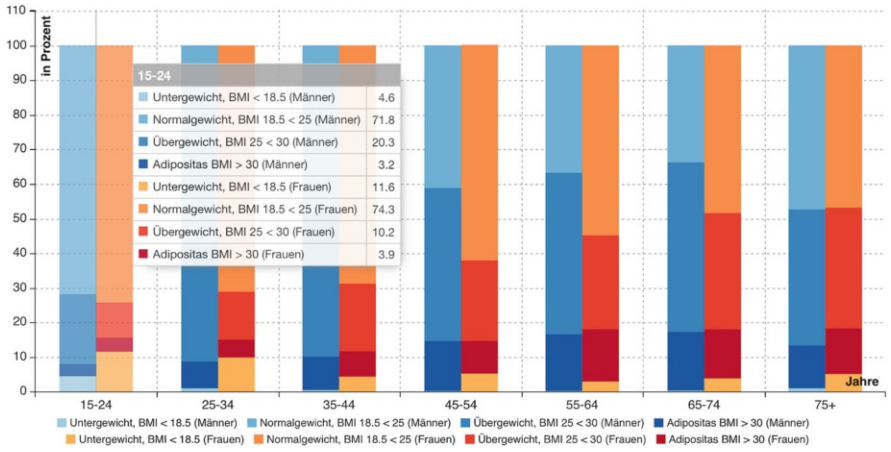

Fig. 14.8 Comparison of body mass indices between different age groups

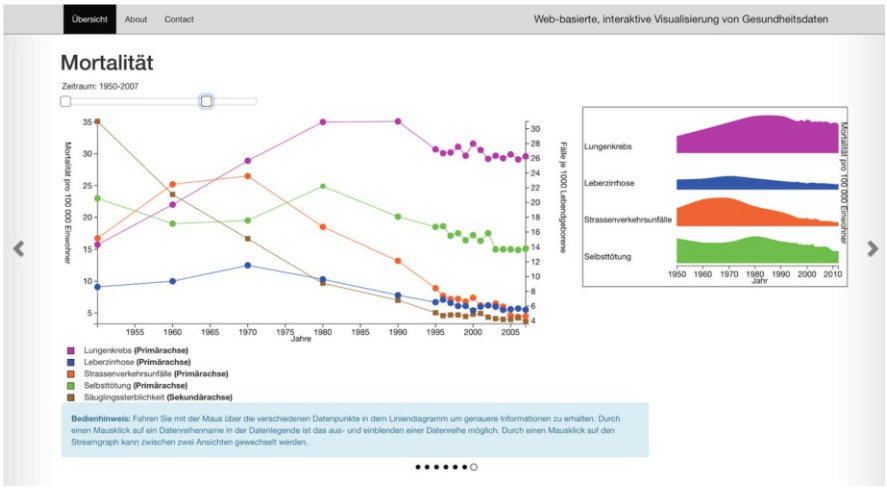

Fig. 14.9 Historical development of mortality reasons

presented in geographical maps. Figure 14.10 shows geographical differences of health care costs at hospital between Swiss cantons in a map of Switzerland. The user can interactively compare costs of geriatric, ambulant, psychiatric, and obstetric health care services. In addition, the visualizations are narrated with background information on hospital treatments as well as on interpretations of the results.

In Fig. 14.10 you see some textual description about the chart (the narrative) as well as geographical and tabular information. All visualization modes are interactive and interlinked and animate the users to explore the information from different visual perspectives.

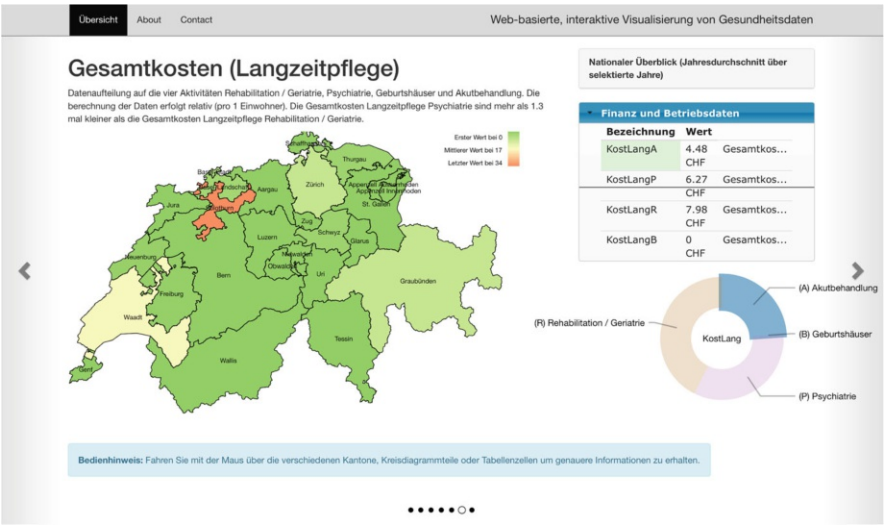

Fig. 14.10 Regional differences of health care costs within Swiss cantons

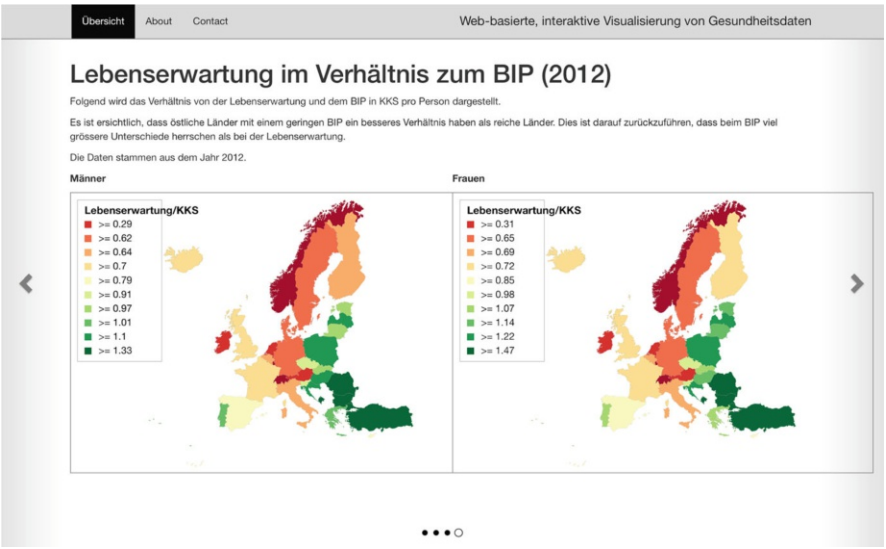

Fig. 14.11 Correlation of life expectancy and gross national product between men and women within EU countries

The next step of the narrative visualization is to analyze life expectancy on a European scale. Figure 14.11 combines regional comparison and gender differences by presenting two maps of the EU side by side. The sequence of visualization guides the user through the examination whether there is clear evidence of a positive

correlation between gross domestic product (GDP) and life expectancy. The results show that the ratio of life expectancy to the gross domestic product is better in eastern European countries than in western European ones—the main exception being Portugal. The reason is that differences in GDP are much higher than the differences in life expectancy.

The interested reader can experiment with more narrative visualizations at the following web page: http://www.visualcomputinglab.ch/healthvis/europeStory.html.

6 Lessons Learned

In the following section, we summarize the major lessons learned about developing interactive information visualizations based on open data:

- **Data preparation:** The most time-consuming aspect of information visualization is the data preparation phase, which follows similar principles as the ones used for building a data warehouse. Even though there is a vast amount of open data available, the datasets are typically loosely coupled collections of data items. Moreover, the datasets have very heterogeneous data formats and often lack a description of metadata. Hence, before data can be visualized, it often needs to be manually transformed, harmonized, cleaned, and brought into a common data model that allows easy visualization.
- **Visualization technology**: Recently, there has been a vast amount of visualization methods developed as part of the D3.js framework that enables quick prototyping. However, in order to develop consistent information visualizations, the produced charts often need to be adopted, for instance, to match dynamic scaling of axes, coherent color coding, and to enable persuasive interactivity. Hence, the high-level visualization frameworks that enable quick prototyping often cannot be used out of the box. In order to get full visualization flexibility, low-level visualization functionality needs to be customized, which requires writing much more code for achieving similar results. As a consequence, interactive information visualization and especially narrative visualization often require a development path from rapid prototyping using "out-of-the-box" data graphics toward "customized" visualizations that require some design and coding efforts.

7 Conclusions

The information visualizations shown in this chapter exemplify the benefit of gaining insight via interactive graphical presentations using open data available from public health authorities. In general, our society demands more transparency

(Taylor and Kelsey 2016). Open data coupled with information visualization will increasingly become an attractive way to communicate fact-based interrelations in economic, social, and political contexts. The awareness is increasing that it is important to provide open data to the public. Even so, it is relevant that the growing information available as open data becomes accessible via interactive visualizations in order to manage the growing complexity we are confronted with. Data scientists use their expertise in applying existing tools to process the vast amount of open data and to create interactive exploration environments for interested users and engaged citizens.

References

Bauer, F., & Kaltenböck, M. (2011). *Linked open data: The essentials*. Vienna: Edition Mono/Monochrom.

Beeley, C. (2016). *Web application development with R using shiny* (2nd ed.). Packt Publishing.

Bostock, M., Ogievetsky, V., & Heer, J. (2011). D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '11), 17*(12), 2301–2309. https://doi.org/10.1109/TVCG.2011.185.7.

Huijboom, N., & Van den Broek, T. (2011). Open data: An international comparison of strategies. *European Journal of ePractice, 12*(1), 4–16.

Montana, M., & Zahner, D. (2015). *Web-based, interactive visualization of health data*. Bachelor Thesis, Zurich University of Applied Sciences, Winterthur, Switzerland.

Orszag, P. (2009). *Open government directive*. https://www.whitehouse.gov/open/documents/open-government-directive

Rosenfeld, L., & Morville, P. (2015). *Information architecture – For the web and beyond* (4th ed.). O'Reilly Media.

Segel, E., & Heer, J. (2010). Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics, 16*(6), 1139–1148.

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages* (pp. 336–343). Washington: IEEE Computer Society Press.

Taylor, R., & Kelsey, T. (2016) *Transparency and the open society: Practical lessons for effective policy*. Policy Press at University of Bristol.

Ware, C. (2012). *Information visualization – Perception for design* (3rd ed.). Morgan Kaufman.

Waser, F. (2016). *How can data analysis be fun? Visualization concept for databases*. Bachelor Thesis, Zurich University of Applied Sciences, Winterthur.

Chapter 15
Security of Data Science and Data Science for Security

Bernhard Tellenbach, Marc Rennhard, and Remo Schweizer

Abstract In this chapter, we present a brief overview of important topics regarding the connection of data science and security. In the first part, we focus on the security of data science and discuss a selection of security aspects that data scientists should consider to make their services and products more secure. In the second part about security for data science, we switch sides and present some applications where data science plays a critical role in pushing the state-of-the-art in securing information systems. This includes a detailed look at the potential and challenges of applying machine learning to the problem of detecting obfuscated JavaScripts.

1 Introduction

Giants like Sony, Yahoo, and Anthem Inc., the second-largest US health insurance company, heavily rely on big data and machine learning systems to efficiently store and process huge amounts of data. But large enterprises are not the only ones; there are more and more startups and SMEs whose business model focuses on data-centric services and products. Unfortunately, where there is valuable data, there are also hackers that want to get it or manipulate it for fun and profit. It is therefore important that data scientists are aware of the fact that new services or data products should be designed with security in mind. Many of the popular technologies and algorithms used in their domain are not secure by default. They have to be used with care. For example, recent research showed that access to the public API of a classification service (e.g., face recognition) might be sufficient to steal or invert the underlying model (Tramèr et al. 2016). We refer to these aspects as *security of data science*, that is, issues related to the security of data science methods and applications.

On the other hand, data science methods and techniques help to address some of the most challenging problems in this field such as the management of huge amounts of log data and the identification of anomalies or other clues that might pinpoint

B. Tellenbach (✉) · M. Rennhard · R. Schweizer
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: bernhard.tellenbach@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_15

activities posing a risk for an organization. It is therefore not surprising that advancements in the field of data science lead to improvements of existing security products. For instance, becoming better at detecting anomalies in credit card transactions, network traffic, user behavior, and other types of data directly results in improved products to protect today's businesses. However, improvements to existing products are not the only outcome of the already fruitful relation of data science and security. It also led to the development of completely new solutions such as next-generation antivirus products (Cylance 2017) . We refer to these aspects as *data science for security*, that is, issues in the security domain that can be approached with data science.

Despite the many benefits of data science, there are also some drawbacks and challenges that come with the rapid evolution of the field. The short development life cycles of new methods and products, be it a software, hardware, or a data product, make it difficult to research whether these methods and products are secure or whether they introduce new security problems and flaws. It is therefore not uncommon (Pauli 2017b) that those methods and products have severe security loopholes. Furthermore, due to the increasingly more centralized storage of large amounts of data, cloud infrastructures and big data applications become attractive targets for attackers. As a result of this, the probability that such infrastructures and applications become the target of an advanced targeted attack with the goal of stealing or manipulating large amounts of data is drastically increased. An advanced targeted attack (ATA) or an advanced persistent threat (APT) (Easttom 2016) is an attack where the attackers put a lot of effort, knowledge, and time into getting and eventually also maintaining access to a system or data. Often, such attacks make use of so-called zero-day exploits. These are exploits that are not yet known to the security industry, which means it is unlikely that signature-based systems can detect them. Detection is further complicated in that the attackers try to be as stealthy as possible.

In addition, data science tools such as machine learning and the growing amount of (publicly accessible) data can also be used by cyber criminals to improve their attack methods and strategies. For example, being able to profile people based on their activities on social media and determining what type and style of social engineering attacks makes them do something they do not want to do would be very useful to cyber criminals.

In the following, we discuss the opportunities and risks of data science in more detail. First, we briefly introduce three key concepts of information security: confidentiality, integrity, and availability. Next, we present a brief overview of important topics related to security of data science and provide more details on some key topics that data scientists should consider to make (applications of) data science more secure. Then, we switch to the topic of data science for security, where we discuss examples of applications of data science in security products. This discussion includes a detailed look at the potential and challenges of applying machine learning to the problem of detecting obfuscated JavaScripts. We then conclude the chapter with a summary of topics every (security) data scientist should keep in mind when working in this field.

2 Key Concepts of Information Security

According to the Federal Information Security Management Act of 2002 (2012), the term "information security" means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability (CIA):

- *Confidentiality* requires the implementation of authorized restrictions on access and disclosure, which includes measures for protecting personal privacy and proprietary information.
- *Integrity* means guarding against improper modification or destruction, and includes information non-repudiation and authenticity.
- *Availability* finally means ensuring timely and reliable access to and use of information.

For a cloud storage provider for example, confidentiality would mean that data must be stored in encrypted form and that there is a key management scheme in place that makes sure that only authorized entities should be able to decrypt it. In its simplest form, a customer would do the encryption, decryption, and key management in his or her own trusted infrastructure and send and receive encrypted files only. However, this way, the cloud cannot look at the data and functionality such as file indexing and searching. Thus, storing the same file submitted by multiple users only once (de-duplication) cannot be done. To be able to do this, the key must be known to the cloud, which means the cloud should be trusted. To keep the attack surface small, access to the key stored in the cloud must happen on a need-to-know basis and access should be logged. Furthermore, data in transit, when transferred from or to the customer or when moved around in the cloud, should be encrypted as well.

Simply encrypting data is not enough, however, as without integrity protections, the employees of the cloud provider could still modify the encrypted files at the bit and byte level without the customer easily noticing this when decrypting the files and looking at them. And without enough resources to handle peak times or denial-of-service attacks, a customer might be cut off from the data (for some time), which could cause significant financial losses.

Hence, if information infrastructures do not have the desired properties with respect to CIA, they might not work as expected. If these infrastructures are in the big data domain, CIA issues might even be magnified by the velocity, volume, and variety of big data (Cloud Security Alliance 2013b). This will be explored in more detail in the next section.

3 Security of Data Science

In this section, we discuss challenges and solution approaches related to the security of data science methods and applications. Since any data product needs an infrastructure to run on, a piece of software that implements it, data that fuels it, and customers that feel comfortable using it, we provide a brief overview and references to more in-depth material on (1) infrastructure security, (2) software security, (3) data protection, and (4) data anonymization. Furthermore, we discuss (5) examples of attacks on machine learning algorithms.

3.1 Infrastructure Security

Infrastructure security is concerned with securing information systems against physical and virtual intruders, insider threats, and technical failures of the infrastructure itself. As a consequence, some of the more important building blocks to secure an infrastructure are access control, encryption of data at rest and in transit, vulnerability scanning and patching, security monitoring, network segmentation, firewalls, anomaly detection, server hardening, and (endpoint) security policies. Resources such as the NIST special publications series (National Institute of Standards and Technology 2017) or the CIS top 20 security controls (Center for Internet Security 2017) provide guidance and (some) practical advice. However, getting all of this right is far from easy and failing might carry a hefty price tag.

In 2007, for example, Sony was accused of having some serious security vulnerabilities. In an interview, Sony's senior vice president of information security stated: "It's a valid business decision to accept the risk of a security breach. I will not invest $10 million to avoid a possible $1 million loss" (Holmes 2007). The data theft and outage of the PlayStation network in 2011 cost Sony $171 million (Schreier 2011). The Sony Pictures hack in 2014 (Risk Based Security 2014), where personal information of employees were stolen, cost Sony $35 million. Nevertheless, as Sony stated, it is indeed a valid business decision to limit investments in security. But such decisions should be made in full awareness of the value of the assets that are at stake, especially in light of the fact that massive amounts of user accounts or data can pose a very attractive target for cyber criminals: they could steal or destroy it and then ask for a ransom to restore it, they might sell it on the black market, or misuse it to perform other crimes.

The fact that many companies have failed to secure their infrastructure can be considered an anecdotal proof that this is a complex task and should not be done without involving security experts. This is even more true when big data systems are involved, since they might require the implementation of new use-case or product-specific security measures (Moreno et al. 2016). For a checklist of what should be considered when building and securing big data systems, check out the top 100 best

practices in big data security and privacy (Cloud Security Alliance 2016). However, note that many of the best practices also apply to "normal" information systems.

Fortunately, data scientists do rarely have to build and secure an infrastructure from scratch. However, they often have to select, configure, and deploy base technologies and products such as MongoDB, Elasticsearch, or Apache Spark. It is therefore important that data scientists are aware of the security of these products. What are the security mechanisms they offer? Are they secure by default? Can they be configured to be secure or is there a need for additional security measures and tools? Recent events have demonstrated that this is often not the case.

In January 2017, 30,000 MongoDB instances were compromised (Pauli 2017b) because they were configured to accept unauthenticated remote connections. The underlying problem was that MongoDB versions before 2.6.0. were insecure by default. When installed, the installer did not force the user to define a password for the database *admin* account, and the database service listened on all network interfaces for incoming connections, not only the local one. This problem was well known and documented (Matherly 2015), but apparently, many operators of such instances didn't know or didn't care. Just one week later, the same hackers started to attack more than 35,000 Elasticsearch instances with ransomware (Pauli 2017a). Most of these instances were located on Amazon Web Services (AWS) and provided full read and write access without requiring authentication.

It is important to keep in mind that many of these new technologies are designed to facilitate easy experimentation and exploration, and not to provide enterprise-grade security by default. The examples mentioned in the previous paragraph are certainly not the only ones that illustrate this problem. A broader study in the area of NoSQL databases revealed that many products and technologies do not support fundamental security features such as database encryption and secure communication (Sahafizadeh and Nematbakhsh 2015). The general advice here is that before setting up such a technology or product, it is important to check the security features it offers and to verify whether the default configuration is secure enough. If problems are identified, they should be fixed before the product is used.

3.2 Software Security

Software security sets the focus on the methodologies of how applications can be implemented and protected so that they do not have or expose any vulnerabilities. To achieve this, traditional software development life cycle (SDLC) models (Waterfall, Iterative, Agile, etc.) must integrate activities to help discover and reduce vulnerabilities early and effectively and refrain from the common practice to perform security-related activities only toward the end of the SDLC as part of testing. A secure SDLC (SSDLC) ensures that security assurance activities such as security requirements, defining the security architecture, code reviews, and penetration tests, are an integral part of the entire development process.

An important aspect of implementing an SSDLC is to know the threats and how relevant they are for a specific product. This allows prioritizing the activities in the SSDLC. For data products, injection attacks and components that are insecure by default are among the biggest threats. Many data products are based on immature cutting-edge technology. They process data from untrusted sources including data from IoT devices, data from public data sources such as Twitter, and various kinds of user input, to control and use the data product.

For instance, if the code assembles SQL queries by concatenating user input and instructions for the database, this can turn out badly. As an example, consider the following line of code where a SELECT query is built and where *userinput* is provided by the user:

```
String query = "SELECT name, description from Product WHERE name
LIKE '%" + userinput + "%'";
```

If the user (an attacker in this case) specifies the following data as *userinput*,

```
' UNION SELECT username, password FROM User--
```

then the following query is built:

```
SELECT name, description from Product WHERE name LIKE '%' UNION
SELECT username, password FROM User--%'
```

This query is syntactically correct (note that—is used in SQL for comments, which means that the part—*%'* will be ignored by the database system) and will not only return all products, but also all usernames and password that are stored in table *User*.

The solution to this so-called SQL injection problem seems simple: input data must be sanitized so that if the data contains SQL commands, it is just interpreted as textual data and not as a potentially harmful SQL command. Another safeguard to protect from SQL injection is to use only minimal access rights for the technical database user that executes the query. This cannot completely prevent SQL injection, but in case of a vulnerability, it serves as a damage control mechanism to make sure that the amount of data that can be accessed by the attacker is limited.

However, although the mechanisms to prevent SQL injection vulnerabilities are well known, history shows that they are not used consistently in practice—even if incidents related to SQL injection regularly make it into the headlines of mass media. For instance, in 2008, SQL injection was used to steal more than 134 million credit card data records from Heartland Payment Systems (Krebs 2009). Three years later, Global Payment Systems faced the same problem and lost about $92.2 million during the incident (Krebs 2012). Even now, the problem is still around. In 2016, data of 55 million voters were stolen from Comelec, the Philippines Commission on Elections (Estopace 2016), and an SQL injection vulnerability might also have played an important role in the incident of the Panama Papers (Burgees and

Temperton 2016), where 11.5 million financial documents about offshore entities were leaked.

Clearly, SQL might not see widespread use in big data systems. New technologies such as NoSQL databases are far more prominent. However, their security history does not look much better, as a recent paper demonstrated similar issues with injection attacks as SQL (Ron et al. 2016).

One reason why it is difficult to get rid of such vulnerabilities is that preventive measures have to be considered by the developers and integrated into the code. If they are not aware of such risks and security is not a crucial part of the SDLC they are employing, it is very likely that vulnerabilities creep into the code because countermeasures are missing completely or are implemented incorrectly. There exists also no magic bullet in the sense of tools or formal proofs that can easily verify whether a piece of software is secure, although there exist tools that can detect some vulnerabilities. A good overview in this context is provided in (Software Testing Help 2017).

In general, the following steps help to address common software security problems when building a (software) product:

- Make sure that third party technology or products used are as mature as possible.
- Make sure that third party technology or products used offer a broad spectrum of security features and access controls options.
- Make sure that you have an SSDLC in place.

A good starting point to learn more about how to develop secure software are the SSDLC models of Microsoft (Microsoft 2017b) and the Open Web Application Security Project OWASP (OWASP 2017a). For more specific advice on what to consider when developing web services and web applications, OWASP (2017b) or Li and Xue (2014) offer well-suited sources. OWASP (2017b) lists the top 10 (web-) application security risks and provides technical guidance on how to test for them and how to avoid them. Five important takeaways from there are that developers should check their web applications and services for the following problems:

- Incorrect or lack of input validation and data sanitation so that an attacker can trick an interpreter or query engine to do things that were not intended.
- Incorrect implementation of authentication and session management.
- Exposure of sensitive data because of problems like (1) insufficient or missing data encryption at rest and in motion, (2) password stores that do not use strong adaptive and salted hashing functions with a work factor (e.g., PBKDF2[1] or bcrypt[2]), or data leakage in log files.
- Incorrect implementation of the mechanisms to restrict what an authenticated user is allowed to do. For example, checks whether a user has the right permissions to execute an action might be done for all actions that a user can trigger via URL

[1] https://tools.ietf.org/html/rfc2898#page-9

[2] https://www.usenix.org/legacy/events/usenix99/provos/provos_html/node5.html

entries that are exposed in the web-interface—but not for actions that could be triggered by accessing portions of a website that are not exposed by such entries (forceful browsing).
- Use of insecure configurations as a result of insecure default configurations, incomplete or ad hoc configurations, outdated configurations, open cloud storage, misconfigured HTTP headers, verbose error messages containing sensitive information, or other root causes.

3.3 Data Protection

A core activity in data science is the processing of (large amounts of) data. For most processing tasks, the data must be available in unencrypted form. This has two main drawbacks. The first one is that when security measures such as access control fail, attackers can easily steal the data and make use of any information it contains. To make this more difficult, the data should always be stored in encrypted form. This way, the attacker must steal the data when it is being processed or manage to steal the keys used to encrypt it.

The second drawback is that the vast amount of processing power available in data centers around the world cannot be exploited if the data contains confidential information or is subject to data protection laws prohibiting the processing by (foreign) third parties. For such cases, it would have to be possible to do the processing in the encrypted space. Searchable encryption and homomorphic encryption (Prasanna and Akki 2015) offer interesting properties with this regard.

Searchable encryption (SE) introduced by Song et al. (2000) [see Bösch et al. (2014)] for an overview of different approaches) can be divided into many different subgroups. The core logic mostly consists of building an encrypted keyword search index on the client side. A search is then performed using trapdoor functions. A trapdoor function is a function that is easy to compute in one direction, but that is difficult to compute in the inverse direction unless one knows a secret. The most basic algorithms allow only queries with a single keyword and have performance issues when new data is added. If data is frequently modified, removed, or added, dynamic data search algorithms are required. Fuzzy-keyword search extends the algorithm to tolerate (some) spelling mistakes. There are also methods that support multiple keywords per query. SE offers methods to perform ranked search, for example, by taking the access history of a user and the access frequency into account. Although some research prototypes have been developed and partly also made available for general use and experimentation [e.g., Popa et al. (2011)], several limitations must be overcome before SE can be used widely in practice. One of these limitations is that algorithms based on secret (symmetric) key cryptography usually require a key exchange over a secured channel and offer only limited search capabilities compared to traditional search engines. Another one is that public key cryptography-based approaches are insufficient for modern big data systems because of substantial computational overhead.

Homomorphic encryption (HE) addresses the challenge to perform general computations on encrypted data. HE allows performing simple operations such as additions, multiplications, or quadratic formulas on ciphertext. It generates an encrypted result, which when decrypted, delivers the same result as if the operations were performed on the plaintext. This offers the ability to run calculations on untrusted devices without giving up on data privacy. Craig Gentry (2009) described the first Fully Homomorphic Encryption (FHE) scheme. This scheme allows performing any desirable function on encrypted data. Unfortunately, FHE is currently far away from practical use, as it increases memory consumption and processing times of even basic operations by about 6–7 orders of magnitude (Brown 2017). Therefore, Somewhat Homomorphic Encryption (SwHE) techniques are proposed. Compared to FHE, they provide better efficiency but do not support all operations [see, e.g., Gentry et al. (2012)]. On the implementation side, there are some HE research prototypes available such as by Halevi (2017). However, given the current state of HE technology, it is expected that several years of further research are required before HE is ready for productive use.

3.4 Privacy Preservation/Data Anonymization

In many cases, data science analyzes data of human individuals, for instance, health data. Due to legal and ethical obligations, such data should be anonymized to make sure the privacy of the individuals is protected. Data anonymization basically means that any data record in the data set should not be easily linkable to a particular individual. Obvious solutions include stripping the real name or the detailed address of individuals from the records, but experience teaches that this is usually not enough to truly anonymize the data.

For instance, in 2006, Netflix started an open competition with the goal to find algorithms that allow predicting user ratings for films. As a basis, Netflix provided a large data set of user ratings as training data, where both users and movies were replaced by numerical IDs. By correlating this data with ratings from the Internet Movie Database, two researchers demonstrated that it is possible to de-anonymize users (Narayanan and Shmatikov 2008). Another example is the Personal Genome Project, where researchers managed to de-anonymize about 90% of all participants (Sweeney et al. 2013). Their basic approach was to link information in the data records (birth date, gender, and ZIP code) with purchased voter registration lists and other publicly available information.

To overcome these issues, a more scientific approach toward anonymization is required. The question is the following: Is it possible to modify data such that the privacy of the participants is fully protected without losing the essence of the data and therefore its utility? In this context, “privacy protection” means that an attacker should not be able to learn any additional information about the individuals than what is directly provided by the data records, even when this data is correlated with other information. Past and more recent research activities have provided several

approaches that can help to achieve this, including generalization (Sweeney 1997) and suppression (Cox 1980), *k*-anonymity (Samarati and Sweeney 1998), and differential privacy (Dwork 2006). Each method has its advantages and drawbacks.

Suppression is a basic form of trying to achieve anonymity by either deleting attributes or substituting them with other values. Generalization describes the approach to blur data by replacing specific values with categories or ranges of values. An attribute containing the age of a person is then translated to a range, so 33 may result in 30–39. Combining these methods can lead to *k*-anonymity, which means that each record cannot be distinguished from at least $k - 1$ other records when considering the personally identifying information in the records.

As an example, assume that a data set includes data records of individual. Each record includes gender, age range, and disease from which the person is suffering. Assume there are three records with gender female and age range 50–59. This basically corresponds to 3-anonymity, as these three records cannot be distinguished from one another based on the attributes gender and age range. *k*-anonymity also has its limitations, especially if the diversity of the non-anonymized attributes is low. In the previous example, let us assume that the disease is heart-related in all three cases. This implies that if an attacker knows that Angela, who is 55 years old, is included in the data set, then he directly knows that she is suffering from heart-related health problems, as all female persons between 50 and 59 in the data set are suffering from it.

The basic idea of differential privacy is that the actual values of the attributes of any single record in the data set should only have a very limited effect on the outcome of any analysis performed on the data. If this is the case, an attacker, when querying the data set, cannot learn anything about a specific individual in the data set as the received outcome is possibly independent of the actual attributes of this individual. This is basically achieved by adding some noise to the result before it is presented to the analyst. For example, let us assume that there are 100 records of 100 persons in a data set and the attacker knows of 99 persons whether they have a heart-related disease or not (we assume that 33 of them have such an issue), but he doesn't know this of the remaining person, which we name Alice. If the attacker performs the query "how many persons have a heart-related disease," then he directly knows Alice's condition: If the query returns 33, Alice has no heart-related problem, if it returns 34, Alice has a heart-related issue. When using differential privacy, the query would not return the actual value, but it would distort it a little bit, that is, the query would return a value in the neighborhood of 33 or 34, such as 30, 32, or 35. What's important is that the returned value does not indicate whether the true value is 33 or 34, which implies the attacker cannot learn anything about Alice's condition.

Obviously, any data anonymization method has its price as it has a negative impact on the quality of the data and the precision of the results when doing data analysis. Suppressing and generalizing data removes information, which means that the results of any analysis performed on the data will become less precise. And in the case of differential privacy, we usually get results that are "close to the correct result," but that usually do not correspond to the exact result. But this is the price of

protecting the privacy of the involved individuals and this also implies that in practice, it is important to carefully balance the privacy of the individuals and the required precision of the analyses. A concise overview about anonymization methods is given by Selvi and Pushpa (2015). Detailed information about privacy-preserving data publishing and corresponding research can be found in the survey by Fung et al. (2010).

3.5 Machine Learning Under Attack

The combination of sophisticated algorithms and untrusted data can open the door for different kinds of attacks. In the 2010 Flash Crash (Kirilenko et al. 2017), the New York Stock Exchange experienced a temporary market loss of one trillion dollar caused by market manipulations. The trader Navinder Singh Sarao rapidly placed and canceled orders automatically so that high-frequency trading firms interpreted the signs incorrectly. In the beginning, they bought the spoof orders and absorbed the sell pressure. Few minutes later, these long-term positions were forcefully sold leading to a feedback loop. In times of big data, trading algorithms often take news feeds like business publications, SEC filings, and Twitter posts into account to make predictions. In 2013, this led to a loss of $130 billion in stock value due to a fake Twitter message from the associated press about an explosion in the White House (Foster 2013).

Mozaffari-Kermani et al. (2015) propose a method to generate data, which, when added to the training set, causes the machine learning algorithms to deliver wrong predictions for specific queries. Thus, this method could, for example, be used to compromise the effectiveness of a system to diagnose cancer or to identify anomalies in computer networks. Their method consists of two algorithms. The first one creates data sets that statistically belong to the attacked class but are labeled like the target class to which a bias should be created. The second algorithm then evaluates which data set has the highest impact on the degradation of the model. For their method to work well, the attacker must know the statistics of the training data set, the feature extraction process, and the machine learning algorithms used. However, the only true requirement is knowledge on the feature extraction process that maps a sample onto a feature vector. If the training set is not public or based on publicly available data and cannot be stolen, an attacker could construct a proxy training data set by querying the predictor with artificial test instances and by observing its responses (Nelson et al. 2008). And if the machine learning algorithm is not known, their approach can be modified to cope with this case at the cost of some of its effectiveness. A good countermeasure to this kind of attack is the use of a threshold value for the returned accuracy metrics. At first, one might think that because an attacker must be able to add training data to the training set, this poisoning attack is rather impractical. However, in many cases, the training data and/or its labels do come from untrusted sources or can at least be influenced by them. And even if the sources

are trusted, consider that an attacker might have hacked one or multiple of those sources because they were easier to hack than the system with the data product itself.

In recent years, machine-learning-as-a-service (MLaaS) has become a huge trend. Tech giants such as Amazon Web Services (2017), Google (2017), Microsoft (2017a), and many others offer customers to create and run machine learning algorithms in the cloud, offering different services like facial recognition and natural language processing. Some of these publicly accessible tools may contain sensitive data within their model that has to be protected. Fredrikson et al. (2015) show how confidential information of a machine learning model can be extracted by inverting it. The authors are able to reconstruct the images of the training data of a facial recognition system. For each image submitted, the system responds with a list of names together with their confidence value. This allows an attacker to treat it as an optimization problem finding the input that maximizes the confidence of a target class. The time for reconstruction depends on the model and varies between 1.4 s and 21 min. The attack is also applicable if nothing about the model is known (black-box) but takes significantly longer. Tramèr et al. (2016) improve the computation time by starting with stealing the model using prediction APIs and then running the inversion attack on the copy of the model. They further show how decision trees, logistic regression-based classifiers, and neural networks can be stolen by just using the provided interfaces and the rich information returned by MLaaS solutions.

4 Data Science for Security

After having discussed some of the security challenges a data scientist might face when developing and using modern data science technologies, this section deals with the opportunities of using data science to help solve major challenges in information security. In this context, we are looking at three general application areas: (1) anomaly detection, (2) malware detection and classification, and (3) threat detection. In the next chapter, we are going to take a more detailed look at a specific case study where machine learning was applied to detect obfuscated JavaScript code.

4.1 Anomaly Detection

The detection of anomalies is a major challenge in information security and has many applications such as network intrusion detection, credit card fraud detection, insurance claim fraud detection, insider trading detection, and many others. An anomaly describes a single point or a set of data points within a large data set that does not match the normal or usual behavior. In a network intrusion detection system, this could be a large amount of login attempts or an attacker who scans systems for open ports to get information about a targeted infrastructure. In credit card fraud detection, this could be an anomalous transaction over a significantly

larger amount than what is usually spent by the credit card holder. Another example is using the credit card in a different context than usual, for instance in a different country. In credit card fraud detection, this is a serious challenge due to the vastly increased amount of online transactions that are difficult to assign to specific locations. A third example of anomalous credit card usage would be a huge amount of small transactions in a short time.

A broader overview about this topic and the performance of different machine learning algorithms for anomaly detection is given in the survey by Chandola et al. (2009). They show how machine learning can contribute to solve different anomaly detection-based challenges. Their core conclusion is that there is currently no "one size fits all" solution. Nearest neighbor and clustering-based techniques suffer when data is high dimensional, because the distance measures cannot differentiate between normal and abnormal behavior anymore. Classification-based algorithms deliver good results but labeled data is often rare. Mahmud et al. (2016) give an overview of machine learning algorithms and their performance in credit card fraud detection. They achieve a classification accuracy of 98.25%, but the fraud detection success rate is below 50% because the fraction of fraudulent credit card transactions in the data set they used was small. According to the results, the highest detection rate is achieved using RotationForest, KStar, and RandomTree models. Finally, Gulenko et al. (2016) have evaluated machine learning algorithms for anomaly detection in cloud infrastructures. They come to the conclusion that high precision and recall rates can be achieved but the models suffer from aging effects. Therefore, models have to be periodically retrained and updated. Specific answers about the required periodicity are not given, however, and left open as future research.

The class imbalance problem that Mahmud et al. (2016) faced when they developed their credit card fraud detection system is fairly common in anomaly detection: the number of normal items, events, or observations is usually much larger than those of anomalous ones. If this imbalance in the distribution of the normal and the abnormal class(es) is not taken into account, a detector might perform poorly. Two examples where this imbalance tends to be quite strong are credit card fraud detection and network intrusion detection. Pozzolo et al. (2015) work with a data set with credit card transactions from European cardholders in September 2013. This data set has only 492 cases of fraud in the total of 2,84,807 transactions. Shiravi et al. (2012) present a reference data set (the ISCX data set) for validating network intrusion detection systems where, according to Soheily-Khah et al. (2017), attack traffic accounts for only 2% of the overall traffic. While 2% is quite low, it might easily be much lower, for example 0.01%, as in the application layer denial-of-service data set of Viegas et al. (2017).

Fortunately, many techniques exist to handle such imbalanced class distributions. One way to address the problem is to resample the training data to turn it into a more balanced data set. In the example with the credit card transaction data mentioned before, Pozzolo et al. (2015) performed a study on the impact of undersampling on classification accuracy and probability calibration. They found that randomly selecting and removing legitimate transactions to get a more balanced data set can indeed increase classification accuracy. However, for some of the other data sets they

used, this was not the case. An overview of this very active area of research—mainly with focus on binary classification—can be found in Branco et al. (2016).

Another way to approach the problem is to make use of machine learning algorithms that can cope better with (strongly) imbalanced class distributions. Traditional methods like support vector machines or decision trees have a bias toward the majority class since "... rules that correctly predict those instances are positively weighted in favor of the accuracy metric, whereas specific rules that predict examples from the minority class are usually ignored (treating them as noise), because more general rules are preferred. In such a way, minority class instances are more often misclassified than those from the other classes" (Galar et al. 2012). Or in other words, if a credit card fraud detector would classify all transactions as not fraudulent, the classifier could achieve 99% accuracy for a data set where 1% of the transactions are fraudulent.

According to Krawczyk (2016), the most popular branch of machine learning algorithms that aims at addressing this problem is cost-sensitive approaches where learners are modified to incorporate penalties for (some) classes. "This way by assigning a higher cost to less represented set of objects we boost its importance during the learning process (which should aim at minimizing the global cost associated with mistakes)" (Krawczyk 2016). However, for most of these approaches, profound theoretical insights into why and how well they perform with arbitrary imbalanced data sets is lacking. An overview over related work on this topic can be found in Branco et al. (2016) or Galar et al. (2012).

The most important takeaway from this discussion is that one should be aware of the imbalance problem when developing anomaly detection solutions.

A good starting point for a more in-depth study is Branco et al. (2016) and/or Krawczyk (2016). Furthermore, another takeaway is that retraining is an overall important task in anomaly detection as the normal behavior defined in the past will usually not sufficiently represent the future. This question is also addressed in general in novelty detection, which is the task of classifying data that differ in some respect from the data that are available during training [Pimentel et al. (2014)].

4.2 Malware Detection and Classification

In the past few years, hundreds of millions of new and unique malware samples have been found every year. However, most of these samples are very similar and belong to a few thousand malware families only (Check Point 2016). One of the reasons for this is that today, most malware authors modify and/or obfuscate their malware on a per victim basis. This way, they can evade simple signature-based antivirus scanners. To mitigate this problem, samples from known families should be recognized and filtered, and only new ones or samples that are "different enough" should have to be analyzed by malware analysts (if at all). Machine learning and big data seem to be capable solutions to handle such a large amount of continuously evolving data and to perform highly accurate classification tasks on it. For example, the next-generation

antivirus software from Cylance makes use of "... data-mining techniques to identify the broadest possible set of characteristics of a file. These characteristics can be as basic as the file size or the compiler used and as complex as a review of the first logic leap in the binary" (Cylance 2017). They claim to extract the uniquely atomic characteristics of the file depending on its type (*.exe*, *.dll*, *.com*, *.pdf*, *.doc*, etc.).

The importance of this task for the research and anti-malware industry was stressed by the fact that in 2015, Microsoft (2015) launched a contest to get new inputs on how to do the classification of malware samples into malware families from the community. The contestants were given a labeled training and a test data set, each consisting of 10,000 samples from nine different malware families. The results of this contest suggested that this task can be solved with very low multiclass loss (around 0.003). However, to achieve this, the contestants had data such as the assembly code of the binaries, which is difficult to extract without using dynamic code analysis. Furthermore, modern malware hides its true nature and unpacks or decrypts its malicious code only when run outside of an analysis environment. This and scalability problems when having to run all suspicious binaries make approaches based on dynamic code analysis less attractive than those based on static analysis.

Static code analysis describes all information about an application that can be gained without running it. On Android systems, this is usually the *apk* file, where security-relevant information such as API calls and even the source code itself can easily be accessed. This is good news since G DATA (2016) reported an average of 9468 new malicious applications for Android per day during the first half of 2016. It seems that due to their increased usage for mobile payment, mobile ticketing, and many other business cases, mobile devices became a very attractive target for cyber criminals.

Tam et al. (2017) provide a comprehensive overview of the challenges encountered when trying to detect and classify malicious Android applications. The authors find that in 2012, popular antivirus systems had a detection rate from around 20–79.6%. In all cases, complex malware was not detected. In particular, the systems often failed when the malware was obfuscated or when malicious Java code was executed after it was dynamically loaded during runtime. They show that new approaches from the data science domain can (easily) surpass traditional ones. This is confirmed by Arp et al. (2014), where the proposed DREBIN method achieves a detection rate of 97% with a low false-positive rate by combining statistical analysis with support vector machines.

On platforms where the source code is not easily available, static analysis gets more difficult. Narayanan et al. (2016) assess the performance of different machine learning and pattern recognition algorithms on imaginary representations of malware binaries. They find that samples from the same families result in a similar image texture. With this approach, it was possible to achieve results that were nearly as good as those of the winners of the Microsoft contest, but without having to extract the assembly code of the malware.

4.3 Threat Detection

Security information and event management (SIEM) (Zuech et al. 2015) technology supports threat detection and security incident response through the (real-time) collection and historical analysis of security events from a wide variety of events and contextual data sources. Such events might include failed and successful authentication attempts, the number of packets dropped by a firewall, or a report by an antivirus program about the identification and blocking of a malicious file.

In 2016, the security operations center of IBM recorded more than 20 billion security events per day (IBM 2016). This is still quite moderate when compared to the numbers from fortune 100 telecommunication providers, which can face up to one million events per second and up to 85 billion events per day (IBM 2013). Traditional SIEM solutions relying on structured databases and (mostly) manual definition of what is normal and malicious and/or abnormal behavior have difficulties scaling up to these large amounts of data.

The use of big data solutions and machine learning is therefore the next logical step in the evolution of such systems. Technologies such as Apache Hadoop and Apache Spark offer fast and scalable methods to analyze vast amount of data. According to Dark Reading (2012),

> in an environment where its security systems generate 3 terabytes of data a week, just loading the previous day's logs into the system can [. . .] take a full day

and

> searching among a month's load of logs could take anywhere between 20 minutes to an hour [. . .]. In our environment within HIVE, it has been more like a minute to get the same deal.

This is why companies such as HP and IBM put a lot of effort into the development of systems using new data science technologies (IBM 2013). However, determining which events are related to activities that are harmless, for example because they stem from an attack that failed, and which are related to real threats, is a challenging problem. In a large-scale experiment from HP, which had the goal to identify malicious domains and infected hosts, more than 3 billion HTTP and DNS Requests where collected from 900 enterprises around the world. They showed that high true-positive rates are possible using decision trees and support vector machines with a very limited amount of labeled data by simultaneously keeping the false-positive rates low (Cloud Security Alliance 2013a). Another work demonstrates the usage of a system called Beehive, which analyzed around 1 billion log messages in an hour and successfully detected violations and infections that were otherwise not noticed (Yen et al. 2013).

5 Case Study: Detecting Obfuscated JavaScripts

To demonstrate the potential and the challenges of applying machine learning to detect malware, this section describes in more detail the results of a recent analysis that was done by Tellenbach et al. (2016).

JavaScript is a common attack vector to probe for known vulnerabilities and subsequently to select a fitting exploit or to manipulate the Document Object Model (DOM) of a web page in a harmful way. The JavaScripts used in such attacks are often obfuscated to make them hard to detect using signature-based approaches. On the other hand, since the only legitimate reason to obfuscate a script is to protect intellectual property, there are not many scripts that are both benign and obfuscated. A detector that can reliably detect obfuscated JavaScripts would therefore be a valuable tool in fighting JavaScript-based attacks.

To evaluate the feasibility and accuracy of distinguishing between different classes of JavaScript code, a classic machine learning approach was used. In the first step, a data set was collected that contains JavaScripts and correct labels (obfuscated or non-obfuscated). Next, 45 features were selected and extracted from the JavaScripts in the data set. These features capture various aspects such as frequency of certain keywords, number of lines, characters per line, number of functions, entropy, and more. Based on this, several classification algorithms were trained and evaluated. The following sample code was used to make visitors of a hacked website connect to a server hosting the CrimePack exploit kit (Krebs 2017). The script is obfuscated to hide the fact that it injects an iframe and to obfuscate the URL it connects to:

```
tmssqrcaizo = "WYTUHYjE3cWYTUHYjE69WYTUHYjE66";
var makvvxmaqgh = "WYTUHYjE72";
var nlsysoyxklj =
"WYTUHYjE61WYTUHYjE6dWYTUHYjE65WYTUHYjE20WYTUHYjE6eWYTUHYjE61WYTU
HYjE6dWYT
UHYjE65WYTUHYjE3dWYTUHYjE22";
var zezugacgoqg =
"WYTUHYjE6eWYTUHYjE6fWYTUHYjE6aWYTUHYjE72WYTUHYjE73WYTUHYjE65WYTU
HYjE72WYT
UHYjE66WYTUHYjE6cWYTUHYjE72WYTUHYjE6f";
var nmcwycmeknp =
"WYTUHYjE22WYTUHYjE20WYTUHYjE77WYTUHYjE69WYTUHYjE64WYTUHYjE74WYTU
HYjE68WYT

(not shown)

var vbvvhagnggg = new Array();
vbvvhagnggg[0] = new Array(
tmssqrcaizo +
makvvxmaqgh +
nlsysoyxklj +

(not shown)
```

```
xmzvkbtpiof);
document[
  "WYTUHYjEwWYTUHYjErWYTUHYjEiWYTUHYjEtWYTUHYjEeWYTUHYjE".replace(
      /WYTUHYjE/g,
      ""
  )
](
  window[
"WYTUHYjEuWYTUHYjEnWYTUHYjEeWYTUHYjEsWYTUHYjEcWYTUHYjEaWYTUHYjEpW
YTUHYjEeWYTUHYjE".
replace(
  /WYTUHYjE/g,
  ""
  )
  ](vbvvhagnggg.toString().replace(/WYTUHYjE/g, "%"))
);
```

The script below is the unobfuscated version of the above script (URL is not the original one). The de-obfuscated code is significantly easier for a security researcher or any programmer to analyze:

```
document.write(
'<iframe name="nojrserflro" width="1" height="0"
src="http://localhost/index.php" marginwidth="1" marginheight="0"
title="nojrserflro" scrolling="no" border="0" frameborder="0"></
iframe>'
);
```

In general, there are many different ways how a script can be made hard to read and understand.

To collect the non-obfuscated samples in the data set, JavaScripts were extracted from the jsDelivr[3] content delivery network, which contains many JavaScript libraries) and the Alexa[4] Top 5000 websites. This resulted in 25,837 non-obfuscated samples, which includes both regular JavaScripts (as they have been written by the developers) and minified JavaScripts (where whitespace have been removed and function- and variable names have been shortened to reduce the overall size). To collect the obfuscated samples in the data set, two different strategies were used. First, a set of truly malicious (and obfuscated) JavaScript samples was received from the Swiss Reporting and Analysis Centre for Information Assurance MELANI.[5] However, this consisted of only 2706 samples. Therefore, additional obfuscated samples were synthetically generated by obfuscating the non-minified JavaScripts from the collected non-obfuscated samples. For this, six different, publicly available

[3] https://www.jsdelivr.com
[4] http://www.alexa.com
[5] http://www.melani.admin.ch

obfuscators were used, which delivered an additional 73,431 samples. Overall, this resulted in 76,137 obfuscated samples.

Based on this data set, the best classification performance could be achieved with a boosted decision tree classifier. With this classifier, only 1.03% of the non-obfuscated scripts were classified as obfuscated (false positives) and only 0.32% obfuscated scripts were classified as non-obfuscated (false negatives). Overall, boosted decision tree was the only classifier that achieved F1-scores above 99% for both classifying obfuscated and non-obfuscated JavaScripts, demonstrating that machine learning works well on this task.

Next, it was analyzed how well classification works to detect obfuscated JavaScripts if the corresponding obfuscator is not used for any of the JavaScripts that are used to train the classifier. The purpose of this analysis was to get an understanding about how well the classifier can "learn about obfuscation in general." The results of this analysis varied greatly depending on the specific obfuscator left out from the training set. For one obfuscator, the F1-score remained almost the same. For the other obfuscators, the F1-score was impacted by a few percent up to almost 100%. Finally, it was analyzed how well the truly malicious JavaScripts can be detected if the training set only includes the non-obfuscated and the synthetically generated obfuscated JavaScripts. In this case, less than 50% of the malicious JavaScripts were classified correctly as obfuscated.

This case study exhibits several interesting results and provides some lessons learned when using machine learning to detect malware or malicious activity in general:

- In general, classifying obfuscated and non-obfuscated JavaScripts works well, provided that the obfuscators used for the obfuscated JavaScripts are also represented in the data set used to train the classifier.
- Detecting obfuscated JavaScripts that use obfuscators not represented in the training set is more difficult. While this might be improved somewhat by using better-suited features, it clearly demonstrates that it is paramount to include samples that use a wide range of obfuscators in the data set so the classifier can learn a wide range of properties employed by different obfuscators. Generalizing this to other scenarios indicates that it is important to use representative malicious samples, whether it is actual malware or malicious activity in general.
- This directly leads to another challenge: It is difficult to get a large number of truly malicious samples. This is not only the case for malicious JavaScripts, but in general for "samples" that capture malicious behavior (e.g., network or system intrusions). While creating synthetic malicious samples may help to a certain degree, this has its limitations as it can usually not capture the full range of truly malicious samples, as demonstrated by the final analysis described above.

6 Conclusions and Lessons Learned

With respect to security, data science is a double-edged sword. On the one side, it offers many new opportunities and a lot of potential to significantly improve traditional security algorithms and solutions. Recent advances in challenging domains such as anomaly detection, malware detection, and threat detection underline the tremendous potential of security data science.

On the other side, it comes with many challenges. Most of them, including questions related to infrastructure and software security, can be addressed with the following practices:

- Protect your information system with suitable security controls. Get an idea of the complexity of the topic by checking out guides like the CIS top 20 security controls (Center for Internet Security 2017) and consult with or hire experts to protect your infrastructure.
- Implement an SSDLC to make sure that the software and services you develop are as secure as possible and that they remain secure.
- Check out the top security problems related to a specific technology or service. For example, the OWASP top 10 (OWASP 2017b) for web applications and services.
- Study the default configuration and all of the configuration options of a component to avoid insecure configurations.
- Keep in mind that anonymization is not perfect; whenever data privacy is critical, one has to choose the anonymization method with care and balance the privacy of the individuals and the required precision of the analyses.
- Check whether your system is susceptible to any of the various ways attackers might try to exploit data-driven applications (data poisoning, model extraction, etc.).

Nevertheless, recent incidents show that these practices are not widely used yet. One of the reasons is that today's security measures for data science heavily rely on security by afterthought, which is not acceptable as security aspects have to be considered during all steps of the development and product and data life cycle.

Other challenges require more research before they can be widely adopted, including questions related to perform computations on encrypted or anonymized data.

References

Amazon Web Services. (2017). *Artificial intelligence on AWS*. Retrieved from https://aws.amazon.com/amazon-ai/

Arp, D., Spreitzenbarth, M., Gascon, H., & Rieck, K. (2014). DREBIN: Effective and explainable detection of android malware in your pocket. *Presented at the 21st Annual Network and*

Distributed System Security Symposium (NDSS). Retrieved from http://dblp.uni-trier.de/db/conf/ndss/ndss2014.html#ArpSHGR14

Bösch, C. T., Hartel, P. H., Jonker, W., & Peter, A. (2014). A survey of provably secure searchable encryption. *ACM Computing Surveys, 47*(2), 18:1–18:51.

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys, 49*(2), 31:1–31:50. https://doi.org/10.1145/2907070.

Brown, B. (2017). *How to make fully homomorphic encryption "practical and usable"*. NETWORKWORLD. Retrieved from https://www.networkworld.com/article/3196121/security/how-to-make-fully-homomorphic-encryption-practical-and-usable.html

Burgees, M., & Temperton, J. (2016). The security flaws at the heart of the Panama Papers. *WIRED Magazine*. Retrieved from http://www.wired.co.uk/article/panama-papers-mossack-fonseca-website-security-problems

Center for Internet Security. (2017). *CIS 20 security controls*. Retrieved from https://www.cisecurity.org/controls

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys, 41*(3), 15:1–15:58.

Check Point. (2016). *Check Point research shows surge in active malware families during first half of 2016*. Retrieved from https://www.checkpoint.com/press/2016/check-point-research-shows-surge-active-malware-families-first-half-2016/

Cloud Security Alliance. (2013a). *Big data analytics for security intelligence*. Retrieved from https://downloads.cloudsecurityalliance.org/initiatives/bdwg/Big_Data_Analytics_for_Security_Intelligence.pdf

Cloud Security Alliance. (2013b). *Expanded top ten big data security and privacy challenges*. Retrieved from https://downloads.cloudsecurityalliance.org/initiatives/bdwg/Expanded_Top_Ten_Big_Data_Security_and_Privacy_Challenges.pdf

Cloud Security Alliance. (2016). *Security and privacy handbook: 100 best practices in big data security and privacy*. Retrieved from https://downloads.cloudsecurityalliance.org/assets/research/big-data/BigData_Security_and_Privacy_Handbook.pdf

Cox, L. H. (1980). Suppression methodology and statistical disclosure control. *Journal of the American Statistical Association, 75*(370), 377–385.

Cylance. (2017). *Math vs. Malware (white paper)*. Retrieved from https://www.cylance.com/content/dam/cylance/pdfs/white_papers/MathvsMalware.pdf

Dark Reading. (2012). *A case study in security big data analysis*. Retrieved from https://www.darkreading.com/analytics/security-monitoring/a-case-study-in-security-big-data-analysis/d/d-id/1137299

Dwork, C. (2006). Differential privacy. In *Proceedings of the 33rd International Conference on Automata, Languages and Programming – Volume Part II* (pp. 1–12). Berlin: Springer.

Easttom, W., II. (2016). *Computer security fundamentals*. Pearson Education.

Estopace, E. (2016). *Massive data breach exposes all Philippines voters*. Retrieved from https://www.telecomasia.net/content/massive-data-breach-exposes-all-philippines-voters

Foster, P. (2013). "Bogus" AP tweet about explosion at the White House wipes billions off US markets. *The Telegraph*. Retrieved from http://www.telegraph.co.uk/finance/markets/10013768/Bogus-AP-tweet-about-explosion-at-the-White-House-wipes-billions-off-US-markets.html

Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 1322–1333). New York, NY: ACM.

Fung, B. C. M., Wang, K., Chen, R., & Yu, P. S. (2010). Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys, 42*(4), 14:1–14:53.

G DATA. (2016). *New ransomware threatens Android devices*. Retrieved from https://www.gdatasoftware.com/news/2016/07/28925-new-ransomware-threatens-android-devices

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches.

IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42 (4), 463–484.

Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing* (pp. 169–178). New York, NY: ACM.

Gentry, C., Halevi, S., & Smart, N. P. (2012). Homomorphic evaluation of the AES circuit. In R. Safavi-Naini, & R. Canetti (Eds.), *Advances in Cryptology – CRYPTO 2012: 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19–23, 2012. Proceedings* (pp. 850–867). Berlin: Springer.

Google. (2017). *Google cloud machine learning engine*. Retrieved from https://cloud.google.com/ml-engine/

Gulenko, A., Wallschläger, M., Schmidt, F., Kao, O., & Liu, F. (2016). Evaluating machine learning algorithms for anomaly detection in clouds. In *2016 IEEE International Conference on Big Data (Big Data)* (pp. 2716–2721). doi:https://doi.org/10.1109/BigData.2016.7840917

Halevi, S. (2017). *HElib: An implementation of homomorphic encryption*. Retrieved from https://github.com/shaih/HElib

Holmes, A. (2007). *Your guide to good-enough compliance*. CIO. Retrieved from https://www.cio.com/article/2439324/risk-management/your-guide-to-good-enough-compliance.html

IBM. (2013). *Extending security intelligence with big data solutions*. IBM. Retrieved from https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=WGW03020USEN

IBM. (2016). *IBM announces new national cyber security centre in Canberra*. Retrieved from http://www-03.ibm.com/press/au/en/pressrelease/50069.wss

Kirilenko, A., Kyle, A. S., Samadi, M., & Tuzun, T. (2017). The Flash Crash: High-frequency trading in an electronic market. *The Journal of Finance, 72*(3), 967–998.

Krawczyk, B. (2016). Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence, 5*(4), 221–232.

Krebs, B. (2009). *Payment processor breach may be largest ever*. Retrieved from http://voices.washingtonpost.com/securityfix/2009/01/payment_processor_breach_may_b.html?hpid=topnews

Krebs, B. (2012). *Global Payments breach window expands*. Retrieved from https://krebsonsecurity.com/2012/05/global-payments-breach-window-expands/

Krebs, B. (2017). *Crimepack: Packed with hard lessons*. Retrieved from https://krebsonsecurity.com/2010/08/crimepack-packed-with-hard-lessons/

Li, X., & Xue, Y. (2014). A survey on server-side approaches to securing web applications. *ACM Computing Surveys, 46*(4), 54:1–54:29.

Mahmud, M. S., Meesad, P., & Sodsee, S. (2016). An evaluation of computational intelligence in credit card fraud detection. In *2016 International Computer Science and Engineering Conference (ICSEC)* (pp. 1–6). doi:https://doi.org/10.1109/ICSEC.2016.7859947

Matherly, J. (2015). *It's the data, stupid!* Retrieved from https://blog.shodan.io/its-the-data-stupid/

Microsoft. (2015). *Microsoft malware classification challenge*. Retrieved from https://www.kaggle.com/c/malware-classification

Microsoft. (2017a). *Azure machine learning studio*. Retrieved from https://azure.microsoft.com/en-us/services/machine-learning-studio/

Microsoft. (2017b). *Microsoft security development lifecycle*. Retrieved from https://www.microsoft.com/en-us/sdl/

Moreno, J., Serrano, M. A., & Fernández-Medina, E. (2016). Main issues in big data security. *Future Internet, 8*(3), 44.

Mozaffari-Kermani, M., Sur-Kolay, S., Raghunathan, A., & Jha, N. K. (2015). Systematic poisoning attacks on and defenses for machine learning in healthcare. *IEEE Journal of Biomedical and Health Informatics, 19*(6), 1893–1905.

Narayanan, A., & Shmatikov, V. (2008). Robust de-anonymization of large sparse datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy* (pp. 111–125). Washington, DC: IEEE Computer Society.

Narayanan, B. N., Djaneye-Boundjou, O., & Kebede, T. M. (2016). Performance analysis of machine learning and pattern recognition algorithms for malware classification. In *2016 IEEE National Aerospace and Electronics Conference (NAECON) and Ohio Innovation Summit (OIS)* (pp. 338–342). doi:https://doi.org/10.1109/NAECON.2016.7856826

National Institute of Standards and Technology. (2017). *NIST Special Publication Series SP 800 and SP 1800*.

Nelson, B., Barreno, M., Chi, F.J., Joseph, A.D., Rubinstein, B.I.P., Saini, U., Sutton, C., Tygar, J. D., Xia, K. (2008). Exploiting machine learning to subvert your spam filter. In: *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, LEET'08* (pp. 7:1–7:9). USENIX Association, Berkeley, CA.

OWASP. (2017a). *OWASP SAMM project*. Retrieved from https://www.owasp.org/index.php/OWASP_SAMM_Project

OWASP. (2017b). *OWASP Top Ten project*. Retrieved from https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Pauli, D. (2017a). *MongoDB hackers now sacking Elasticsearch*. The Register. Retrieved from https://www.theregister.co.uk/2017/01/13/elasticsearch_mongodb/

Pauli, D. (2017b). *MongoDB ransom attacks soar, body count hits 27,000 in hours*. The Register. Retrieved from http://www.theregister.co.uk/2017/01/09/mongodb/

Pimentel, M. A. F., Clifton, D. A., Clifton, L., & Tarassenko, L. (2014). A review of novelty detection. *Signal Process, 99*, 215–249. https://doi.org/10.1016/j.sigpro.2013.12.026.

Popa, R. A., Redfield, C. M. S., Zeldovich, N., & Balakrishnan, H. (2011). CryptDB: Protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles* (pp. 85–100). New York, NY: ACM. https://doi.org/10.1145/2043556.2043566.

Pozzolo, A. D., Caelen, O., Johnson, R. A., & Bontempi, G. (2015). Calibrating probability with undersampling for unbalanced classification. In *2015 IEEE Symposium Series on Computational Intelligence* (pp. 159–166). https://doi.org/10.1109/SSCI.2015.33.

Prasanna, B. T., & Akki, C. B. (2015). *A comparative study of homomorphic and searchable encryption schemes for cloud computing*. CoRR, abs/1505.03263. Retrieved from http://arxiv.org/abs/1505.03263

Risk Based Security. (2014). *A breakdown and analysis of the December, 2014 Sony Hack*. Retrieved from https://www.riskbasedsecurity.com/2014/12/a-breakdown-and-analysis-of-the-december-2014-sony-hack

Ron, A., Shulman-Peleg, A., & Puzanov, A. (2016). Analysis and mitigation of NoSQL injections. *IEEE Security and Privacy, 14*, 30–39.

Sahafizadeh, E., & Nematbakhsh, M. A. (2015). A survey on security issues in big data and NoSQL. *Advances in Computer Science: An International Journal, 4*(4), 68–72.

Samarati, P., & Sweeney, L. (1998). *Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression*. Computer Science Laboratory, SRI International. Retrieved from http://www.csl.sri.com/papers/sritr-98-04/

Schreier, J. (2011). Sony estimates $171 million loss from PSN hack. *WIRED Magazine*. Retrieved from https://www.wired.com/2011/05/sony-psn-hack-losses/

Selvi, U., & Pushpa, S. (2015). A review of big data and anonymization algorithms. *International Journal of Applied Engineering Research, 10*(17).

Shiravi, A., Shiravi, H., Tavallaee, M., & Ghorbani, A. A. (2012). Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers and Security, 31*(3), 357–374.

Software Testing Help. (2017). *37 most powerful penetration testing tools (security testing tools)*. Retrieved from http://www.softwaretestinghelp.com/penetration-testing-tools/

Soheily-Khah, S., Marteau, P.-F., & Béchet, N. (2017). *Intrusion detection in network systems through hybrid supervised and unsupervised mining process – A detailed case study on the ISCX benchmark dataset*. Retrieved from https://hal.archives-ouvertes.fr/hal-01521007

Song, D. X., Wagner, D., & Perrig, A. (2000). Practical techniques for searches on encrypted data. In *Proceedings of the 2000 IEEE Symposium on Security and Privacy* (pp. 44–). Washington, DC: IEEE Computer Society. Retrieved from http://dl.acm.org/citation.cfm?id=882494.884426

Sweeney, L. (1997). Guaranteeing anonymity when sharing medical data, the Datafly System. *Proceedings: A Conference of the American Medical Informatics Association. AMIA Fall Symposium* (pp. 51–55).

Sweeney, L., Abu, A., & Winn, J. (2013). *Identifying participants in the Personal Genome project by name (a re-identification experiment)*. CoRR, abs/1304.7605. Retrieved from http://arxiv.org/abs/1304.7605

Tam, K., Feizollah, A., Anuar, N. B., Salleh, R., & Cavallaro, L. (2017). The evolution of Android malware and Android analysis techniques. *ACM Computing Surveys, 49*(4), 76:1–76:41. https://doi.org/10.1145/3017427.

Tellenbach, B., Paganoni, S., & Rennhard, M. (2016). Detecting obfuscated JavaScripts from known and unknown obfuscators using machine learning. *International Journal on Advances in Security, 9*(3&4). Retrieved from https://www.thinkmind.org/download.php?articleid=sec_v9_n34_2016_10.

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K., & Ristenpart, T. (2016). *Stealing machine learning models via prediction APIs*. CoRR, abs/1609.02943. Retrieved from http://arxiv.org/abs/1609.02943

Viegas, E. K., Santin, A. O., & Oliveira, L. S. (2017). Toward a reliable anomaly-based intrusion detection in real-world environments. *Computer Networks, 127*(Suppl. C), 200–216. https://doi.org/10.1016/j.comnet.2017.08.013.

Yen, T.-F., Oprea, A., Onarlioglu, K., Leetham, T., Robertson, W., Juels, A., & Kirda, E. (2013). Beehive: Large-scale log analysis for detecting suspicious activity in enterprise networks. In *Proceedings of the 29th Annual Computer Security Applications Conference* (pp. 199–208). New York, NY: ACM. https://doi.org/10.1145/2523649.2523670.

Zuech, R., Khoshgoftaar, T. M., & Wald, R. (2015). Intrusion detection and big heterogeneous data: A survey. *Journal of Big Data, 2*(1), 3.

Chapter 16
Online Anomaly Detection over Big Data Streams

Laura Rettig, Mourad Khayati, Philippe Cudré-Mauroux, and Michał Piorkówski

Abstract In many domains, high-quality data are used as a foundation for decision-making. An essential component to assess data quality lies in anomaly detection. We describe and empirically evaluate the design and implementation of a framework for data quality testing over real-world streams in a large-scale telecommunication network. This approach is both general—by using general-purpose measures borrowed from information theory and statistics—and scalable—through anomaly detection pipelines that are executed in a distributed setting over state-of-the-art big data streaming and batch processing infrastructures. We empirically evaluate our system and discuss its merits and limitations by comparing it to existing anomaly detection techniques, showing its high accuracy, efficiency, as well as its scalability in parallelizing operations across a large number of nodes.

1 Introduction

Data-intensive systems in many domains require an understanding of the quality of data. Two factors go into the estimation of the trustworthiness of data sources and analytics derived from these sources: the content and the processing methods prior to the analyses. There is a high degree of control over the correctness of the processing, as it lies in the hands of those developing the analytics. The quality of the data coming from the various sources lies beyond our control and is prone to various types of error. We therefore need to discover data quality on the existing data. Data quality issues often present themselves as anomalies in the data. While not all anomalous data are in fact linked to data quality, certain types of anomalies can be linked directly to faulty data sources producing data of poor quality. It is hence necessary to detect and understand anomalies in the data that serve as a foundation for analyses. Assessing

L. Rettig · M. Khayati · P. Cudré-Mauroux (✉)
University of Fribourg, Fribourg, Switzerland
e-mail: pcm@unifr.ch

M. Piorkówski
Philip Morris International, Lausanne, Switzerland

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_16

data quality in real time on data streams is important for being able to react quickly whenever real-time services are provided based on the data.

Data anomalies can manifest themselves in many different ways—for instance, via missing values or outliers—and can be caused by erroneous procedures, system failures, or unexpected events. While the two former causes are linked to data quality issues, the latter is typically a property of the data. It is thus necessary to be able to distinguish between anomalous cases. Using anomaly detection for measuring data quality follows the assumption that the majority of data are of high quality and non-anomalous, such that anomalies are directly linked to data quality problems.

In this chapter, we focus on detecting anomalies on the signaling traffic of Swisscom's mobile cellular network, where any mobile terminal attached to the cellular network produces signaling messages. High-level anonymized messages are subsequently captured by the network infrastructure for the sake of quality assurance. The characteristics of the signaling traffic we consider fall into the class of big data streams, as (1) the cumulative daily volume we consider is in the order of terabytes (TBs), (2) the signaling data is multidimensional (high variety), and (3) the number of events per time unit is in the order of hundreds of millions per second (high velocity).

Our system needs to meet the following properties: (1) *Generality*: The system needs to adapt to different types of data, for example, multidimensional or categorical. (2) *Scalability*: Since we are dealing with big data streams with a high velocity, we are interested in a system that scales to a larger number of machines for parallel processing. (3) *Effectiveness*: We would like to be able to quantify the statistical soundness of the detected anomalies.

The following section provides background information on the data and technologies being used in this framework. This is followed by the presentation of the anomaly detection system in Sect. 3. In Sect. 4, we evaluate the accuracy and performance of the system on real-world and simulated data, and discuss the results in Sect. 5. Section 6 presents related work on data streams and anomaly detection. Finally, Sect. 7 concludes the work and presents the lessons learned.

2 Background Information

For a general overview, this chapter first provides a detailed introduction to the various open source technologies that are being used in the implementation of our anomaly detection system as well as their place in the specific domain context at Swisscom. This is followed by a description of the two anomaly detection measures, namely, relative entropy and Pearson correlation, which are leveraged in the system.

2.1 Technologies

Multiple open source streaming platforms have emerged in recent years, including Apache Storm,[1] Apache Samza,[2] Apache Spark's Streaming library,[3] and Apache Flink.[4] This project uses Apache Spark (Zaharia et al. 2010) and Spark Streaming (Zaharia et al. 2012a), the latter offering real-time processing in the form of micro-batches with data parallelism. Data parallelism implies that the same task is performed in parallel on multiple machines, each on a partition of the data.

2.1.1 Apache Spark

Apache Spark is a general-purpose engine for large-scale in-memory data processing that handles both batch and stream data processing. Spark offers several advantages over MapReduce (Dean and Ghemawat 2004), including faster in-memory execution and a high-level API that facilitates the expression of complex processing pipelines.

Spark's main abstraction is Resilient Distributed Datasets (RDDs) (Zaharia et al. 2012b) for representing distributed datasets. An RDD is an immutable abstraction of distributed data; materialization of the data is done in a lazy manner that also minimizes data shuffling between the executors over which the data are distributed. Each executor maintains a proportion of the data in memory. In batch processing mode, RDDs are created by loading data from storage or as a result of transforming another RDD. Deterministic transformations of the distributed dataset, such as *map*, *filter*, or *reduce* operations, yield a new RDD.

2.1.2 Spark Streaming

Apache Spark includes a streaming library called Spark streaming. The underlying execution engine, the Spark engine, is the same for both streaming and batch modes. Spark Streaming provides the ability to consume real-time data from various sources, including Apache Kafka. Stream processing is based on micro-batch computations and introduces a second core abstraction in addition to RDDs, Discretized Streams (DStreams) (Zaharia et al. 2013). Micro-batch computation implies that instances of data are not processed individually as they arrive, but instead are buffered into very small batches that are then processed jointly. There are different models for micro-batches: A micro-batch can either be of a fixed size (e.g., n data points), or contain all data collected during a fixed time period (e.g., n seconds).

[1] http://storm.apache.org
[2] http://samza.apache.org
[3] http://spark.apache.org/streaming
[4] http://flink.apache.org

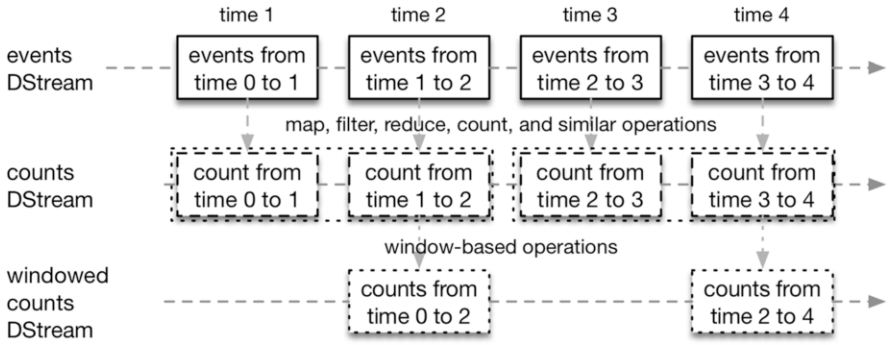

Fig. 16.1 DStream micro-batch model. Each box corresponds to one RDD. Operators that are applied directly to each RDD in the DStream. Window operations that group together data from multiple RDDs over a period of time transform one DStream to another [adapted from the Spark Streaming Programming Guide (The Apache Software Foundation 2015)]

In Spark Streaming's programming model, the continuous data on the stream are treated in micro-batches of fixed durations. DStreams are continuous sequences of RDDs (cf. Fig. 16.1), with one RDD containing all the data belonging to one micro-batch. DStreams can be transformed much as RDDs. A transformation on a DStream will be applied to each incoming RDD that contains all data for the duration of one micro-batch. Figure 16.1 shows an incoming DStream of events as a sequence of micro-batch RDDs and the application of operators to each RDD. Windowing groups together multiple micro-batches into batches over longer periods of time. Output operations are performed on each RDD in the stream and include printing results, saving data to disk, or writing data to a queue for consumption by another application, leading to materialization of the current dataset in the stream.

A DStream's RDDs are processed sequentially in the order in which they arrive. It is important that any processing terminates in less than the micro-batch interval duration. If the processing takes less than the batch duration, one micro-batch can be processed while the receivers collect the data for the next micro-batch. Once the next micro-batch is ready to be processed, the previous processing has completed and the computational resources are available. If the processing takes longer than the batch duration, new data will have to be stored until they can be processed. This way, data add up and increasingly delay the processing. Eventually, processing will no longer be possible, because old stream data, which have not yet been processed, had to be removed in order to receive and store newer data.

As code can be reused with minimal adaptation between batch and stream processing, Spark is well-suited for cases where both batch and streaming data are to be jointly processed, or where similar pipelines exist for both real-time and batch processing.

2.1.3 Apache Kafka and Real-World Data Streams

In order to perform stream processing with Spark Streaming, we retrieve data streams in real time from Apache Kafka,[5] a high-throughput distributed publish/subscribe system. Kafka is run as a cluster and can handle multiple separate streams, called *topics*. *Producers* publish messages to a Kafka topic and *consumers*, such as our application, subscribe to topics and process the streams. Topics may be partitioned over multiple machines, called *brokers* in the Kafka cluster, enabling data consumers to receive data in parallel.

As part of Swisscom's big data infrastructure, the so-called *Firehose* system provides a streaming pipeline from raw binary sources to the application layer. For the purpose of quality assurance, a passive monitoring system collects nonpersonal signaling events as probes on the links between local controllers and Swisscom's central core network. An event is triggered by any action of an anonymized network user. Each protocol among 2G, 3G, and 4G has a separate infrastructure setup and has its particular probe. Firehose ingests the signaling events that are obtained from these probes in real time in binary format. All telecommunication protocols' data are treated as separate input streams. Specifically, 2G and 3G are each separated into two interfaces: one monitoring voice events (e.g., phone calls) and the other monitoring data traffic events. In total, we are considering five monitoring interfaces in this work: *A* for 2G voice events, *Gb* for 2G data traffic events, *IuCS* for 3G voice events, *IuPS* for 3G data traffic events, and *S1-MME* for all 4G events. For each of these input streams, Firehose parses the events and then sends them to separate Kafka topics, one per interface. From there, the events, each having a series of attributes associated with it, are available for use in Spark Streaming applications.

The joint use of Spark Streaming and Kafka provides at least once and exactly once processing guarantees for received records. Kafka's fault tolerance allows a real-time application to recover after a brief failure.

2.2 Anomaly Detection Measures

Two measures are computed over the streams in order to perform anomaly detection: relative entropy between consecutive batches of data over a stream and Pearson correlation between multiple streams. In this section, these two measures are briefly explained.

Relative Entropy, or *Kullback–Leibler Divergence* (Kullback and Leibler 1951), $D(P\|Q)$, is a nonsymmetric measure of information loss. Specifically, it measures the difference between two probability distributions P and Q representing two datasets, for example for the purpose of detecting anomalies (Lee and Xiang 2001). In our context, $D(P\|Q)$ is used to measure changes between successive time windows over multidimensional data streams (Dasu et al. 2006). It is defined on two probability distributions P and Q as follows:

[5] http://kafka.apache.org

$$D(P\|Q) = \sum\nolimits_{i\in A} P(i)\log\frac{P(i)}{Q(i)} \tag{16.1}$$

where $P(i)$ and $Q(i)$ are the probability of item i in the respective probability distribution, given by

$$P(i) = \frac{m_i}{\sum_{a\in A} m_a} \tag{16.2}$$

where A is the set of all possible items i in the probability distributions and m_i and m_a are the number of items i and a, respectively, in the current distribution P.

The values of $P(i)$ and $Q(i)$ are defined over $[0, 1]$. D is not defined over a fixed range. In order to be able to interpret the value, it is therefore necessary to determine a baseline as a range of normal values for relative entropy. Under the premise that there exists a normal profile of the data, low relative entropy is linked to regularity. Low relative entropy indicates that the two distributions P and Q are similar, 0 meaning identical P and Q. Anomalies are detected when the relative entropy increases, that is, when D increases significantly compared to the baseline.

Pearson correlation coefficient is a statistical value measuring the linear dependence between two vectors X and Y, which we assume are normally distributed and contain n elements each. Pearson correlation is defined over X and Y as

$$r(X,Y) = \frac{\sum_{i=1}^{n}(x_i - \underline{x})\left(y_i - \underline{y}\right)}{\sqrt{\sum_{i=1}^{n}(x_i - \underline{x})^2}\sqrt{\sum_{i=1}^{n}\left(y_i - \underline{y}\right)^2}} \tag{16.3}$$

where $\underline{x}$ and $\underline{y}$ stand for the mean of X and Y, respectively.

The coefficient $r(X, Y)$ ranges between 1 and -1. Positive values from $(0, 1]$ indicate positive correlation between X and Y, while negative values from $[-1, 0)$ indicate negative correlation. A positive $r(X, Y)$ occurs when an increase or decrease in the values in X is met with the same trend, increase or decrease, in Y. A negative $r(X, Y)$ occurs when changes in X and Y are opposing, for example, a decrease in one vector is met with an increase in the other vector. When the Pearson correlation coefficient is 0, there is no linear correlation between X and Y.

3 Anomaly Detection System

The system we designed for anomaly detection and its integration within the telecommunications monitoring data pipeline is depicted in Fig. 16.2. In this high-level overview, we show the entire data pipeline starting from cell towers on the left-hand side to the anomaly detection results on the right-hand side.

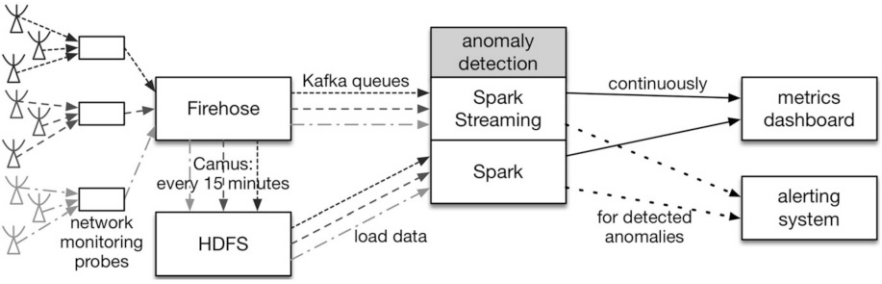

Fig. 16.2 Overview of the stream architecture and the integration of the anomaly detection system therein, showing the entire pipeline from telecommunication cell towers through staging and processing to anomaly detection results

Signaling traffic is received from network probes. The different shades of grey in the data flow in Fig. 16.2 coming from antennas to the monitoring probes and then into the system represent the streams of different data types, which are received and processed separately. For the purpose of simplicity, Fig. 16.2 has been limited to show one network monitoring probe for each interface only. In reality, there are multiple (physical) probes per interface type, of which the events are collected by Firehose, the data stream enabling infrastructure. Firehose stages the data in real time and then writes to a dedicated queue for real-time consumption. Periodically, data are also staged to HDFS for longer-term storage and processing. As each event is timestamped, it can be treated in the same way as real-time data and can also be used to simulate streams. In that sense, our system emulates the so-called lambda architecture (Marz and Warren 2013) with analogous pipelines for batch and real-time processing on the same data source.

The anomaly detection component consumes the data (either in streaming from a Kafka queue or in batch mode from HDFS), processes them, and sends the output to both the metrics dashboard and the alerting system.

3.1 Stream Processing

In Spark Streaming, data are processed periodically after a fixed duration as micro-batches. The duration of the micro-batches is chosen experimentally, as it depends on the volume of data and the processing complexity. Longer micro-batches require more storage since more data need to be cached in-memory until the next processing interval. On the other hand, a shorter micro-batch duration requires faster online algorithms and cannot amortize the network overhead from shuffling data. It is necessary that all computation finishes on average within the duration of a micro-batch, as these are processed in sequence and will otherwise accumulate and eventually fail.

Nine parallel streaming receivers, which represent the input interface of the Spark Streaming component, connect as consumers to Kafka, located in Firehose. As output from the anomaly detection component, metrics are written continuously, whereas alerts are triggered upon detection of an anomalous event.

The actual anomaly detection—in Spark and Spark Streaming, depending on the use case and data at hand—consists of computing measures over short time windows and comparing the outcome to expected values. In order to perform anomaly detection, two measures are continuously maintained over the streams: relative entropy on individual data streams and Pearson correlation across multiple streams obtained from different interfaces. These metrics form a constant baseline over non-anomalous data, such that anomalous data are detected as deviations from typical baseline values.

3.1.1 Relative Entropy Pipeline

Relative entropy is computed separately on each interface, for example, A or IuCS, by comparing the empirical distributions of event types. The data pipeline for computing relative entropy is shown in Fig. 16.3. As a first step, optionally, the stream is filtered to include only events originating from an area of interest. Each batch in the incoming DStream is mapped onto a new DStream of *((location, event type), 1)* tuples, where the identifier for the location and the type of the event form a composite key.

By summing up the values per key in a *reduce* operation, the number of events per location and event type get counted. Grouping per location yields a new RDD containing the event histograms, that is, the counts per event type and per location. While looking at lower levels in the topology, that is, more fine-grained location identifiers, facilitates the detection of local anomalies, a more global model is faster to compute due to the smaller number of distinct computations.

These event histograms are interpreted as aggregates of anonymized user-triggered actions since they capture various aspects of human activity (making a phone call, moving across the network, etc.). $P(i)$ represents the relative frequency of event type i in the current distribution. Finally, for each location indicator, the relative entropy $D(P_t||Q_{t-\Delta t})$ between the current distribution P_t at time t and the previous distribution $Q_{t-\Delta t}$ at time $t - \Delta t$ is computed by summing the comparison of each possible event type i. A higher D than in the baseline indicates the occurrence of change. In streaming mode, the probability distribution from the previous RDD is stored for comparing adjacent windows, yielding a distance measure between the two time periods per location.

We now give a simplified example of how relative entropy is computed using sample data.

Example 1 We consider streams of messages of the form $[(t_1, c_1, e_1), (t_2, c_2, e_2), \ldots]$ with c_i coming from the set of cell identifiers $\{B, C\}$ and e_i coming from the set of possible event type identifiers $A = \{1, 2\}$. A micro-batch $[(t_0, C, 1), (t_1, C, 2), (t_2, C, 1), (t_3, B, 2), (t_4, C, 1), (t_5, C, 1)]$ is obtained at a time t, with timestamps t_i in the

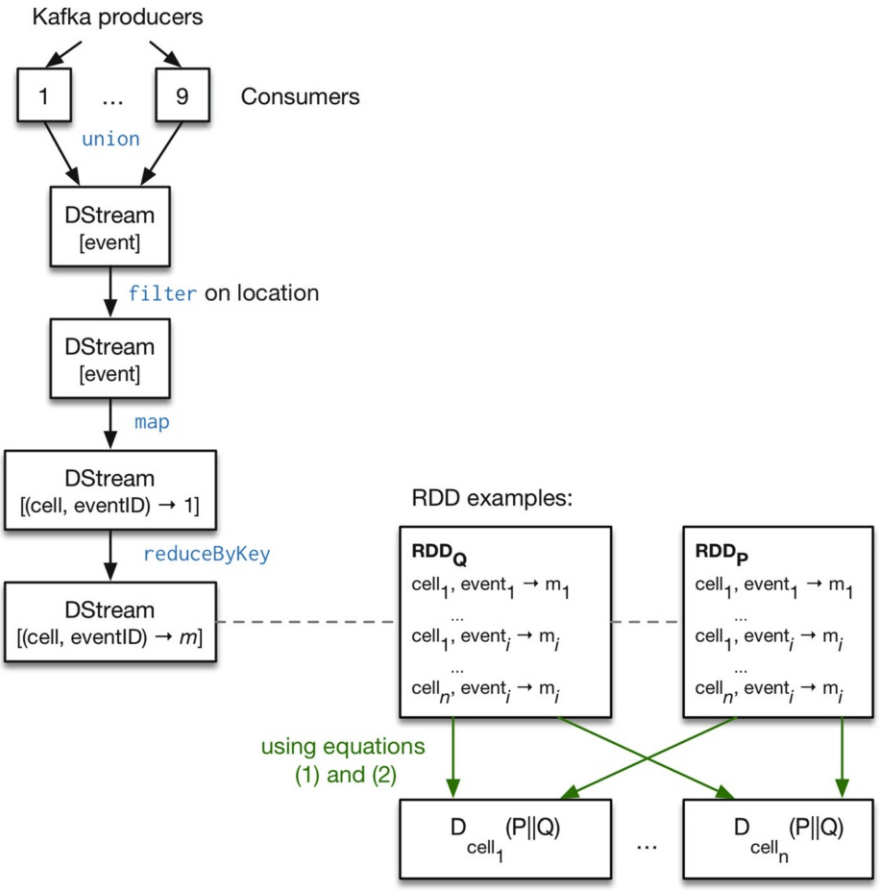

Fig. 16.3 Relative entropy $D(P\|Q)$ computation pipeline showing the parallel receiving of the stream, transformations, and computation of the measure from the data

range of this micro-batch. The partition of the stream at time t is mapped onto $[((c_1, e_1), 1), ((c_2, e_2), 2), \ldots]$. We apply a reduce operation on the composite key consisting of the cell and the event type to transform this micro-batch into tuples containing the count for each key as follows: $[((C, 1), 4), ((C, 2), 1), ((B, 2), 1)]$. Since we compute the relative entropy for each cell individually, we illustrate the computation for cell C only (similar computations are applied to all other cells). At time t in cell C, the histogram's counts are respectively 4 for event type 1 and 1 for event type 2. Using Eq. (16.2), the probabilities in P_t are respectively $P(1) = 4/5$ and $P(2) = 1/5$. We compare the distribution P_t to that from a previous micro-batch $Q_{t-\Delta t}$ with, say, $Q(1) = 2/3$ and $Q(2) = 1/3$. By applying Eq. (16.1), we obtain $D(P\|Q) = \frac{4}{5}\log\frac{\frac{4}{5}}{\frac{2}{3}} + \frac{1}{5}\log\frac{\frac{1}{5}}{\frac{1}{3}} = 0.044$.

3.1.2 Pearson Correlation Pipeline

In order to compute the Pearson correlation coefficient $r(X, Y)$ between vectors X and Y obtained from windows at time t over two streams S_X and S_Y, the implementation consumes events from at least two separate interfaces. As depicted in the data pipeline in Fig. 16.4, both streams are treated separately, mapping each stream onto a DStream containing anonymized user IDs and then counting the number of distinct IDs per micro-batch such that we obtain one value per micro-batch. Since we cannot compute the correlation coefficients directly between two unbounded streams, we opt for windowing over the stream in order to create finite vectors, between which we are able to compute the Pearson correlation as per Eq. (16.3). Windowing yields RDDs containing multiple counts—essentially DStreams containing, as RDDs, the vectors X (on the windowed stream S_X) and Y (on the windowed stream S_Y). At this point, both streams are combined as a DStream of pairs of RDDs, (X, Y), with corresponding timestamps. Using the pairs of RDDs, containing the unique input counts $x_1, \ldots, x_n$ and $y_1, \ldots, y_n$, respectively, a correlation coefficient for the particular time window t is computed.

A simple example on sample data illustrates the pipeline.

Example 2 Given two distinct streams of messages mapped onto a stream of anonymized user identifiers u_i of the following form $[u_1, \ldots, u_n]$, we collect the data during a short period of time (e.g., 10 s). Let us assume stream S_X contains $[A, B, B, A, B]$ and stream S_Y contains $[C, B, C, C, D, C, D, E, F, A]$ in this short window. By applying a *distinct* operation on each stream (yielding $[A, B]$ on S_X and $[A, B, C,$

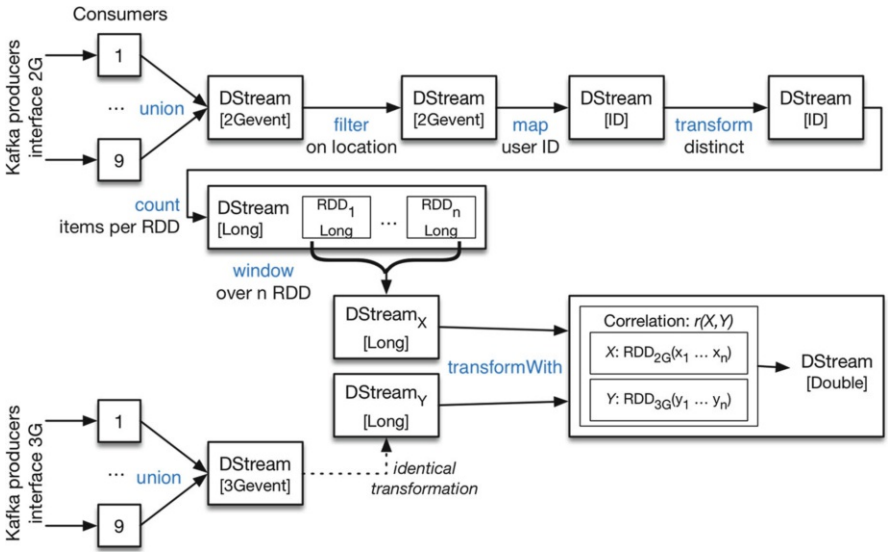

Fig. 16.4 Pipeline for computing the Pearson correlation $r(X, Y)$ between windows containing the vectors X and Y over two streams. Two streams are received and transformed separately, then joined to compute the correlation between corresponding windows at the same time

D, E, F] on S_Y) and then retrieving the length of this window, we obtain the count of distinct users per micro-batch. These two counts, respectively 2 and 6, are then written to the respective result streams RS_X and RS_Y. After 40 s, that is, after receiving four micro-batches on each stream, each yielding one count, the streams contain: RS_X, $[2, 1, 1, 3]$, and RS_Y, $[6, 5, 4, 6]$. These 40 s windows of counts are treated as vectors X and Y for this timespan, both of length 4. We group these vectors into pairs as $RS_{X,\ Y}$, $[(2, 6), (1, 5), (1, 4), (3, 6)]$. Equation (16.3) yields a Pearson correlation of $9/\sqrt{55} = 0.94$ for this example. Consider the case where the network monitoring probe producing events on S_Y fails, such that we no longer receive events from one area. Then, by reducing the events on S_Y and the count of distinct users on RS_Y at a certain time, for example, after 20 s, an increase in x_i meets a decrease in y_i. Thus, the stream of grouped pairs is as follows: $[(2, 6), (1, 5), (1, 2), (3, 5)]$ so that the correlation $r(X, Y)$ for this pair of vectors is lower at $5/(3\sqrt{11}) = 0.5$.

4 Empirical Evaluation of the Anomaly Detection System

To evaluate the efficiency and the effectiveness of our anomaly detection pipeline, we conducted experiments on real-world data in the form of big data streams and data loaded from HDFS. Both data sources are provided by Swisscom's big data infrastructure. The data we focused on for our experiments are captured at the A and the IuCS interfaces by the probes monitoring 2G voice and 3G voice links, respectively, which report network events on the telecommunication network.

We are interested in evaluating the system both in terms of its accuracy—how well it detects anomalies—and its scalability—how well it can adapt to an increase in computing parallelism.

4.1 Anomaly Detection Accuracy

4.1.1 Relative Entropy Accuracy

As explained previously in Sect. 3, event histograms are interpreted as an aggregate of anonymized mobile phone users' nonpersonal activity within the network. In the experiments in this section, each histogram is constructed over the counts of events per event type. Under non-anomalous circumstances, human behavior is mostly regular, that is, there is no major change in the relative proportion of the counts for each event type and thus a low value for relative entropy between distributions. However, large-scale anomalous events such as natural catastrophes disrupt movement patterns and lead to sudden changes lasting over an extended period in time and cause a change in the distribution of the events that is visible through high relative entropy values.

As a real-world example of an anomaly relating to a human event, we consider the flood that took place in Geneva on May 2, 2015. This event caused a significant change in the movement patterns of telecommunication network users as several

bridges had to be closed and users had to pick new routes to get to their usual destinations. The change in behavioral patterns implies higher relative entropy in the anomalous scenario (on May 2) compared to the baseline scenario. This baseline is constructed from other days with no known major disruption.

In the experiment summarized in Fig. 16.5, the relative entropy $D(P||Q)$ is computed on a per-cell level between histograms of adjacent 1 h windows, yielding one value per cell per hour, filtered to cells within the city of Geneva. The two distributions that are being compared are hence P_t, that is, the distribution of the event types in all events during the 1 h period before the time t; and $Q_{t-\Delta t}$, where Δt is 1 h, that is, the distribution of the event types occurring between 2 h and 1 h before t.

Figure 16.5a shows the distribution of the per-cell mean relative entropy values; that is, the mean overall hourly values during one day. For the baseline, the mean entropy overall cell's values $\underline{D}$, which is approximately 0.15, determines the ranges for the bins on the x-axis. The y-axis shows the relative proportion of cells' daily mean relative entropies falling into the range given on the x-axis.

The results show that the majority of cells have mean relative entropy values in the low ranges both on normal and anomalous days. Normal days' entropy values are more strongly centered on $2\underline{D}$ with lower variance, and there are few cells with means greater than 0.3 (or $2\underline{D}$), unlike on the anomalous day. Figure 16.5b displays the difference between the proportion of cells on baseline and anomalous days for each range. The figure supports the previous observation: We see fewer cells with D within the lower ranges [0.07,0.3) on the anomalous day than in the baseline, but for the higher ranges, starting at a mean relative entropy of 0.3, we observe an increase in D for the anomalous day, and thus relative entropy can be used in combination with a suitable threshold to detect anomalies in the form of human behavior changes. The high frequency of relative entropy values within the lowest bin on the anomalous day is due to inactivity in some cells that were in areas no longer accessible due to the flood, as the relative entropy between two windows with no activity will be very low.

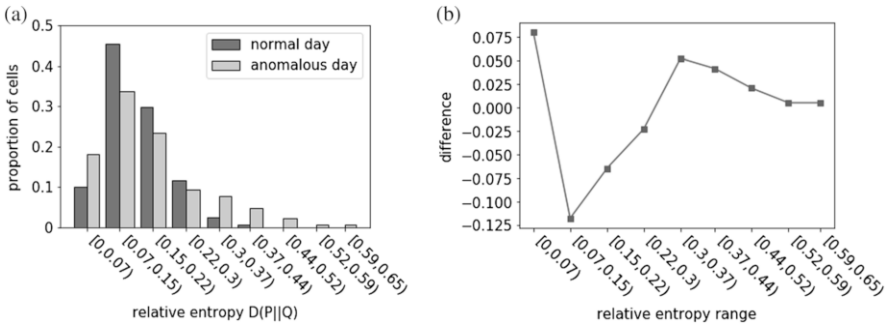

Fig. 16.5 Distribution of cells' mean relative entropy between adjacent windows throughout one day. Baseline from multiple normal days compared to the anomalous scenario on May 2, 2015. (**a**) Distribution of the proportion of cells where the mean relative entropy falls into the respective range, for the baseline and the anomalous day. (**b**) Difference between the proportion of anomalous cells' means and the proportion of baseline cells' means per range

4.1.2 Pearson Correlation Accuracy

The physical network monitoring probes from which we obtain the data are located close to Swisscom's core network and each probe is responsible for monitoring an aggregate of cells. Probe failures are therefore detectable by looking at large-scale changes. As the probes are separate per interface and one failure only affects one interface, these changes can be observed by maintaining global Pearson correlation coefficients between two interfaces. In the non-anomalous scenario, the data streams coming from different telecommunication interfaces (2G and 3G, specifically) are highly correlated in the counts of users on the interfaces during a period in time.

Since we have no data of a failure scenario, we resort to simulations that aim to imitate a realistic failure of a network monitoring probe. We consider two realistic types of failure scenarios: *hardware failures*, where one probe ceases to transmit events for the area it monitors, and *software failures*, where a gradually increasing duplication of transmitted events takes place.

Abrupt Infrastructure Failures typically result from hardware failures. In this case, no events get transmitted from the respective monitoring probe, which leads to lower user (i.e., input) counts since no users are counted for cells in the area monitored by the failed probe. For simulation, we filter out a proportion of the received events after a certain time, resulting in a sudden drop in the Pearson correlation $r(X, Y)$ during one window.

Figure 16.6 displays the results of computing the Pearson correlation $r(X, Y)$ between windows over the counts of distinct users on the 2G voice (A) and the 3G voice (IuCS) streams during 1 h, with one count every 10 s and one correlation score

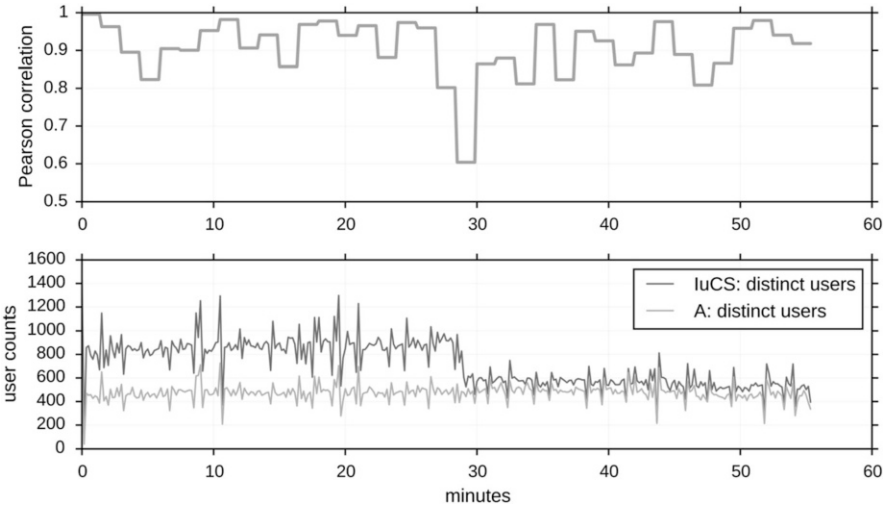

Fig. 16.6 Simulating the impact of the cessation of data transmission from one probe, that is, losing a fraction of the events on one interface, on the global correlation between the distinct user counts on the A and the IuCS stream

every 90 s. X and Y are vectors, each containing nine counts, corresponding to the counts in one 90 s window on the stream originating from the A and IuCS interface, respectively. After 30 min, we filter out one-third of the IuCS stream's events.

The results show that both before and after the failure, the correlation between the counts over the two streams is high (ranging between 0.8 and 1). At failure time, there is a momentary decrease of the correlation to 0.6 during one 90 s window. Because the event loss is uniform, the correlation remains high even when parts of the stream are lost, but the score is impacted at the time of change. This momentary decrease of 0.3 is significant considering the baseline's mean of 0.91 having a standard deviation σ of 0.06. We detect anomalous cases by identifying correlation coefficients that deviate from the average by $k\sigma$; in our deployment, picking k to be 4 yields an accurate detection of infrastructure failures in time.

Gradual Infrastructure Failures occur in the software running on monitoring probes, which is vulnerable to manual misconfiguration. In previous real-world failure cases, events have been transmitted multiple times, that is, duplication occurred. The amount of duplication increased gradually over time, making them hard to detect through visual monitoring tools.

The simulation of gradual increase in distinct user counts (e.g., as a result of re-emission of previous events) has been achieved by gradually increasing the counts on the IuCS interface after a period of 20 min. Figure 16.7 shows the counts as well as the Pearson correlation coefficients. We observe that, although the counts on the IuCS interface increase greatly (significantly faster than in a realistic scenario), the correlation remains within the range that we consider highly correlated (greater than 0.8).

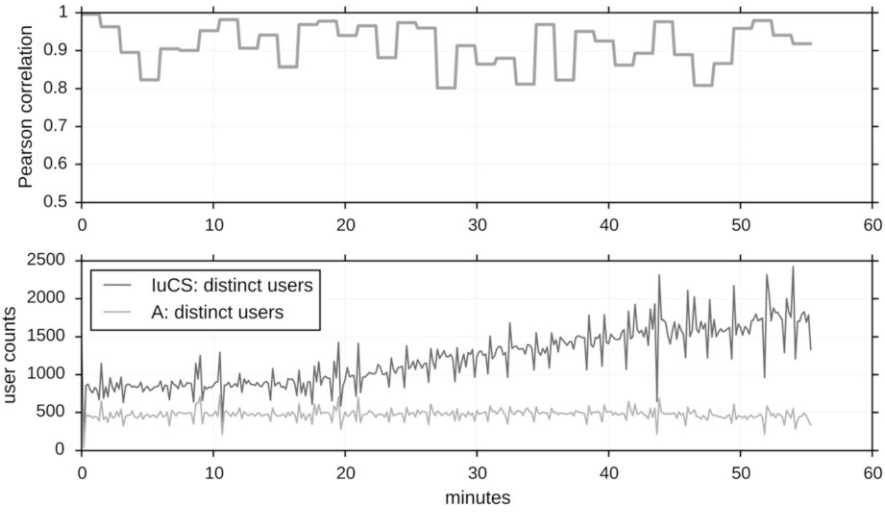

Fig. 16.7 Simulating the impact of a gradual increase in event duplication as an increase in the distinct user counts on the global correlation between the A and the IuCS stream in terms of the number of distinct users

4.2 Comparison to State-of-the-Art Anomaly Detection Techniques

In the following, we apply state-of-the-art anomaly detection methods (Münz et al. 2007; Young et al. 2014) to our data and the specific anomaly detection scenarios—real-world events—in order to evaluate their applicability and to compare our methods to these in the following section.

4.2.1 Volume of Telecommunication Activity

One approach to detecting local anomalies over telecommunication data is through quantifying the volume of activity on a local scale (Young et al. 2014). We count the number of events per 30 min window in a scenario where a fire at the Lausanne train station disrupted the train traffic, and compare the counts to average counts on normal days in Fig. 16.8.

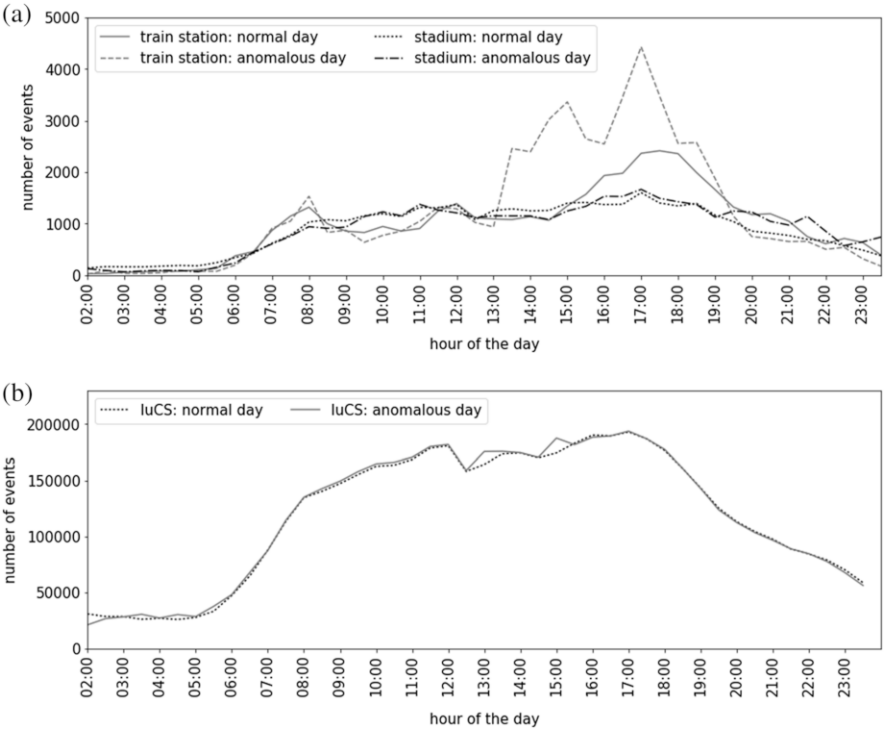

Fig. 16.8 Count of events per 30 min, per cell and per city, comparing between a baseline for normal days and an anomalous day. (**a**) Activity around two IuCS base stations in Lausanne: train station and Olympic stadium. (**b**) Activity on the IuCS interface in the entire area of the city of Lausanne

For monitoring on a per-cell level, two locations are considered: one at the Lausanne train station and another at the Olympic stadium, located approximately 2 km from the train station, which is not an important point for railway traffic. Additionally, we attempt to detect the event at a coarser granularity for the entire city of Lausanne.

It can be observed in Fig. 16.8a that there is an increase in activity in the afternoon of the anomalous day around the location of the event but no anomalous activity at the nearby location. For comparison, Fig. 16.8b shows the event counts for the entire city of Lausanne. There is no observable difference between the normal and the anomalous day at this scale. From these results we reason that the event is only recognizable by monitoring counts at a very fine granularity, which is costly in terms of memory and computations considering the large number of cells.

4.2.2 *k*-Means Clustering

As clustering is a common method for anomaly detection (Münz et al. 2007), we validate the accuracy of our system against an approach using *k*-means. Once more we consider the event where there was a fire at the Lausanne train station. In order to be comparable to our previously described system, we summarize temporal windows of data by computing features that are similar to the stream summaries used in computing relative entropy and Pearson correlation; specifically, we consider event type distributions on the A and IuCS interfaces as well as user counts. We obtain 15 features for each nonoverlapping window of 10 min. We choose *k*, the number of clusters, to be 7 by empirically evaluating different values and choosing a number where additional clusters do not add information.

For the purpose of analyzing the clusters in a visual manner, the high-dimensional space is reduced to three most discriminative dimensions.

For each day of data, a separate *k*-means model has been built and the centroid for each cluster of each day is shown in Fig. 16.9. Most cluster centroids are positioned around a diagonal line. For the anomalous day, two outlying cluster centroids with a higher distance to the diagonal can be observed. Thus, *k*-means is able to detect known anomalies, but it requires many processing steps and iterations over the dataset to find and analyze the results, making it too slow for real-time applications (minutes as opposed to seconds).

4.3 Scalability of the Algorithms

We now turn to evaluating the scalability of our anomaly detection system. Since the focus of this work is on real-time anomaly detection, this experiment is conducted on the stream implementation. Parallel processing is a key feature of big data infrastructures. In order to evaluate the scalability of our implementation, we conduct an experiment with a varying number of processing executors and an approximately constant data rate. The computation for any micro-batch, including any aggregates of

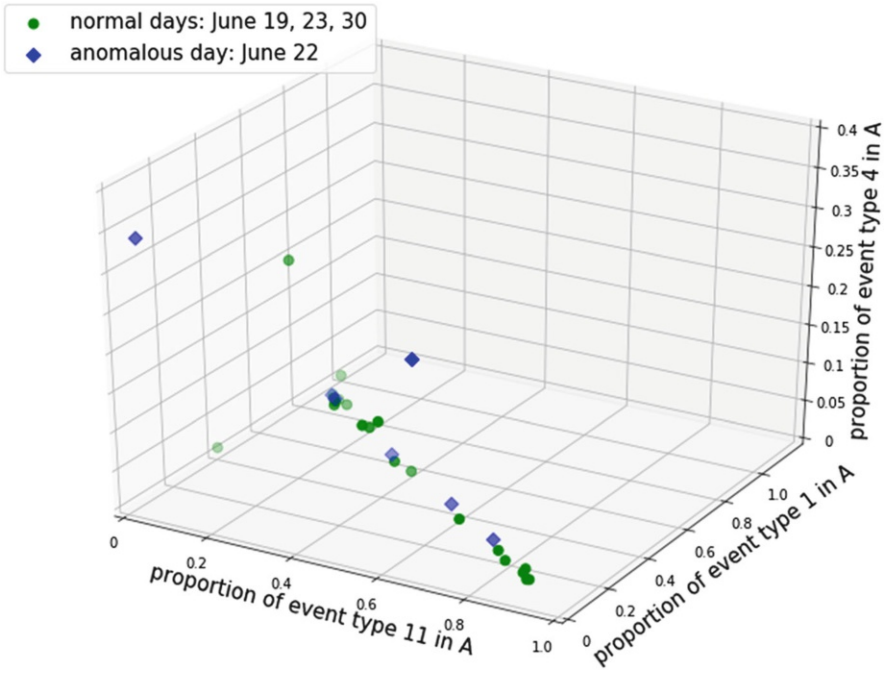

Fig. 16.9 Cluster centroids in *k*-means for models built from the data for normal and anomalous days, reduced to the three most discriminant dimensions for cluster centroid positions

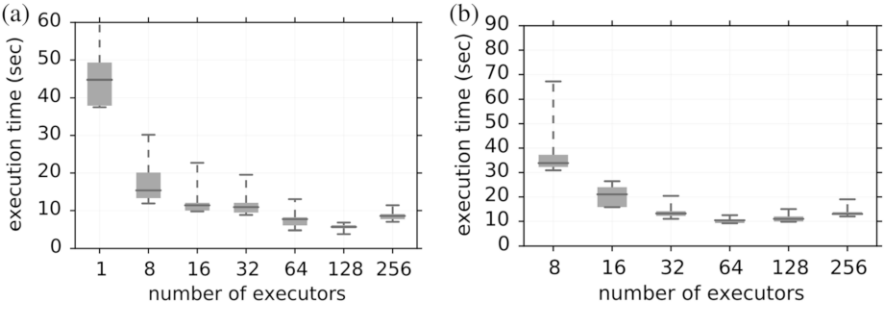

Fig. 16.10 Streaming mode: micro-batch processing times per number of executors. (**a**) Computing relative entropy on the A stream with a micro-batch duration of 60 s. (**b**) Computing Pearson correlation between the A and IuCS streams with windows of 90 s

micro-batches in the form of windowing, should require only sufficiently short periods of time; on average well below the duration in which data are collected, that is, the length of the micro-batch or window itself.

The experiment in Fig. 16.10a displays the processing time for the relative entropy algorithm with 60 s micro-batches given an increasing number of executors.

This experiment shows that the algorithm terminates on average below the duration of a micro-batch, such that the relative entropy on the stream is computed in real time with any configuration.

The experiment in Fig. 16.10b displays the micro-batch processing time as a function of the number of executors for computing the Pearson correlation. In this case, the processing of the streams using fewer than eight executors was consistently unable to terminate within the given 90 s window duration and are therefore not shown.

Both experiments show that the execution time and the variance decrease by adding more executors, indicating that the algorithms are scalable to increasing parallelism. We observe that the execution time increases after reaching an optimum at 128 and 64 executors, respectively, due to network overhead caused by node management and data shuffling.

5 Discussion

In Sect. 4, we experimentally evaluated the accuracy and scalability of two proposed measures for anomaly detection, relative entropy and Pearson correlation coefficients. We showed that one of our proposed measures—relative entropy—allows us to detect anomalous events related to users' mobility in the form of a high proportion of higher values for relative entropy. We found that Pearson correlation is not a suitable measure for detecting anomalies that are of a gradual nature. On the other hand, when simulating an abrupt change in the form of a hardware component's failure, Pearson correlation coefficients show a significant decrease against the highly correlated baseline for normal behavior. Hence, by continuous monitoring over the available data streams, the detection of different types of anomalies is feasible by using both proposed measures simultaneously.

When comparing to baseline methods for anomaly detection, specifically count-based anomaly detection and clustering, our two proposed methods were more efficient and more accurate for detecting anomalies given a threshold.

In order to evaluate the suitability and compare the different approaches, four dimensions were considered that are relevant for choosing a suitable long-term anomaly detection technique. These dimensions for comparison are the following: (1) *gradual change*: ability to detect changes happening gradually over time, such as human-behavior-induced events or gradual failures; (2) *abrupt change*: ability to detect abrupt changes; (3) *spatial granularity*: anomaly detection accuracy for different levels of geographical granularity—global, regional, and local—with an impact on the efficiency; (4) *efficiency*: focusing on stream processing, this aspect includes the processing duration, the scalability to parallelization and to larger quantities of data, as well as the efficiency of the algorithms in terms of resource usage.

5.1 Type of Change: Gradual or Abrupt

We have evaluated the suitability of our metrics for detecting known possible anomalies. Relative entropy and both state-of-the-art methods are suitable for detecting gradual change. Both Pearson correlation and measuring the volume of activity allow us to detect anomalies as a result of abrupt change, for example, from infrastructure failures. This results from the quantification of activity in absolute terms for both methods.

Relative entropy would not be suited for detecting abrupt change, as it considers the proportions of event types, which would remain the same due to uniform downscaling in a hardware failure scenario.

5.2 Spatial Granularity

Both proposed techniques, relative entropy and Pearson correlation, are able to detect anomalies on a local, regional, and global scale. For Pearson correlation, the granularity depends on the area that is affected by the failure; typically, larger aggregates of cells. For the sake of efficiency, a regional partitioning or global computation of the correlation is preferable.

The volume-based approach is limited to detecting anomalies locally at the cell level, that is, even major anomalies are not visible on the regional scale or at nearby cells. This is an inefficient approach that makes it difficult to detect true anomalies.

While k-means clustering did produce observable outliers for anomalous data on a regional scale, this approach was not able to produce distinguishable outliers at a global scale for the same event that we were able to observe on a global scale using relative entropy.

5.3 Efficiency

We evaluated the computation times for our proposed methods in streaming. Using the ideal number of executors—128 and 64 respectively—we reach median processing times far below the micro-batch duration (respectively, 55 and 80 s less), such that fault recovery is feasible within the given time. We also obtained a low variance with an increased number of executors, which helps to guarantee an upper bound for the processing time under normal circumstances and less vulnerability in case of failures. When parallelizing to more than the ideal number of executors, the network connections become a bottleneck.

While the event counting approach has only been implemented over batch data for evaluation using known anomalous events, its implementation—counting the number of distinct events during a period—is simpler than our proposed methods.

On the other hand, the need to maintain a large number of distinct values, one for each cell, makes this approach inefficient in terms of resource usage.

k-means is a relatively costly algorithm that requires a large number of iterations over the data. In addition to the issues with efficiency, it should be further noted that there are some limitations to performing k-means clustering on the monitoring data. The monitoring data are essentially multidimensional time series, which are highly correlated. The network traffic logs that were used in the related work proposing this method, on the other hand, have discrete entries.

In summary, by comparing our proposed system against the existing methods, it can be observed that relative entropy detects similar anomalies to *k*-means—in large parts due to the similar choice of features. Relative entropy offers an improvement over *k*-means regarding the efficiency (relative entropy requires only one iteration over the data) and regarding the ability to detect anomalies at a global scale.

Correspondingly, Pearson correlation and the approach using event counts detect similar types of anomalies. It should however be pointed out that by counting the number of total events without correlating between streams we can expect to observe false positives, for example, as a result of highly local phenomena that do not correspond to real events.

5.4 Limitations

As a result of the lack of ground-truth data, especially anomalous events, our work has some limitations and aspects that could not be addressed. When identifying anomalies, we used parametric thresholds as deviations from the standard deviation of the measure. These thresholds and parameters vary greatly between the different events and setups, meaning that the choice of parameters needs to be determined within the given context. It is at this time difficult to determine the significance of the respective differences when comparing between anomalous data and a baseline. In order to draw meaningful conclusions and perform statistically sound automated anomaly detection, further validation and training data are necessary.

6 Related Work

6.1 Data Streams

Streams frequently have the property that data arrive at a high velocity, posing problems in the areas of transmitting input to a program, applying functions to large input windows, and storing data, both temporarily and long term (Muthukrishnan 2005). Statistical metrics and probabilistic data structures that represent sliding windows in streams have been proposed for summarizing streams. Datar et al. (2002) introduce approximate stream summary statistics for sliding windows.

Since regularities in streams may evolve over time, the issue of data decay is handled by giving more weight to recent objects, aggregating previous windows, and eventually discarding older data. The authors store information using exponential histograms. This data structure uses timestamps as the bins and the count of an item in the stream as the value for each temporal range. While their work is suitable for computing approximate statistics with bounded errors to summarize aspects of the content of a stream, they do not address the issue of detecting change.

Frugal streaming was introduced by Ma et al. (2013) providing first-order statistics over data streams. These frugal streaming algorithms are able to treat streams one item at a time, requiring no memory of previous data, and only a maximum of two pieces of information are maintained in memory. Flajolet et al. (2007) proposed the *HyperLogLog* structure, a sketch suitable for counting distinct elements with bounded errors in a single pass over the data, making the algorithm highly suitable for stream data. While very simple and efficient, both approaches are restricted to streams of a single dimension.

6.2 Anomaly Detection on Time Series and Data Streams

In this chapter, we define and identify data quality issues as anomalies, that is, deviations from the expected model of the data. Related work on anomaly detection for time series data can also be applied to data streams. While time series do not always require real-time systems, both time series and data streams provide in fact temporal data, as data streams naturally carry the notion of time (Papapetrou et al. 2012) (either by means of time of arrival of a data point or from a timestamp associated with it). A number of techniques have been proposed to detect anomalies in multidimensional data streams or for multidimensional time series data.

A general method for detecting anomalies in datasets consisting of distributions is proposed by Lee and Xiang (2001). The authors use relative entropy amongst other information-theoretic measures to detect anomalies. Their measures are suitable for describing the characteristics of a dataset, but they do not address the data stream notion, requiring real-time computability. Based on the proposed information-theoretic measures, Dasu et al. (2006) present an approach to detect sudden changes in multidimensional data streams. In their approach, multidimensional stream instances are represented as *kdq-trees* (a combination of *kd-trees* and *quadtrees*), while relative entropy is used as a similarity measure. To detect changes on unknown distributions, the method resamples the data from one window using the so-called bootstrap technique in order to obtain expected distributions of the data. The relative entropy between the distributions gives a bound for the relative entropy between different windows (under the assumption that the data originate from the same distribution), allowing for a statistically sound detection of significant changes. The authors propose two different window comparison models. The first model compares adjacent windows, which is well-suited for detecting abrupt changes. The second model compares a sliding window to a previous window, which is

convenient to detect more gradual changes. We use similar techniques to measure changes between successive time windows over multidimensional data streams. However, we do not rely on complex and multidimensional data structures that would be very difficult to distribute and efficiently update on clusters of machines.

Zhang et al. (2004) propose a solution that detects outliers in multidimensional data. The proposed approach performs anomaly detection by measuring the distance of a data point in various subspaces. The authors show that for multidimensional data, changes may be observable on one dimension, over a subset of dimensions, or overall. However, the proposed techniques based on indexing and subspace pruning are not applicable to real-time scenarios due to the high number of iterations over the data.

Young et al. (2014) detect and classify emergency and nonemergency events using annotated telecommunications network data, specifically, call detail records. Similarly to our work, they compare normal and anomalous days to detect deviations from a baseline representing average behavior. The known events in their dataset are detectable when plotting the call volume throughout a day for the anomalous event compared to an average for this day of the week. They observed that events change the users' activity at the location of the event, such that the difference—in terms of activity profile—to nearby cells, where activity is as normal, increases. Unlike our proposed system, they use a metric that observes the anomaly only at the closest cell tower to the known event. Their work uses autoregressive hidden Markov models in order to classify time frames and detect the precise onset of an event. Furthermore, the applied matrix factorization is computed on data at rest and not in real time, unlike our high-velocity streams.

Clustering algorithms are frequently used to detect outliers or anomalous instances that have been assigned to anomalous clusters. In their survey of anomaly detection techniques for temporal data, Gupta et al. (2014) note that different clustering algorithms, such as *k*-means, can be used to detect point outliers, as well as to create dynamic models for anomaly detection in streaming. Münz et al. (2007) detect anomalies from network monitoring data as part of an intrusion detection system by using the *k*-means clustering algorithm. Instances are created by computing features on the traffic data per time interval. *k*-means forms *k* distance-based clusters based on unlabeled training data and assigns normal and anomalous instances each to a different cluster. In their setting, *k* is configured to 2 in order to assign normal and anomalous instances each to a different cluster. The clusters' centroids are then deployed to classify new instances as either normal or anomalous. This is a highly generic approach that is suitable for many scenarios, however, it is significantly slower than the approach presented in this chapter.

7 Conclusion

This chapter presented a system for the purpose of performing anomaly detection over high-velocity streams of telecommunications monitoring data. In the implementation of the system, we leveraged general measures from statistics and information theory and applied them for the purpose of anomaly detection. These measures have been implemented in Spark and Spark Streaming, thus enabling data quality testing in the form of anomaly detection both on data streams and on data at rest. The implementation is flexible and robust in terms of detecting anomalies that occurred on different spatial and temporal scales, since we can consider any subset of the network topology, as well as varying subsequences of the stream or the batch data. We showed that the implementation scales with the number of parallel nodes until reaching an optimum.

The results of the empirical evaluation show that (1) *relative entropy* is suited to detect gradual changes in human behavioral patterns caused by a disruption at one point in time, with the effect gradually increasing and lasting for multiple hours; (2) *Pearson correlation* enables the detection of abrupt hardware failures but does not detect any gradual changes; and (3) compared to state-of-the-art techniques, the proposed system for anomaly detection is superior in terms of accuracy and efficiency.

7.1 Lessons Learned

In implementing and testing this project using Spark and Spark Streaming, we learned the importance of understanding the underlying system, that is, the way data are treated by Spark, in order to fully optimize the implementation. One important point is making sure that data processing is efficient, particularly during streaming, as data loss results when micro-batches are not processed within the duration of the data collection, as the system can only buffer a limited amount of data to be processed.

From the different measures that have been implemented and compared, we observed that Pearson correlation and the simplistic approach in the form of event counting are well-suited for abrupt changes. On the other hand, relative entropy, when computed between the proportion of event types as done in this work, does not recognize abrupt changes, but is better at handling gradually occurring change, especially when comparing against a recent ground-truth non-anomalous sample.

This system does not yet support fully automated anomaly detection. Tests were done using recent manually annotated ground-truth for non-anomalous data. Thresholds and parameters for alerting about anomalies should be based on bigger sets of ground-truth data in order to be more robust and reliable. For example, one might consider using resampling techniques to determine the statistical significance of an anomalous measure given the previous information. Another option could be the use of machine learning techniques, such as classification rules, that are learned from annotated ground-truth data. As a side effect, this would allow the system to automatically output the type of anomaly along with the alert.

References

Dasu, T., Krishnan, S., Venkatasubramanian, S., & Yi, K. (2006). An information-theoretic approach to detecting changes in multi-dimensional data streams. In *Proceedings of the 38th Symposium on the Interface of Statistics, Computing Science, and Applications*.

Datar, M., Gionis, A., Indyk, P., & Motwani, R. (2002). Maintaining stream statistics over sliding windows. *SIAM Journal on Computing, 31*, 1794–1813.

Dean, J., & Ghemawat, S. (2004). MapReduce: Simplified data processing on large clusters. In *OSDI'04: Sixth Symposium on Operating System Design and Implementation*.

Flajolet, P., Fusy, É., Gandouet, O., & Meunier, F. (2007). HyperLogLog: The analysis of a near-optimal cardinality estimation algorithm. In *Conference on Analysis of Algorithms, AofA*.

Gupta, M., Gao, J., Aggarwal, C. C., & Han, J. (2014). Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics, 22*, 79–86.

Lee, W., & Xiang, D. (2001). Information-theoretic measures for anomaly detection. In *IEEE Symposium on Security and Privacy* (pp. 130–143).

Ma, Q., Muthukrishnan, S., & Sandler, M. (2013). Frugal streaming for estimating quantiles. In *Space-Efficient Data Structures, Streams, and Algorithms* (Vol. 8066, pp. 77–96). Berlin: Springer.

Marz, N., & Warren, J. (2013). *Big Data: Principles and best practices of scalable realtime data systems*. Greenwich, CT: Manning Publications Co.

Münz, G., Li, S., & Carle, G. (2007). Traffic anomaly detection using k-means clustering. In *GI/ITG Workshop MMBnet*.

Muthukrishnan, S. (2005). Data streams: Algorithms and applications. *Foundations and Trends in Theoretical Computer Science* (Vol. 1).

Papapetrou, O., Garofalakis, M., & Deligiannakis, A. (2012). Sketch-based querying of distributed sliding-window data streams. In *Proceedings of the VLDB Endowment* (Vol. 5, pp. 992–1003).

The Apache Software Foundation. (2015). *Spark Streaming programming guide*. Retrieved from http://spark.apache.org/docs/1.0.0/streaming-programming-guide.html

Young, W. C., Blumenstock, J. E., Fox, E. B., & Mccormick, T. H. (2014). Detecting and classifying anomalous behavior in spatiotemporal network data. In *The 20th ACM Conference on Knowledge Discovery and Mining (KDD '14), Workshop on Data Science for Social Good*.

Zaharia, M., Chowdhury, M., Das, T., Dave, A., McCauley, M., Franklin, M. J., et al. (2012b). Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*.

Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. In *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*.

Zaharia, M., Das, T., Li, H., Shenker, S., & Stoica, I. (2012a). Discretized streams: An efficient and fault-tolerant model for stream processing on large clusters. In *Proceedings of the 4th USENIX Conference on Hot Topics in Cloud Computing*.

Zaharia, M., Das, T., Li, H., Hunter, T., Shenker, S., & Stoica, I. (2013). Discretized streams: Fault-tolerant streaming computation at scale. In *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*.

Zhang, J., Lou, M., Ling, T. W., & Wang, H. (2004). HOS-Miner: A system for detecting outlying subspaces in high-dimensional data. In *Proceedings of the 30th International Conference on Very Large Databases*.

Chapter 17
Unsupervised Learning and Simulation for Complexity Management in Business Operations

Lukas Hollenstein, Lukas Lichtensteiger, Thilo Stadelmann, Mohammadreza Amirian, Lukas Budde, Jürg Meierhofer, Rudolf M. Füchslin, and Thomas Friedli

Abstract A key resource in data analytics projects is the data to be analyzed. What can be done in the middle of a project if this data is not available as planned? This chapter explores a potential solution based on a use case from the manufacturing industry where the drivers of production complexity (and thus costs) were supposed to be determined by analyzing raw data from the shop floor, with the goal of subsequently recommending measures to simplify production processes and reduce complexity costs.

The unavailability of the data—often a major threat to the anticipated outcome of a project—has been alleviated in this case study by means of simulation and unsupervised machine learning: a physical model of the shop floor produced the necessary lower-level records from high-level descriptions of the facility. Then, neural autoencoders learned a measure of complexity regardless of any human-contributed labels.

In contrast to conventional complexity measures based on business analysis done by consultants, our data-driven methodology measures production complexity in a fully automated way while maintaining a high correlation to the human-devised measures.

Lukas Hollenstein and Lukas Lichtensteiger have contributed equally.

L. Hollenstein (✉) · L. Lichtensteiger · T. Stadelmann · M. Amirian · J. Meierhofer · R. M. Füchslin
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: hols@zhaw.ch; licn@zhaw.ch; stdm@zhaw.ch; amir@zhaw.ch; meeo@zhaw.ch; furu@zhaw.ch

L. Budde · T. Friedli
Institute of Technology Management, University of St. Gallen, St. Gallen, Switzerland
e-mail: lukas.budde@unisg.ch; thomas.friedli@unisg.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_17

1 Introduction

One of the most important aspects of a data science project is the data itself. Its availability is *the* necessary condition for any successful data product that at its core relies on a successful analysis of this data. This fact seems obvious enough to be considered a truism, but nevertheless is the proverbial "elephant in the room" of countless data analytics projects. A recent survey among 70 data scientists showed that on average 36% of their projects have been negatively impacted by the unavailability of the data to be analyzed. The following paragraphs summarize the results from this poll.[1]

The survey has been conducted among the associates of the ZHAW Datalab.[2] The typical negative impact reported has been a delay of the project in the order of months, sometimes leading to a change of scope and goal up to the cancellation of the complete project (see Fig. 17.1). "Unavailability of data" here refers to the situation in which a data science project has been started under the requirement that specific data will be available at a certain point in the timeline of the project. The analysis of this data is the main part of the project and crucial to reach its goal, and all reasonable measures have been taken upfront to secure its availability. According to the survey, failing this requirement has usually one of the following reasons:

- *Measurement issues:* the data was meant to be collected in the course of the project but resource problems for staff to conduct measurements, the absence of specific events to be measured, or the unavailability of respective hardware hinder its collection.
- *Privacy issues:* the data is there but cannot be shared among the project partners due to new or unforeseen legal constraints.
- *Quality issues:* the raw data is available and shareable but the measurements themselves or the human-provided labels lack the required precision.

The effect of the unavailability of data is manifold: usually, it not only stretches the duration of an affected project by several weeks to years, it also leads to much more work on data curation at the expense of less time for the actual analysis and decreases the motivation on all parts of the project team, as was mentioned several times in the survey. It forces the data scientist to revert to suboptimal methods (with respect to the aspired project goal), and usually leads to lowered overall project goals up to a total cancellation of the endeavor. The matter is even more severe if data is not absent altogether, but some crucial parts are missing or its quality is far below the necessary standard. This ultimately leads to the same issues as outlined above; the

[1] While the survey and its evaluation have been conducted under controlled circumstances specifically for this chapter, we explicitly point out the small return rate of 10 questionnaires and hence the limited generality of conclusions; we report them because of their good correlation with our overall impression from numerous experiences with colleagues inside and outside our respective institutions.

[2] See www.zhaw.ch/datalab for a list of associates.

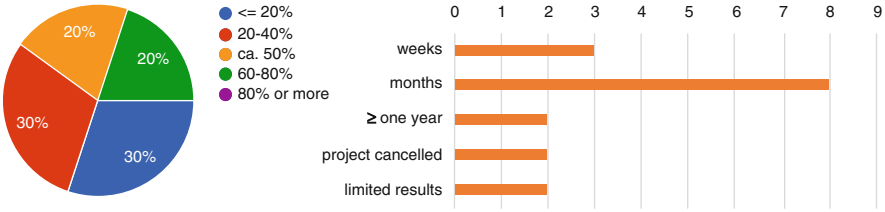

Fig. 17.1 (Left) Survey results for the question "Which percentage of projects you worked on has roughly been affected by the unavailability of the data?". (Right) Answers to the question "How long have the affected projects typically been delayed (multiple answers allowed) by the unavailability of data?." Overall, the survey produced a 17% return rate (12 people), out of which 10 answered the above questions

more subtle form of the problem, however, may hinder project management to take appropriate measures early on, as was pointed out by several participants in the survey.

In this chapter, we provide a twofold contribution: first, we discuss a specific approach to partially overcome the above-mentioned issues of data unavailability by producing detailed data for machine learning out of a simulation model informed only by high-level process descriptions. Second, we introduce a novel measure of business operations complexity that can be evaluated fully automatically based on data and holds potential to inform business owners on how to reduce unwanted complexity. It exploits the idea that complexity and compressibility are highly anticorrelated (Schmidhuber 2008). Our case study from the area of Industry 4.0 is motivated by the assumption that in the time of growing mass customization in production (Fogliatto et al. 2012), variability in the product range leads to increased production complexity, which is a major driver of costs. (Note that there are scenarios where this assumption does not hold, e.g., cases where the variability of the product range enables the compensation of variabilities in the flow of resources. For more about this discussion, see Sect. 2.)

The goal of this case study hence has been twofold: first, to measure the inherent complexity in the production processes of certain industrial goods based on the analysis of production data; second, based on the complexity measurement, to propose changes to these processes that reduce the complexity while being feasible from a business and operations perspective. However, the necessary raw data from the shop floor turned out to be largely unavailable in the expected form.

In this situation, the methodology of coupling simulation and learning (Abbeel 2017) proved useful. Simulating the known physical properties of the production processes on an abstract level leads to many "observations" of the production of goods. Training an unsupervised learning algorithm like a neural autoencoder (Goodfellow et al. 2016) on this data converts the model from a physics-based simulation to a machine learning model with similar content, but different properties. The interesting property of the learned model with respect to the goal of the case study is the following: it has learned a compressed representation (Bengio et al. 2013) of the patterns inherent in the data, which is in the best case (a) able to generalize

(Kawaguchi et al. 2017) beyond the limitations and discontinuations of the abstract simulation; and (b) allows conclusions on how the original processes might be compressed (i.e., simplified) out of an analysis of its own way of compressing the learned information. Note that our two contributions—the suggestion to use simulation to overcome data scarceness, and the novel complexity measure—are independent of each other and only linked by the necessity of the case study under consideration.

The remainder of this chapter is organized as follows: Sect. 2 introduces the case study with its business background, showing the necessity and merit of a learned complexity measure. Section 3 details our methodology of linking simulation to unsupervised learning. Section 4 discusses the results of the case study before Sect. 5 concludes with several lessons learned on the problem of the unavailability of the analysis data in general.

2 Case Study: Complexity Management in Business Operations

The problem statement and solution approach described were applied in an industrial shop floor environment of a large international enterprise based in Switzerland. The factories are challenged with decisions about expanding the product portfolio for a higher degree of differentiation and an extended skimming of market segments, which is expected to yield higher revenues. However, it is obvious that even in the context of a modular production strategy in which new product versions are based on existing modules, increasing the product portfolio results in an increased complexity of the business operations in production, therefore resulting in increased production costs. Thus, there is a trade-off between higher revenue and higher costs. The availability of a tool to assess the complexity of a given production scheme based on measurable input data can provide a relevant support for the corresponding management decisions. Such a tool matches the definition of a so-called data product in the sense that it generates value from data for the benefit of another entity (i.e., the shop floor management) by the application of data science skills (Meierhofer and Meier 2017).

Product variety or complexity increase is often the outcome of the differentiation strategy of companies to enter market niches and to achieve higher revenues and market shares (Tang 2006). Beside the fulfillment of individual customer requirements and the outperforming of competitors (Lancaster 1990), researchers as well as practitioners in various studies reveal that an increase of product complexity does not equally lead to higher profitability and sustainable growth (Ramdas and Sawhney 2001). On the contrary, complexity is often associated with various negative effects that come attached and are built up over years (Fisher et al. 1999; Kekre and Srinivasan 1990). Several researchers claim the existence of an optimal level of product complexity that companies need to approach (Budde et al. 2015; Orfi et al.

2012; Krishnan and Gupta 2001). But the definition of the optimal level of complexity is not a trivial task because multiple factors need to be considered (Fisher et al. 1999; Budde et al. 2015). Product portfolio decisions (e.g., new product variants or new product developments) affect all steps along the value-chain, for example, development, production, and even service operations. Even minimal changes at the product architectures can have multiple impacts on the production or service side. This is also why decision-making around the product portfolio, such as decisions for new product development projects, product variants, or product architectures, is seen as one of the most critical tasks of management due to its uncertain and changing information, dynamic opportunities, and multiple and strategic considerations from different stakeholders along the value-chain (Closs et al. 2008).

Managers struggle to evaluate complexity out of a broader multifunctional perspective due to a lack of system interdependency knowledge and information asymmetries (Budde et al. 2015). This results in decisions that may be optimal for one functional perspective but not always optimal for the company along the product life cycle (Fisher and Ittner 1999; Closs et al. 2008; Lancaster 1990). Closs et al. (2008) recognized the need of metrics that measure the relational and combinatorial dimensions of complexity. These metrics should be able to predict various performance outcomes. Developing such a metric and deriving decision support from it for the case at hand was a central goal of our work.

In the given case study of the shop floor, data was available on the number of product alternatives and how they are composed as well as on the number of production steps required to produce those product types. However, within the practically given time frame of the project, it was not possible to gather the detailed data of the shop floor, for example, data about the sequence of the raw material or semifinished products across the machines or data about the load fluctuations of the individual machines. Higher effort than originally expected would have been necessary to generate all required information out of the different IT systems: the information was not directly available and not connected. Additionally, it was not possible to conduct different interviews with product managers as well as with experts from production or supply-chain departments due to organizational constraints.

Still, the project pursued the goal to make the resulting complexity of different production schemes measurable and thus to enable the assessment of different scenarios of product and production constellations. As stated in the introduction, the approach chosen here and explained in the following sections is based on training an unsupervised learning algorithm on data from simulations, which in turn are based on the scarcely available data and expert knowledge, thus transforming the physical model into a machine learning model that can provide insights into the inherent complexity on a more abstract level.

3 Linking Simulation and Learning

Even if there is no or insufficient data available to successfully train a machine learning model, some knowledge of the underlying nature of the system is often available from the domain experts. Here we discuss how for our case study we define a simulation model that can provide the data needed for a proof of concept of our complexity measure based on a neural autoencoder (Goodfellow et al. 2016).

3.1 Simulation Models Can Provide Data

Simulations, as opposed to machine learning, are based on expert knowledge of the dynamics and rules of the complex system under analysis (Zeigler et al. 2000). Thus, in the absence of observations of the system (the desired data) we can simulate the behavior of the system by means of modeling its dynamics, running it (maybe many times), and gathering the observational data. Clearly, a simulation model needs data, too, but typically that data is of a higher level of abstraction, for example, the number of processing steps and the duration of each step for a given product. So, even if we do not have exact data for some of these values, like the durations, we can make some reasonable assumptions by talking to shop floor domain experts.

Many different simulation modeling approaches are known and the choice depends strongly on the system to be described and the knowledge we have about it (Zeigler et al. 2000). Roughly speaking, models can be characterized with respect to the following features: discrete versus continuous time/space, global versus local decisions/behavior, and deterministic versus stochastic decisions/data. Some examples are as follows:

- Physical and chemical systems are often continuous in time and space, have local forces (decisions), and are only rarely stochastic; this is why they are often described by differential equations.
- Production, supply-chain and logistics systems are discrete in time and space, decisions are often global, and they can be stochastic; thus, they are well-described by discrete-event simulations.
- Economic and sociological systems are also discrete in time and space and can be highly stochastic, however, often decisions and behavior is determined mainly locally, which is why agent-based simulation models are well-suited in this case.

With the decision for the simulation approach at hand, one can go ahead and determine the details of the model and what data is needed or needs to be generated to feed it. Validation of the simulation model is just as important as in any other simulation study. Since there is insufficient data available for direct validation, like when simulating systems that do not exist yet, one has to validate by means of consistency conditions provided by shop floor domain experts.

Finally, running the model (maybe many times) will generate the synthetic data for the subsequent machine learning step. The machine learning model is then trained on the generated data to faithfully reproduce the inputs, but with different properties than the simulation model: *our hypothesis (using the method detailed in* Sect. 3.3*) is that a successfully trained network will abstract essential features of the data used for training, and measuring the minimally required network complexity for successfully learning a given dataset would be a good measure for the complexity of the data itself.*

Once the network model is established on synthetic data, the case study that lacked data in the first place can now continue: the model trained on synthetic data can be embedded in its application and one can start testing, using, and refining it, until a freshly trained model can replace it once the real data is available.

Clearly, the fact that domain knowledge and high-level descriptions/data are needed for this approach can be seen as a drawback. On the upside, in many cases

- domain knowledge will anyway be needed for a successful data analytics project;
- higher-level descriptions/data are either already available or are not so hard to come by or estimate stochastically;
- the simulation modeling process leads to a deeper understanding of the domain and its dynamics;

which is why we argue that building a simulation model can successfully mitigate the issue of "unavailability of data" in the first place. The results obtained in that way will then have to be validated by different means, for example, through investigation by the original data owners.

3.2 A Concrete Example: The Job Shop Model

The goal of our case study was to measure the complexity of the manufacturing processes based on production data rather than based on business analysis. The production data desired by us would have been provided as an event log that traces the processing steps that each order undergoes on its way through the production system. Manufacturing systems like this are best modeled by discrete-event simulations that model the orders being passed from one process to the next, producing a discrete series of events in time—the exact data that we need in our case study.

In order to validate our complexity measure based on a neural autoencoder, we chose to implement a relatively simple model of a production facility, called the *job shop model* (Pinedo 2009); see the example depicted in Fig. 17.2. The job shop model describes all production steps as so-called *machines* that are visited by *jobs* that represent the orders for products to be produced. Machines can process only one job at a time. Each job has a list of *tasks*, which are combinations of machines and processing times of that job on the given machine. The tasks must be completed in the given order. Different products may visit machines in different orders and the number of tasks can vary as well. We do allow for recirculation, that is, a given job

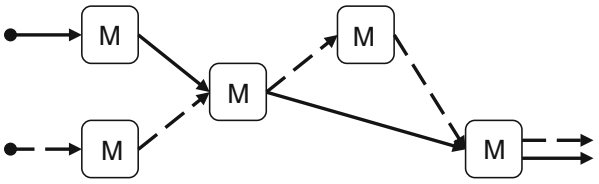

Fig. 17.2 A job shop model with five machines, M1–M5, and two jobs, the sequences of solid and dashed arrows, respectively

may visit a given machine several times on its route through the system, and we allow a changeover time to be accounted for before a new job can be processed on a given machine. Determining the optimal sequence for a job shop is a classical NP-hard optimization problem in operations research (Pinedo 2009).

Once the sequence of the jobs to be processed and their task lists with the processing times are fixed, the model is fully deterministic. The simulation yields a log of events, each with timestamp, job ID, machine ID, and event-type, for example,

- Job entered in waitlist, job selected from waitlist
- Capacity blocked, capacity released
- Changeover started, changeover ended
- Processing started, processing ended

Thus, for given job sequences, the simulation model provides raw production data from a synthetic shop floor that can be fed into the machine learning model.

3.3 A Novel Neural Net-Based Complexity Measure of Industrial Processes

In this section, we propose a novel measure to estimate complexity in production lines, based on a neural network, as well as an unsupervised approach to compute the measure. The goal is that for a given production line this complexity measure can be evaluated completely automatically without any human intervention and in (near) real time. The concept of complexity can be followed in compression theory (Henriques et al. 2013), learning theory (Zhu et al. 2009), and computational complexity theory (Park and Kremer 2015). The complexity of production lines is evaluated statically (Park and Kremer 2015) and dynamically (Fischi et al. 2015) in state-of-the-art research in order to improve manufacturing performance. Moreover, complexity can be evaluated for an entire dataset (Bousquet et al. 2004) or samples (Pimentel et al. 2014).

In our view, the complexity of a system can be quantified by how much a dataset containing an implicit full description of that system can be compressed without losing information about the system. For example, if the data describing all ongoing processes in a factory is very redundant, it can easily be compressed into a much shorter description, and the complexity of such a factory would be low. On the other

hand, for a factory where most ongoing processes are random, the description given by the data would be close to random and thus very hard to compress, and we would quantify this as a highly complex system.

In other words, our complexity measure for a system is the minimum description length or, equivalently, the maximum compression factor that can be achieved on datasets fully describing that system, without loss of information.[3] In principle, any compression algorithm could be used; however, the compression performance of those algorithms generally depends on the nature of the input data. For example, a compression algorithm that can achieve high compression factors for still images might perform quite badly on data consisting of moving images (i.e., video sequences). Since we are interested in the maximum compression rate, we need compression algorithms that are working well for the specific kind of input data we have. One way would be to hand-design good compression algorithms for our data; however, this would require obtaining a deep understanding of the underlying structures in our data by hand, which would be very labor-intensive.

For this reason, we chose to use neural networks for data compression. Unsupervised training of neural networks provides a fully automated way to extract such underlying structures from data, which are needed for good compression performance. The system is adaptive to a large degree, that is, for data with different characteristics it will automatically find the underlying structures that are better suited there. There is no need to hand-tune the compression algorithm as would be the case with classical, nonadaptive algorithms. Of course, training the network on a specific dataset requires time, but hand-tuning algorithms—in addition to time—would also require deep knowledge about the underlying data structures. Neural networks, on the other hand, once trained, can be used to *discover* such high-level structures and features in underlying data (while this promises to be a very interesting extension of our approach, this is beyond the scope of the current chapter and referred to future work).

In this chapter, we hence propose to measure the complexity of a production line using neural networks, specifically autoencoders (Goodfellow et al. 2016). The proposed measure evaluates the complexity of an entire production process. This approach is fully unsupervised and does not need any labeled data. However, a sufficient amount of data is required in order to train the autoencoder.

An autoencoder is trained to produce a replica of its input at the output layer. The structure of this type of neural network consists of a number of hidden layers connecting the input and output layer. In general, autoencoders contain a code layer as well as an encoding and a decoding function. The code is a representation of the input learned through the unsupervised training procedure. The dimensionality of the code is smaller than both input and output in undercomplete ("compressing")

[3] In principle, the compression does not need to be lossless in the strict meaning of the word. While on the noise-free simulation data used in experiments below, maximum compression while maintaining losslessness provides a natural threshold for the degree of compression in our measure, some degree of loss might even be desirable on real-world data to get rid of inherent noise from measurement errors, etc.

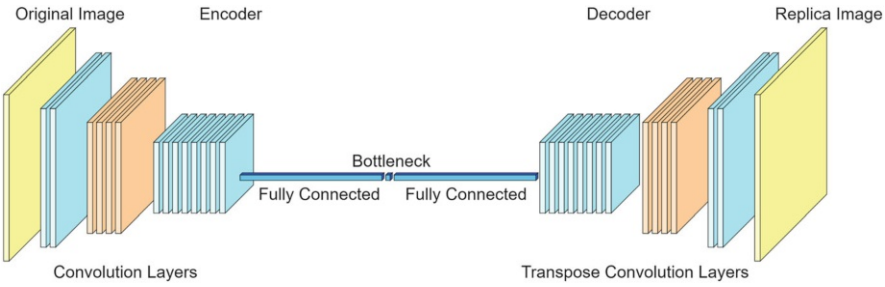

Fig. 17.3 The structure of a deep autoencoder with encoder, decoder, and the code (bottleneck) in between. As in the picture, we also encoded our data in image format (see Sect. 4.2)

autoencoders. In this case, the code forms an information bottleneck between the encoding and decoding networks, as depicted in Fig. 17.3.

The complexity measure we propose here is the minimal bottleneck dimensionality for the autoencoder to yield lossless reconstruction. Its value does not directly represent any features of the production line process but rather reflects its overall complexity in an abstract way. The production line data from the simulation (see Sec. 3.2) is considered a source of information for lossless compression and reconstruction. Each job in the job shop is represented as a certain temporal sequence of processing steps on several machines and is encoded as a two-dimensional matrix, where one dimension is the discretized process time and the other dimension is the ID of the process machine (see Fig. 17.5). An entry in the matrix is set to one if the corresponding machine is active at that time step, and set to zero otherwise. These matrices can be interpreted as patterns or images, and the set of all patterns of all jobs occurring in a given job shop provides a representation of the complete activity of this job shop. The autoencoder tries to compress the set of all these patterns as much as possible, without loss of information. The minimum code length (i.e., the size of the smallest bottleneck layer) that can achieve this is related to the information content in the activity patterns, and is chosen as our complexity measure for this job shop.

Based on Shannon's source coding theorem (Shannon 2001), it is possible to asymptotically obtain a code rate that is arbitrarily close to the Shannon entropy (Shannon 2001) in lossless compression of a source of information. The code rate refers to the average number of bits per symbol (products in production lines) in this definition. Importantly, lossless compression of a source is not possible with a code rate below the Shannon entropy (Shannon 2001). The dynamics of the production line is initially represented in the form of images containing temporal information as well as machine identification numbers. The autoencoder subsequently performs a lossless compression of these images. Therefore, the code rate in this context corresponds to the compression ratio of images (information of production lines) to code (bottleneck of the autoencoder).

The Shannon entropy of a source of information determines the lower band of the code rate. Therefore, the minimum code rate can be used as an approximation for the Shannon entropy. Assuming a source with fixed input length in the encoder, and specifically the autoencoder, the code rate only depends on the code length or the

bottleneck dimensionality. Therefore, the Shannon entropy of the source of information (production line) is proportional to the minimum dimensionality of the bottleneck in the autoencoder. Feldman and Crutchfield (1998) explain why the Shannon entropy is a measure of statistical complexity. Recently, Batty et al. (2014) used this measure to analyze spatial information and complexity. The proposed measure of complexity in this work, minimum dimensionality of the autoencoder bottleneck (code), is directly proportional to the entropy, which is a measure of complexity. It reflects the temporal usage patterns of the machines in the production line; the more different patterns that are needed to represent the system dynamics, the more complex it is.

4 Experiments and Discussion

Here we provide and discuss a proof of concept for our neural-network-based complexity measure for production systems. To show its validity, we generate a series of instances of the job shop model, produce the simulation event logs, and measure the complexities of each scenario in two ways: first, using a conventional complexity measure based on business analysis (see Sect. 4.3), and second, computed with our neural-network-based measure discussed in Sect. 3.3. Since the data for the original case study was not available, we used the job shop simulation model to produce the data required for the proof of concept.

4.1 Scenarios

We investigate the complexity of a series of instances of the job shop model. Starting from a simple base scenario, we vary several features in different directions, targeting different complexity drivers, namely, the number of processing steps, the number of products, the percentage of dedicated production lines, and the manufacturing stability. Introducing these variations leads to different production complexities for each scenario.

The base scenario has machines grouped in three stages that all jobs visit in the same order, reflecting the realistic situation where products typically go through stages like setup, assembly, and packaging. We generate 800 jobs that have tasks sampling all possible combinations of machines in those three stages, all with the same processing times. See Fig. 17.4 for an illustration of the base scenario with two machines per stage.

To produce different scenarios, we focus mainly on the second stage and change the number of machines available, the processing times of jobs on individual machines, the availability of machines, or the processing time depending on the choice of machine in the first stage. In addition, we enlarge the base scenario to encompass three and four machines per stage, respectively, and generate variations analogous to those described above.

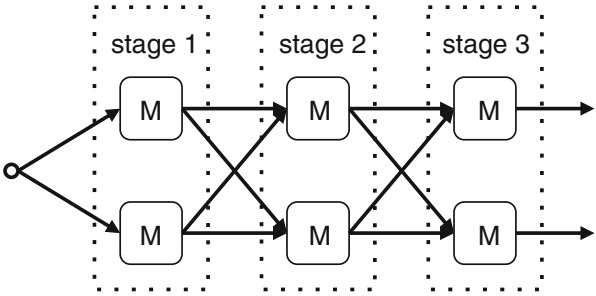

Fig. 17.4 The base scenario with two machines per stage. Each job visits the three stages in the same order and can be assigned to either machine per stage. Here, the arrows represent all possible paths of jobs through the system

4.2 Data Preprocessing and Autoencoder Network Topology

Before being fed into the autoencoder for complexity analysis, the data from the job shop has to be preprocessed. We do this in a way that allows automated integration in real factory settings in an Industry 4.0 environment later, namely, using timestamps of process steps. When jobs pass through the simulated job shop, each process step produces a timestamp when it is started and when it is stopped, together with the ID of the machine on which it is run (see Sect. 3.2).

For each job, we generate a two-dimensional matrix from this information, where one dimension is indexed by machine ID and the other by elapsed (discrete) time steps since the job started. If the machine with ID j is processing the given job at time step k, then in the corresponding matrix the entry at position (j, k) is set to +1, otherwise to -1. In other words, the processing of a job in the job shop can be represented as a two-dimensional pattern of black and white pixels, where white pixels indicate active machines at the corresponding time step, black pixels indicate inactive machines, and process steps are represented by horizontal white lines of different lengths (see Fig. 17.5). Each of these patterns constitutes a training pattern for the autoencoder, and each pixel position in the pattern is fed into a corresponding neuron in the input layer of the autoencoder. In our simulations, we use a maximum number of 16 machines and 61 time steps, so all our input patterns have a fixed size of 16×61 pixels. Note that not every machine or time step is used in every scenario; unused entries will simply be zero. We chose to keep the input dimensions fixed over all scenarios so that the number of weights in the neural networks would not depend on the scenario, allowing better comparability between scenarios. Thus, the input dimensions are just chosen large enough to accommodate the maximum number of machines and time steps in any of the scenarios.

While it would be possible to use a classical autoencoder (with fewer and fully connected hidden layers as the one depicted in Fig. 17.3) directly on these input data, for these fully connected networks the relatively large number of inputs ($16 \times 61 = 976$) leads to a rather large number of weights, resulting in slow learning and large training data requirements. Therefore, we decided to use a different network topology for our autoencoder: immediately after the input layer we use a stack of three convolutional layers, followed by two fully connected layers with a central hidden layer (the actual autoencoder), and finally a stack of three "transpose

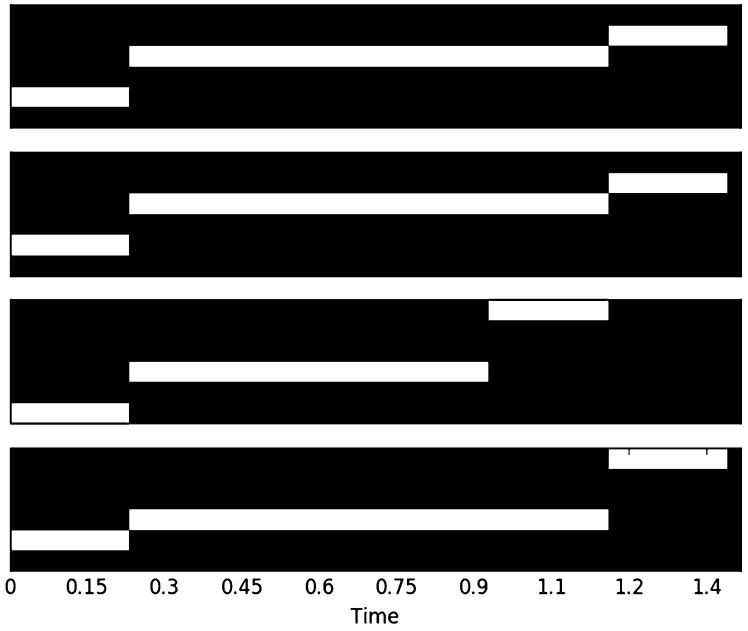

Fig. 17.5 Example of four input patterns for the autoencoder, generated from four different jobs from the simulated job shop. In each pattern, each row of pixels corresponds to the activity of a certain machine during the processing of that job (only 6 out of 16 potentially active machines per job are shown in this illustration). The horizontal axis represents (discrete) time passed since the start of that job. White pixels denote the corresponding machine being active at the corresponding time step, black pixels represent inactive machines

convolutional" layers to revert the action of the convolutional layers (Stadelmann et al. 2018).

The convolutional layers are very good at extracting information from two-dimensional pictures with geometric features such as our horizontal process step lines, while requiring relatively few weights due to weight sharing. Furthermore, since we use a stride of 2 in each layer, also the dimensions of the input patterns are reduced accordingly. Using 3×3 filters, we compared different filter numbers and found that for 2, 4, and 8 filters in the 1st, 2nd, and 3rd convolutional layer, respectively, the network could learn to map all input patterns for all scenarios to the correct output patterns, using less than 10 neurons[4] in the central hidden layer in all cases. It should be pointed out that in spite of the larger number of layers, our network actually has much fewer weights than a traditional autoencoder: for example, for a case with 8 neurons in the bottleneck layer, the simplest traditional three-layer autoencoder would require 16,600 weights, whereas our network topology only requires 2961 weights to be learned for that case.

[4]This specific number depends on the concrete data used and can be determined experimentally for any real data.

For comparability, we decided to fix the convolutional layers at the configuration described above and only vary the number of neurons in the central hidden layer. The minimum number of hidden neurons for which the network could still learn the correct input–output mappings for all patterns (jobs) in a given scenario was then chosen as the complexity number for that scenario. In other words, the network had to map all input patterns (representing all jobs in the production line) successfully back to the *same* input patterns while passing this information through a small bottleneck layer, and the minimum size of the bottleneck layer for which this was possible was chosen as the complexity number. It should be noted that this is only a *relative* complexity measure, since changing the network configuration of the convolutional layers will affect the minimal number of hidden neurons required. In other words, how the data is preprocessed affects how easily it can be learned (Lichtensteiger and Pfeifer 2002). Here, having less filters in the convolutional layers will require more neurons in the central hidden layer for still being able to learn successfully. However, since we are not yet able to quantify this influence appropriately, for this study we decided to fix the convolutional network topology at a configuration that was shown to work well and focus only on the number of neurons in the central hidden layer for our complexity measure.

In order to verify the self-consistency of our complexity measure, we did a second full run of experiments where we used different weight initializations for the networks and added strong multiplicative random noise in the neural activities of the bottleneck layer. In addition, we varied the size of input patterns by adding different amounts of zero padding. Our first results show that in spite of these rather substantial changes to the network, the resulting complexity measures do not change significantly, indicating the robustness of our approach. With regard to computational runtime, on a desktop PC equipped with an Intel Xeon Processor E5-2620 running at 2.40 GHz and an NVIDIA Quadro M4000 GPU, learning the correct input–output mappings for all patterns (jobs) in a given scenario required around 1–5 min. When the number of neurons in the bottleneck layer was changed, the system had to learn again. Since calculating complexity required finding the minimum number of neurons in the bottleneck layer for which learning was successful, using, for example, binary search around 5–10 variations of neuron numbers were needed. Therefore, calculating our complexity measure for a given scenario took around 10–50 min on our hardware configuration.

4.3 Results

To validate our neural-network-based complexity measure we compare it to a state-of-the-art conventional method (Friedli et al. 2013). It is computed as a weighted sum over contributions from the following factors (complexity drivers): number of process steps, percentage of dedicated production lines, number of changeovers, flexibility upside, and number of batches. These factors are measured for all simulated job shop scenarios and normalized to the interval between 0 and 1. Note that we

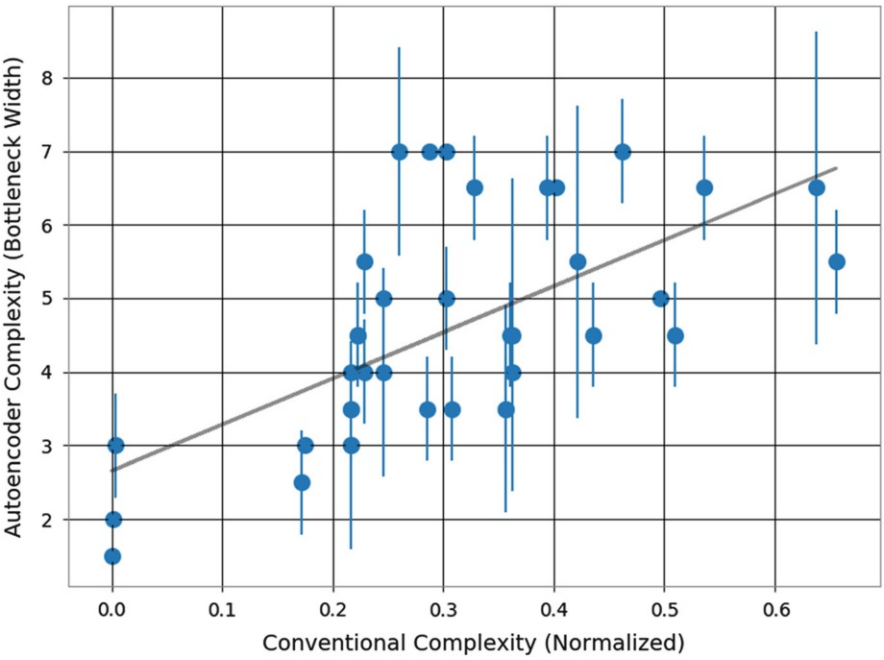

Fig. 17.6 The complexity values for all scenarios from the autoencoder (minimal number of bottleneck neurons) is plotted against the conventional complexity normalized over all scenarios. The Pearson correlation coefficient is $p = 0.637$ ("fair correlation"). The error bars show the standard deviation of two different series of experiments (see end of Sect. 4.2)

did not consider manufacturing stability as a driver of complexity here, since the scheduling of the job shops is static and therefore the simulations are deterministic. For the weights of the individual complexity drivers we take the results from Friedli et al. (2013), and renormalized them to 1 after neglecting the weight for manufacturing stability.

Figure 17.6 shows the averages of our autoencoder-based complexity values from the two experimental runs plotted against the complexity values obtained using the conventional method. The error bars show standard deviations as conservative indicators of the variability of our approach, see discussion in the end of Sect. 4.2. The Pearson correlation coefficient is $p = 0.637$, which indicates a fair correlation. This shows that our autoencoder-complexity measures at least partly the same features as the conventional method does, rendering it a valuable tool in the analysis of production and process analysis while being determined completely in a data-driven manner. This result is to be understood as a first proof-of-concept. To improve the understanding of the relation between the two complexity measures and the dependency of the autoencoder complexity on the features of the production processes and product architectures, a complete study based on larger job shops and, preferably so, real data is needed and aimed for.

5 Conclusions

We claim that data analytics projects need data to be analyzed. Often taken for granted and not seriously planned as a potential showstopper, the unavailability of data of the right quality, at the right granularity, and in a reasonable project time frame may put entire projects at risk. The message is clear: gathering the right data is not to be underestimated and can make up by far the majority of the project time.

Lesson Learned #1 For future projects, special attention needs to be paid to the measurement and gathering of the specifically required data out of the production systems.

The research conducted in this chapter showed a feasible way of how to deal with unavailable data when one is hit by it. Specifically, available high-level data can be turned into a simulation model (using extra help from domain experts) that produces finer-grained synthetic data in arbitrary quantity (but in quality bound to the explicitly modeled aspects of the simulation). This finer-grained data (independent of originating from direct measurements or simulations) can in turn be used to train a machine-learning model with intriguing properties: it inherits the properties of the simulation model while being able to generalize beyond its discontinuities. This study used state-of-the-art unsupervised learning schemas (deep convolutional compressing autoencoders) for this task.

Lesson Learned #2 Coupling simulation and machine learning to "convert" models of the real world and thus get access to the intriguing properties of each method is a powerful tool. In the presented scenario we show how simulation can be used to provide missing input data, at least until the real data can be provided. In an age where data is considered extremely valuable, yet sometimes still scarce if too specialized, this is an important methodology in many domains from sociology to traffic, energy, and health.

We specifically introduced a novel complexity measure for industrial product architectures and process topology based on the minimum dimensionality of the bottleneck layer of our trained autoencoder. We computed this complexity measure for a range of production line scenarios, inspired by real situations in our case study. Comparing those values to the state-of-the-art complexity measures based on conventional complexity drivers suggested by business experts, we find that the two measures are fairly correlated (see Fig. 17.6), which we interpret as a proof of concept for the autoencoder approach. As opposed to the conventional measure that is based on expert knowledge and extensive human effort (qualitative interviews and subsequent work of economists), our measure has the advantage of being learned completely in an unsupervised fashion from timestamped process data alone. Note that we are not suggesting to always use this complexity measure in conjunction with a respective simulation model of the production system in question. On the contrary, the aim for further work is to establish our complexity measure by testing it in real-world situations with real shop floor data, using it as a tool to identify unwanted complexity and suggest changes in process structures and product architecture that reduce this complexity and the associated costs.

Lesson Learned #3 The paradigm of data-driven decision support can even enter the domain of a highly qualified business consultant (that would usually estimate the classical complexity measure manually), delivering the quantitative results necessary to ponder informed management decisions.

Neither the complexity measure itself, nor the neural autoencoder architecture, or the necessary data, are highly sophisticated. They are based on available information and common-sense ideas, implemented and thoroughly verified but not much changed from the original idea. While a first prototype like this case study shows the traits of a research project, nothing hinders its direct application by engineers in business in the next scenario that is somewhat similar.

Lesson Learned #4 It is merely the knowledge of what methods and technologies are possible and available that currently hinders the faster adoption of the data-driven paradigm in businesses.

Neither the involved simulation methods, nor the used machine learning techniques, nor the idea of bootstrapping machine learning with simulation per se are novel. Nevertheless, the data-driven complexity measure is new and arises simply as a straightforward combination of available technologies and methodologies. Innovation in this project arose from the collaboration of experts, not from individual novel developments [see also Swiss Alliance for Data-Intensive Services (2017)].

Acknowledgments The authors are grateful for the support by CTI grant 18993.1 PFES-ES, and for the participation of our colleagues from the ZHAW Datalab in the conducted survey.

References

Abbeel, P. (2017). *Pieter Abbeel: Deep learning-to-learn robotic control* [video-file]. Retrieved from https://youtu.be/TERCdog1ddE

Batty, M., Morphet, R., Masucci, P., & Stanilov, K. (2014). Entropy, complexity, and spatial information. *Journal of Geographical Systems, 16*(4), 363–385.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1798–1828.

Bousquet, O., Boucheron, S., & Lugosi, G. (2004). Introduction to statistical learning theory. In *Advanced lectures on machine learning* (pp. 169–207). Heidelberg: Springer.

Budde, L., Faix, A., & Friedli, T. (2015). From functional to cross-functional management of product portfolio complexity. In *Presented at the POMS 26th Annual Conference*, Washington, DC.

Closs, D. J., Jacobs, M. A., Swink, M., & Webb, G. S. (2008). Toward a theory of competencies for the management of product complexity: Six case studies. *Journal of Operations Management, 26*(5), 590–610. https://doi.org/10.1016/j.jom.2007.10.003.

Feldman, D. P., & Crutchfield, J. P. (1998). Measures of statistical complexity: Why? *Physics Letters A, 238*(4–5), 244–252.

Fischi, J., Nilchiani, R., & Wade, J. (2015). Dynamic complexity measures for use in complexity-based system design. *IEEE Systems Journal, 11*(4), 2018–2027. https://doi.org/10.1109/JSYST.2015.2468601.

Fisher, M. L., & Ittner, C. D. (1999). The impact of product variety on automobile assembly operations: Empirical evidence and simulation analysis. *Management Science, 45*(6), 771–786.

Fisher, M., Ramdas, K., & Ulrich, K. (1999). Component sharing in the management of product variety: A study of automotive braking systems. *Management Science, 45*(3), 297–315.

Fogliatto, F. S., Da Silveira, G. J., & Borenstein, D. (2012). The mass customization decade: An updated review of the literature. *International Journal of Production Economics, 138*(1), 14–25.

Friedli, T., Basu, P., Bellm, D., & Werani, J. (Eds.). (2013). *Leading pharmaceutical operational excellence: Outstanding practices and cases*. Heidelberg: Springer. https://doi.org/10.1007/978-3-642-35161-7.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, MA: MIT Press. Retrieved December 22, 2017, from http://www.deeplearningbook.org

Henriques, T., Gonçalves, H., Antunes, L., Matias, M., Bernardes, J., & Costa-Santos, C. (2013). Entropy and compression: Two measures of complexity. *Journal of Evaluation in Clinical Practice, 19*(6), 1101–1106.

Kawaguchi, K., Kaelbling, L. P., & Bengio, Y. (2017). Generalization in deep learning. *CoRR, 1710*, 05468. Retrieved December 22, 2017, from https://arxiv.org/abs/1710.05468

Kekre, S., & Srinivasan, K. (1990). Broader product line: A necessity to achieve success? *Management Science, 36*(10), 1216–1232.

Krishnan, V., & Gupta, S. (2001). Appropriateness and impact of platform-based product development. *Management Science, 47*(1), 52–68.

Lancaster, K. (1990). The economics of product variety: A survey. *Marketing Science, 9*(3), 189–206.

Lichtensteiger, L. & Pfeifer, R. (2002). An optimal sensor morphology improves adaptability of neural network controllers. In J. R. Dorronsoro (Ed.), *Proceedings of the International Conference on Artificial Neural Networks (ICANN 2002)*, Lecture Notes in Computer Science LNCS 2415 (pp. 850–855).

Meierhofer, J. & Meier, K., (2017). From data science to value creation. In St. Za, M. Drăgoicea, & M. Cavallari (Eds.) *Exploring Services Science, 8th International Conference*, IESS 2017, Rome, Italy, May 24–26, 2017, *Proceedings* (pp. 173–181). Cham: Springer.

Orfi, N., Terpenny, J., & Sahin-Sariisik, A. (2012). Harnessing product complexity: Step 2—measuring and evaluating complexity levels. *The Engineering Economist, 57*(3), 178–191. https://doi.org/10.1080/0013791X.2012.702197.

Park, K., & Kremer, G. E. O. (2015). Assessment of static complexity in design and manufacturing of a product family and its impact on manufacturing performance. *International Journal of Production Economics, 169*, 215–232.

Pimentel, D., Nowak, R., & Balzano, L. (2014, June). On the sample complexity of subspace clustering with missing data. In *2014 IEEE Workshop on Statistical Signal Processing* (SSP) (pp. 280–283). IEEE.

Pinedo, M. L. (2009). *Planning and scheduling in manufacturing and services* (2nd ed.). Dordrecht: Springer.

Ramdas, K., & Sawhney, M. S. (2001, January 1). *A cross-functional approach to evaluating multiple line extensions for assembled products* [research-article]. Retrieved January 15, 2014, from http://pubsonline.informs.org/doi/abs/10.1287/mnsc.47.1.22.10667

Schmidhuber, J. (2008). Driven by compression progress: A simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes. In *Workshop on anticipatory behavior in adaptive learning systems* (pp. 48–76). Heidelberg: Springer.

Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review, 5*(1), 3–55.

Stadelmann, T., Tolkachev, V., Sick, B., Stampfli, J., & Dürr, O. (2018). Beyond ImageNet - deep learning in industrial practice. In M. Braschler, T. Stadelmann, & K. Stockinger (Eds.), *Applied data science: Lessons learned for the data-driven business*. Heidelberg: Springer.

Swiss Alliance for Data-Intensive Services. (2017). Digitization & innovation through cooperation. *Glimpses from the digitization & innovation workshop at "Konferenz Digitale Schweiz"*. Retrieved April 26, 2018, from https://data-service-alliance.ch/blog/blog/digitization-innovation-through-cooperation-glimpses-from-the-digitization-innovation-workshop

Tang, C. S. (2006). Perspectives in supply chain risk management. *International Journal of Production Economics, 103*(2), 451–488. https://doi.org/10.1016/j.ijpe.2005.12.006.

Zeigler, B. P., Kim, T. G., & Praehofer, H. (2000). *Theory of modeling and simulation* (2nd ed.). Orlando, FL: Academic Press.

Zhu, X., Gibson, B. R., & Rogers, T. T. (2009). Human rademacher complexity. In *Advances in neural information processing systems* (pp. 2322–2330). Cambridge, MA: MIT Press.

Chapter 18
Data Warehousing and Exploratory Analysis for Market Monitoring

Melanie Geiger and Kurt Stockinger

Abstract With the growing trend of digitalization, many companies plan to use machine learning to improve their business processes or to provide new data-driven services. These companies often collect data from different locations with sometimes conflicting context. However, before machine learning can be applied, heterogeneous datasets often need to be integrated, harmonized, and cleaned. In other words, a data warehouse is often the foundation for subsequent analytics tasks.

In this chapter, we first provide an overview on best practices of building a data warehouse. In particular, we describe the advantages and disadvantage of the major types of data warehouse architectures based on Inmon and Kimball. Afterward, we describe a use case on building an e-commerce application where the users of this platform are provided with information about healthy products as well as products with sustainable production. Unlike traditional e-commerce applications, where users need to log into the system and thus leave personalized traces when they search for specific products or even buy them afterward, our application allows full anonymity of the users in case they do not want to log into the system. However, analyzing anonymous user interactions is a much harder problem than analyzing named users. The idea is to apply modern data warehousing, big data technologies, as well as machine learning algorithms to discover patterns in the user behavior and to make recommendations for designing new products.

1 Data Warehouse Architecture

Modern enterprises typically have dozens of databases that store different types of data such as information about customers, products, processes, marketing data, financial data, etc. In a global organization, different regions might store products in different databases with different schemas and sometimes conflicting information. Moreover, parts of the database might be stored in a relational database, in a

M. Geiger · K. Stockinger (✉)
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stog@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_18

NoSQL-database or even in Excel-sheets. Analyzing these diverse, heterogeneous datasets individually might lead to different results among the different databases due to data redundancies or data quality issues. The goal of data warehousing is to integrate these diverse datasets into one coherent database that can be considered as the single source of truth for subsequent analytics pipelines.

Traditionally, there are two main approaches of building a data warehouse:

- Inmon-Approach: three-layer architecture with Staging Area, Integration Layer, Data Marts (Inmon 1992)
- Kimball-Approach: two-layer architecture with Staging Area and Data Marts (Kimball 2002)

We will first analyze these two approaches and afterward discuss alternative solutions.

1.1 Inmon-Approach

According to Bill Inmon, a data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process (Inmon 1992). Let us analyze this definition in more detail:

- **Subject-oriented:** Datasets are not just randomly spread all over the data warehouse but they are separated by specific topics according to their business functions. For instance, all database tables of a customer are stored in a dedicated area called customer. All information related to products is stored in a product area.
- **Integrated:** This is the most important aspect of the Inmon-Approach. Assume that a global company has a customer database in Zurich and one in New York. Both databases have different schemas to store their customers. Integrating these datasets means to design a data model that covers all aspects of the customer in one generic schema. We will provide an illustrative example below.
- **Time-variant:** Data warehouses typically do not delete any data but store data changes over various points in time to enable historical data analysis.
- **Nonvolatile:** Data is stored permanently and cannot be changed any more.

The architecture of the Inmon-Approach consists of three layers as shown in Fig. 18.1. We will first describe the layers very briefly and afterward show concrete examples about the functionality and challenges of each layer.

- Source Systems (shown on the left side of Fig. 18.1): These systems provide the data that should be stored in the data warehouse. They are not considered a dedicated data warehouse layer.
- **Staging Area:** Data is imported from various sources into one database system.
- **Integration Layer:** Data is integrated, harmonized, cleaned, and stored into an enterprise-wide data model.

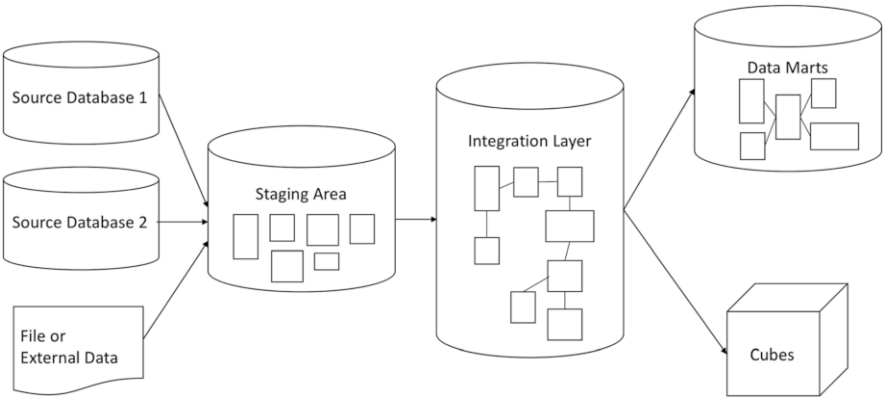

Fig. 18.1 Data Warehouse Architecture according to Inmon. The left side shows the source systems. The remaining parts show the three layers of the data warehouse: Staging Area, Integration Layer, and Data Marts. Note that Cubes are special types of Data Marts

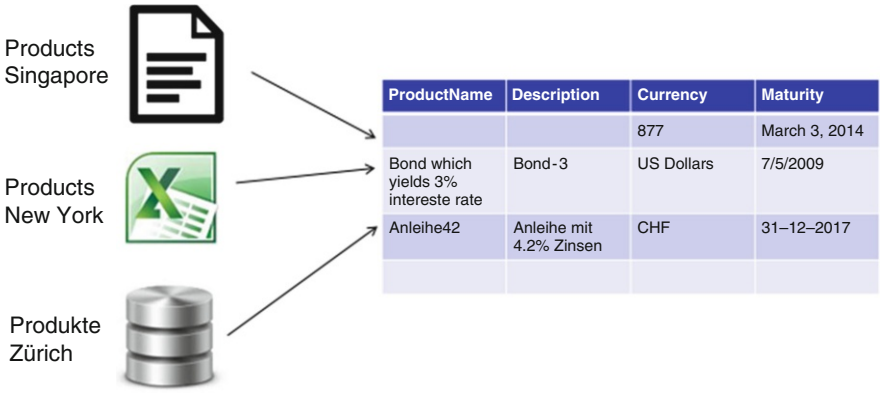

Fig. 18.2 Illustration of a Staging Area. Datasets from different sources need to be stored in a common database system. Note the data quality issues in the resulting table

- **Data Marts:** Data is physically reorganized to enable fast query response times for analytics. Cubes are special types of Data Marts that physically store database tables as multidimensional arrays as opposed to storing them in a relational database.

We now discuss the three data warehouse layers by means of concrete examples.

Staging Area Figure 18.2 illustrates the functionality of a Staging Area. On the left side we see datasets about products from three different locations of the same company, namely, Singapore, New York, and Zurich. Also note that the formats are different—text file, Excel-sheet, and database, respectively. These three datasets need to be stored into tables of the Staging Area. However, note the different data quality issues of the three systems. For instance, some data values are missing, for

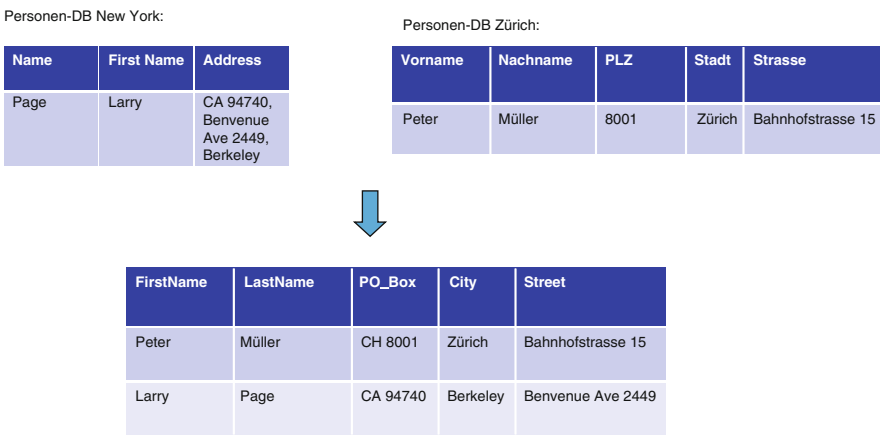

Fig. 18.3 Excerpt from an Integration Layer. Two tables of the Staging Area with different schemas (top part of the figure) need to be modeled and integrated into a common table (bottom part) that captures the information of both customer tables

example, ProductName, the currencies are expressed in different ways, for example, 877 versus US Dollars, and dates are formatted differently, for example, 7/5/2009 versus 31-12-2017. One task of the Staging Area is to convert the date information into a common format while data modeling tasks like harmonizing the different expressions of the currency information are handled in the subsequent Integration Layer.

Integration Layer Figure 18.3 shows an illustration of designing an integrated data model. Assume that the Staging Area contains two tables about customers that are originally stored and maintained in separate systems in New York and Zurich. Note that on the one hand, the schema is different. On the other hand, the language of the attributes is also different. A major task is now to design a common data model that is able to capture the complete information from both customer tables of the Staging Area. Note that this exercise has to be repeated for every table in the Staging Area. In large corporations it is not uncommon to have several thousands of tables in the Staging Area (Ehrenmann et al. 2012). The goal is to build a common enterprise data model where attributes are aligned, data quality issues are resolved, and different versions of data are stored over time.

According to Inmon, the data model adheres as much as possible to third normal form (Bernstein 1976), which is common to online transaction processing systems, that is, database systems that support insert, update, and delete operations. Simply put, third normal form guarantees that there are no redundancies in the data and insert as well as update operations can be performed efficiently.

Data Mart Figure 18.4 illustrates the functionality of a Data Mart. The main purpose of this layer is to enable fast data analysis. Hence, the data is logically and physically reorganized such that query response times are minimized. This kind

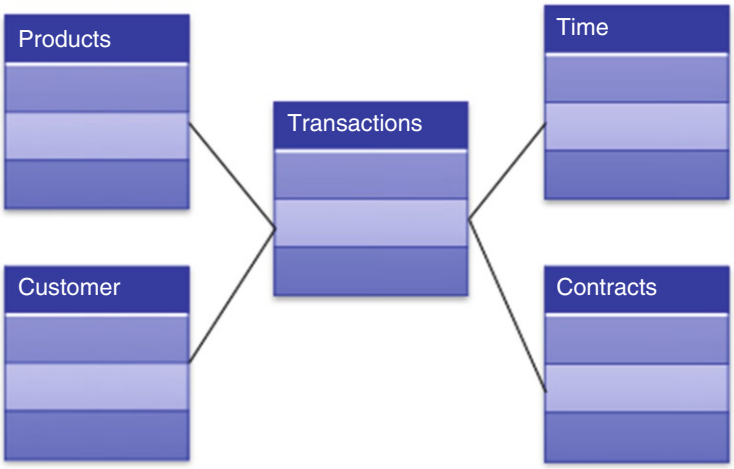

Fig. 18.4 Illustration of a Data Mart modeled as a denormalized Star Schema consisting of a fact table (Transactions) and four dimension tables (Products, Customer, Time, and Contracts)

of data modeling technique is called *Star Schema* (Kimball 2002) where the fact information is in the center of the star, namely, the transaction information. A typical example is to store each purchase item as a separate transaction. The outer part of the star are called *dimensions*, which contain information about the context of a transaction, such as what products were bought, by which customer, when, and under which contracts.

One of the main differences of a Star Schema model as opposed to third normal form is that data is *denormalized*, that is, data is stored redundantly. This has the advantage that fewer database tables are required and hence fewer typically expensive join-operations are executed during query processing. Since the purpose of Data Marts is to enable analysis and does not require updates, having data redundancy does not imply potential update problems Note that the Star Schema design is still relevant even though there have been significant changes in both hardware and software, such as main-memory databases (Larson and Levandoski 2016) or hardware accelerators (Casper and Olukotun 2014). The main reason is that Star Schema design is a good data modeling practice and should not be confused with a technique for performance optimization. A good overview on the main arguments can be found in the following blog post by Uli Bethke.[1]

[1] https://sonra.io/2017/05/15/dimensional-modeling-and-kimball-data-marts-in-the-age-of-big-data-and-hadoop/

1.1.1 Discussion of Inmon-Approach

From a development perspective, an Inmon data warehouse is built bottom-up, that is, first the Staging Area, then the Integration Layer, and afterward the Data Marts. In other words, the development process is according to the data flow.

The advantage of the Inmon-Approach is that the Integration Layer harmonizes and cleans the data, such that Data Marts can access data of high quality. However, the major disadvantage is that it takes quite some time until first analysis results are achieved, since Data Marts are built at the end of the development cycle. Moreover, designing an enterprise-wide data model is a challenging task since one needs to interact with various business units of the enterprise to correctly model the real world.

1.2 Kimball-Approach

The Kimball approach tries to tackle the main problems of the Inmon-Approach by starting with developing the Data Marts first. Similar to the Inmon-Approach, Data Marts are based on denormalized Star Schemas. Moreover, the Kimball data warehouse does not have an Integration Layer with an enterprise-wide data model and typically only consists of a Staging Area and Data Marts with Star Schema (see Fig. 18.5). The actual data integration needs to be done by each Data Mart separately or via so-called *Conformed Dimensions*, which can be considered as shared dimension tables that are used among Data Marts.

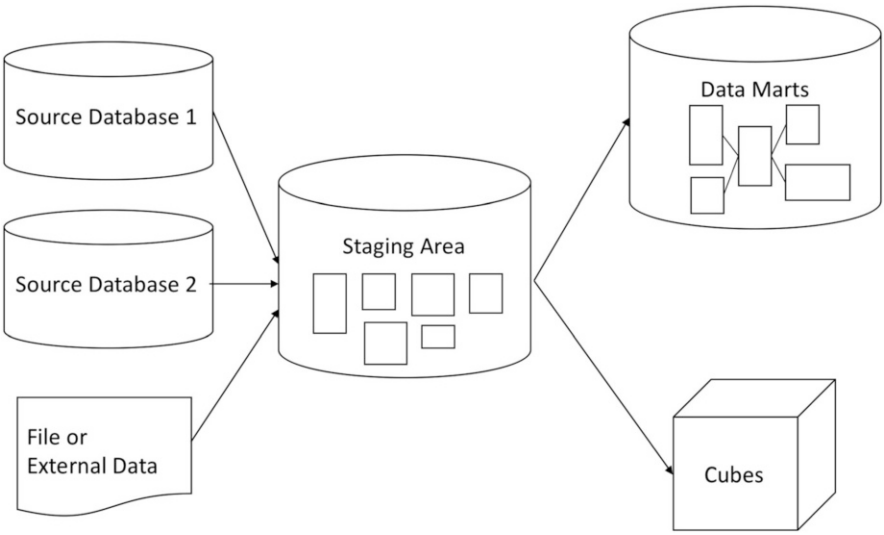

Fig. 18.5 Data Warehouse Architecture according to Kimball. The left side shows the source systems. The remaining parts show two layers of the data warehouse, namely, the Staging Area and the Data Marts. Cubes are special types of Data Marts and are thus considered to be part of the Data Mart Layer

1.2.1 Discussion of Kimball-Approach

The Kimball approach has clearly the advantage of delivering analytics results early on since Data Marts are built first. The architecture is also less complex since it often consists only of two layers. However, the major disadvantage is that each Data Mart needs to do data integration separately and often independently, which could result in Data Marts that produce noncoherent results.

1.3 Alternative Approaches

Alternative approaches to the Inmon and Kimball data warehouses are to not fully model all entities in the Integration Layer but only focus on the most important business entities, such as customers, products, and organizational information (Ehrenmann et al. 2012). The other entities could then be directly loaded from the Staging Area into the Data Mart.

Another approach is to model the Integration Layer as a *Data Vault* (Hultgren 2012). In short, a Data Vault does not fully integrate all data entities into one common data model but simply adds entities together via bridge tables. The advantage of this approach is that data can be loaded independently and does not require full integration and hence potential expensive schema update operations do not occur. The disadvantage, however, is that the data model typically results in more tables, which in turn requires more join operations when loading the Data Mart. For a more comprehensive discussion see Hultgren (2012).

2 Data Warehouse Use Case: Market Monitoring

In the following sections, we describe a use case of building an e-commerce platform based on a data warehouse that services as the basis for machine learning to recommend better products. The e-commerce platform includes information on tens of millions of products that are entered and curated by tens of millions of users via a crowd-sourcing approach. End-users are provided with detailed product information of goods with focus on groceries and cosmetics. The users can browse the product catalog, compare the product ingredients, or explicitly search for specific products. In case some product information is missing or some products are not in the system, end-users have the possibility to enter the missing information and thus contribute to the product database. See Fig. 18.6 for an example of a typical product showing the nutritional value of a certain type of yogurt.

The product database as well as the user community has been growing over the years. However, the company that is running the e-commerce portal had no information about the needs of their customers. Moreover, that company also did not

- **Nutritional value** (based on 100g of product)
  - Fat: 3g (average)
  - Sugar: 15.5g (high)
  - Salt: 0.2g (low)
- **Average daily use** (based on 100g of product)
  - Calories: 5%
  - Proteins: 8%
  - Fat: 4%
  - Carbohydrates: 7%
  - Sugar: 32%

Fig. 18.6 Example of typical product contained in the product database showing the nutritional value of a certain type of yogurt along with a comparison of the average daily use of the major product ingredients

know which products were popular or unpopular since no evaluation of the user behavior was performed, even though the user access logs were stored in an anonymized form over several months. Hence, the goal of this use case was to analyze the user behavior to recommend healthier products to the end-users or products with sustainable production. Another goal of this use case was to find out which products are requested by users and what kind of product ingredients are indicators for successful or unsuccessful products on that platform.

Besides entering new products, the platform allows the users to rate the ingredients for each product. Hence, a first step of this use case was to perform exploratory analytics to identify the main characteristics of the user behavior. In other words, one of the concrete tasks was to find out which product ingredients are popular and which ones are unpopular. Figures 18.7 and 18.8 give a brief overview on these kinds of analytics on cosmetics products. We have chosen cosmetics products since they are among the most popular ones on the respective e-commerce platform.

Figure 18.7 shows which ingredients users want be included in certain products (such as water, glycerin, or phenoxyethanol), while Fig. 18.8 shows which ingredients should not be included (such as silicon oil, disodium etha, and methylparaben). In order to run these kind of analytics on a daily basis with reasonable query performance (i.e., response times below 5 s), it was important to design and implement a state-of-the-art data warehouse as described in Sect. 1.

Figure 18.9 gives an overview of the data warehouse architecture that we implemented. This architecture follows a layered approach with *Staging Area*, *Integration Layer*, and *Data Mart*. Our design basically follows the Inmon-Approach with a reduced Integration Layer. We discuss the particularities of each of these layers below.

The Staging Area is basically a copy of the products stored in the product database along with the user access statistics. Moreover, the Staging Area also holds access statistics provided by Google Analytics that we will discuss in Sect. 3.

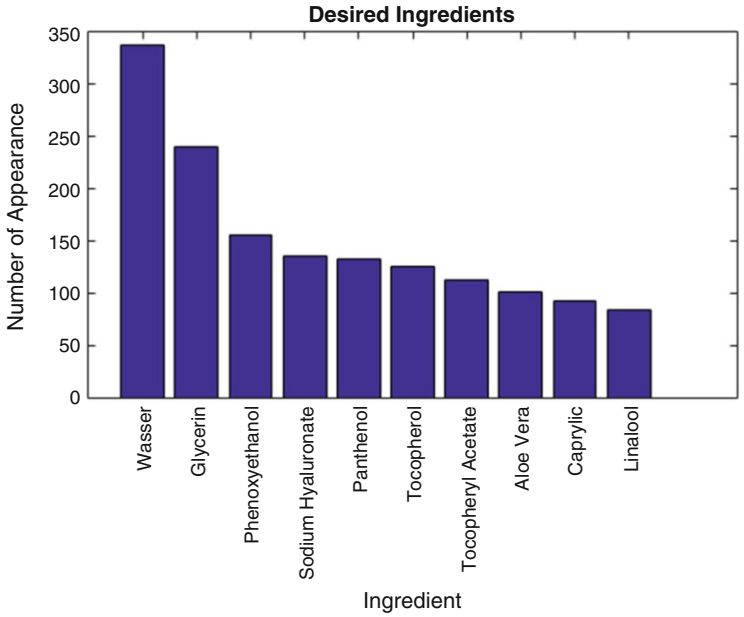

Fig. 18.7 Aggregated user behavior about cosmetics products with particular ingredients. The figure shows which ingredients **should** be included

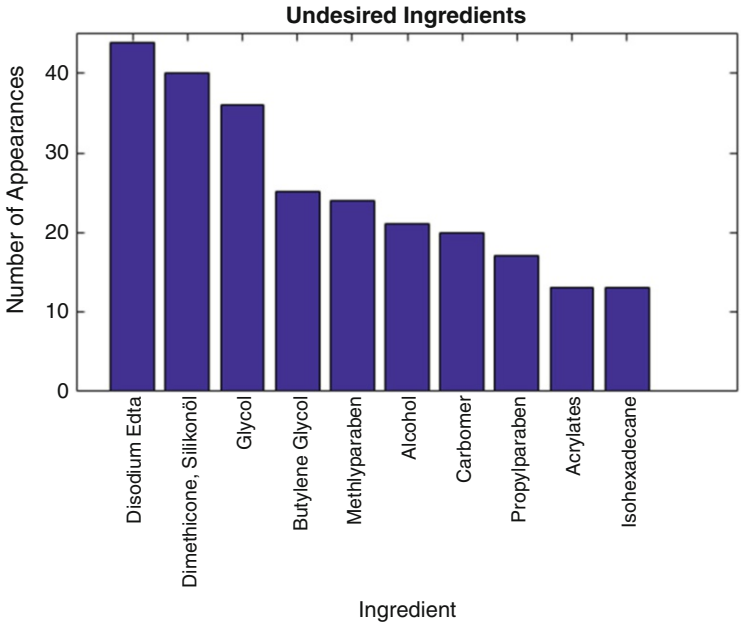

Fig. 18.8 Aggregated user behavior about cosmetics products with particular ingredients. The figure shows which ingredients **should not** be included

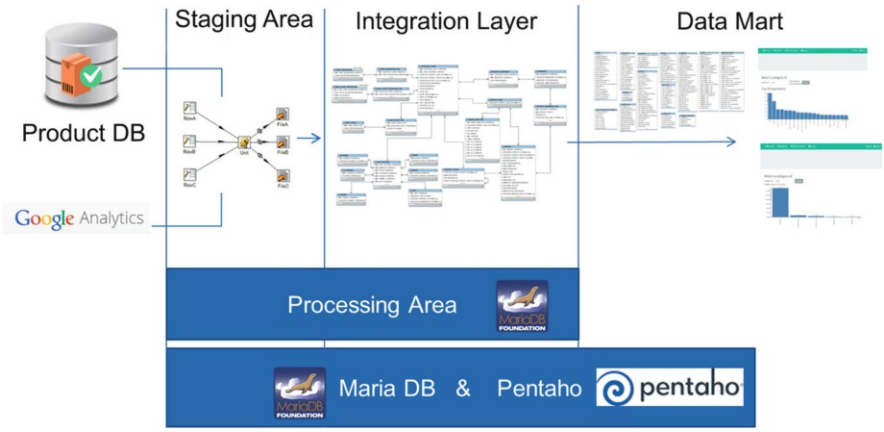

Fig. 18.9 Data Warehouse Architecture as implemented for our use case. The data warehouse consists of the classical three layers, namely, Staging Area, Integration Layer, and Data Marts

The Integration Layer is the core of the data warehouse since it integrates and harmonizes the various independent database tables from the Staging Area. Hence, a carefully designed data model is required that reduces data redundancies or ambiguities. Jointly with our industry partner we performed several iterations to fully understand and correctly model the data.

In order to analyze the history of products over time, we applied data historization as commonly used in data warehouses in industry (Ehrenmann et al. 2012). In other words, changes in the information of products are stored over time with specific timestamps, in order to be able to analyze changes over time.

Finally, the Data Mart stores the results of complex calculations and physically reorganizes the data in order to enable fast query processing.

For this use case, we only used open source technology since we worked with a small startup company that could not afford expensive, commercial solutions. The data is stored in the relational database management system MariaDB (MariaDB). MariaDB is the fully open source version of MySQL, which in turn contains some closed source modules. We have deliberately chosen the open source version since we wanted to avoid being tracked into some company-specific codes. A major alternative open source database system to MariaDB/MySQL would be Postgres (Postgres). We have chosen MariaDB since it provides a richer set of plug-ins for open source projects and is part of the LAMP web service stack (Linux, Apache, MySQL/MariaDB, php) (Lawton 2005).

The processes for data extraction, transformation, and loading (ETL) are implemented with the data warehousing tool Pentaho (Pentaho), which is one of the most widely used open source ETL tools with a wide range of functionalities. The advantage of using Pentaho for database transformations is the graphical user interface, which enables developers to build ETL processes visually. The resulting process flows enable nontechnical users to better understand data flows and

transformations without the need to know domain-specific programming languages such as SQL or PL/SQL (Feuerstein and Pribyl 2005). Alternatives to Pentaho are Talend (Talend) and JasperSoft (JasperSoft) that provide similar functionality. We have chosen Pentaho since we had already a positive development experience from previous projects. However, we see no obvious reasons why we could not have chosen another ETL tool.

One of the main challenges of building the data warehouse was to achieve acceptable load (less than 12 h to potentially load a data warehouse twice a day) and query performance (below 5 s for end-user queries). Some tables had on the order of 10^7 and 10^8 records. The tables needed to be joined in various ways in order to perform analytical calculations. Hence, in order to reduce the query response times below 5 s, we needed to perform a detailed analysis of the query access paths and to build dedicated database indexes. However, building a database index is always a trade-off between query performance and load performance: On the one hand, database queries are typically accelerated by database indexes. On the other hand, the load performance could deteriorate, since loading a database table also requires updating the respective indexes.

We could overcome this problem by monitoring the behavior of the database (loading and querying of data) over a certain period of time and carefully studying the impact of introducing database indexes. In particular, we analyzed the query plans produced by the query optimizer and studied the resulting access paths (Ioannidis 1996). A typical query that requires joining two tables could have several access paths, for example,

- Sequential scan over both tables followed by hash join
- Sequential scan over both tables followed by nested loop join
- Index scan followed by hash join

In addition to studying the queries, we also needed to study the database operations for loading and transforming tables into the various layers. Hence, a significant time spent on building the data warehouse is to physically tune the database.

Since database tuning is a very challenging task and requires expert knowledge both of the database and the data content, recent approaches use machine learning algorithms to optimize the database performance (Wang et al. 2016).

3 Enrichment with Google Analytics Data

As already mentioned previously, in order to use the e-commerce platform, no specific user account is required that explicitly characterizes each individual. In other words, the access logs that are stored by the platform contain only anonymized information. However, in order to perform some kind of user analysis, statistical access information is required. Hence, we used Google Analytics (GA) to better understand the user population (Clifton 2012).

GA provides statistical information on users that access certain web pages. GA requires a snippet of JavaScript code that needs to be added to the website that should be tracked. The visitor data is then sent to a Google server that analyzes the data. However, various ad filtering programs can block Google's tracking code, which potentially leads to holes in the access statistics. Moreover, user can delete or block GA cookies and hence no data can be collected. All these issues show that GA is a good additional source of information but one needs to accept certain data quality issues that cannot be thoroughly quantified.

Let us now discuss how we used GA in our use case. Since each product of our e-commerce application is described on a dedicated web page, we could directly leverage GA. For instance, GA indicates whether a certain user is male or female. Moreover, GA provides information about age groups of users. Figure 18.9 shows some statistical user information about accessing various products. For instance, the product with ID 324404 (first row in Table 18.1) has been accessed 7890 times; 1681 accesses were by male users and 1686 by female users. Note that the number of male and female users does not necessarily sum up to the total number of GA views. One reason is that GA only provides statistical information if at least ten users have accessed a certain page. We will revisit this fact later on. Figure 18.9 also shows which users are estimated to be in the age range of 15–24, 25–34, etc.

We used this information to estimate the gender of users for whole access paths of product groups. Assume, for instance, that a user accessed three different products as shown in Fig. 18.10. Further assume that GA estimates the probability that product 1 was accessed by a female user to be 68%. The probabilities provided by GA for products 2 and 3 are 62% and 80%, respectively.

We can now calculate the total probability for all three products by applying Bayesian inference as shown in Fig. 18.1. Hence, the total probability that a female user has accessed all three products is 93%.

Equation 18.1 Bayesian inference to calculate probability of genders for users accessing certain products.

$$p = \frac{p_1 p_2 p_3 \cdots p_N}{p_1\, p_2 \cdots p_N + (1-p_1)(1-p_2)\cdots(1-p_N)}$$

$$p = \frac{0.68 \cdot 0.62 \cdot 0.8}{0.68 \cdot 0.62 \cdot 0.8 + (1-0.68)(1-0.62)\cdots(1-0.8)}$$

In order to use GA effectively, the following information needs to be kept in mind:

- GA only provides statistical information for a certain web page if this page was at least accessed ten times, otherwise GA does not provide any information at all.
- There are a limited number of queries that one can submit to GA for retrieving statistical results. If the number of queries is above a certain threshold, then GA provides summary information. Hence, there is a trade-off between the number queries submitted and the result quality.

Table 18.1 Statistical user information provided by GA about accessing a certain product

Product_Id	GA_Views	Male	Female	Age_15_24	Age_25_34	Age_35_44	Age_45_54	Age_55_64	Age_65_Plus
324404	7890	1681	1686	1107	799	538	110	63	84
11186594	5790	381	2757	799	856	381	355	125	47
11294251	4993	402	2308	626	778	423	345	89	57
347074	4661	204	2297	517	705	360	345	125	21

Fig. 18.10 Gender probabilities of a user who accessed three different products. The green arrows indicate the access path from product A to B and C

In summary, GA is a good additional source to gain additional information about anonymous users that do not require a login to access an e-commerce web page.

4 Unsupervised Machine Learning Approach to Cluster Users and Products

In order to analyze the access patterns of the end-users and to better understand their needs, we applied various clustering algorithms, that is, unsupervised machine learning approaches. The most commonly known clustering algorithms usually are either centroid-based or density-based. Centroid-based clustering algorithms such as *K*-means (MacQueen 1967) and canopy clustering (McCallum et al. 2000) represent their data points as well as the cluster centroids as feature vectors. In density-based clustering algorithms such as DBSCAN (Ester et al. 1996), on the other hand, clusters are defined as areas of higher density than the rest of the dataset and therefore only the distance between the data points has to be known. For the user clustering and the product clustering based on ingredients, where we have feature vectors, we decided to use *K*-means as well as its adaptation canopy clustering due to their simplicity as well as due to their scalable implementation provided by Apache Mahout (Apache Mahout). For the product clustering using click paths, where we only defined a distance between the products, we decided to use the most commonly known density-based clustering algorithm, DBSCAN.

In the remainder of this section, we will explain these approaches in more detail, provide insights why we have chosen these algorithms and report on the experiences we gained by applying them to our use case.

4.1 User Clustering Using K-*Means*

In order to make particular recommendations about better or more sustainable products, we clustered the users based on their browsing behavior.

Before we could apply a clustering algorithm, we needed to perform feature engineering. In particular, we represented each user with a feature vector that consists of two parts (see Fig. 18.11). The first part is built by 12 complex,

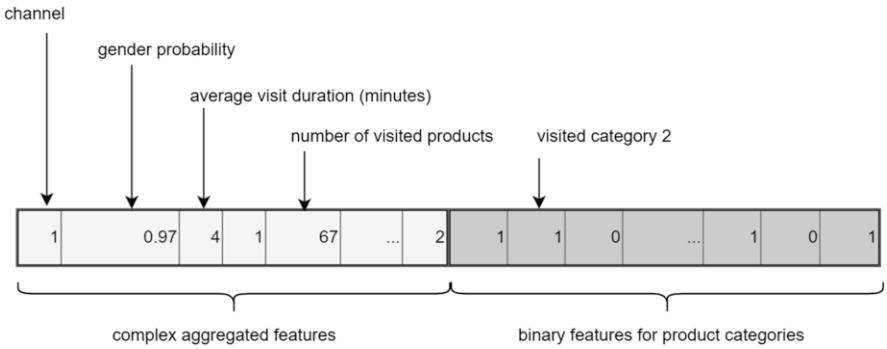

Fig. 18.11 Feature vector that represents a user

aggregated features, such as a flag to denote the channel,[2] the gender probability (provided by GA), the number of visited product categories, the number of generated events, the number of visited products, etc. The second part of the feature vector consists of binary features for each product category, which is 1 when the category was visited and 0 otherwise.

We used *K*-means applying the cosine distance measure. We will now explain the choice of this measure.

In *K*-means every cluster is represented by its centroid, which is the mean vector of the data points that belong to the cluster. Hereby, the cluster assignments can be computed as a nearest neighbor classifier that assigns each data point to its nearest centroid. The objective function of *K*-means minimizes the sum of squares of the data points to their closest centroid. The distance function between the data points and the centroids can be chosen based on the characteristics of the feature vectors of the data points. Since our feature vectors contain different kinds of feature types, it is not obvious which distance function fits best. Therefore, we ran and evaluated experiments with the Euclidean distance as well as with cosine distance and found that the latter led to better results. This is in accordance with theory, that predicts a worse performance for the Euclidean distance due to the curse of dimensionality, since our feature vector is quite high-dimensional. We also evaluated different numbers of clusters and found that using 20 clusters results in reasonable customer groups and is still small enough that it is possible to grasp them as a human.

The evaluation of the user clustering (and unsupervised learning tasks in general) is challenging since no ground truth exists and the quality assessment has to be done by a domain expert. However, already when inspecting a few clusters, the suggested groupings could help us in better understanding the users, as we will show below.

In the following, we describe 3 of the 20 customer groups extracted by the *K*-means algorithm. The descriptions are basically the centroids converted to natural language.

[2] A channel indicates whether the user has accessed the web page via an app or via a web page.

- Cluster 1: Predominantly women that spend on average 50 min in the mobile phone application. They compare products of the categories "Media and Books" as well as "Hobby and Leisure Time."
- Cluster 2: Solely women that search for cosmetic products such as personal hygiene, face care, hair styling, and hair care with an average visit duration of 40 min.
- Cluster 3: Women as well as men quickly (average of 10 min) checking the ingredients of candies, nuts, and other snacks.

These findings can be used to recommend certain products that are popular in a particular user cluster.

4.2 Product Clustering Based on Click Paths

The products in the application are already categorized manually into very broad categories. However, the categories are not narrow enough to be used in a recommendation algorithm that suggests healthier product similar enough to be a true alternative for the user. Therefore, we introduced a method that clustered the products of a specific category into several smaller subclusters.

We observed that users already quite often use the application to compare products and their ingredients. We therefore grouped products together that have been compared to each other in the past, that is, appear in the same click path. To achieve this, we defined a distance formula between two products (A and B) as follows:

$$d(A,B) = \frac{\#(A \cup B) \in \text{Clickpath}}{\frac{\#A \in \text{Clickpath} + \#B \in \text{Clickpath}}{2}}$$

It is basically the ratio between how many times the two products A and B appear in the same click path and the average number of click paths that contain either of the products. The average ensures that the distance measure is symmetric, that is, $d(A, B) = d(B, A)$.

The distances between the products can either be represented as a distance graph or to compute a distance matrix that contains the distance of each product to each other product in the category. We use DBSCAN to cluster the products in the matrix into groups.

Figure 18.12 shows a visualization of the clustering for the "Makeup" category. The category contains 4479 products that were subdivided into 70 clusters. The figure shows the subset of the largest clusters. Herein, we selected the maximum distance between two samples to be considered in the same neighborhood (eps), so that the number of generated clusters is maximized. The minimal number of points (minPoints) in a cluster was set to two. Those DBSCAN parameters help to find a lot

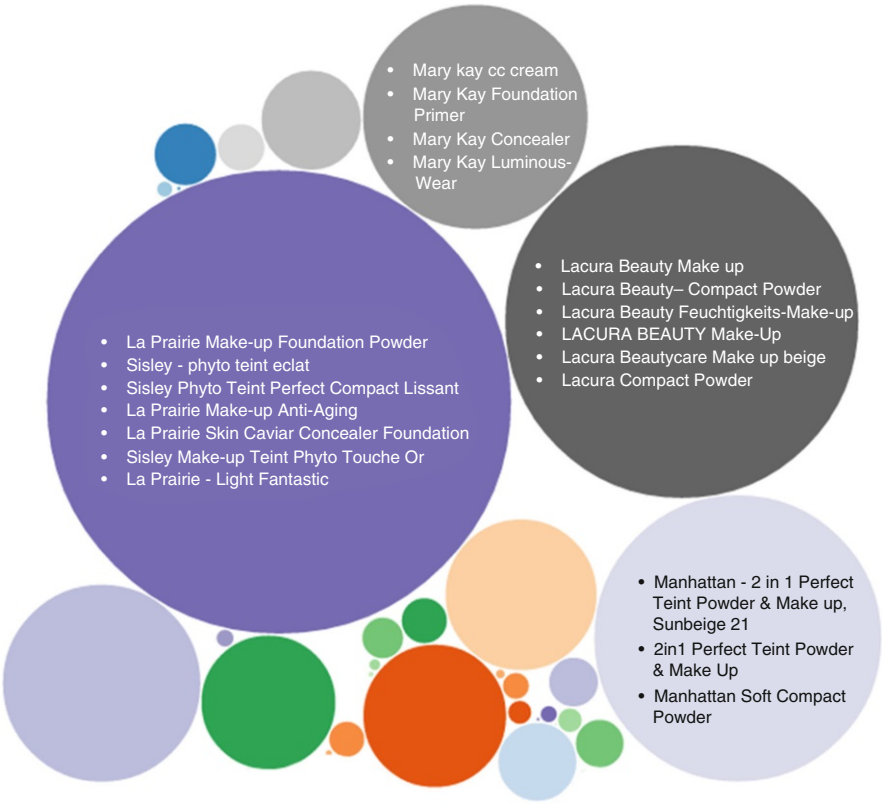

Fig. 18.12 Results of product clustering based on DBSCAN

of rather small clusters with very similar products, which was the main goal of the clustering.

The clustering resulted in 69 small clusters and one large cluster that contains all the products that are rarely visited and therefore the similarity to other products cannot be determined by the algorithm. The other clusters often contain a lot of products with the same brand, which mostly also defines a price segment. Apparently, when choosing a makeup, customers like a specific brand and then try to find the best product within that brand. Other clusters contain multiple brands that offer products in the same price segment and with a similar target audience.

5 Conclusions and Lessons Learned

The market monitoring use case was very challenging (see lessons learned below) and a good example of a data science project requiring many different skills ranging from data warehousing to data analysis and machine learning. The main success factors have been in designing and implementing an efficient data warehouse and in applying machine learning methods that can cope with various data quality issues. The main benefits of the project were that the industry partner got a very powerful end-to-end system that enables analyzing both the popularity of products as well as the customer behavior. Moreover, the system also helped in engaging new customers and laid the groundwork for designing healthier or more sustainable products based on a detailed analysis on customer preferences.

The main lessons learned of implementing this use case are as follows:

- **Data warehouse design and implementation:** Plan considerable amount of time and resources for the design and implementation of the data warehouse. It turned out that some 80% of the time over the whole use case was spent on building the data warehouse, including tuning the performance of SQL statements for loading and querying the data. In short, designing and building a small database application is pretty straightforward. However, efficiently managing a database with dozens of tables that contain more than 10^7 records is nontrivial and requires careful database tuning and query optimization.
- **Data quality:** Since the products of our e-commerce platform were entered manually by the end-users according to a crowd-sourcing approach, the data quality differs substantially between the products. For instance, some products contain detailed information on ingredients while other products contain very little or wrong information. Moreover, since both the number of unique users and products is in the order of tens of millions, a large percentage of products were only accessed a few times. Hence, we had to deal with very sparse data, which had significant impact on the analysis methods. In particular, we only provided analysis results for products having at least ten clicks.
- **Data enrichment to get information about anonymized users**: Google Analytics is a good additional source to gain additional information about anonymous users that do not require a login to access an e-commerce web page. However, one needs to accept certain data quality issues that cannot be thoroughly quantified. Hence, the statistical results of gender and age groups can only be considered as rough estimates with unknown error bars. However, to get a better insight into the platform users, this information is certainly helpful.
- **Unsupervised machine learning:** In order to evaluate the access patterns of our end-users, we applied various unsupervised machine learning algorithms. The main challenge with these approaches is that a human needs to interpret and evaluate the results. For instance, in our case the clustering algorithms produced some 25 clusters where each of the clusters contained between ten and a few thousand products. For a human it is very challenging to evaluate clusters with thousands of entries. We thus applied a sample-based cluster verification

approach and analyzed in detail a subset of the clusters jointly with our industry partner. We have chosen the samples based on products that were accessed most frequently.

Acknowledgment The work was funded by the Swiss Commission for Technology and Innovation (CTI) under grant 16053.2.

References

Apache Mahout. Retrieved August 24, 2018., from http://mahout.apache.org/

Bernstein, P. A. (1976). Synthesizing third normal form relations from functional dependencies. *ACM Transactions on Database Systems, 1*(4), 277–298.

Casper, J., & Olukotun, K. (2014). Hardware acceleration of database operations. In *Proceedings of the 2014 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays* (pp. 151–160). ACM.

Clifton, B. (2012). *Advanced web metrics with Google analytics*. Hoboken, NJ: Wiley.

Ehrenmann, M., Pieringer, R., & Stockinger, K. (2012). Is there a cure-all for business analytics case studies of exemplary businesses in banking, telecommunications, and retail. *Business Intelligence Journal*, *17*(3). TDWI.

Ester, M., & Kriegel, H.P., & Sander, J. & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *International Conference on Knowledge Discovery and Data Mining*, 1996.

Feuerstein, S., & Pribyl, B. (2005). *Oracle Pl/SQL programming*. O'Reilly Media, Newton, MA.

Hultgren, H. (2012). *Modeling the agile data warehouse with data vault*. Denver, CO: New Hamilton.

Inmon, B. (1992). *Building the data warehouse*. Hoboken, NJ: Wiley.

Ioannidis, Y. E. (1996). Query optimization. *ACM Computing Surveys (CSUR), 28*(1), 121–123.

JasperSoft. Retrieved July 21, 2017, from https://www.jaspersoft.com/

Kimball, R. (2002). *The data warehouse toolkit*. Hoboken, NJ: Wiley.

Larson, P. Å., & Levandoski, J. (2016). Modern main-memory database systems. *Proceedings of the VLDB Endowment, 9*(13), 1609–1610.

Lawton, G. (2005). LAMP lights enterprise development efforts. *Computer, 38*(9), 18–20.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In *Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press.

McCallum, A., Nigam, K., & Ungar, L. H. (2000). Efficient clustering of high dimensional data sets with application to reference matching. In *SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Pentaho. Retrieved July 21, 2017, from http://www.pentaho.com/

Postgres. Retrieved July 21, 2017, from https://www.postgresql.org/

Talend. Retrieved July 21, 2017, from https://www.talend.com/

Wang, W., Zhang, M., Chen, G., Jagadish, H. V., Ooi, B. C., & Tan, K. L. (2016). Database meets deep learning: Challenges and opportunities. *ACM SIGMOD Record, 45*(2), 17–22.

Chapter 19
Mining Person-Centric Datasets for Insight, Prediction, and Public Health Planning

Jonathan P. Leidig and Greg Wolffe

Abstract In order to increase the accuracy and realism of agent-based simulation systems, it is necessary to take the full complexity of human behavior into account. Mobile phone records are capable of capturing this complexity, in the form of latent patterns. These patterns can be discovered via information processing, data mining, and visual analytics. Mobile phone records can be mined to improve our understanding of human societies, and those insights can be encapsulated in population models. Models of geographic mobility, travel, and migration are key components of both population models and the underlying datasets of simulation systems. For example, using such models enables both the analysis of existing traffic patterns and the creation of accurate simulations of real-time traffic flow. The case study presented here demonstrates how latent patterns and insights can be (1) extracted from mobile phone datasets, (2) turned into components of population models, and (3) utilized to improve health-related simulation software. It does so within the context of computational epidemiology, applying the Data Science process to answer nine specific research questions pertaining to factors influencing disease spread in a population. The answers can be used to inform a country's strategy in case of an epidemic.

1 Introduction

Person-centric information generated by emerging data sources (e.g., IoT, sensors, mobile devices, and web traffic) represents a new and entirely different form of information than previously available. Content such as large, aggregated datasets of Call Data Records (CDRs) generated from mobile phones provide a reliable source of real-time, verified observations. CDR datasets provide extremely high-resolution, geo-temporal details that are not found in other human-centric sources. In comparison with the historical utilization of static information as collected via surveys,

J. P. Leidig (✉) · G. Wolffe
Grand Valley State University, Allendale, MI, USA
e-mail: jonathan.leidig@gvsu.edu

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_19

census, and focus groups (United States Department of Commerce 2017), these emerging dynamic data sources provide a more detailed and verifiable basis for making predictions, identifying events, and measuring the effects of policy decisions and actions.

In the health and life sciences, the goal is to optimize solutions and policies for problems such as successfully mitigating epidemics (Barrett et al. 2005). Large CDR datasets can be mined for patterns of human demographics, population density, and geospatial mobility (Vogel et al. 2015). In public health research, models built from CDR datasets can be used to improve the modeling and simulation of disease spread by using these more detailed and verifiable representations of human behavior (Jiang et al. 2015). Conducting well-informed simulations and analyzing the results directly impacts the prediction of epidemics and is critical for planning appropriate mitigation strategies.

In this chapter, we describe how data processing and advanced analytics were utilized to gain insights and generate models from latent data in large mobile phone datasets. The following sections on modeling and simulation, human-centric datasets, data processing, clustering, model building, and visualization demonstrate our approach to provide answers to the kind of questions that are required in order to build population models and simulation systems.

2 Modeling and Simulation in Health

Computational epidemiology involves the use of computing to study the health of a population. In this field, simulation systems provide a means of predicting real-world outcomes based on a given input scenario. An example input scenario might be a single, initially infected individual arriving at an airport in a large city; the simulation would attempt to predict the potential diffusion of the disease throughout a geographic region. Simulation systems are developed for modeling and simulating a specific disease type (e.g., contagious, vector-borne, or sexually transmitted). The simulation system requires an epidemiology disease model, generally a variant of a compartmental model with Susceptible, Infected, and Removed (SIR) disease states. Some simulation systems are based on ordinary differential equations (ODE), which are used to estimate the prevalence and rates of change between disease states for a given population. Many other simulation systems are based on stochastic, agent-based, and discrete event models (Barrett et al. 2008; Bisset et al. 2009; Chao et al. 2010). Agent-based simulation algorithms rely on graph theory, and the agents in these models (e.g., humans, livestock, wildlife) require individualistic information and decision-making abilities. Calibrating these agent-based models with the real-world scenarios they represent requires realistic networks and intelligent behavior by the agents.

The *Data for Development Challenge* health project that serves as a use case here utilized human-centric datasets to generate realistic agent-based models. With verified and quantitative mobile phone datasets, we employed modern mining techniques to answer specific questions and gain the insights that are needed to develop

accurate population models. CDR datasets provided details that informed population and activity models (e.g., where people live, travel, and migrate). These models were combined with additional information such as reports and statistics of monitored disease prevalence at regional and national levels to create more realistic simulations. Simulations based on these improved models were expected to provide more realistic results in terms of disease progression and spread.

This project consisted of an effort to model and simulate the spread of infectious diseases (primarily Ebola) in West Africa (Côte d'Ivoire and Senegal) in order to provide quantitative support for public health policies being set by the Ministry of Health of several countries and world organizations (Vogel et al. 2015). The effort required datasets for the populations that reside in West Africa along with a simulation engine that predicts the potential spread of a disease based on a given scenario. Preventing and mitigating an epidemic is heavily dependent on public health planning and policies.[1] Local and national governments' responses to an emerging outbreak are guided by research on the predicted impact of a given set of policies. The project improved simulation software that is utilized to set public health priorities and policies (via developing more realistic human agents). Governmental and private agencies have long been tasked with decision making in this area. The improved simulation software provides quantitative evidence from large studies (thousands of simulations for each specific scenario) based on more rigorous and realistic health, mathematical, and computational models.

3 Data Characteristics

Because of its promise in illuminating human mobility patterns, mobile phone communication (i.e., CDR) data has the potential to aid in understanding human behavior. It can inform researchers as to where people actually live, work, and travel, how they react to major events, and provide temporal data about daily commutes, seasonal migrations, and population shifts. However, there are inherent difficulties in obtaining and using this new form of data. Obvious issues emerge regarding privacy, given the capability of using the data to track movements of individuals and to potentially reveal sensitive information. Questions arise about the ethics of using such data, especially when applied to private information such as health and medical records. There is also the challenge of establishing that these new types of data, and

[1] Computational epidemiology aims at discovering novel insights, predicting events, experimenting with and optimizing scenarios, planning strategies, and setting policies. Mitigation strategies are used to limit and potentially eradicate a given disease from a population based on selective interventions. These strategies might include vaccine and antiviral distributions, isolating infected individuals, closing schools, closing sporting and large events, closing political borders, and limiting economic activity. Responding to an epidemic in practice requires advanced planning to set the mitigation policies, determine when and where to react, what to stockpile, how to distribute allocations from stockpiles, and many other factors.

hence the benefits that may accrue from analyzing it, become openly and globally accessible.

To address this latter issue, Orange Telecom instituted the *Data for Development (D4D) Challenge*, an open competition intended to contribute to the socioeconomic development and well-being of several West African countries (Côte d'Ivoire and Senegal, to date) by providing access to large, anonymized mobile phone datasets for purposes of scientific research.[2] The datasets are based on billions of anonymized CDRs of phone calls and SMS exchanges between millions of Orange Telecom's customers in West Africa, providing high-resolution temporal and geospatial detail. This dynamic, up-to-date surveillance supersedes much existing knowledge. Traditional population modeling, in any country, is often based on outdated, static, or nonexistent census and survey data; it uses on-the-ground contacts in specific, individual locations that may or may not be generalizable across a large region; and it is often not available from remote areas.

The datasets were collected from a subset of Orange Telecom customers over a period of 5 months (Côte d'Ivoire) and one full year (Senegal). Some preprocessing was conducted to completely anonymize the identity of users, eliminate redundant information, filter "uninteresting" users (i.e., those who never traveled out of range of their home cellular tower), and slightly obfuscate exact antennae locations (to preserve commercially proprietary information). The resulting datasets were also considered proprietary information, which required the use of Non-Disclosure Agreements and measures such as secure server hosting and restricted access.

CDRs typically have the following format:

Timestamp	Caller id	Callee id	Call duration	Antenna code

The records are usually organized chronologically. Because of size constraints, they are stored in a number of flat files that span the observation period. They represent various customer activity recordings.

The recorded spatial and temporal metadata were composed of discrete, recorded events and did not continuously sample user location. In other words, the datasets contain spatial observations of users only at those moments when calls and texts were placed or received. From this raw data, information can be mined as follows:

- Antenna-to-antenna traces on an hourly basis
- Individual user trajectories for a short duration (2-week time windows) with high-resolution spatial information (antenna locations)
- Individual user trajectories for a long duration (over the entire observation period) with low-resolution spatial information (sub-prefecture or arrondissement locations)
- Short duration (2-week time windows), limited (2-hop) customer communication graphs

[2] www.d4d.orange.com

The use of short duration observations or low spatial resolution is intended to further anonymize subscriber identity. Subsequent sections describe how CDRs can be mined at a variety of geospatial and temporal granularities to derive population models, activity models, and human mobility patterns.

4 Analyses and Mining

Human-centric content contains both obvious and subtle patterns. Finding obvious patterns reinforces trust in and understanding of a dataset—for example, most subscribers within a mobile phone dataset would be expected to place and receive far fewer phone calls in the middle of the night than during business hours. Less obvious patterns can be identified when analysts use algorithms capable of revealing connections without user guidance. These less obvious patterns are often unexpected, novel, or poorly understood prior to the analysis. For example, mobile phone datasets directly provide insight into the specific commercial areas of a large city that spectators frequently visit before and after a major sporting or entertainment event.

In this project, a wide range of data science approaches were applied as a workflow of increasing complexity. Each component built upon prior components, resulting in a composition of processes that provided output as input to subsequent components (see Fig. 19.1 for the workflow utilized in this project). The following three subsections detail the (1) data processing, (2) clustering, and (3) modeling and visualization aspects of our project. In order to construct realistic population models, these three tasks were performed to answer a series of questions, grounded in latent patterns observed within CDR records. The following sections are structured based on these nine research questions the project attempted to answer.

Similar analyses were conducted for each country within the West Africa datasets. The following discussion interleaves representative examples and visualizations from each of the countries that were studied.

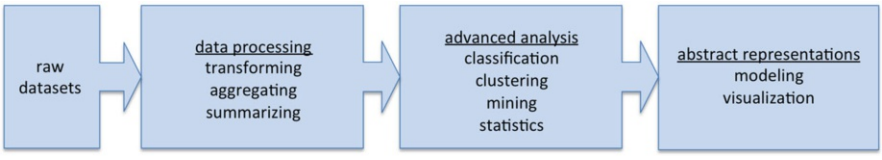

Fig. 19.1 Pipeline of tasks (workflow) that led to improved understanding of human activity and generated models of human behavior from the patterns latent in raw mobile phone datasets

4.1 Data Processing

Human-centric data includes numeric values and free text, contains continuous ranges and discrete values, is gathered automatically by sensors or is provided directly by humans, may be human or machine readable, and may be highly accurate to fuzzy or inexact. One collection may consist of the exact times and cell tower locations of an individual's phone call as collected by the commercial cellular infrastructure. Another collection might contain short, unstructured tweets about influenza, sickness, and illness as collected via social media or web scraping within a health and epidemic monitoring application, for example, Google and Twitter (Ginsberg et al. 2009). Data processing is often the first step in making sense of large volumes of varying types of data.

Question 1: Can the Data Tell Us Where Individuals Reside?
A technique used in data processing is the selection, extraction, and storage of subsets of a full raw dataset for more efficient analyses by subsequent processes. Studying a specific user's behavior as captured in a mobile phone dataset is computationally faster if that user's records have previously been indexed and aggregated separately from the datastream of the entire subscriber base. Of course, this efficiency comes at the expense of a larger storage footprint, requiring analysts to balance the storage costs against the computational costs of regenerating the content again given the likelihood of potential reuse of the intermediate information.

Data processing provided a quantitative means of attempting to answer the first research question. An (imperfect) assumption is that calls made late in the evening, overnight, and in the early morning hours likely originate from an individual's home location. As implied by Fig. 19.2, the process involved question formation, extraction of the required information, storage of the intermediate subset data, analysis, human review of the results for the purpose of gaining insights, and staging of the question answering procedure and code for reuse in the future.

In the course of the project, a subset of phone records corresponding to a given subscriber (ego) was extracted through data processing. The extracted subsets constituted the underlying data necessary for answering the following scenarios:

- Where does an individual (ego) spend most of her/his time?
- Where does the ego travel throughout the working hours of the day?
- How often does the ego travel?

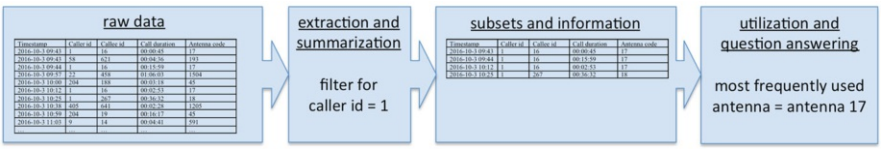

Fig. 19.2 Extraction of pertinent subsets from a raw dataset in support of a given question. In this example, a specific user's most frequently used mobile antenna is identified, providing individualized location information

- When does the ego travel?
- Did the ego migrate to a new location sometime during the year?
- What are all of the ego's weekday and weekend movement patterns (i.e., the series of movements between antennas that comprise an ego's behavior)?

Due to the multiple question scenarios and expected reuse of the content, it was advantageous to generate and store intermediate, ego-specific subsets of records. This step involved simple tasks, as demonstrated in Fig. 19.2: filtering the raw data by user ID and call location/time. Although this type of mining could have been performed via a database, the choice was made to employ a suite of Python scripts. Python is free, intuitive, and benefits from a wide range of powerful libraries and modules. It is well-suited for the large, flat-file format of the raw data. In addition, because the task involved relatively simple processing of extremely large data files, it is also quite suitable for distributed processing. Hence, the original Python scripts were converted to run under Spark,[3] an in-memory version of the Hadoop distributed computing framework.[4] This approach allowed for concurrent execution on a small (~64 node) virtual cluster, using aggregated distributed memory, resulting in significant speedups for the processing of billions of records. An additional benefit is that the PySpark scripts are "cloud-ready" in the event more processing power is required in the future.

4.2 Advanced Analysis

Going beyond basic data processing, analysis techniques such as clustering (Jain et al. 1999) were used to discover relationships in the data. In the *D4D* analysis of Côte d'Ivoire, the goal was to mine the dataset for information that could help public health officials develop more effective strategies for limiting the spread of infectious diseases. After using cell tower proximity data to situate subscribers (i.e., to determine their "home" location), clustering algorithms were then applied to identify groups of individuals expressing similar mobility patterns. The idea was that discovering knowledge about dynamic population densities could lead to better-informed public health interventions such as quarantine and isolation decisions.

Question 2: What Hidden Patterns Exist in the Data?
Since we were looking for hidden patterns and did not know in advance a good value for the number of clusters (k), hierarchical clustering (Sneath and Sokal 1971) was applied as a first step. The analysis targeted the prefecture administrative units of Côte d'Ivoire. As described above (see Sect. 4.1), the large raw datasets had previously been mined and filtered using Spark-based Python scripts to distill a much smaller file containing aggregate counts of calls between prefectures.

[3] spark.apache.org
[4] hadoop.apache.org

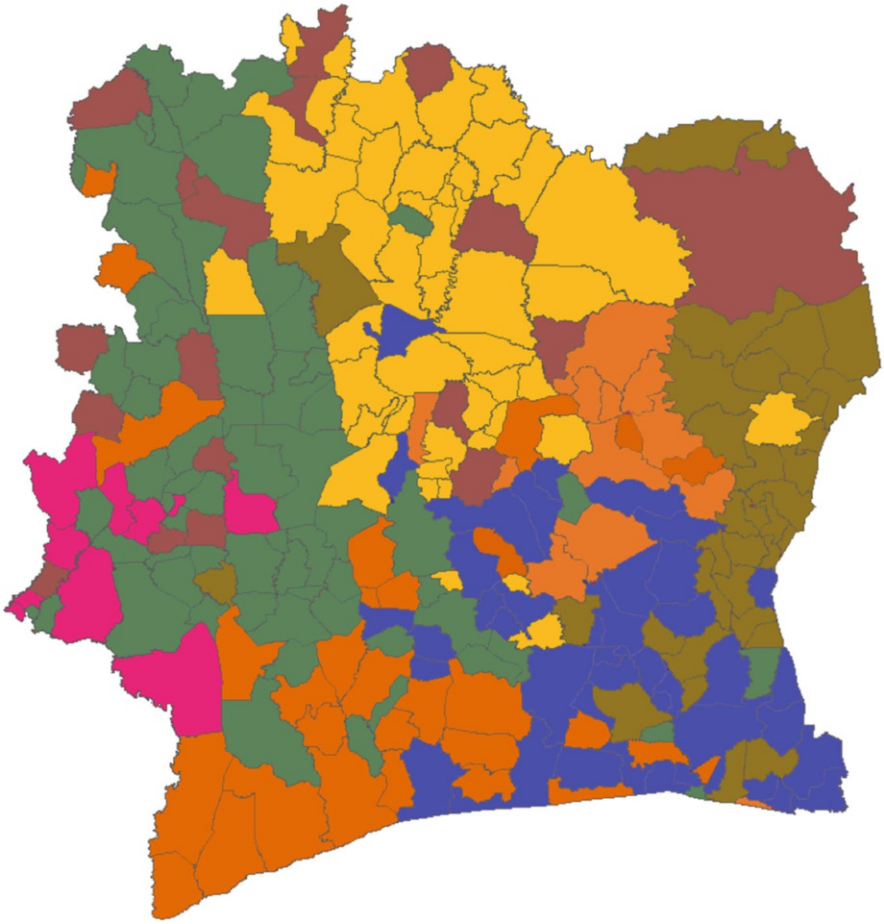

Fig. 19.3 K-means clustering (where $k = 7$) of Côte d'Ivoire administrative units (prefectures). The distance function between two locations is based on the number of calls between the two locations

In effect, this distributed approach was necessary only for the preprocessing steps (mining, filtering, aggregation). The resultant file, of manageable size, could then be efficiently clustered using the existing Python module `scipy.cluster.hierarchy`. The distance metric was defined as the total number of calls made between any two prefectures. The output of the agglomerative clustering was used to create color-coded maps; ranging from $k = 1$ cluster (the entire country) to $k = 255$ clusters (each prefecture is its own cluster). This resulted in one clustering ($k = 7$) that roughly corresponded to third-party maps identifying the ethno-linguistic groups in the country (see Fig. 19.3). The objective, quantitative CDR-based metric (number of calls between regions) revealed the same patterns of connectedness as groupings based on culture (e.g., religion, ethnicity, and language). This provided

some validation of the dataset and of the efficacy of using calls/texts as indicators of human behavior, and is also directly related to our health analysis project: given that people tend to rely on family and other social groups in times of crisis, this information helped identify likely group migration and disease diffusion effects in the event of an epidemic.

Question 3: Does the Data Show Travel Patterns?
In observation of the CDR population densities by prefecture, as estimated by subscriber base, it was evident that the population in some sub-prefectures was more mobile than other sub-prefectures. Therefore, estimates of the risk of disease spread between sub-prefectures should not rely on population density only but should also consider mobility and social mixing patterns. CDR datasets were first preprocessed using Python scripts to distinguish two major population groups: *static* users who made all of their calls in a single sub-prefecture and *dynamic* users who made a call in at least two sub-prefectures. The dynamic population is of particular interest in terms of potential geographic disease transfer.

Next, key sub-prefectures (those containing a border, airport, major city, hospital, clinic, or pharmacy) were identified. A pattern matrix of the user IDs of the dynamic population indexed by "key sub-prefectures visited" was generated and subsequently analyzed using Python's `sklearn.cluster.KMeans` module. K-means clustering (MacQueen 1967) of the dynamic population by shared, frequented sub-prefectures was performed. The goal was to find common subscriber mobility patterns (i.e., to identify sub-prefectures with a high degree of travel between them). Therefore, the definition of distance used in the clustering algorithm was based on the number of individual subscribers observed to have traveled between two sub-prefectures. The number of clusters, $k = 7$, was chosen based on the results of the agglomerative clustering described above.

The clusters of highly mobile, commonly visited sub-prefectures were seen to overlap at multiple locations with the key sub-prefectures containing important infrastructure. By identifying prefectures with a high degree of intra-prefecture mixing and travel, this analysis helped identify the most effective locations for interventions in the event of an epidemic. In terms of health policy, well-formed, targeted containment and mitigation strategies slow disease spread and buy time for inoculation and treatment of the population.

Question 4: How Can We Reduce "Noise" in the Data?
Other types of clustering were used for different purposes, such as cleaning the data. As a specific example, `sklearn.cluster.DBSCAN`, a density-based spatial clustering algorithm (Ester et al. 1996), was used to cluster antenna locations in an effort to reduce noise. Consider a user who lives right on the border between two antenna ranges. Because CDRs report the antenna at which a call originates, this individual might be seen to have "traveled" between cell towers without actually doing any physical movement other than walk around their house. This was considered noise in the dataset as it did not represent true user mobility. DBSCAN was employed with a minimum threshold of 500 m to discover and merge any spatially

close antennas. In effect, they were then considered a single antenna location for purposes of defining active mobile subscribers (i.e., the dynamic population).

4.3 Abstract Representation: Modeling and Visualization

The ultimate goal of the project was to model and then simulate, in silico, the progression of disease within a single human host and the propagation of that disease across a dynamically moving and mixing population of hosts. In order to accurately simulate these processes, models were needed at multiple levels of the host population. Unfortunately, standardized tools or formats for population models did not exist. We therefore created population models consisting of a variety of structures such as statistical distributions, matrices, network graphs, finite state machines, and Markov chains. As an example, statistical distributions were utilized to record the time at which calls were made throughout the day. Matrices were often used to aggregate data points, such as the number of calls made between antennas *i* and *j*. Graphs were used to highlight the interconnectivity between data points, such as the social network of users that call each other. Finite state machines were used to store abstract states, such as the progression and viral load of an individual infected with the Ebola virus as they became infected, remained infectious, and recovered from the disease. Markov chains were used to capture the probability of transitions between states, such as the likelihood of moving between all possible pairs of antenna *i* to antenna *j*. These structures provided a mechanism for first capturing population dynamics and later analyzing a variety of latent patterns within the datasets. The population models were built upon earlier data processing, clustering, and noise reduction tasks and then served as the underlying input files for several public health simulation systems. The dynamic, time-varying population and mobility models were constructed via answering the following questions.

Question 5: What Population Density and Census Levels Were Observed Throughout the Year?
The raw datasets were processed (as outlined in Sect. 4.1) in order to calculate the number of concurrent individuals near each cellular antenna at 10 min intervals. Individuals were assumed to remain at their last observed location. Infographics were used to verify our basic assumptions of the CDR datasets and our derived insights before incorporating data mining results into population models—see Fig. 19.4 for a high-level overview of the census model, aggregated to countrywide home and work locations. A day-by-day recalculating of an ego's home identified long-term travel and migration and was then stored in a vector of the dynamic ego's time-varying home location. The vectors for all egos' time-varying home locations were combined into a matrix containing time-varying features (columns) for all egos (rows). This full matrix contained all of the "permanent" movements by individuals over the course of a year. Similar visualizations and underlying matrices were produced for travel, transportation, migration, holidays, and weekend leisure.

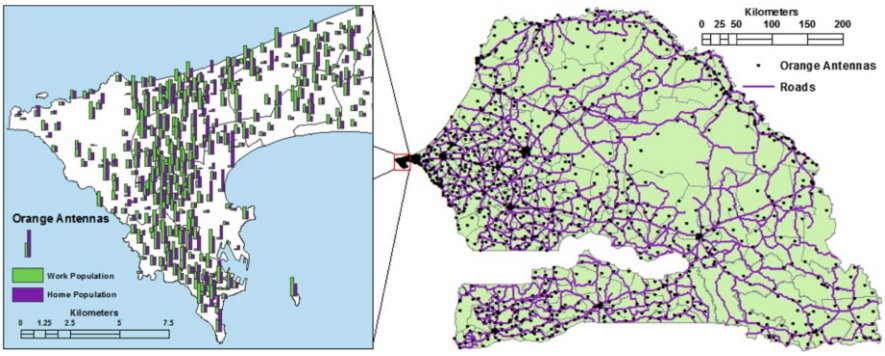

Fig. 19.4 Histogram on left displaying the home (dark) and work (light) locations of individuals at each cell tower throughout the Dakar capital region of Senegal, generated in ArcGIS (http://www.esri.com/arcgis). This visually illustrates large-scale daily commute patterns as people move between work and residential areas

Visualizations were used to validate the underlying data and matrices. The density and census insights formed the underlying location models for the simulation system case study, at fine-grained 10 min intervals for an entire year.

Question 6: What Were the Daily Behavior Patterns for Each Ego?
The full set of observations for a single ego was extracted from the raw dataset. The geo-temporal nature of an ego's behavior was used to classify antenna locations, trips, and activities (see Fig. 19.5). From an ego's records, it was possible to define and determine attributes for the ego. As examples, records were used to determine the ego's home location (defined as the place most often located from 7 pm to 7 am), workplace, work schedule, routes taken throughout the local area, holiday travel, weekend social trips, etc. These numerous individual-based patterns formed the underlying daily schedules and activity model for the simulation system case study.

Question 7: What Were the Mobility Patterns of Egos and the Aggregated Population?
Geospatial mobility indicates commercial activity, leisure activity, transportation and movement patterns, trips taken, seasonal migration, and long-term migration. The interactive chord diagram in Fig. 19.6 gives a complete view of aggregated migration between arrondissements in Senegal. The interactive visualization allows for selectively viewing a specific arrondissement, providing both a "big picture" and a detailed view down to the level of a single individual's movements. As an example, 1120 egos traveled to 21 other locations from Parcelles Assainies, Senegal during the time period selected in Fig. 19.6. The movement patterns for an ego were then assigned to a set of agents in our synthetic population models. The daily behavior patterns from the previous question provide details on every trip and path taken by individuals. A call at location A followed by a call at location B indicated a trip from A to B occurred during the elapsed time period. These patterns were extracted to identify daily schedules at the ego level and travel patterns at the population level

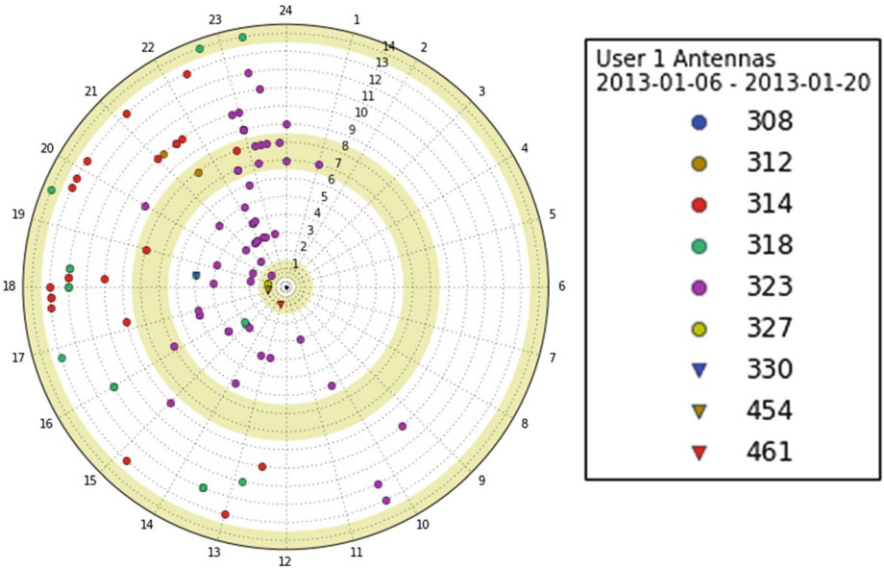

Fig. 19.5 Temporal visualization displaying the day, time, and location of every call for a given user—a polar coordinate graph generated in Python based on SpiralView (Bertini et al. 2007). Day 0 is represented as the innermost ring of the diagram; day 14 as the outermost ring. The 24 h are represented as a "clock." The antenna-based location of the user is represented by distinct colors. The proliferation of calls made at a specific antenna between the hours of 9 pm and midnight suggest the user's home was at that location

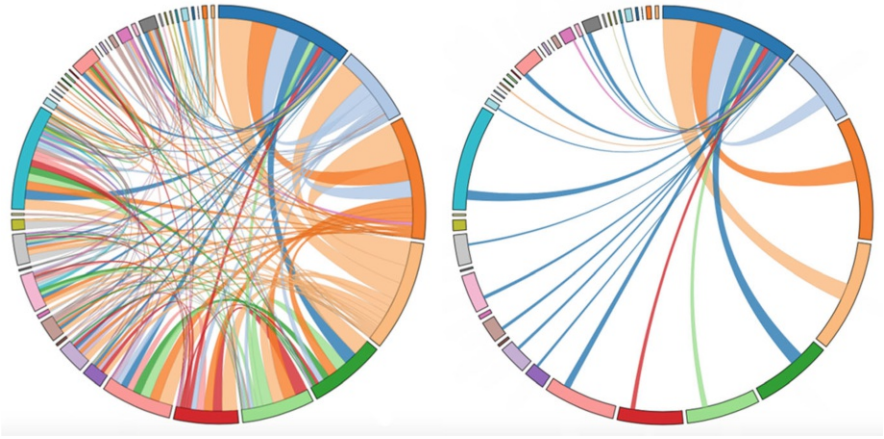

Fig. 19.6 An interactive chord diagram displaying migration from one arrondissement (labels hidden for clarity) to another in Senegal, generated with JavaScript and D3.js (https://d3js.org). The fully connected graph (left) was used to investigate the movement, mixing, and migration relationships between geographic areas. Selecting a specific arrondissement (Parcelles Assainies, right) applies a filter, showing incoming and outgoing migration for the selected arrondissement only. The width, color, and tooltips (not shown) provide additional features and statistics for each edge in the graph, allowing researchers to, for example, selectively investigate seasonal migration between agrarian arrondissements

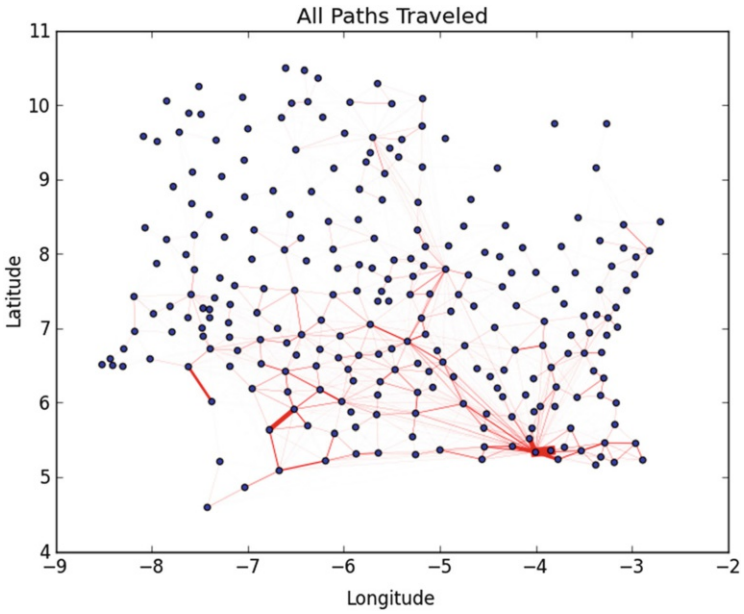

Fig. 19.7 Geospatial layout of the graph of cellular antennae in Côte d'Ivoire, generated with Python and matplotlib (http://matplotlib.org). Edges were weighted based on the observed travel between locations. Much of the country's traffic was centered around the port city, Abidjan (bottom right), and moving up to the capital city, Yamoussoukro

(see Fig. 19.7). These patterns were composed of both daily mobility patterns (e.g., commuter traffic in large cities and public transportation routes) and long-term mobility (e.g., seasonal relocation and population shifts). These numerous individual-based patterns formed the underlying movement models for the simulation systems case study.

Question 8: How Did Egos Come into Contact and Mix?
Networks (also called graphs) were well-suited for representing natural human circumstances and mixing. In mobile phone datasets, social networks are formed based on communication and colocation between users. Location networks are formed to describe an ego's probabilistic patterns of geo-temporal movements. Graphs were developed in this project for social connections, disease progression states, and travel. These graphs made up a large component of our population models. Figure 19.8 displays the two-hop social network for a given ego in the Côte d'Ivoire dataset, providing insight into the mixing, communication, and interaction between this individual and other agents. Similar graphs were utilized to determine the set of users colocated at a given antenna for a particular time period. These social networks formed the underlying mixing models for the simulation systems case study.

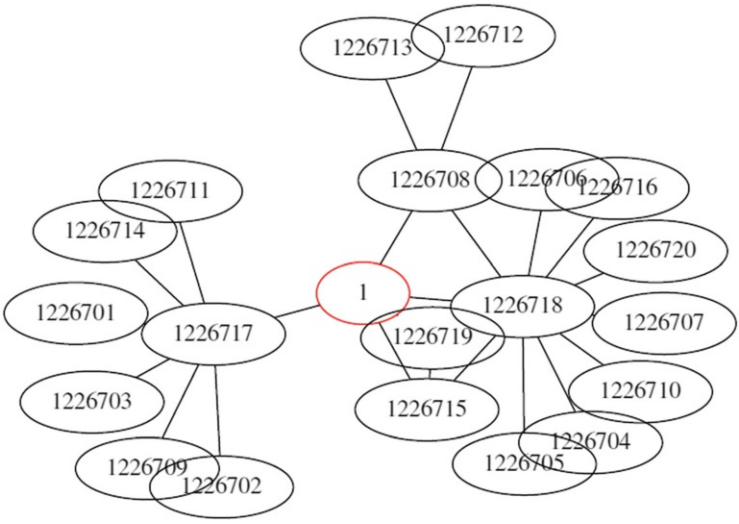

Fig. 19.8 Graph visualization of the 2-hop network based on direct phone calls between pairs of users (i.e., the friends and friends of friends) for ego "1" using Python visualization packages (https://networkx.github.io and https://pygraphviz.github.io). Note that this individual has several well-connected friends

Question 9: How Suitable Were the Constructed Models for Typical Simulation Systems?

The ultimate purpose of the abstract models was to serve as input to large-scale, agent-based simulation systems (Barrett et al. 2008; Bisset et al. 2009; Chao et al. 2010; Vogel et al. 2015). The models were designed to contain the daily movements of 13 million synthetic agents, based on real-world CDR datasets. Using the abstract models, synthetic agents were assigned a statistically valid profile (e.g., living within a particular suburban cell tower range and traveling with a given pattern within a nearby city). Thus, each of the abstract models "drove" a specific aspect of the simulation.

Animations were used to gain a better appreciation of the trends expressed in the raw data and its extracted patterns over time. Animations of simulation results demonstrated and verified that the underlying populations models were sufficient for simulation software requirements. They were automatically generated and integrated as part of our analysis pipeline (see Fig. 19.9). However, animations were time-consuming and human-intensive to review, without knowing beforehand which individuals or simulation runs would be interesting. For example, some egos were extreme outliers in terms of the number of locations they visited throughout the day, the differences in their weekday versus weekend behavior, or where they lived throughout the year. After identifying outlier egos, geospatial animations provided a better understanding of their unique movement patterns and effect on simulations.

Fig. 19.9 Geo-temporal animation of a sample simulation result modeling a potential epidemic in Senegal based on a specific scenario (generated with Google Maps and D3.js). The size and color of the points on the map indicate the number of infections at that location for a given simulated day. Compare (Leidig and Dharmapuri 2015)

5 Lessons Learned

Constructing high-resolution models that are statistically equivalent to a real-world population remains a challenging task. Latent patterns of geospatial, temporal, and behavioral significance were contained in CDR datasets. Data processing, data mining, machine learning, and visual analytics techniques were used to generate synthetic populations and models that served as inputs to simulation software.

- **Data processing**: Exponential combinations and permutations made it difficult to find a suitable balance between storage costs and computational costs. With millions of individuals and billions of records, loading and processing even subsets of the raw dataset required several hours of compute time. Simple calculations and transformations on the full dataset took several days when performed on sequential environments. Therefore, parallel algorithms and toolkits (e.g., Spark) were employed to reduce the computational costs of transforming the records of streaming, real-time observations into a format suitable for further analysis. Storing all of the possible trips or routes between antennas based on actual ego travel would have made trip pattern identification easier at the cost of ~100 TB of storage. Condensed intermediate datasets that summarized raw data points were required for most tasks.
- **Clustering**: Using multiple, unrelated datasets illuminated hidden patterns. For example, travel, migration, and communication were highly correlated with the dialects and cultural norms found in different subregions of the country. Thus, maps of the linguistic subregions of a country validated the selected value of $k = 7$ in clustering tasks. Also, techniques for reducing noise in the data were required. Noise would have caused model overfitting, misinterpretations, and falsely identified movements due to artifacts in the way the dataset was collected. In this case study, these techniques eliminated some of the noise caused by cellular antenna switching that was not actually due to significant physical movement. As always, it was important to select the right clustering algorithms

for each task (e.g., DBSCAN for a task where clusters are expressed in different densities of the data points, and K-Means where clusters are defined by distances).

- **Modeling and visualization**: There is a wealth of unexpected, latent information and patterns in human-centric datasets. Human-intensive programming, scripting, and data mining were used to build models that implicitly contained these patterns. Models constructed from person-centric sources were a noticeable improvement over historical approaches. As an example, previous mobility models largely assigned two approximate locations (home and work) for a synthetic agent through archived survey and questionnaire results. In contrast, person-centric sources in this case study were able to provide all 15 verified locations that a specific ego traveled in a given day. Explicit practical insights were identified via question answering. As the nature of the insights were not known a priori, visualizations were required to determine the next set of questions that could potentially be asked. Visualizations also served to evaluate the validity of scripts, algorithms, intermediate results, and generated models.

In summary, this workflow demonstrated the feasibility of developing population models for public health simulations that incorporate human behavior complexity by mining latent patterns found in large CDR datasets.

References

Barrett, C., Eubank, S., & Smith, J. (2005). If smallpox strikes Portland. *Scientific American, 292*, 54–61.

Barrett, C., Bisset, K., Eubank, S., Feng, X., & Marathe, M. (2008). EpiSimdemics: An efficient algorithm for simulating the spread of infectious disease over large realistic social networks. In *Proceedings of the 2008 ACM/IEEE Conference on Supercomputing (SC '08)* (pp 1–12). IEEE: Piscataway, NJ.

Bertini, E., Hertzog, P., & Lalanne, D. (2007). SpiralView: Towards security policies assessment through visual correlation of network resources with evolution of alarms. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST)* (pp 39–146) 2007.

Bisset, K., Chen, J., Feng, X., Vullikanti, A., & Marathe, M. (2009). EpiFast: A fast algorithm for large scale realistic epidemic simulations on distributed memory systems. In *Proceedings of the 23rd International Conference on Supercomputing (ICS '09)* (pp 430–439). ACM: New York, NY.

Chao, D. L., Halloran, M. E., Obenchain, V., & Longini, I. M., Jr. (2010). FluTE, a publicly available stochastic influenza epidemic simulation model. *PLoS Computational Biology, 6*(1), e1000656.

Ester, M., Kriegel, H.P., Sander,J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, Portland, OR.

Ginsberg, J., Mohebbi, M. H., Patel, R. S., Brammer, L., Smolinski, M. S., & Brilliant, L. (2009). Detecting influenza epidemics using search engine query data. *Nature, 457*(7232), 1012–1014.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys, 31*(3), 264–322.

Jiang, S., Ferreira, J., & González, M. (2015). Activity-based human mobility patterns inferred from mobile phone data: A case study of Singapore. In *International Workshop on Urban Computing*.

Leidig, J.P., & Dharmapuri, S. (2015). Automated visualization workflow for simulation experiments. In *IEEE Symposium on Information Visualization (InfoVis)*, Chicago, IL.

MacQueen, J. (1967) Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on Mathematical Statistics and Probability* (pp. 81–297). University of California Press.

Sneath, P. H., & Sokal, R. R. (1971). *Numerical taxonomy*. San Francisco: Freeman.

United States Department of Commerce: Bureau of the Census. (2017). *American community survey (ACS): Public use microdata sample (PUMS)*.

Vogel, N., Theisen, C., Leidig, J. P., Scripps, J., Graham, D. H., & Wolffe, G. (2015). Mining mobile datasets to enable the fine-grained stochastic simulation of Ebola diffusion. *Procedia Computer Science, 51*, 765–774.

Chapter 20
Economic Measures of Forecast Accuracy for Demand Planning: A Case-Based Discussion

Thomas Ott, Stefan Glüge, Richard Bödi, and Peter Kauf

Abstract Successful demand planning relies on accurate demand forecasts. Existing demand planning software typically employs (univariate) time series models for this purpose. These methods work well if the demand of a product follows regular patterns. Their power and accuracy are, however, limited if the patterns are disturbed and the demand is driven by irregular external factors such as promotions, events, or weather conditions. Hence, modern machine-learning-based approaches take into account external drivers for improved forecasting and combine various forecasting approaches with situation-dependent strengths. Yet, to substantiate the strength and the impact of single or new methodologies, one is left with the question how to measure and compare the performance or accuracy of different forecasting methods. Standard measures such as root mean square error (RMSE) and mean absolute percentage error (MAPE) may allow for ranking the methods according to their accuracy, but in many cases these measures are difficult to interpret or the rankings are incoherent among different measures. Moreover, the impact of forecasting inaccuracies is usually not reflected by standard measures. In this chapter, we discuss this issue using the example of forecasting the demand of food products. Furthermore, we define alternative measures that provide intuitive guidance for decision makers and users of demand forecasting.

1 Introduction

1.1 Sales Forecasting and Food Demand Planning

Accurate demand forecasts are the backbone of successful demand planning. In particular, for food products with short life cycles the choice of the most suitable

T. Ott (✉) · S. Glüge · R. Bödi
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: thomas.ott@zhaw.ch

P. Kauf
PROGNOSIX AG, Richterswil, Switzerland

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_20

forecasting method is of central concern for business and hence the question is a driver for applied research activities. It does not come as a surprise that a plethora of different forecasting methods have been developed and suggested for food demand planning (e.g., Da Veiga et al. 2014; Žliobaitė et al. 2012; Taylor 2007). The most prevalent methods are based on time series models or state space models, notably, exponential smoothing, Holt-Winters method, ARIMA models, Kalman filters or regression models [see, e.g., De Gooijer and Hyndman (2006) for an overview of the most common methods]. Furthermore, artificial neural networks have been used for demand planning for a long time (Doganis et al. 2006), while only more recently other classes of machine learning techniques such as regression trees or random forests (Bajari et al. 2014) have been utilized. Common commercial software solutions for demand planning, such as Inform add*ONE or SAP APO (Vasumathi and Shanmuga Ranjani 2013), typically employ one or more of these methods.

Demand planning takes more than good forecasts. For the actual planning, a number of boundary conditions such as inventory constraints have to be considered. Sales forecasting should focus on the demand of a product irrespective of these constraints as they often distort the figures about the actual demand. In an operational setting we often face the problem of one-step-ahead forecasting, that is, for a product we want to predict the demand at time step t based on the demand observations from times $t-1, t-2, \ldots, t-n$. In the following, we use X_i for the actual demand and F_i for the respective forecast. In order to estimate the past demand values $X_i(i = t-1, t-2, \ldots, t-n)$, the actual sales data is used. Special care has to be taken in stock-out situations, as sales data underestimates the real demand of the product. At the same time, the real demand of some substitute product might be overestimated. Hence, the availability of accurate past demand data is nontrivial. For the following considerations we will ignore this problem and assume that X_i closely reflects the actual demand.

1.2 *Successful Demand Forecasting: The Past and the Future Inside*

Statistical forecasting algorithms try to capture past sales patterns and project them into the future. However, these patterns can be disturbed or can even undergo disruptive changes. An experienced procurement manager has some intuition and beliefs about the driving factors of structural interruptions and their impact on sales quantities. Hence, she or he may adjust the forecasts manually, in accordance with the assumed impact of the driving factors that she or he considers relevant in advance. In practice, a manual intervention is often made when promotions are planned or when an upcoming event or specific weather conditions are supposed to influence sales. Clearly, human intuition can be an important source to incorporate the impact of information about the future, and yet, human intuition is limited. For instance, for humans it is often difficult to grasp cross-effects of many factors and, as

a consequence, humans often tend to overestimate the influence of a single factor. Hence, especially when dealing with a large product portfolio, a need for supporting software solutions arises that evaluate and employ external drivers for enhanced sales forecasting.

An example of such a software solution is PROGNOSIX Demand, which combines various forecasting approaches and additionally incorporates the experience of human experts in cases where not enough (or unreliable) data is available. The methodology is based on the common experience that there is not a single best forecasting method for everything. Depending on the product, available data and the current sales situation, different methods are more or less suitable. Hence, it is important to evaluate the methods in terms of performance, where the performance is usually put into relation with the forecast error, or forecast accuracy, respectively, evaluated over a certain period of time. Subsequently, there is a need for suitable error or accuracy measures. In the following, we will thus first discuss common error measures. However, in practice, one has to decide for one measure in order to judge the performance of different methods and to select the best one. Does it matter which error measure is used? What is the economic significance of the error? The answer is not always clear when using conventional measures, as we will illustrate in the subsequent sections.

1.3 Traditional Measures of Forecast Accuracy

The goal of good forecasting is to minimize the forecasting error(s).

$$e_t = F_t - X_t \tag{20.1}$$

The error is positive, if the forecast is too high, and negative, if the forecast is too low. Usually, the error is defined with opposite signs. Here, in the context of sales forecasting, we prefer the convention in Eq. (20.1), as a positive error means that we have some unsold products left (oversupply).

Traditional measures of forecast accuracy, also referred to as forecast error metrics, can be subdivided into four categories (Hyndman 2006). We will quickly review each by providing the most popular metrics for one-step ahead forecasts.

1. **Scale-dependent metrics** are directly based on the forecast errors e_t
   The most popular measures are the mean absolute error (MAE):

$$\mathrm{MAE}(n) = \frac{1}{n}\sum_{t=1}^{n} |e_t| \tag{20.2}$$

and the root mean square error (RMSE):

$$\mathrm{RMSE}(n) = \sqrt{\frac{1}{n}\sum_{t=1}^{n} {e_t}^2} \tag{20.3}$$

Here and in the following we assume that the forecasting series is evaluated over a period $t = 1, \ldots, n$.

2. **Percentage error metrics** aim at scale-independence, such as the widely used mean absolute percentage error MAPE:

$$\mathrm{MAPE}(n) = \frac{1}{n}\sum_{t=1}^{n} \left| \frac{e_t}{X_t} \right| \tag{20.4}$$

MAPE has the disadvantage of being asymmetric, as for a given forecast value F_t and $|e_t|$, the penalty is heavier if $e_t < 0$. Therefore, a symmetric form of the MAPE is used sometimes, where the denominator is replaced by $\frac{(X_t+F_t)}{2}$, or alternative measures have been suggested (e.g., Kolassa and Schütz 2007).

3. **Relative error metrics** compare the error of the forecasting with the error of some benchmark method. Usually, the naïve forecast (i.e., X_{t-1} for F_t) is used as benchmark, where the forecast value for a one-step ahead forecast is simply the last observed value. One of the measures used in this context is the relative mean absolute error (RelMAE), defined as

$$\mathrm{RelMAE}(n) = \frac{1}{n}\sum_{t=1}^{n} \frac{|\, e_t \,|}{|\, X_t - X_{t-1} \,|} \tag{20.5}$$

Here we assume that X_{t-1} is also available. Similarly, we can define the relative RMSE, also known as Theil's U (De Gooijer and Hyndman 2006).

4. **Scale-free error metrics** have been introduced to counteract the problem that percentage error metrics and relative error metrics are not applicable if zeros occur in the denominators. The mean absolute scaled error MASE (Hyndman 2006) introduces a scaling by means of the MAE from the naïve forecast, where the last value is used as forecast:

$$\mathrm{MASE}(n) = \frac{1}{n}\sum_{t=1}^{n} \left(\frac{|\, e_t \,|}{\frac{1}{n-1}\sum_{i=2}^{n} |X_i - X_{i-1}|} \right) \tag{20.6}$$

All these measures come along with certain advantages and disadvantages. For example, percentage error metrics are often recommended for comparing forecast performance across different time series. Drawbacks are the inapplicability if a demand value is zero and the vagueness of percentage values regarding the interpretation of the economic impact. For sales forecasting, a MAPE of 1% may be economically significant or insignificant, depending on the sales volume. The next

sections will address the issue of the economic significance of errors on the basis of concrete examples.

2 Cost Error Metrics

2.1 Which Metric Is Best: A Toy Example

We first study a prototypical situation in demand forecasting by means of a slightly caricatured toy example. For this, we created a random sequence of $n = 100$ samples from a Gaussian distribution with mean $\mu = 10$ and standard deviation $\sigma = 1$ (arbitrary units). This sequence is interpreted as the sales baseline. We then added five random peaks with height of $\Delta h = 4\mu$, which represent the increased demand due to external factors. Real-world examples of such factors are sales promotions/special offers, holidays, special weather conditions etc. The generated sequence is shown in Fig. 20.1. Furthermore, the output of two different forecasting models is depicted. The first model is a perfect baseline model that, however, cannot anticipate the peaks. The second model is able to perfectly predict the peaks, but is always slightly overestimating the sales otherwise. We modeled this situation by a slight upshift of the original time series by 1 unit.

Imagine a planner that has to decide which model to choose for future predictions. She or he has to resolve the trade-off between hitting the peaks while being slightly

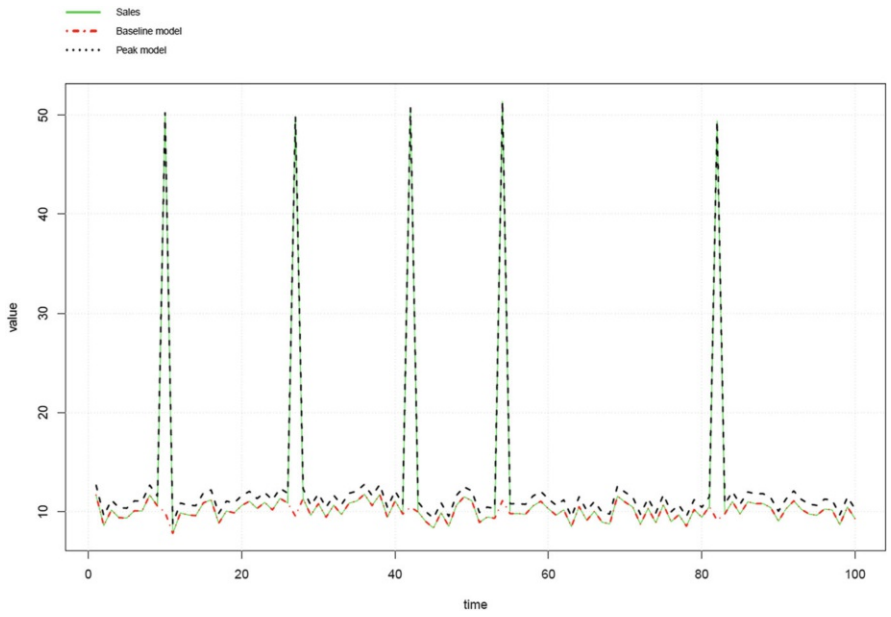

Fig. 20.1 Sales sequence (green) with five disruptive peaks, a perfect baseline model (red) that misses the peaks and a perfect peak model (black) which is slightly shifted in between peaks

Table 20.1 Results of four different error metrics

	MAE	RMSE	MAPE	relMAE
Baseline model	2.0	8.9	0.04	0.05
Peak model	0.95	0.97	0.10	2.74

The best results are highlighted

wrong in the meantime and being accurate most of the time but missing the peaks. For her or his decision, the planner evaluates the observed sequence using MAE, RMSE, MAPE, and relMAE. The results of the evaluation are shown in Table 20.1.

The result is ambiguous. MAE and RMSE speak in favor of the peak model, while MAPE and relMAE favor the baseline model. According to MAE, the time series model seems to be about "twice as bad" as the peak model; according to the RMSE, it seems to be even about "nine times as bad." Similar arguments can be produced for the comparison between MAPE and relMAE in favor of the baseline model.

The example illustrates the limitation of forecast error metrics for decision making. How can we resolve this issue? At the end of the day an economically relevant metric is defined by cost, that is, the financial consequences that come along with the prediction errors. Costs, however, can be highly product-specific and market-specific. Moreover, they depend on stock-keeping processes, an aspect we will discuss later.

2.2 Constructing Cost-Based Error Metrics

For now, let us assume that forecasting errors and costs are in direct relation. This is typically the case for fresh food products that cannot be stored and for which cost are directly related to sales. In consequence, a forecast that is too high results in costs for food waste and a forecast that is too low yields costs for stock-out. For goods that can be stored for an (un)limited time, there are storage costs instead of waste costs. In most practical cases, there will be a mixture of these types of costs. In any case, we assume that the forecast error e_t can be directly translated into costs $c(\cdot)$ and the costs do not depend on the history, that is,

$$c((X_t, F_t), (X_{t-1}, F_{t-1}), (X_{t-2}, F_{t-2}), \ldots) = c(e_t). \tag{20.7}$$

In the following, we will explain to what extend the metrics discussed above are able to reflect these costs and what kind of adaption would be needed to better account for real costs. We propose a generalized Mean Cost Error (MCE) of the following form:

$$\mathrm{MCE}(n) = s\left(\frac{1}{n}\sum_{t=1}^{n} c(e_t)\right), \tag{20.8}$$

where $c(\cdot)$ is a cost function and $s(\cdot)$ is a scaling function. Obviously, MCE defines a general form of a scale-dependent metric; MAE and RMSE can be considered special instances of MCE (see Fig. 20.2a, b).

If MAE and RMSE are interpreted in the framework of MCE, then it becomes apparent that these metrics impose some specific assumptions on the costs that may not be very natural in practice.

From the perspective of cost, a natural first approach is to neglect economies of scale and assume proportionality. Hence, excess stock cost or food waste cost ($e_t > 0$) increase proportional to the volume of the leftovers, that is, proportional to the forecasting error. For instance, costs may increase proportional to the manufacturing cost per unsold item or to the storage cost per unsold item (a: costs per item for $e_t > 0$). Similarly, stock-out cost increases proportional to the stock-out, for example, proportional to the unrealized profit or margin per item (b: costs per item for $e_t < 0$). Consequently, a first model is a piecewise linear cost model.

As a special class of MCE, we thus define the linear MCE (linMCE) as

$$\mathrm{linMCE} = \frac{1}{n}\sum_{t=1}^{n} c_{ab}(e_t) \quad \text{with} \quad c_{ab}(e_t) = \begin{cases} ae_t & \text{if } e_t \geq 0 \\ -be_t & \text{if } e_t < 0 \end{cases} \tag{20.9}$$

The measure is usually asymmetric as $a \neq b$ in general. In this setting, MAE is a special symmetric instance of linMCE (c.f. Fig. 20.2a, c).

Furthermore, we define a generalized class of scale-independent metrics that we call Mean Cost Percentage Error MCPE, as follows:

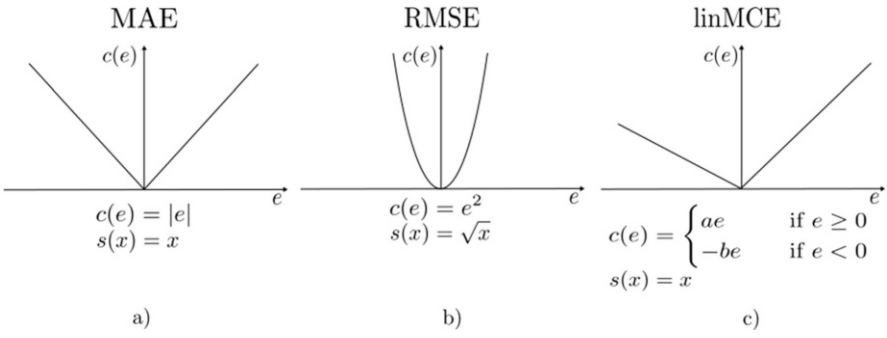

Fig. 20.2 Cost function of MAE, RMSE, and linMCE as special instances of MCE

$$\mathrm{MCPE}(n) = s\left(\frac{1}{n}\sum_{t=1}^{n}\frac{c(e_t)}{p(X_t)}\right), \tag{20.10}$$

where $p(\cdot)$ is given by some reference costs that are in connection with the real demand at time t. In the linear approach, we may for instance assume that $p(X_t)$ is proportional to the sales price of a product and to the number of items sold, that is, $p(X_t) = p \cdot X_t$. Hence, we define the linear MCPE as

$$\mathrm{linMCPE} = \frac{1}{n}\sum_{t=1}^{n}\frac{c_{ab}(e_t)}{p \cdot X_t} \quad \text{with} \quad c_{ab}(e_t) = \begin{cases} ae_t \text{ if } e_t \geq 0 \\ -be_t \text{ if } e_t < 0 \end{cases} \tag{20.11}$$

The measure can be interpreted as the mean of the costs due to the forecasting error in relation to the sales volume per forecasting period. MAPE is a special case of linMCPE with $a = b = p = 1$.

2.3 Sensitivity Analysis for linMCE

In order to calculate linMPE, we need to specify the parameters a (oversupply cost) and b (stock-out cost) for each product. Therefore, the costs per item for oversupply and for stock-out have to be made explicit. In practice, the parameters may be difficult to quantify exactly as, for instance, the oversupply cost can consist of a variable mixture of costs for food waste and storage. Thus, we may be interested in a more general comparison of forecasting methods or models with respect to the parameters a and b. For this sensitivity analysis we dissect the linMCE in an a-part and a b-part [i.e., using the Heaviside step function $h(\cdot)$]:

$$\mathrm{linMCE} = \frac{1}{n}\sum_{t=1}^{n} c_{ab}(e_t) = a \cdot \underbrace{\left(\frac{1}{n}\sum_{t=1}^{n} e_t h(e_t)\right)}_{\mathrm{linMC}\ \ E_a \geq 0} - b \cdot \underbrace{\left(\frac{1}{n}\sum_{t=1}^{n} e_t (1 - h(e_t))\right)}_{\mathrm{linMC}\ \ E_b \leq 0} \tag{20.12}$$

We can then study the relative performance of two forecasting models, $M1$ and $M2$, in dependence on the ratio x of a and b as follows:

$$f\left(x=\frac{a}{b}\right) = \frac{\text{linMCE}^{M1}}{\text{linMCE}^{M2}} = \frac{a\cdot\text{linMCE}_a^{M1} - b\cdot\text{linMCE}_b^{M1}}{a\cdot\text{linMCE}_a^{M2} - b\cdot\text{linMCE}_b^{M2}} = \frac{x\cdot\text{linMCE}_a^{M1} - \text{linMCE}_b^{M1}}{x\cdot\text{linMCE}_a^{M2} - \text{linMCE}_b^{M2}} \tag{20.13}$$

with the restriction that $x \geq 0$. Model 1 outperforms model 2 if $f(x) < 1$. Hence, as a critical condition for x we obtain

$$x_{\text{crit}} = \frac{\text{linMCE}_b^{M1} - \text{linMCE}_b^{M2}}{\text{linMCE}_a^{M1} - \text{linMCE}_a^{M2}} \tag{20.14}$$

In practice, one has to perform a case-by-case analysis to decide whether the critical point is in the relevant range $x \geq 0$ and to determine the values of $f(x)$. Hence, it is more convenient to plot this function, as we will discuss in the next section.

3 Evaluation

3.1 Calculating the Linear MPE: Toy Example Revisited

For our toy example we calculate the function $f(x)$ in a straightforward manner. The time series consists of $n = 100$ observations and 5 peaks with peak height $\Delta h = 4\mu = 40$. The peak model is shifted by $\Delta v = 1$ off the peaks. Hence, for the comparison of the baseline model ($M1$) and the peak model ($M2$), we get

$$f\left(x=\frac{a}{b}\right) = \frac{\text{linMCE}^{\text{baseline}}}{\text{linMCE}^{\text{peak}}} = \frac{2b}{0.95a} = \frac{2.11}{x} \tag{20.15}$$

which is derived from the following analysis of the linMCE (Table 20.2).

As a cross-check we see that the values for the MAE in Table 20.1 are retrieved for $a = b = 1$. The function $f(x)$ in Eq. (20.15) is continuously decreasing and the critical point is at $x = 2.11$. Therefore, the baseline model should be preferred if $a > 2.11 \cdot b$ ($x > 2.11$) and the peak model is the right choice if $a < 2.11 \cdot b$. That is, the peak model performs better if the oversupply (food waste/storage) costs per item are smaller than about two times the stock-out costs per item. This is due to the fact that a larger b in comparison to a, and hence a smaller x, puts a heavier penalty on stock-out situations that occur for the baseline model during peaks.

Table 20.2 Dissection of linMCE for the toy example (cf. Fig. 20.1)

	$a\cdot\text{linMCE}_a$	$-b\cdot\text{linMCE}_b$	linMCE
Baseline model ($M1$)	0	$5\cdot\Delta h\cdot\frac{b}{n} = 2b$	$2b$
Peak model ($M2$)	$(n-5)\cdot\Delta v\cdot\frac{a}{n} = 0.95a$	0	$0.95a$

3.2 Real World Example

In this section, we turn the focus on a real world example. Figure 20.3 depicts the demand data for a food product (weekly sales of a fruit juice) from a retail supplier. The sales sequence (blue curve) comprises of $n = 146$ values with mean $\underline{x} = 18{,}266$ and standard deviation $\sigma = 3783$ (units). The time series shows some characteristics that are typical for many food products, that is, there are characteristic peaks and dents due to promotions and the series exhibits a falling trend and, hence, is not stationary.

Following the toy example introduced above, we chose and fitted two models for one-step-ahead predictions. Both models are based on regression trees. However, they show a rather complementary behavior comparable to the models in the toy example before (cf. Fig. 20.3). One model can be considered as baseline model (red curve, model 1). It is able to predict the general development of the time series, but misses the peaks. In contrast, the second model (peak model; black dotted curve, model 2) takes into account additional external information and hence, is able to predict most peak demands. The price to pay is a reduced reliability between peaks. The model even predicts peaks that do not occur at all in the actual sales sequence.

With regard to the traditional error measures we observe the same picture as for the toy example (cf. Table 20.1). MAE and RMSE favor the peak model, whereas

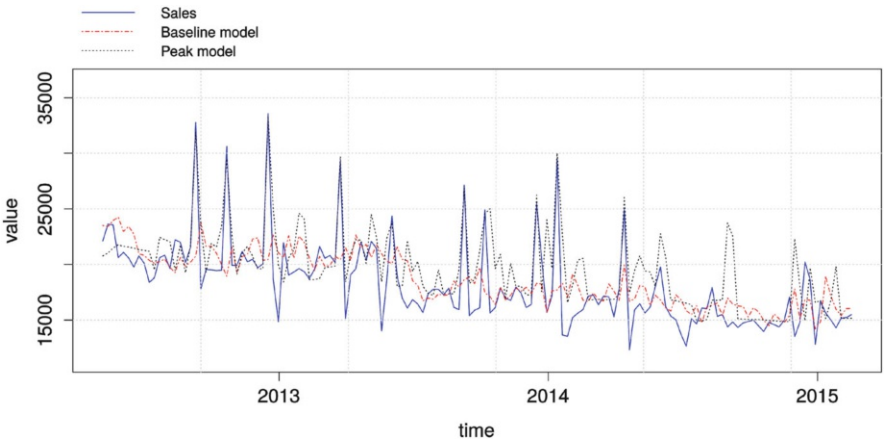

Fig. 20.3 Typical sales sequence and two different forecasts for a food product

Table 20.3 Results of four different error metrics for the real world example

	MAE	RMSE	MAPE	relMAE
Baseline model	2134.85	3234.45	0.1125	3.7248
Peak model	2080.34	2951.00	0.1253	6.2250

The best results are highlighted

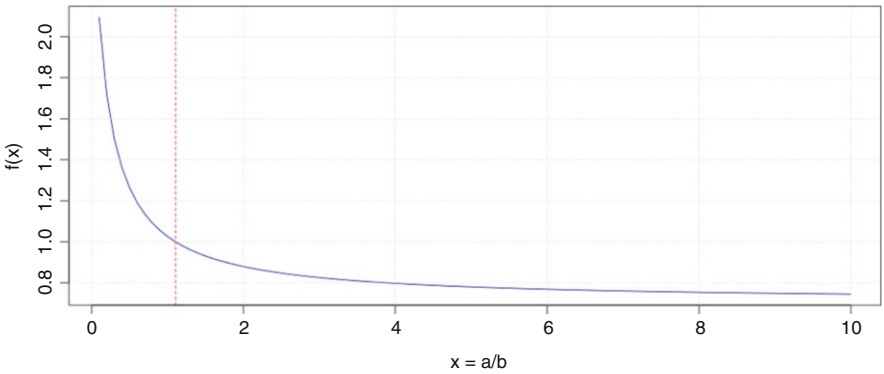

Fig. 20.4 Comparison of baseline model versus peak model as a function of the ratio a/b. On the right side of the critical point (red line), the baseline model should be preferred

MAPE and relMAE suggest using the baseline model (Table 20.3). In fact, relMAE even indicates that the naïve model should be used. The error measures were computed over the whole sequence.

The sensitivity analysis based on linMCE again allows for a clearer picture. In Fig. 20.4, the function f according to Eq. (20.13) is depicted. The critical point, highlighted by a red vertical line, is at $x_{\text{crit}} = 1.105$. We can conclude that the baseline model should be used in case $a/b > 1.105$, that is, if the stock-out cost per item is clearly smaller than the oversupply cost per item. In case $a/b < 1.105$, the peak model performs better since the stock-out costs per item are almost equal or larger than the oversupply costs per item. Again, for ratios $a/b < 1.105$, stock-out situations that occur for the baseline model during a peak are penalized more heavily and the costs for the baseline model are increased accordingly.

For the comparison of more than two models we suggest pairwise comparisons of each model with a benchmark, for example, the naïve prediction (last value is used as predicted value), which allows for a ranking of the models for each value of x by comparing the functions:

$$b_{\text{model}}\left(x = \frac{a}{b}\right) = \frac{\text{linMCE}^{\text{model}}}{\text{linMCE}^{\text{benchmark}}} \tag{20.16}$$

The result of this comparison for the baseline model and the peak model is shown in Fig. 20.5. We can identify three different regimes:

1. $0 < x < 1.105$, that is, if the oversupply costs per item are less than 1.105 times the stock-out costs, the peak model outperforms the baseline model and the benchmark model, the benchmark model is the worst choice.
2. $1.105 < x < 2.050$, the baseline model outperforms the peak model and the benchmark model, and the benchmark model is the worst choice.

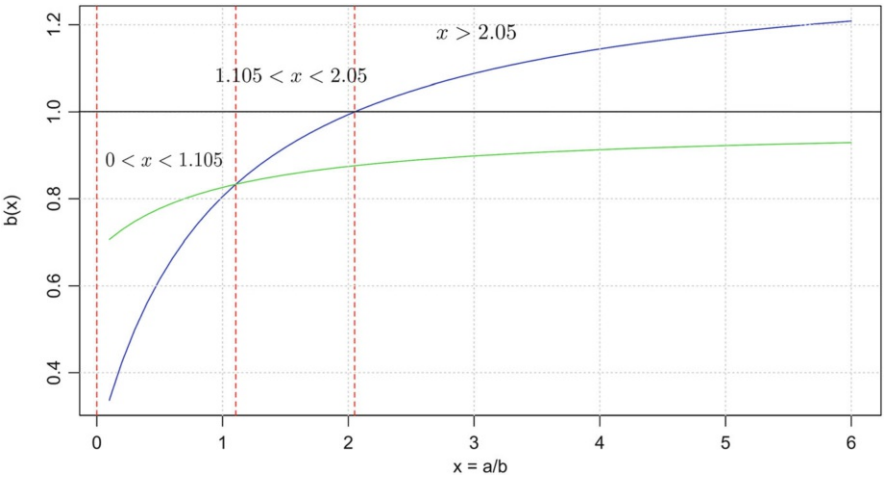

Fig. 20.5 Comparison of models versus naive model: green line: b_{baseline}, blue line: b_{peak}, black line: benchmark. The red lines indicate preference regimes of different models

3. $x > 2.050$, that is, if the oversupply costs are more than 2.050 times larger than the stock-out costs, the baseline model is best, the peak model is worst.

If we finally want to decide which model to use for our product, we need to make assumptions about the parameters a and b. For our example product (fruit juice), a per unit price of 1 CHF has to be paid to the producer and the product is sold for 1.2 CHF. Hence, the margin per unit is 0.2 CHF and this value is used to estimate the stock-out cost ($b \approx 0.2$). The oversupply parameter a is a bit harder to estimate in this case. As the time of permanency of this product is relatively large in comparison to the supply circle, an oversupply leads to an increase in stock rather than to food waste. The stock-keeping costs are estimated to be 10% of the average stock value, that is, 0.1 CHF per unsold unit per cycle. Hence, according to a first approximation, we choose $a \approx 0.1$.

In conclusion, we have $x = a/b \approx 0.5$ and hence the sensitivity analysis suggests to use the peak model.

3.3 Stock-Keeping Models: Beyond Simple Cost Measures

The measures for forecast costs presented so far were functions of demands/sales X_t and forecasts F_t. These measures are applicable in straightforward manner for goods with short expiration times as in this case, the parameters a and b are relatively easy to estimate (a corresponds to the production/base prize and b corresponds to the margin per unit). The estimations become more complicated for products with a longer time of permanency. In this case, as we have seen in the example above, we

have to make further assumptions about the stock-keeping process as the proposed measures do not take into account storage capacities. In general, for goods that can be stored, storage capacities, storage cost, and logistics strategies should be taken into consideration for a more reliable evaluation of the economic impact of forecasting algorithms. In the following, we present a simple stock-keeping model including stock capacities, service level, storage cost, ordering times, and product margins.

An important key figure in logistics is the *beta* service level, defined as the probability that an arbitrary demand unit is delivered without delay. Typical target values for the *beta* service level are at 0.99 or even higher. From the service level, a safety stock level can be derived, depending on assumptions about the demand distribution and the forecast accuracy. Intuitively, the more reliable the forecasts are, the lower the safety stock level can be, given a certain *beta* service level. Typically, the demand distribution is not known, but has to be estimated indirectly through the sales distribution (not yielding information about, e.g., potential stock-outs), as we pointed out earlier.

To compute the safety stock level in practice, normally distributed forecast errors $e_t = F_t - X_t$ are usually assumed (Brown 1967). From these errors, the root mean square error RMSE(e_t) can be computed. Defining

$$t_{\text{beta}} = \frac{((1 - \text{beta})\ D)}{\left(\text{beta}\ \sqrt{\text{Lt}}\ \text{RMSE}(e_t)\right)}, \tag{20.17}$$

where D is the average demand and Lt the average lead time, we compute $w = \sqrt{\log\left(\frac{25}{t_{\text{beta}}{}^2}\right)}$ and approximate the safety stock factor k_{beta} as (according to Brown 1967)

$$k_{\text{beta}} = \frac{-5.3926 + 5.6611 \times w - 3.8837 \times w^2 + 1.0897 \times w^3}{1 - 0.725 \times w + 0.5073 \times w^2 + 0.06914 \times w^3 - 0.0032 \times w^4}. \tag{20.18}$$

From k_{beta}, the safety stock level is computed as safety stock level $= k_{\text{beta}}$ sigma $\sqrt{\text{Lt}}$, with sigma denoting the standard deviation of the forecast errors e_t. Details on the derivation of the safety stock level can be found in Brown (1967).

With these foundations (simplifying Lt = 1), a stock-keeping model can be defined through

$$\text{Orders for time } t+1 = F_{t+1} + \text{safety stock level} - \text{stock at time } t.$$

To evaluate different forecasting strategies, X_{t+1} can be used as simulated demand and costs for stock-keeping and lost sales can be simulated for each forecasting model.

Table 20.4 Stock-keeping model results for the example presented in Fig. 20.3

Quantity	Baseline model	Peak model
Average stock level (units)	22,544	23,164
Safety stock level (units)	3886	3415
Effective *beta* service level (%)	98.08	99.92
Stock-keeping costs (CHF)	6329	6504
Opportunity costs (CHF)	10,266	385
Stock-keeping + opportunity costs (CHF)	16,595	6889

At a margin of 20% (0.2 CHF), stock-keeping cost differences are by magnitudes lower than the differences in opportunity costs for the two models

Applied to the example presented in Fig. 20.3, assuming again a per unit price of 1 CHF paid to the producer, a per unit price of 1.2 CHF paid by the customer, an average lead time Lt = 1 time period, a safety stock factor $k_{\text{beta}} = 0.99$, and annual stock-keeping costs of 10% of the average stock value, Table 20.4 shows a comparison between the baseline model and the peak model (146 periods). Note that more complex inventory models would allow for further parametrizations of expiration times for a product and correspondingly for estimations of waste cost.

As expected from Fig. 20.3, the peak model is more valuable in terms of opportunity costs than the baseline model. For stock-keeping cost, the baseline model is slightly more profitable. The effective beta service level "effective *beta* service level" is close to 99% for both models, indicating that forecast error distributions are in accordance with the assumptions stated above. The decision upon which model should be used can now be based on total costs. In this example, the peak model is to be preferred. This finding is in line with our result based on the linMCE analysis, where we found $\frac{a}{b} \tilde{} 0.5$. The stock-keeping model, however, allows for a more robust estimate of the economic significance of the two forecasting models. From Table 20.4 we see that choosing the right model helps reduce the costs by almost 60%, when changing from the baseline model to the peak model. Or in other words, if the decision would have been based on either MAPE or relMAE, the cost due to forecasting errors of the chosen model would have been at least 2.4 times as high as necessary in the case of the optimal decision.

4 Conclusions and Lessons Learned

Error metrics are used to evaluate and compare the performance of different forecasting models. The traditional, most widely used metrics such as MAE, RMSE, MAPE, and relMAE come along with certain disadvantages. As our examples from food demand forecasting illustrated, their values are often difficult to interpret regarding the economic significance and they may yield incoherent accuracy rankings. In practice, economically relevant metrics are linked to the costs that are caused by the prediction errors. We introduced a class of such measures that allow for

considering different weights for oversupply/excess stock costs and stock-out costs. It turns out that traditional measures can be interpreted as special cases in this class with specific cost configurations. In practice, costs for oversupply or stock-out might be difficult to determine. In order to cope with this issue, we introduced a method that enables a straightforward sensitivity analysis. It allows for choosing the optimal forecasting method on the basis of a relatively rough estimate of cost ratios.

The proposed cost error metrics, however, have no memory. That is, they assume that there is no stock available at the beginning of each step and the demand is equal to the supply of goods. This assumption is reasonable for the approximate evaluation of a forecasting method. However, real costs may not always directly reflect this performance, for example, for stocked goods a too low demand forecast does not necessarily lead to stock-out cost. It might even help reducing stocks and hence a bad forecast can have a positive effect on the costs. In order to better approximate real costs, simplified stock-keeping models can be used.

We illustrated the discussed aspects by means of a toy and a real world example. From these case studies we learned the following:

- The choice of the best forecasting model depends on the ratio of oversupply costs and stock-out costs.
- In particular, a baseline model should be preferred over a peak model if the oversupply costs are much higher than the stock-out costs and vice versa.
- Common error metrics do not account for this observation and can lead to bad model decisions.
- A bad model decision can easily result in an increase of the cost or the nonrealized earning potential by a factor of 2.4 for a single product.

An important aspect regarding the choice of optimal models that has not been discussed is the length of the evaluation time window. On the one hand, if the evaluation window is too short, random deviations without any significance can be predominant. On the other hand, if this window is too long, the good performance of a model in the distant past might masque structural disruptions that can cause a poor performance in the near future. For the model selection process, we thus generally suggest introducing an additional optimization loop that regularly adjusts the optimal length of the evaluation window. There is clearly not a unique optimally performing forecasting algorithm for everything. Similarly, to assess the qualities and economic values of forecasts, there is not a unique best forecast error measure. Different aspects, mainly involving costs of stock-keeping, stock-out, and waste, but also supply chain and marketing strategies (customer satisfaction, ecologic reputation, transport optimization, etc.) should be considered when evaluating forecasting procedures. The strategies presented here may provide a contribution to the goal of creating more economic value from demand forecasting.

References

Bajari, P., Nekipelov, D., Ryan, S., & Yang, M. (2014). Machine learning methods for demand estimation. *American Economic Review, Papers and Proceedings, 105*(5), 481–485.

Brown, R. (1967). *Decision rules for inventory management*. New York: Reinhart and Winston.

Da Veiga, C. P., Da Veiga, C. R. P., Catapan, A., Tortato, U., & Da Silva, W. V. (2014). Demand forecasting in food retail: A comparison between the Holt-Winters and ARIMA models. *WSEAS Transactions on Business and Economics, 11*, 608–614.

De Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International Journal of Forecasting, 22*, 443–473.

Doganis, P., Alexandridis, A., Patrinos, P., & Sarimveis, H. (2006). Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing. *Journal of Food Engineering, 75*(2), 196–204.

Hyndman, R. J. (2006). Another look at forecast-accuracy metrics for intermittent demand. *Foresight, 4*, 43–46.

Kolassa, S., & Schütz, W. (2007). Advantages of the MAD/Mean ratio over the MAPE. *Foresight: The International Journal of Applied Forecasting, 6*(6), 40–43.

Taylor, J. W. (2007). Forecasting daily supermarket sales using exponentially weighted quantile regression. *European Journal of Operational Research, 178*(1), 154–167.

Vasumathi, B., & Shanmuga Ranjani, S. P. (2013). Forecasting in SAP-SCM (Supply Chain Management). *International Journal of Computer Science and Mobile Computing, 2*(7), 114–119.

Žliobaitė, I., Bakker, J., & Pechenizkiy, M. (2012). Beating the baseline prediction in food sales: How intelligent an intelligent predictor is? *Expert Systems with Applications, 39*(1), 806–815.

Chapter 21
Large-Scale Data-Driven Financial Risk Assessment

Wolfgang Breymann, Nils Bundi, Jonas Heitz, Johannes Micheler, and Kurt Stockinger

Abstract The state of data in finance makes near real-time and consistent assessment of financial risks almost impossible today. The aggregate measures produced by traditional methods are rigid, infrequent, and not available when needed. In this chapter, we make the point that this situation can be remedied by introducing a suitable standard for data and algorithms at the deep technological level combined with the use of Big Data technologies. Specifically, we present the ACTUS approach to standardizing the modeling of financial contracts in view of financial analysis, which provides a methodological concept together with a data standard and computational algorithms. We present a proof of concept of ACTUS-based financial analysis with real data provided by the European Central Bank. Our experimental results with respect to computational performance of this approach in an Apache Spark based Big Data environment show close to linear scalability. The chapter closes with implications for data science.

1 Introduction

The financial sector is a challenging field of application for cutting-edge ICT technology. Indeed, its raw material being money and capital in different forms, which mostly is represented by mere numbers stored in computers, financial institutions can be viewed as applied IT companies. Still, their way of using the technological tools present severe shortcomings. Some of them have become evident during the 2008 financial crisis. Back then, a total financial collapse was averted only by massive public sector intervention. As to the underlying reasons, the Basel Committee on Banking Supervision recognizes in a recent report (Bank for International Settlement 2013)

W. Breymann (✉) · N. Bundi · J. Heitz · K. Stockinger (✉)
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: wolfgang.breymann@zhaw.ch; stog@zhaw.ch

J. Micheler
European Central Bank, Frankfurt, Germany

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_21

> . . . that banks' information technology (IT) and data architectures were inadequate to support the broad management of financial risks. [. . .]

The report identifies banks' poor capability to "quickly and accurately" aggregate risk exposures and identify concentrations of risk at "*the bank group level, across business lines and between legal entities.*" The same holds true for supervisory and regulatory institutions. In essence, we learned that both the public and the private sector lack the capability for flexible, near-time risk measurement that would allow us to understand quickly enough what is happening. This is especially true in the face of more complex, rapidly unfolding events, such as a global liquidity freeze, which can cause major disruptions and impose very high costs in a short span of time.

The state of financial data is emblematically illuminated by the weaknesses in "stress testing" the largest financial institutions. Stress tests are important risk assessment tools for judging the soundness of financial institutions and (ultimately) the financial system as a whole. One of the failings is the *impossibility of carrying out stress tests speedily and consistently across different large financial institutions*, much less the whole financial system. Currently, it takes a major bank several months to perform a regulator-required stress test. In addition to the lack of timeliness, the results are of limited value because the analyses completed for different institutions are not comparable with one another. The state of data in finance means that a near real-time and consistent understanding of finance is impossible today. The aggregate measures produced by traditional methods are rigid, infrequent, and not available when needed. The current limitations are illustrated by the *inability to measure, even imprecisely, the risks inherent in the development of subprime lending, securitization, and risk transfer in the run-up to the 2007 crisis* (UBS AG 2008).

This should be contrasted with what has already been achieved in other fields of human technology that require worldwide integration. Weather forecasting is now based on complex physical models fed by a constant stream of data from multiple observation sources (Vasquez 2011) and, as a result, has become quite reliable compared to forecasting of only a few decades ago. The data stream generated by all experiments of CERN's Large Hadron Super Collider together is reported to attain about 25 GB/s (CERN 2017) and is analyzed by a thousand particle physicists all over the globe; that is, CERN has a truly amazing Big Data challenge.[1] Important in all these activities are large-scale simulations, a scientifically recognized approach for understanding complex systems and establishing effective control. The most promising approach for simulating a complex system often starts with granular data. Simulating (parts of) the financial system on a granular level is a formidable computational task similar to weather forecasting or other large volume data processing tasks.

[1] It is interesting to compare the volume of this data flow to the bandwidth of the AEConnect cable, one of the newest transatlantic cables that make up the backbone of the Internet. It consists of 130 optical fiber with a bandwidth of 100 Gbps/fiber, thus adding up to 13 Tbps or about 1.3 TB/s for the whole cable (Lightwave 2017).

To avert in the future events such as the 2007–2009 financial crisis, which bear the risk of long-term damage to the economy and collapse of social well-being, it is essential to speedily address the failings revealed by this crisis (Shadow Banking 2017) and to raise the use of IT technology to a level already attained in other domains. It is not the absence of bandwidth or computing power that impedes the transformation of financial measurement and analytics comparable to the advances in weather forecasting and particle physics. It is the absence of a standardized representation of the basic elements of financial system that inhibits data integration and the efficient use of the full potential of the available ICT technology in the financial sector. Indeed, while the meteorological infrastructure evolved over decades in a collaborative effort, similarly powerful infrastructure does not yet exist for the financial system. Accordingly, we focus on two crucial aspects of the IT technology needed to remedy the situation and be prepared for the future:

1. The establishment of a combined standard for both, core financial data and core algorithms using these data as input for financial calculations
2. The use of an Apache Spark–based Big Data environment to massively speed up the calculations

As to the first point, standardization at the deep, technical level of the contract data is a necessary precondition for coherent financial measurement at the individual bank level and, a fortiori, the financial system level (Jenkinson and Leonova 2013). This standardization is addressed by the project ACTUS (Algorithmic Contract Type Unified Standards 2017), which is at the basis of the results presented in this chapter. Indeed, Francis Gross, former head of the European Central Bank's External Statistics Division, has recently pointed out (Gross 2014) the dire need of data standardization and singled out project ACTUS as one of three promising initiatives. This project centers on representing a financial contract as a set of contingent cash flow elements. In clear terms: who pays how much, to whom, when, and under what circumstances?"

More specifically, **ACTUS**, which stands for **Algorithmic Contract Types Unified Standards,** is creating an open-source standard representation of *financial contracts* (FCs), which are the basic building blocks of the financial system. The goal is to provide a common basis for forward-looking financial analysis of granular transaction and position data based on a rigorous way of modeling financial contracts. FCs are well-defined special-purpose legal contracts—also called financial instruments or securities—that control the cash flows exchanged between counterparties. Examples are stocks, bonds, futures, swaps, and options. The rules enshrined in an FC define the cash flows to be exchanged between the parties involved. The generated cash flow stream depends, however, also on external factors whose future evolution is not known, such as market interest rates or, more generally, the state of the economic and financial environment called risk factors in the following. It is these risk-factor *state-contingent cash flows generated by an FC that are the starting point for all financial analysis*. This is why we propose that a reliable and flexible analysis should always go back to the cash flow streams generated by the FCs.

ACTUS develops and makes publicly available (1) a universal data model covering the various terms of financial contracts, and (2) a machine-executable method to evaluate the financial implications of their legal obligations in the form of contract algorithms that use the contract terms and the risk factors as input and generate the contracts' future cash flows conditional to the state of the risk factors. ACTUS is pursuing the aim to make its contract types a global data standard for financial contract representation.

As two the second point, a Big Data environment is required because (1) financial institutions hold millions of FCs on their books and the whole financial system is estimated to consist of billions of FCs and (2) in order to assess the riskiness of the current situation, the cash flows generated by all those FCs must be simulated many times for different possible future states of the risk factors by means of so-called *stress tests* or by *Monte Carlo simulations* (Glassermann 2013), the latter method requiring thousands of different risk factor scenarios for an acceptable analytical quality. We are thus confronted with the task of calculating and analyzing the cash flow streams of up to trillions of FCs in a manageable time and of managing the tremendous data volumes of the order of Petabytes of cash flow data generated by the approach outlined above. It is obvious that such tremendous data volumes must be drastically reduced in the process of financial analysis before being presented to human analysts.

In the following, we present two types of results using ACTUS contract types and state-of-the-art Big Data technology:

- An ACTUS proof-of-concept provided by the prototype implementation of a stress test by means of a stylized Monte Carlo simulation using real contract data from the Centralized Securities Database (CSDB) of the European Central Bank. The CSDB contains this information for all securities issued in euros or in the European Union or held by European resident entities. This adds up to about seven million financial securities. We have mapped a part of these instruments on the ACTUS standard and analyzed a pilot portfolio using the ACTUS algorithms in a prototype environment.
- We describe how to design and implement a framework for financial calculations based on state-of-the art Big Data technology to enable large-scale, data-intensive computations. Afterward, we discuss our performance results of a parallel execution of the core ACTUS algorithms in an Apache Spark Big Data environment. Indeed, until now it has not been demonstrated that existing algorithms can (1) scale to Terabyte or even Petabyte-scale data sizes and (2) finish the calculations within a reasonable amount of time. The results presented here for a prototype system similar to the one used for the proof of concept indicate that this goal can indeed be achieved in all likelihood.

The remainder of this chapter is organized as follows. In Sect. 2 we present the ACTUS concept. Section 3 is devoted to the presentation of the financial analysis and the stress testing results while the results of ACTUS in the context of Big Data technology are described in Sect. 4. Section 4 can be directly accessed by the reader

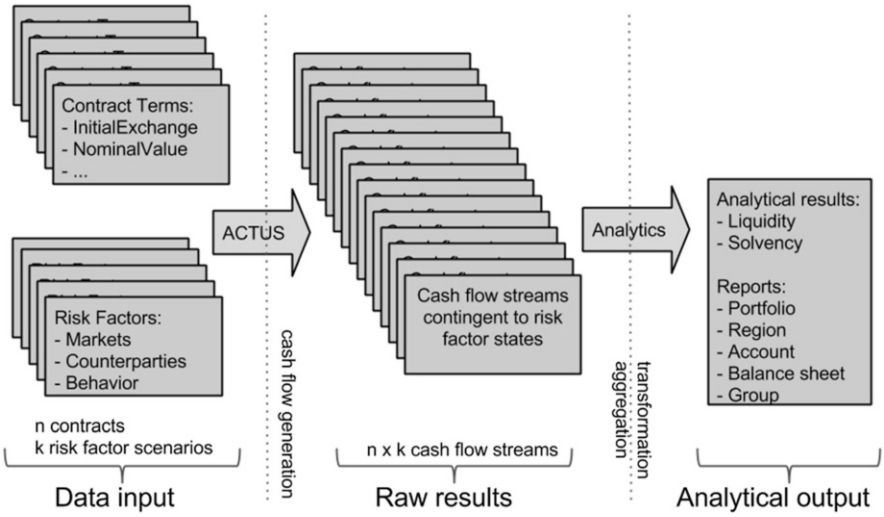

Fig. 21.1 Overview of data flow in the ACTUS framework

without a deeper reading of Sect. 3, if so desired. Section 4.2.2, finally, discusses our findings and conclusions.

2 The ACTUS Approach

In the following, we sketch the ACTUS concept of standardization of financial contract modeling and analysis (Algorithmic Contract Type Unified Standards 2017), which is based on the methodology presented in Brammertz et al. (2009). The analytical process is organized in the form of a data supply chain as depicted in Fig. 21.1. The main parts are as follows:

- **Financial contracts** play the central role in this model. They consist of contract data and algorithms. The contract algorithms encode the legal contract rules important for cash flow generation (who is paying how much, when, to whom, and under which circumstances), while the contract data provide the parameters necessary for the full contract specification. A typical example is a government bond with a principal value of US$1000, a 2% coupon rate, time to maturity of 7 years, and yearly coupon payments. For the holder, this bond generates a yearly cash flow of US$20 for 7 consecutive years and an additional cash flow of US $1000 at maturity for paying back the principal values.
- **Risk factors** determine the state of the economic and financial environment. They are further divided into factors for market risk, for counterparty risk, and for all the remaining risk factors lumped together in a third catch-all category called "Behavior." The important property of risk factors is that their future state is

unknown. The most important market risk factors are *interest rates, foreign exchange rates, and stock and commodity indices*. Counterparty risk factors typically consist of *credit ratings* and/or *default probabilities*.

- In order to generate the cash flows encoded in a contract, both contract data and risk factor information are needed. The reason is that the contract rules often refer to market information such as interest rates in the case of a variable rate bond. We explicitly emphasize that the separation of risk factors and contracts is important because it separates the known from the unknown: the contract rules are deterministic (known), while the future development of risk factors is unknown. For a given risk factor scenario, that is, an assumed future development of the values of the risk factors, the future development of the state of a contract and the cash flows it generates is completely determined and can be derived by the ACTUS contract algorithms.
- The **raw results** are **cash flow streams** together with some auxiliary information obtained as output of the contract algorithms. Assuming n contracts and k risk factor scenarios, there will be $n \times k$ cash flow streams consisting on average of about 20–50 events each. Since there are millions of contracts on a bank's balance sheet and an MC simulation does contain up to 10,000 or even more risk factor scenarios, the size of the data can easily be of the order of tens or even hundreds of terabytes for large institutions. This requires the use of Big Data technologies.
- Different types of financial analysis such as **liquidity and solvency calculations** are carried out on top of the raw results, referred to as **analytical outputs** or **analytical results**. This encompasses income analysis, sensitivity analysis and different kind of risk measures. Important is the possibility to flexibly transform and aggregate the raw data according to different analytical criteria.

3 Stress Testing of a Fixed Income Test Portfolio

In this section, we describe the ACTUS proof-of-concept that consists of the following steps: (1) mapping selected data from the data source onto the ACTUS format in order to establish the portfolio to be analyzed; (2) defining a suitable risk-factor environment and running the ACTUS algorithms on all contracts in the portfolio and all risk factor scenarios, which results in the raw results; (3) using the raw results for different types of financial analysis. Notice that the Big Data Spark environment has so far only been used for step 2. The corresponding results are presented in Sect. 4. Those readers who are interested mainly in the data science aspects and less in the details of the financial analysis may skip Sect. 3.3.

3.1 Mapping CSDB Data into the ACTUS Format

The data source for this proof of concept has been the centralized securities database (CSDB), which is jointly operated by the European System of Central Banks (ESCB) and aims to provide accurate and complete reference information of fixed income, equity, and fund instruments on a security-by-security basis. The CSDB has been designed as a multi-versatile system to provide data for multiple usages, spanning from analysis of data at its most granular level to aggregated time series data. It has not been tuned for only one single purpose, namely, supporting stress tests. Indeed, stress-testing securities requires a number of key information fields at the level of the single security spanning from instrument and issuer classifications and cash flow information to individual ratings. For our exercise, most of the fields could have been used with high quality from the CSDB. However, it turned out that the information on the underlying reference rate and the fixing formula for floating/variable rate instruments was not available with sufficiently high quality and coverage. Thus, the proof of concept used only fixed rate securities from the CSDB to be mapped into the format of ACTUS.

A first step of the mapping consists of identifying the ACTUS contract type that matches the cash flow patterns of the source securities. In our case, this turns out to be the Principal At Maturity type (Algorithmic Contract Type Unified Standards 2017). Once the Principal At Maturity type is identified, the ACTUS data dictionary (ACTUS Data Dictionary 2017) provides the associated data elements, that is, the contract terms such as *ContractID*, *LegalEntityIDCounterparty*, *InitialExchangeDate*, *MaturityDate*, *NominalInterestRate*, etc. Then, in a second step, a mapping between these data elements and the respective data fields in the source database (i.e., the CSDB) is required.

3.2 Running the ACTUS Algorithm with Suitable Interest Rate Scenarios

Fixed rate government bonds pay a fixed interest rate (so-called coupon rate) during their lifetime and, in addition, pay back the principal at maturity, except when the issuer defaults (fails) on its obligations. Thus, the generated cash flow streams do not depend on the interest rate curve and are only subject to credit risk. Valuation, however, requires discounting with the prevailing yield curve and thus is also subject to interest rate risk. Financial analysis, therefore, requires an interest rate model and a credit risk model. In addition, these models must be formulated relative to the analysis date, which has been fixed as of May 1, 2015.

At this step of the analytical process, we take into account only interest rate risk. Credit risk is taken into account only at a later state of the analysis (cf. Sect. 3.3). Since all securities are denominated in EUR-currency, we used the Euro-Area AAA-rated government bond implied yield curve published daily by the ECB (European Central Bank 2017) as an interest rate model for discounting. As base

Table 21.1 Euro-Area AAA-rated government bond implied yield curve published daily by the ECB (European Central Bank 2017)

Tenor	1 Yr	2 Yrs	3 Yrs	4 Yrs	5 Yrs	6 Yrs	7 Yrs
Rate	−0.26	−0.21	−0.13	−0.05	0.03	0.115	0.2

scenario we used the spot interest curve (IRC) as of the analysis date (see Table 21.1). In addition to these "base scenarios," we define a set of 100 different modifications of the IRC that we use as economic shock scenarios under which the value of the government bonds has to be reevaluated.

Then, the cash flow simulation by means of the ACTUS algorithms have been executed for the combinations of all the 3809 Euro-denominated fixed-rate government bonds in the portfolio and all the interest rate scenarios (which consist of the base interest rate scenario as well as the 100 shocked scenarios). The ACTUS framework is implemented in Java; it can be accessed through an *R*-interface that also offers a convenient means to carry out complex statistical operations on the ACTUS data. The simulation has been controlled by an *R*-script that uses the *R* interface to the ACTUS Java library provided by the package rActus (rActus 2017). The result is a table of about 3.8 million rows containing the raw results in form of the state-contingent cash flows of all the contracts and for all the risk factor scenarios. Table 21.2 displays the rows for two contracts and the base scenario.

3.3 Financial Analysis by Aggregation of the Raw Results

From the raw results the following types of financial analyses have been computed: (1) liquidity analysis without credit risk; (2) solvency analysis consisting of valuation under consideration of interest rate risk; (3) liquidity analysis under consideration of credit risk, which is taken into account only at the level of the analysis and results in a modification of the results obtained under item (1). The influence of credit risk on the valuation results (i.e., the so-called Credit Valuation Adjustment) has been omitted at this stage.

3.3.1 Liquidity Analysis Without Credit Risk

Since our test portfolio has long positions only, there are only incoming cash flows. Since our portfolio consists only of fixed rate bonds and credit risk has not been considered at simulation level, the simulated cash flows are fully deterministic. Figure 21.2 displays the yearly cash flows, aggregated by type (principal or interest payments) at the left and by countries at the right. Notice the diminishing interest payments over time (green part) as more and more bonds mature. The increase in

Table 21.2 Contract events with cash flows as of May 1, 2015

Contract ID	Event date	Event type	Event value	Time (in years)	Nominal value	Nominal rate	Nominal accrued	Currency	Country	Sector
DE0000000001	2015-05-01 T00:00Z[UTC]	AD0	0	0.086111	50,000,000	0.0352	151555.6	EUR	DE	S_1312
DE0000000001	2015-12-02 T00:00Z[UTC]	IP	1,183,111	0.586111	50,000,000	0.0352	0	EUR	DE	S_1312
DE0000000001	2016-12-02 T00:00Z[UTC]	IP	1,760,000	1	50,000,000	0.0352	0	EUR	DE	S_1312
DE0000000001	2017-12-04 T00:00Z[UTC]	IP	1,769,778	1.05556	50,000,000	0.0352	0	EUR	DE	S_1312
DE0000000001	2018-12-03 T00:00Z[UTC]	IP	1,755,111	0.997222	50,000,000	0.0352	0	EUR	DE	S_1312
DE0000000001	2019-12-02 T00:00Z[UTC]	IP	1,755,111	0.997222	50,000,000	0.0352	0	EUR	DE	S_1312
DE0000000001	2019-12-02 T00:00Z[UTC]	MD	50,000,000	0	0	0	0	EUR	DE	S_1312
GR0000000001	2015-05-01 T00:00Z[UTC]	AD0	0	0.038889	3,000,000,000	0.0475	5,541,667	EUR	GR	S_1311
GR0000000001	2016-04-18 T00:00Z[UTC]	IP	142,895,833	0.963889	3,000,000,000	0.0475	0	EUR	GR	S_1311
GR0000000001	2017-04-17 T00:00Z[UTC]	IP	142,104,167	0.997222	3,000,000,000	0.0475	0	EUR	GR	S_1311
GR0000000001	2018-04-17 T00:00Z[UTC]	IP	142,500,000	1	3,000,000,000	0.0475	0	EUR	GR	S_1311
GR0000000001	2019-04-17 T00:00Z[UTC]	IP	142,500,000	1	3,000,000,000	0.0475	0	EUR	GR	S_1311
GR0000000001	2019-04-17 T00:00Z[UTC]	MD	3,000,000,000	0	0	0	0	EUR	GR	S_1311

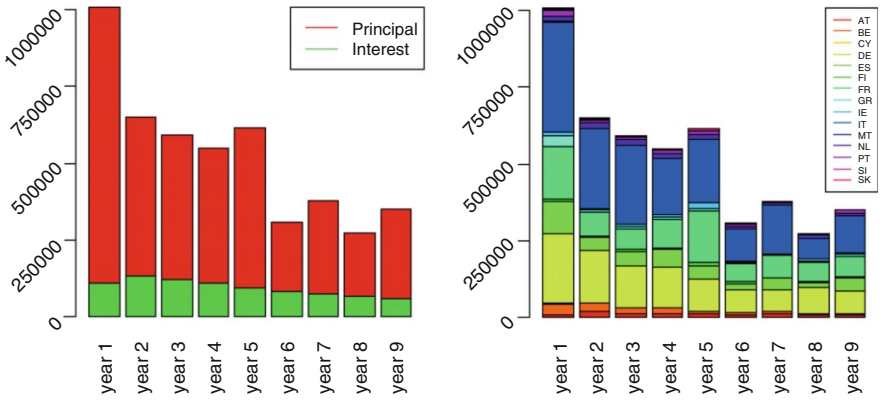

Fig. 21.2 Aggregate liquidity generated by central government issued bonds ordered by (left) cash flow type and (right) country of issuance

interest payments from the first to the second year is due to the fact that the aggregation has been carried out with respect to calendar years so that only 8 months were left from the analysis date till the end of 2015.

3.3.2 Valuation

Valuation requires discounting the cash flows with the prevailing IRC. Given the base interest rate scenario, that is, the IRC introduced in Table 21.1, this yields the following result for the example of the bond mentioned above with 1000 € principal, yearly coupon payment of 2% and 7 years' time-to-maturity:

$$20\left(\frac{1}{0.9974}+\frac{1}{0.9979^2}+\frac{1}{0.9987^3}+\frac{1}{0.9995^4}+\frac{1}{1.0003^5}+\frac{1}{1.00115^6}\right)$$
$$+\frac{1020}{1.002^7}=1125.92$$

This is an example for the type of calculations that are so fundamental for banks and must be carried out for any of the millions of contracts on a large bank's balance sheet in order to compute the market value of its portfolio.

A shock resulting in an increase of the interest rates of 1 percentage point will increase the denominators in the fractions and result in a decrease of the NPV to 1054.29 €, that is, a loss of its mark-to-market value of more than 6%. Taking into account 100 different shock scenarios results in a distribution for the bond's value. Carrying out this computation for all bonds and all the interest rate scenarios defined in Sect. 4.2 results in the histogram of the portfolio value displayed in Fig. 21.3. From this figure a risk measure such as the 99% Value-at-Risk (cf. red line) can be derived. Notice, however, that the cash flows have not changed because, as already

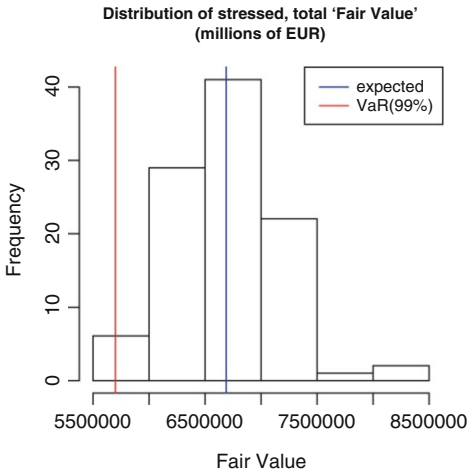

Fig. 21.3 Histogram of portfolio values created by 100 interest rate scenarios

mentioned above, the interest to be paid is set to fixed rates in the contracts so that the cash flows are not subject to interest rate risk.

3.3.3 Liquidity Analysis with Credit Risk

To take into account credit risk, a suitable credit risk model is needed. According to the general ACTUS logic, this should be defined at input level. However, due to the particular nature of credit risk it is possible to discard credit risk for the cash flow simulation and take it into account only when carrying out the analysis.[2]

Credit risk has been accounted for by weighting every cash flow with (1—default probability), where the 1-year default probability has been derived from the credit risk rating of the issuing country. One-year default probabilities are listed in the last column of the credit risk migration matrix displayed in Table 21.3. Default probabilities for n years are obtained by using the migration matrix as transition matrix in a Markov model [often referred to as CreditMetrics™ model (CreditMetrics 2017)], which uses the credit risk migration matrix as transition matrix for the changes of the credit rating from 1 year to the next. The second line of this table, for example, contains the transition probability of an AA rating to the other rating classes. In particular, the probability of an AA-rated counter-party to default within 1 year is 0.01%.

The result of the n-year liquidity is displayed in Fig. 21.4. The heights of the bars indicate the liquidity without credit risk as in Fig. 21.2 but here the cash flows are ordered with respect to their credit ratings. The red line shows the expected yearly

[2]To be precise, this is true only for simplified models of the kind considered here. The technical reason this can be done is that this type of credit risk models only require evaluating simple functions on the cash flows without the need to use the ACTUS algorithms.

Table 21.3 Migration matrix for credit risk ratings

	AAA	AA	A	BBB	BB	B	CCC	Default
AAA	93.66	5.83	0.40	0.08	0.03	0.00	0.00	0.00
AA	0.66	91.72	6.94	0.49	0.06	0.09	0.02	0.01
A	0.07	2.25	91.76	5.19	0.49	0.20	0.01	0.04
BBB	0.03	0.25	4.83	89.26	4.44	0.81	0.16	0.22
BB	0.03	0.07	0.44	6.67	83.31	7.47	1.05	0.98
B	0.00	0.10	0.33	0.46	5.77	84.19	3.87	5.30
CCC	0.16	0.00	0.31	0.93	2.00	10.74	63.96	21.94
Default	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00

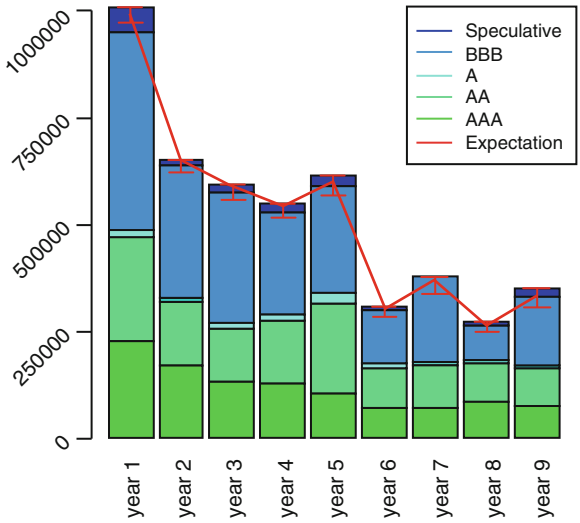

Fig. 21.4 Aggregate, state-contingent cash (in-) flows in millions of EUR of "central government" debt by credit rating and year as well as its expectation under a stochastic default model (red line)

cash flows computed with the probabilities of the migration matrix. Notice that they are smaller than those indicated by the bars and that the difference increases with time. Indeed, after 1 year, only about 20% of the speculative (CCC rated) bonds have defaulted while due to the credit migration effect, after 10 years practically all those bonds have defaulted. The error bars have been obtained through a simulation of a CreditMetrics™-type Markov model governed by the same credit migration matrix.

4 ACTUS in a Big Data Context

In this section, we describe our performance experiments of ACTUS in a cloud computing environment with different numbers of compute nodes. The main goal is to demonstrate the scalability of the ACTUS framework, which is indispensable for

the use of ACTUS for large banks or the whole financial system. We first discuss the design considerations for building a system based on Big Data technology. Then, we experimentally evaluate our approach based on two different Apache Spark (Zaharia et al. 2016) implementations. The first implementation is based on Spark-R, while the second is based on Spark-Java.

4.1 System Architecture and Design Considerations

We will first revisit the main components of the ACTUS framework and use them as the basis for discussing the architecture of the Big Data system. As shown in Fig. 21.1, the main components are:

- **Contract terms**: These quantities contain intrinsic information about financial contracts. They can be modeled as a database table consisting of n contracts and m attributes, such as initial exchange date, nominal value, etc.
- **Risk factors**: These quantities model the relevant part of the environment external to the contracts technically, they can be considered as a set of multivariate time series events of various interest rates, foreign exchange rates, or other market observations. Typically, financial analysis means evaluating the financial contracts under a specific or multiple scenario paths for these risk factors. Hence, similar to contract terms, risk factors can be modeled as a database table of *k* risk factor scenarios for t risk factor variables where each data point describes, for instance, the interest rate for a certain term at a certain point in time and in a certain scenario.

In order to calculate the cash flows of all financial contracts, the contract terms need to be combined with the risk factor scenarios as input for contract-specific calculations. In more abstract database terminology, the combination of contract terms and risk factor scenarios requires the calculation of a *cross product* (Kent 1983), that is, all pairs of contract terms combined with risk factor scenarios. Moreover, the calculation of the cash flows can be considered as a *user defined function* (Linnemann et al. 1988) that has to be applied on the cross product. Note that the user defined function can be any financial calculation of arbitrary complexity.

The combination of contract terms and risk factor scenarios results in complex calculations typically yielding a large number of cash flows, making it hard to calculate the cash flows in a reasonable time frame. For instance, a large bank might have on the order of 10^7 contract terms and 10^3 risk factor scenarios, which result in 10^{10} cash flow streams. If we assume that the output size of one cash flow stream is about 5 KB, then the total number of all cash flow streams is of the order of

50 TB.[3] If we further assume that we do not only want to calculate the cash flows for a single bank but for all the banks in the Euro Area or the European Union, the total size of the simulated cash flow streams is expected to be two to three orders of magnitude larger, that is, resulting in up to 50 PB.

Let us analyze what kind of Big Data approach we can use to tackle the above-mentioned challenge. In general, we can choose among the following **parallel processing paradigms** (Subhlok et al. 1993; Ramaswamy et al. 1997; Kambatla et al. 2014):

- **Task parallelism**: This kind of paradigm splits a task into subtasks and executes each subtask on a potentially different compute node[4] of the computer cluster. In other words, each node potentially executes a different task. This approach assumes that tasks can be modeled mathematically based on a certain cost model and allows analyzing the impact of splitting the tasks in an optimal way such that the workload is evenly distributed among the compute nodes. The main challenge with this approach is that it is often nontrivial to optimally schedule the subtasks in such a way that some compute nodes are not underutilized while others are not overutilized, thus hampering the scalability of the approach.
- **Data parallelism:** Rather than splitting the tasks, this kind of parallelism splits the data and distributes it among the compute nodes in the computer cluster. In this approach, each compute node executes the same task but potentially on a different part of the whole data set. The main challenge with this approach is that distributing the data can result in significant communication costs when input data or intermediate results need to be shipped from one compute node to another over a computer network.

When designing a Big Data architecture, one needs to keep these two basic paradigms in mind. What is more, the design choice largely depends on the **type of (parallel) computing problem**. For instance, some types of computing problem are so-called **embarrassingly parallel** (Wilkinson and Allen 2005); they do not require any particular communication between the parallel tasks. A typical example is a Monte Carlo analysis, where each Monte Carlo path can be calculated independently from the others and the final analysis only requires the aggregation of the individual outcomes into the final result. On the other hand, some types of **computing problems require intensive communication** between the tasks. Consider, for instance, a simulation of atom models where atoms interact with each other. This

[3]Obviously, the size of a cash flow stream differs for different financial contracts. For example, while a zero-coupon bond produces essentially two cash flows, or events more generally, a 30-year government bond with annual interest payments results in 2 (initial notional payment and repayment of notional at maturity) + 29 (annual interest payments) = 31 events. Further, for variable rate instruments interest rate resets must be taken into account, which further increases the number of events and consequently the size of the cash flow stream. Hence, the value of 5 KB for the size of a contract's cash flow stream is meant to be an average while the actual size of a contract's cash flow stream will strongly depend on the exact configuration of the contract under consideration.

[4]In Apache Spark, a compute node is referred to as worker node or simply worker.

kind of problem is the hardest to optimize since it requires modeling complex dependencies and hence it is often not easy to evenly balance the workload among the compute nodes in the cluster.

Luckily, the problem of calculating cash flows for financial contracts is an embarrassingly parallel problem that can relatively easily be implemented with a data parallelism paradigm.

We have chosen to implement our approach with **Apache Spark**, which is based on the data parallelism paradigm. The main idea of Spark is that data can be distributed among the compute nodes by leveraging a parallel file system such as Hadoop Distributed File System, HDFS (Borthakur 2008), Amazon's S3[5] file system, or others. According to the paradigm of data parallelism, Spark distributes the tasks among the compute nodes such that each compute node executes the same task but on a different subset of the data.

4.2 Experiments with SparkR

4.2.1 Experimental Setup

For our experiments, we have implemented the ACTUS functionality with Apache Spark. In particular, we have used SparkR data frames as the main building blocks for parallel computation. We have chosen SparkR since the majority of the financial calculations by end users are implemented in *R* while the ACTUS backend is implemented in Java.

The experiments have been executed on Amazon Web Services using up to 8 nodes with Elastic Map Reduce. Each node has 2 CPUs with 4 GB of RAM.

We measured the performance of generating ACTUS cash flows for various fixed-income instruments of the ACTUS Principal-At-Maturity type described in Sect. 2 and under different interest rate scenarios (shocks) (cf. the setup described in Sect. 3.2). In particular, we have chosen fixed and variable rate bonds with different coupon frequencies. Further, we have used the same interest rate model with the 100 shock scenarios as described in that section. Obviously, the simulation can be extended to any type and number of financial contracts and to more general risk factors and arbitrary shock scenarios.

In order to implement ACTUS in a parallel computing framework such as Apache Spark, the first design choice is the granularity of parallelism. The seemingly most natural way to parallelize the computation is to split the number of contracts by the number of nodes and have each node calculate the cash flows for a certain number of contracts. Unfortunately, this obvious approach turned out to be very inefficient for SparkR. The reason is the large overhead generated by the communication of the

[5] https://aws.amazon.com/s3/

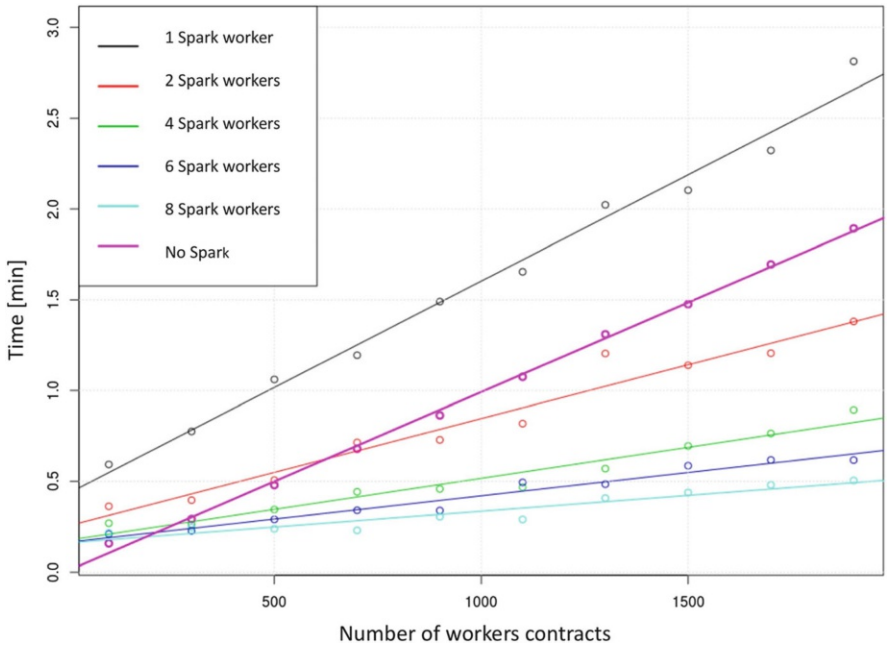

Fig. 21.5 Performance results for SparkR parallelizing by shock scenario. The execution time is a linear function of the number of contracts for different number of workers

R objects with the Java objects as implemented in the *R*-package rActus.[6] As an alternative, we partitioned the number of shock scenarios. This approach turned out to scale much better (Prenaj and Rizza 2016) since all contracts of the test portfolio could be read in one go, which reduced the communication overhead between *R* and Java significantly. However, the disadvantage is that the parallelism scales only to the number of shock scenarios—which was 100 in our case.

4.2.2 Results

Before carrying out the scalability analysis, we measured the time needed for generating the cash flows on a single compute node in the traditional *R* environment. In particular, we varied the number of contracts between 100 and 2000, and used 100 interest rate scenarios (shocks). Next, we measured the performance of ACTUS with SparkR on 1, 2, 4, and 8 compute nodes (Prenaj and Rizza 2016).

[6]Note that the additional overhead of converting objects between R and Java is a generic problem when building programs that interact with two different programming languages.

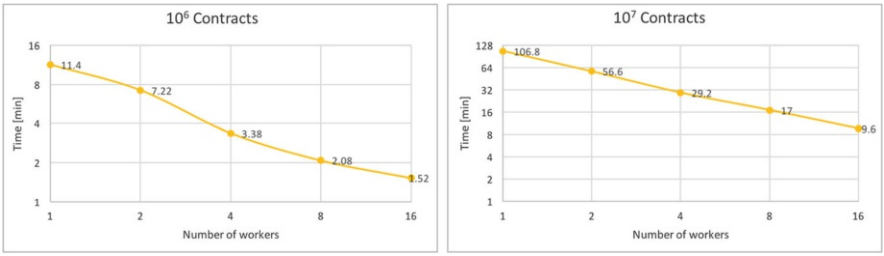

Fig. 21.6 Performance results for generating cash flows for 10^6 (left) and 10^7 (right) contracts. The code is implemented using the Java-API of Spark. Parallelization is per contract

Figure 21.5 displays the results where the code is parallelized by shock scenarios. The execution time as function of the number of contracts is shown in the left panel. For all cases the points essentially align on straight lines, indicating that execution time increases linearly with the number of contracts, as expected. The line depicting the behavior for the traditional (scalar) *R* environment (light blue) only has a small positive y-intercept indicating a short initialization time of about 2 s. Running ACTUS with SparkR on one worker node (black line) results in nearly the same slope but with a significantly increased initialization time of about 30 s, which is due to the buildup overhead of SparkR during the initialization phase of the RDD (resilient distributed data set, i.e., one of the main parallel abstractions in Apache Spark). As expected, the slopes decrease with increasing number of compute nodes (workers).

4.3 Experiments with Spark Java

4.3.1 Experimental Setup

For our second set of experiments we implemented ACTUS cash flow generation in Java using Spark. Rather than parallelizing the code by shock scenario, we now parallelized the code by contract. Since we have used the Java-interface of Spark, there is no overhead for converting contract objects between *R* and Java. Hence, we expect a significant performance improvement.

The experiments were executed on a Spark cluster with four nodes. Each node has 4 CPUs with 16 GB of RAM.

4.3.2 Results

Figure 21.6 shows the performance of generating cash flows for 10^6 (left) and 10^7 (right) contracts. In both cases, we see a behavior similar as in Fig. 21.5b, even though here, the data points do not align on a perfect straight line. A fitted straight

line has a slope of −0.87, again shallower than the perfect slope of −1, indicating a less than perfect scaling behavior. Indeed, given that the calculation of cash flows for 10^7 contracts takes 106.8 min for one worker, the ideal performance for 16 workers would be 6.7 min rather than 9.6. However, the scaling is better than for SparkR with a 45.2% decrease of computation time when the number of workers is doubled.

However, the current implementation allows processing more than 16,000 contracts per second. Assuming that a large bank has some ten million contracts, we can calculate the resulting cash flows for one risk-factor scenario in only about 10 min.[7]

5 Discussion and Conclusion

We have shown results concerning the use of the ACTUS standard for future financial and risk analysis. In particular, we presented (1) a proof of concept with real bond data obtained from the securities database of the European Central Bank; and (2) performance results of using ACTUS in a cloud-based Big Data environment. In particular, the following has been achieved:

- Cash flow results have been derived for a test portfolio of 3809 government bonds and a Monte-Carlo-like simulation consisting of 100 interest rate scenarios. The simulation has been carried out and compared with two different Spark-APIs by means of an *R*-interface (rActus) to the ACTUS Java library, and a Java-interface linking in the ACTUS Java library directly.
- Different types of financial analyses have been carried out using these raw cash flow results. Even though relatively simple, the results show that flexible aggregation according to different criteria as well as drilling down to individual instruments is possible. Extending this test case to more and different contracts as well as more sophisticated risk-factor models is straightforward. From a purely technological point of view, the ACTUS technology is sufficiently general in order to provide the basis for modeling the whole financial system. This would make it possible to forecast the financial system's risk state or part of it using Monte Carlo simulations.
- The greatest challenge is the task of convincing all the different, very powerful players to adopt a unique standard or, at least, to provide the information necessary to carry out the mapping from the variety of bespoke data formats to the ACTUS standard. This is illustrated by the fact that a mapping of more complex contract types contained in the CSDB described in Sect. 3 to the ACTUS standard is currently hampered by the fact that the information required by ACTUS is not fully available. Similar difficulties would occur with most existing data warehouses.

[7]We are currently running experiments with 100 s of workers on Amazon Web Services where we observed similar scalability characteristics. These results demonstrate that our approach of calculating cash flows is very promising.

- The cash flow generating part of the *R*-Script has been executed on Apache Spark/SparkR with up to eight worker nodes. Surprisingly, the ACTUS framework could be parallelized very easily when avoiding excessive communication between the *R* objects and the Java objects. The latter could be achieved by selecting the right type of granularity for parallelization when transforming the input data set into SparkR data frames. Since *R* already supports the concept of data frames, only minimal code changes were required. Unlike other parallel computing frameworks such as MPI or threads, the Spark abstraction is much more productive to turn existing code into a scalable application.
- We have implemented two different parallelization strategies (by shock scenario and by contract) using two different Spark-APIs, namely, SparkR and Java. However, our results show that SparkR involves significant overhead in the communication between *R* and Java objects through the Java Native Interface (JNI) and has significant scalability issues.
- The scaling behavior of the execution time in a Big Data environment using up to 16 Spark worker nodes is very encouraging. Recent experiments with 100 s of worker nodes on Amazon web Services show promising results and demonstrate that our approach is able to scale to hundreds of machines. We assume that the suboptimal scaling is due to the fact that the communication overhead increases more than linearly, which is often the case in parallel computing. Optimizing the scaling behavior requires more thorough work on the parallelization.

To conclude, the results presented here corroborate the belief that ACTUS has the potential of being a disruptive innovation in the domain of financial and risk analysis. ACTUS provides the necessary computational infrastructure for an efficiency revolution in financial risk assessment. The adoption of the ACTUS standard would make it possible to carry out large-scale simulation in finance similar to those in other fields such as physics, engineering, and supercomputing. However, in order to deliver on its full potential, ACTUS must be used together with other current development complementing its capacities, such as the Legal Entity Identifier, which improves our understanding of who our counterparties are. A system built on these new technologies will ultimately enable near-time stress tests and Monte Carlo simulation of financial institutions and ultimately the whole financial system on a daily basis, similar to the daily weather forecast. Notice that the analogy with meteorology can be pushed a step further. During the last years, so-called nowcast methods have been developed for very detailed weather forecasts up to 6 h based on the extrapolation of the current weather conditions, which are monitored by a dense network of automated sensors without the use of physical models. Similar techniques relying on the extrapolation of the current state of the economic-financial environment by a so-called forward-rate model could be used to generate very detailed short-term forecasts of the financial system's risk state without the need of costly Monte Carlo simulations.

5.1 Lessons Learned for Data Scientists

- Data science is a data-driven field. In finance, and in particular financial risk analysis, we often use analytical shortcuts because either granular data is not available or computational resources have not been sufficient in order to work at the granular level. Nowadays, the latter is not really a problem anymore. We have shown in this work that working at the level of granular data offers big potential for financial analysis and reporting. In particular, the complexities in risk aggregation can be reduced greatly, which is important in the light of the latest regulatory and accounting requirements.
- Modern technology enables conducting financial (risk) analysis at scale (business unit, department, organization, system) at the granular contract level. Our scaling analysis has shown that the risk analysis conducted in this work can be parallelized and distributed among multiple computing units (workers) showing linear scalability.
- ACTUS provides a formidable basis for data-driven financial (risk) analysis and reporting.
- An additional point, which could not be discussed in depth, is the separation of the raw results and the analytical output. Producing the raw results is very time-consuming, while carrying out the aggregation needed to generate the analytical outputs requires mainly linear operation. Thus, if the raw results are stored (which is possible with the current Big Data technologies) special purpose analytical results can be created quickly on demand. There could even be special companies providing such services, similar to the companies that currently offer special purpose meteorological reports.
- To summarize, modern data science methodologies in combination with the ACTUS data and algorithmic standard provide the technological basis for frequent risk assessment of financial institutions and ultimately the whole financial system with a potential similar to that of modern weather forecasts.

Acknowledgments One of the authors (W.B.) thanks the External Statistics Division of the European Central bank for having received him as visitor during his sabbatical. Without the ensuing collaboration this study could not have been carried out. We further thank Drilon Prenaja and Gianfranco Rizza for running the performance experiments as part of their bachelor thesis. The work is funded by the Swiss Commission for Technology and Innovation under the CTI grant 25349.1.

Appendix: The European Central Bank's Centralized Securities Database

The centralized securities database (CSDB) is jointly operated by the European System of Central Banks (ESCB) and aims to provide accurate and complete reference information of fixed income, equity, and fund instruments on a security-

by-security basis. The reference data of the CSDB is collected from multiple sources including National Central Banks and various commercial data providers. The CSDB is using compounding technologies that reconcile the information of instruments (banking products) and their issuers and selects for each single attribute the value with the highest data quality. The CSDB contains all instruments that are issued in the European Union or held by European resident entities.

The data model of the CSDB follows a relational structure that allows storing the data historically and on the most granular level possible. For each single instrument, the CSDB provides classification and identification measures, reference data including credit ratings and time-dependent measures such as prices, outstanding amounts, number of securities, and detailed cash flow related information like the underlying coupon and redemption payment structure or dividend data. In addition to the instrument data, each data source provides information on the respective issuer. Contrary to instruments, where the International Securities Identification Number (ISIN) is the established identifier, for issuers no globally unique and complete identification measure is available. Therefore, the CSDB has implemented a multipartite entity resolution algorithm that consistently combines the issuer data from multiple data sources identified by various proprietary IDs.

In addition to the collection and processing of instrument and issuer reference data, the CSDB also applies standard algorithms to estimate prices and to derive standard financial measures such as the yield to maturity, duration, or accrued interest.

The usages of the instrument and issuer reference data of the CSDB are manifold. The granular data of the CSDB allows full flexibility in aggregation and drill down operations. In the field of statistics, the CSDB data can be used to generate time series of aggregated stocks and transactions (gross issues and redemptions) of issued securities grouped by various dimensions such as country or economic sector. The data can be used to monitor refinancing needs of individual issuers observing the expected replacement costs of outstanding debt that needs to be redeemed.

The availability of credit rating information together with the coupon and redemption payment structure has the potential to create yield curves and to estimate yield spreads between different credit rating quality steps.

The flexibility of the CSDB data model was especially useful to map the data into the ACTUS format, thus serving as input of an ACTUS-based simulation approach.

References

Algorithmic Contract Type Unified Standards. (2017). Retrieved from http://actusfrf.org/

ACTUS Data Dictionary. (2017). Retrieved from http://actusfrf.org/index.php/data-dictionary-descriptions/

Bank for International Settlement. (2013, January). *Basel committee on banking supervision, principles for effective risk aggregation and risk reporting*.

Borthakur, D. (2008). *HDFS architecture guide*. *Hadoop apache project* (p. 53).

Brammertz, W., Akkizidis, I., Breymann, W., Entin, R., & Rustmann, M. (2009). *Unified financial analysis – the missing links of finance*. Hoboken, NJ: Wiley.

CERN. (2017). Retrieved August 31, 2017, from https://home.cern/about/computing/processing-what-record

CreditMetrics. (2017). *Technical document*. RiskMetrics Group, Inc. Retrieved from https://www.msci.com/resources/technical_documentation/CMTD1.pdf

European Central Bank. (2017). *Euro area yield curve*. Retrieved from https://www.ecb.europa.eu/stats/money/yc/html/index.en.html

Glassermann, P. (2013). *Monte Carlo methods in financial engineering*. New York: Springer.

Gross, F. (2014). Setting the standard in a complex world. *Financial World*. Retrieved from https://fw.ifslearning.ac.uk/Archive/2014/march/features/setting-the-standard-in-a-complex-worldG14

Jenkinson, N., & Leonova, I. S. (2013). The importance of data quality for effective financial stability policies–legal entity identifier: A first step towards necessary financial data reforms. *Financial Stability Review, 17*, 101–110.

Kambatla, K., Kollias, G., Kumar, V., & Grama, A. (2014). Trends in big data analytics. *Journal of Parallel and Distributed Computing, 74*(7), 2561–2573.

Kent, W. (1983). A simple guide to five normal forms in relational database theory. *Communications of the ACM, 26*(2), 120–125.

Lightwave. (2017, May 16). *RETN to use capacity on AEConnect submarine cable to cross Atlantica*. Retrieved August 31, 2017, from http://www.lightwaveonline.com/articles/2017/05/retn-to-use-capacity-on-aeconnect-submarine-cable-to-cross-atlantic.html

Linnemann, V., Küspert, K., Dadam, P., Pistor, P., Erbe, R., Kemper, A., et al. (1988, August). *Design and implementation of an extensible database management system supporting user defined data types and functions*. In VLDB (pp. 294–305).

Prenaj, D., & Rizza, G., (2016). *Performance evaluation of financial algorithms with big data technologies*. Bachelor thesis, Zurich University of Applied Sciences.

rActus. (2017). *rActus - an R-Interface to the ACTUS contract types*. Zurich University of Applied Sciences.

Ramaswamy, S., Sapatnekar, S., & Banerjee, P. (1997). A framework for exploiting task and data parallelism on distributed memory multicomputers. *IEEE Transactions on Parallel and Distributed Systems, 8*(11), 1098–1116.

Shadow Banking. (2017). *Committee on economic and monetary affairs*. Retrieved from http://www.europarl.europa.eu/sides/getDoc.do?type=REPORT&reference=A7-2012-0354&language=EN

Subhlok, J., Stichnoth, J. M., O'hallaron, D. R., & Gross, T. (1993). Exploiting task and data parallelism on a multicomputer. *ACM SIGPLAN Notices, 28*(7), 13–22.

Wilkinson, B., & Allen, C. (2005). *Parallel programming: Techniques and applications using networked workstations and parallel computers*. Upper Saddle River, NJ: Pearson/Prentice Hall.

Vasquez, T. (2011). *Weather analysis and forecasting handbook*. Garland, TX: Weather Graphics Technologies.

UBS AG. (2008, April). *Shareholder report on write-downs*. Zurich. Retrieved from http://maths-fi.com/ubs-shareholder-report.pdf

Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., et al. (2016). Apache spark: A unified engine for big data processing. *Communications of the ACM, 59*(11), 56–65.

Chapter 22
Governance and IT Architecture

Serge Bignens, Murat Sariyar, and Ernst Hafen

Abstract Personalized medicine relies on the integration and analysis of diverse sets of health data. Many patients and healthy individuals are willing to play an active role in supporting research, provided there is a trust-promoting governance structure for data sharing as well as a return of information and knowledge. MIDATA.coop provides an IT platform that manages personal data under such a governance structure. As a not-for-profit citizen-owned cooperative, its vision is to allow citizens to collect, store, visualize, and share specific sets of their health-related data with friends and health professionals, and to make anonymized parts of these data accessible to medical research projects in areas that appeal to them. The value generated by this secondary use of personal data is managed collectively to operate and extend the platform and support further research projects. In this chapter, we describe central features of MIDATA.coop and insights gained since the operation of the platform. As an example for a novel patient engagement effort, MIDATA.coop has led to new forms of participation in research besides formal enrolment in clinical trials or epidemiological studies.

1 Introduction

Developing new kind of data-science-based products and services relies more and more on the analysis of personal data, especially in the medical domain (Dhar 2013; Jeff Leek 2013). A key question about personal data is by whom and how they are controlled. Medical data generated in the course of healthcare services are stored by the healthcare providers and cannot be used for research without the explicit informed consent of the individual. Health-related data generated by mobile apps, smartphones, and sensors are collected and stored by the service provider. By

S. Bignens (✉) · M. Sariyar
Institute for Medical Informatics, University of Applied Sciences Bern, Bern, Switzerland
e-mail: serge.bignens@bfh.ch

E. Hafen
Institute of Molecular Systems Biology, ETH Zürich, Zürich, Switzerland

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_22

accepting the general terms of use of the devices or apps, the users usually sign off the rights of their data reuse to the service provider, thereby ceding control over the secondary use of their data. Services providers and in general those who control big amounts of data have little incentive to abandon the siloed approach to data, which in turn hinders true integration of different personal data sets.

Effective integration of personal data for personal health and precision medicine will be facilitated if the individuals regain the control over the secondary use of their personal data (the individual can then decide to hand this control over to a third party). Article 15 of the new EU general data protection regulation (EU GDPR: https://www.eugdpr.org) introduces the right to get a copy of such data. In addition, Article 20 introduces the right to data portability, which enables individuals to transfer at least parts of their personal data from one service provider to another and obliges data controllers to have systems in place that allow individuals to realize their data portability right (Datenschutzbeauftragte 2018). In Switzerland, the federal council has accepted to evaluate a parliamentary motion for the "Right to a Digital Copy" of one's personal data (Derder Fathi 2015). The Swiss Right to a Digital Copy, if approved, would grant individuals the right to control data reuse. Therefore, it can potentially empower citizens to actively control the secondary use of their personal data (Hürlimann and Zech 2016).

As the availability of comprehensive health data is more and more crucial for better health outcomes, there are worldwide initiatives targeting the patient-controlled integration of health data. A synonymous term to personal health record, which is used especially in the USA is "health record banking" (Yasnoff and Shortliffe 2014), used for example by the Health Record Banking Alliance (Health Record Banking Alliance 2018). The main goal of this alliance is to increase public awareness of personal health records and share lessons learned and best practices in personal health records. A systematic literature review on this topic is provided by A. Roehrs et al. (2017) that reviewed over 5000 articles. Especially the difference in governance structures (there are also personal health record initiatives, where the data management is not in the responsibility of the citizen) and in the goals (just recording or also manipulating and analyzing the data) are worth mentioning.

It is still a challenge make personal health record data available for research. The following activities are essential for implementing applied data science projects based on personal data:

- Definition of **benefits** resulting from new forms of data management and analytics
- Specification of the data to be **collected** and its source
- Consideration of legal issues relating to the usage of the data, particularly when personal data of **individual citizens** are involved
- Implementing a **governance** and business model for the collection and use of the data
- Operation of secure and scalable **IT platform** that can be augmented by third-party applications
- and finally, taking advantage of the present and future **opportunities** while managing the risks.

This chapter describes these different aspects and illustrates them with the help of the citizen science initiative named "MIDATA.coop"[1] [for a general introduction to citizen science, we refer to Irwin (2001) and Bonney et al. (2009)] that has addressed those challenges and is solving them in a technical, economical, and ethical way.

MIDATA.coop embodies an IT platform managing personal data and the governance needed to operate it. As a not-for-profit, citizen-owned cooperative, its vision is to allow citizens to collect, store, visualize, and share specific sets of their health-related data[2] with friends and health professionals, and to make anonymized parts of these data accessible to medical research projects in areas that appeal to them. The value generated by this secondary use of personal data is managed collectively to operate and extend the platform and support further research projects.

2 Utility and Benefits in Using Personal Health Data

The key to successful data science projects is addressing a specific need and producing direct benefits for all stakeholders involved in the process of generating, gathering, managing, using, and analyzing data.

Regarding personal data in general and health data in particular, the stakeholders are manifold (see Fig. 22.1). There are patients or just citizens (if we consider the healthy ones as well), health professionals as individuals or healthcare institutions, researchers, industry (in particular the pharmaceutical, medtech, and IT industries), and public health policy makers. In the following, we describe the kind of benefits that can be targeted for each stakeholder group when using MIDATA as a platform for applied data science.

2.1 Benefits for Citizens and Patients

Collecting health data and making them available to citizens and patients empowers and transforms them from rather passive objects in the healthcare system to active subjects that are empowered to influence the resource allocation in the healthcare system by governing the aggregation and distribution of their data (Woolley et al. 2016). The citizens can be supported in collecting and visualizing their data. As an extension of the quantified-self movement (Appelboom et al. 2014), in which fitness

[1] https://www.midata.coop

[2] While in a first phase MIDATA.coop targets health-related data, its governance and IT architecture allow to extend its use to other personal data, for instance, education data, whose secondary use value is rapidly growing.

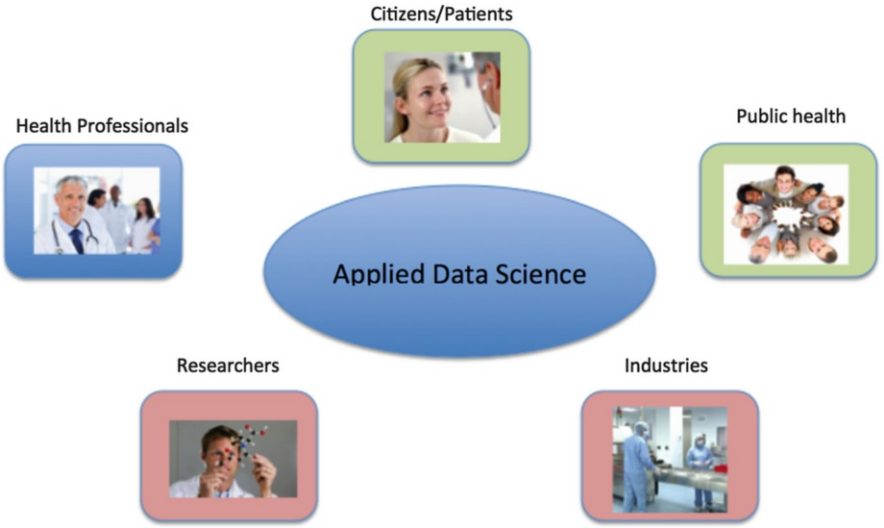

Fig. 22.1 Stakeholders involved in data science in healthcare

conscious people monitor their performance (steps, speed, altitude, heart rate, etc.), nowadays, thanks to the rapid spread of mobile sensors and tracking devices, people can also monitor several types of health data, including but not limited to blood pressure, glycemic index, pain level, and sleep quality.

Harnessing these new possibilities of broad data collection and combining them with (big) data analytics approaches, data science projects aim to generate new knowledge about risk factors, the development of new diagnostic tools, the impacts of new treatments or medications. Citizens and patients will benefit from and become active contributors to this new knowledge.

2.2 Benefits for Health Professionals

Health professionals benefit directly from good, accurate, and continuous health data from their patients. This allows them to take more informed and therefore better decisions, thereby increasing the quality of the treatment. They also benefit from the aforementioned new knowledge and new diagnostic tools and possible treatments that result from data science projects.

2.3 Benefits for the Researchers

Researchers are among the main actors in data science projects, which they mostly lead. They benefit from new ways of recruiting study participants, obtaining additional data and faster access to information, as well as new and more powerful data analytics tools. Furthermore, they benefit from the possibilities to aggregate data from multiple sources.

2.4 Benefits for the Pharmaceutical, Medtech, and IT Industries

Data are key for developing new drugs, devices, IT services as well as for achieving regulatory approval required to bring them on the market. The improved ability to obtain access to the necessary data, to link and aggregate the data from different sources, to gain and manage the consent of the data owners in a defined time frame are important assets.

2.5 Benefits for Public Health

Policy making, in particular in the health and public health domains, has to rely on evidence and data. That is not only true in industrialized countries, but also in low- and middle-income ones. These countries do also suffer from the burden of the rise in chronic diseases. Their scarce resources force them to optimize the actions they plan and can afford to take. Availability of data in sufficient quality allows such optimization.

3 Which Data?

We know that our health depends not only on our genetic constitution and the effectiveness of the treatments that our healthcare system can provide, but also on the environment we live and work in, and on our lifestyle. “Environment” encompasses many aspects including air quality, noise level, traffic intensity, pollen density, weather conditions, etc. For health data science initiatives, data concerning all those determinants are relevant and must be considered in specific combinations depending on the targeted benefits.

Generally, data about our **health status** and the treatments provided are generated mainly by healthcare professionals and their institutions (hospitals, home care, radiology institutes, etc.). They may include vital signs, blood analysis, diagnostics,

medications, radiology imaging, and omics data. These data are managed by a very large set of applications in structured and unstructured format.

In addition to that, **environmental data** are widely available such as weather data (UV light, air pressure, and temperature) and to some extent air quality data (small particles, NOx, etc.), even though much less for noise, exposure to viruses, and other environmental data. Environmental data can be linked to an individual based on geolocation, but geolocation is not always known/collected and environmental data are not available for all localizations with the necessary precision.

The collection of **lifestyle data** has seen a rapid development, starting from the *quantified-self* movement mainly around sports activities and extending to the daily monitoring of physical activity, weight, blood pressure, and respiratory capacity. The emergence of low-cost sensors, the possibility to connect these sensors either to the Internet (through Internet of things, IOT) or to smartphones, and the availability of health (mobile) applications to visualize the values and evolution of these data have accelerated the adoption by the population (Takacs et al. 2014). This results in a large amount of data collected first by the users and then stored in a multitude of cloud services.

One important field that acquires and requires all these kinds of data is **precision medicine**, which aims at refining our understanding of disease onset and progression, treatment response, and health outcomes through the more precise measurement of genetic, treatment, and environmental and behavioral factors that contribute to health and disease (National Research Council (US) Committee on A Framework for Developing a New Taxonomy of Disease 2011). Some forms of precision medicine have already been established, particularly in the field of cancer therapy, where certain drugs are given depending on individual tumor genetics. Including self-collected data on environmental and behavioral factors will foster the extension of individualized medicine (Swan 2012).

One of the challenges in data science projects is the ability to collect the necessary data from various sources (Töpel et al. 2008). The main challenge, however, lies in the correct interpretation of the data. Nowadays, most of the clinical data are represented in medical reports in free text form. Interpretation of these data necessitates either manual coding, which is very resource-consuming and therefore limited, or natural language interpretation through machine learning algorithms that need many well-annotated training data in order to generate good results (Savova et al. 2010; Uzuner et al. 2010). The availability of data in structured form is a large advantage for data science, but it is not sufficient. The format and the semantics used to structure these data are most important. In the healthcare domain, international standards for ontologies and classifications have emerged, like, for instance, *SNOMED* (SNOMED International 2018) for coding medical data or *LOINC* (LOINC 2018) for coding laboratory results. Unfortunately, some standards are not for free use and most of them are not widespread in the current information systems of the healthcare providers.

4 Trust-Promoting Frameworks for Data Science Projects

Data science initiatives are run today in the private and the public sectors with different governance and business models and promote different levels of trust. Both, the business and public sector models, relying on profit and the general public good, leave very little participation space for the citizens, the owners of the personal data. Therefore, these models do not allow a relationship with the citizens that would facilitate the extension of data science projects beyond the initial goal targeted by the company or the research institution. One important mechanism for a long-term commitment of citizens to data sharing is trust (Kaye et al. 2009).

To have citizen that engage with trust and active participation in data science initiatives and to develop a new and fair data economy, the following pillars are important from our experience-based point of view: democratic organization, not-for-profit model, transparency, and data security (Hafen et al. 2014):

- **Democratic organization**: The cooperative form is quite adequate for the entity in charge of collecting and managing personal data. The democratic principle "one member one vote" of cooperatives fits particularly well as every citizen has similar amount of personal data to be managed. In this way, the individual is able to make various sets of personal data (genome, nutrition, geolocation, medical, and fitness data) accessible for novel data services and research projects. The data cooperative as a representation of its users ensures that the data made accessible by the data subjects will only be used for the purpose of the service or data science project and there is a fair financial compensation for data use that goes to the not-for-profit cooperative, and can be used for actions the cooperative decides to be useful, for example, extending the information dissemination regarding scientific results.
- **Not-for-profit**: This model finds its justification in the fact that the economic value of the data is not in the individual data sets but in the whole data set resulting from the aggregation of the data of all persons. Therefore, the value of the data should not belong to a private organization or be paid to individuals but managed in a not-for-profit manner to the benefit of the society, represented by the cooperative. Financial incentives to share personal data should not be distributed. For instance, in the case of blood donation, numerous studies have shown that the quality of the blood is worse when the donor receives a direct payment (Mellström and Johannesson 2008). The large economic value resulting in the aggregation of data under the control of the citizen flows back into the cooperative, and its members can decide how to allocate those resources for the development of new services and for conducting further data science projects in the same not-for-profit framework.
- **Transparency**: The organization of the cooperative with its general assembly of its members is one of the elements of transparency. The cooperative should also nominate an ethics committee, which overlooks and validates the initiatives, projects, and new services that the cooperative targets to operate. On the

information technology side, the software should be open source, so that its quality and purpose can be verified at any time by any person.
- **Data security**: In order to protect privacy and increase trust the latest data security mechanisms, such as advanced encryption technologies, should be used and validated with regular security audits. Having a public-key infrastructure allows the user to have the data encryption key under his or her control. In this way, even database administrators have no access to the data.

The combination of transparent, democratic, not-for-profit governance and secured open-source infrastructure is the basis and condition for the emergence of a new sustainable data economy supporting fair data science projects. Not only traditional services and research projects can be run on such a framework but also innovative services from third parties can be offered.

The MIDATA.coop is an example of a cooperative that is based on these pillars. Its articles of association imply citizens as members, one cooperative share per member, and one vote per share. As any cooperative, it is managed by its board members. To assure that the goals are followed and the means used by the cooperative are aligned with its vision and ethical rules at any time, an ethics committee has the task to review all data science projects before they are allowed to be operated on the framework. In addition, an international and multidisciplinary advisory board has been constituted.

In the MIDATA ecosystem, the IT platform (data storage and access management governed by the member-owned cooperative) is separated from the data applications. This forms a novel type of personal data innovation framework. As data account holders, the citizens can choose from a variety of different data services and research activities in which they decide to participate. Startup companies, corporations, and research groups can offer services in the form of mobile applications that analyze and collect data (e.g. medical, activity, and nutrition data)—data that hitherto have resided in noncompatible data silos. Access authorization always resides with the data owner.

As mentioned earlier and illustrated in Fig. 22.2, the cooperative acts as the fiduciary of the account holders' data. It negotiates with research organizations, data service providers, and pharmaceutical companies the financial conditions for accessing the data that individuals have authorized for these specific use cases. Such use cases may include the recruitment of patients for clinical trials, providing mobile app-based health services or drug side effect reporting (outcomes-based medicine). The revenues generated will be allocated to the maintenance of the platform, security checks, upgrades, and on additional services for the account holders. Members of the cooperative will decide how additional profits will be invested in projects (e.g. research projects) that will profit society. Value is created collectively; there is no pressure on individuals to sell their individual data.

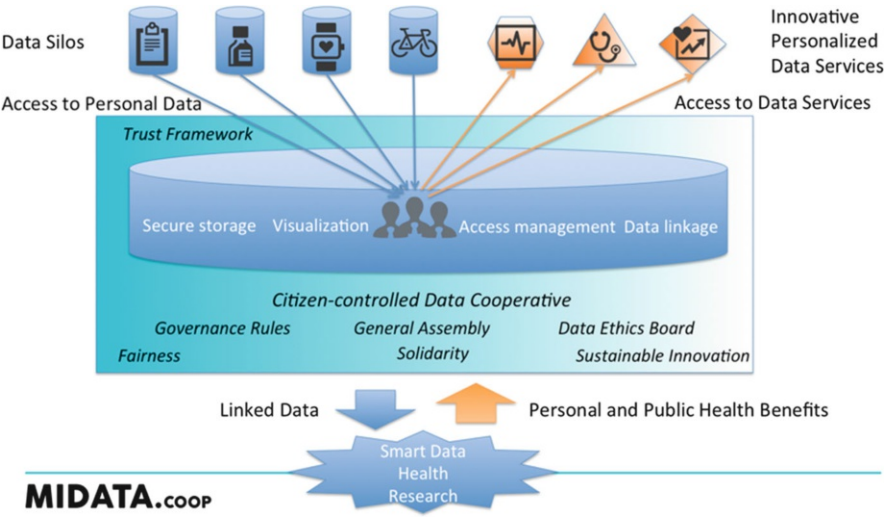

Fig. 22.2 MIDATA.coop governance and ecosystem

5 IT Platform

The ability to make use of a right to a digital copy of personal data requires a suitable and trusted IT infrastructure to securely store, manage, and control access to personal data. Such a trust-promoting framework for active data sharing for personal data services and medical research rests on data security, the individual's control over their data, and a fair as well as transparent governance.

A suitable IT infrastructure that meets these criteria builds on a cloud-based data storage platform. Individuals open their own data account in which individual personal data records are securely stored. Only the data owners control to whom they grant access, to which data sets, and for what purpose.

To illustrate this, Fig. 22.3 shows the component architecture design of the MIDATA IT platform. This architecture has a clear separation of data acquisition, data management, and data analytics.

The data acquisition can be done interactively through mobile applications or a web portal or indirectly by importing data from external data sources. Sensors can either be integrated with mobile applications, which are directly connected to the MIDATA server or communicate with third-party servers, like the platforms provided by large activity tracker vendors.

The interface to the MIDATA server is provided through an application programming interface (API) that complies with the new standard FHIR (FHIR v3.0.1 2018) (Fast Healthcare Interoperability Resources framework). This standard, developed by the standardization organization HL7.org finds a rapidly growing acceptance for the connection of mobile health applications (mHealth) with back-end systems like the secured cloud services provided by MIDATA. The FHIR standard not only

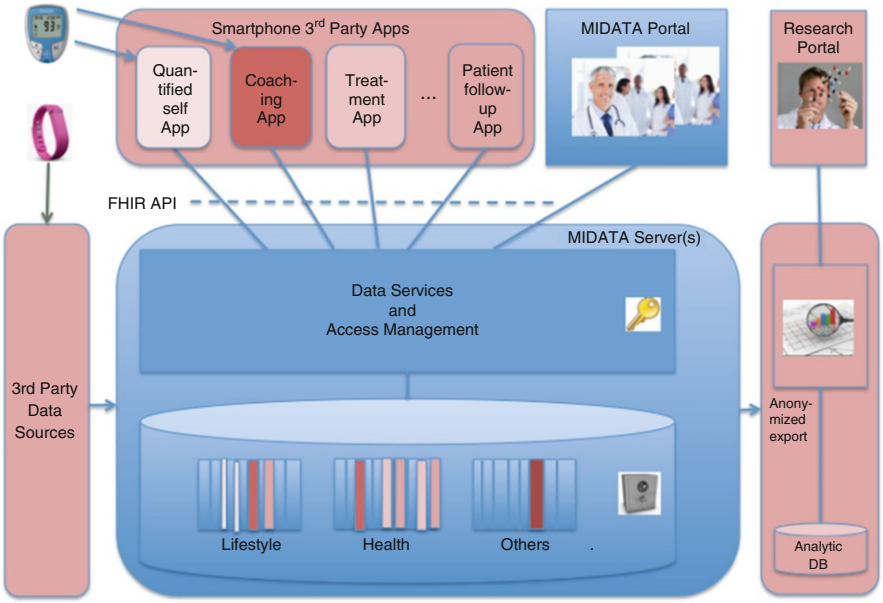

Fig. 22.3 Architecture of the MIDATA IT platform and third-party integration

defines the syntax of the messages exchanged between the mHealth Apps and the server but also the syntax and most importantly the semantics of the information to be stored and managed by the server. FHIR does not define "yet another semantics" but allows to reference well-known and widespread coding systems from the healthcare domain like *LOINC* or *SNOMED*.

The core of the MIDATA IT platform is the data management part. The data management is offered and implemented as "secured cloud services" and interacts, on the one hand, with the different data acquisition channels and, on the other hand, it can export data for analysis in anonymous or nominative form depending on the requirements and on the consent type. The MIDATA server is composed of a first layer, handling the identification and authentication of the users and also handling the access management and the cryptography to protect the data. Past this layer each data element is stored as a single object (in JSON format). Within the JSON object, the syntax and semantic of FHIR is used. These JSON objects are stored in a NoSQL database, thus allowing great flexibility in handling the large diversity of data types that are typically encountered in the health domain.

The data analytics part needs dedicated tools for the researchers who work on the data that the data owners have shared with them and which is therefore implemented in separate components.

The clear separation between mobile applications and data management allows third parties to develop and distribute mobile applications, which can be for-profit. Thus, an ecosystem can emerge where profit can be generated from value-added services for collecting and visualizing data, while the management of the data would

always remain a not-for-profit activity, with citizens retaining control about sharing and using their data. To allow the ecosystem to grow with development done by third-parties, MIDATA focuses on the development, operation, and maintenance of the components painted in blue in Fig. 22.3. Moreover, MIDATA enables third-party actors to develop, operate, and maintain the pink components.

6 Data Protection and Security

Data protection is key to any data science project. Data protection is not only required by legislation but is also mandatory to build and maintain trust with the data owners. Data protection means in particular allowing access to data only to parties that have received explicit informed consent by the data owners. In addition to that, data security implies management of the data based on security aspects, regular verification that security is continually assured, and protection against data loss.

The MIDATA IT platform enforces data protection by:

- Allowing a citizen to register to the platform and become an account owner
- Authenticating each data owner using the platform
- Securely managing the data of each data owner
- Allowing a data owner to share data with another user or with a third party conducting a data science project
- Managing the access to the data of each data owner
- Allowing a data owner to delete his/her data
- Allowing a data owner to withdraw from the platform and have all data optionally exported and then deleted
- Identifying each researcher using the platform
- Managing descriptions provided by researchers of each of their data science project as a basis for receiving explicit informed consent
- Managing the consent of each data owner willing to participate in the data science project and sharing part of his or her data in nominative, coded, or anonymized form
- Allowing each participant to withdraw consent to MIDATA-related aspects of a project

In addition to the services provided by the MIDATA IT platform, additional organizational measures have been taken, such as:

- Identifying users as real persons in order to prohibit fake users
- Managing the register of the researchers using the MIDATA IT platform
- Managing and vetting the MIDATA administrators of the MIDATA IT platform
- Review of the ethical quality of services by a dedicated ethics committee

On the MIDATA IT platform, each data item is stored and managed as a single record. Each record is encrypted with a first key, which is stored with other similar

keys in an access permission set. This access permission set is encrypted with a second key. In a third step, this second key is encrypted with the public key of the data owner.

A data owner willing to access his/her data will use their primary key to decrypt the second key that allows them to decrypt and read the access permission containing the keys to finally decrypt, access, and read the data. All those operations are triggered by the user but executed by the MIDATA IT platform, thus hiding this complexity to the user.

For a data owner giving consent to share data (referenced in one of his or her access permission sets) with a researcher or with another user, the second key that had been used to encrypt that access permission set will then be encrypted with the public key of the researcher or of the other user.

In this way, the researcher or the other user uses his/her primary key to decrypt the second key that allows him/her to decrypt and read the access permission containing the keys to finally decrypt, access, and read the data.

Security audits are run by external independent and recognized security expert organizations on an annual basis. These audits check that no unauthorized access to the platform and the managed data is possible. Some of those intrusion tests are run with no user login available to attempt access to any data; other tests are run with a user login with the intent to access more data than allowed.

7 Example of a Data Science Project Running on MIDATA

The MIDATA IT platform is used for one data science project in which patients with Multiple Sclerosis (MS) are monitored at home. It is a collaboration between MIDATA, University of Applied Sciences Bern and Andreas Lutterotti from the Neurology Clinic at the University Hospital in Zürich. MS is a prototypic autoimmune disease characterized by recurrent areas of inflammation in the brain and spinal cord (Sospedra and Martin 2005). With more than 2 million patients worldwide and more than 10,000 in Switzerland, MS is one of the leading causes of neurological disability in early adulthood. Like many other diseases, MS is a complex disease, where both etiology and disease course are strongly influenced by an interplay of genetic, environmental, and lifestyle factors (Olsson et al. 2017).

The whole project is set to measure both, (1) objective neurological function by implementing different tests for cognition, hand and arm function, as well as gait; (2) patient reported outcome measures using standardized questionnaires or visual analog scales to assess neurologic function, cognition, fatigue, and mood disorders; and (3) gather information on lifestyle and environmental factors that can influence the course of the disease. A smartphone app has been developed that allows continuous monitoring of patients with Multiple Sclerosis (MitrendS App; see Fig. 22.4). This app has two tests: one assesses the speed and correctness in reproducing a symbol-sign assignment, and the other the ability to memorize a path through a graph. It will be used by MS patients in order to monitor their motoric

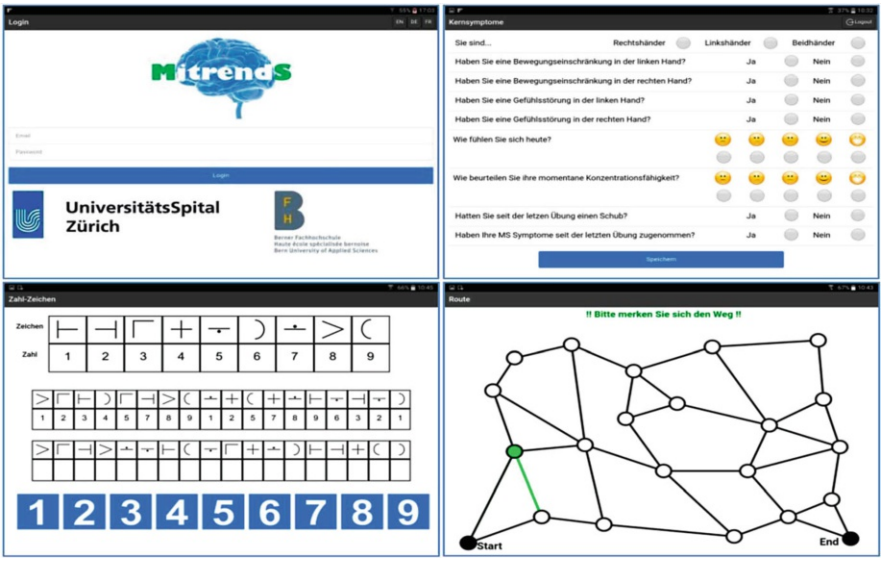

Fig. 22.4 Screenshots of the tests implemented in the MitrendS App

and cognitive capabilities and compare the results of the tests throughout time. Up to now, the app has been successfully tested with healthy persons (results are reproducible) and the recruitment of patients has started.

Tools to predict clinical evolution of MS for individual patients at the time of diagnosis or to identify the most effective treatment are major unmet medical needs for MS patients and their treating physicians. All currently available clinical/imaging measures or biomarkers used for risk assessment perform relatively well at the group level, whereas individual predictions for single patients are not yet available. Concerning the analysis of disease progression, the project intends to study the correlation between external factors such as sun exposure (there is no need for the patients to enter such information with geolocalization) and MS symptoms. Particularly, climatic and environmental variables in combination with lifestyle factors are the most easily accessible external factors that could influence the progression of MS and therefore those are the ones that will be used in the analysis.

8 Conclusion and Lessons Learned

MIDATA.coop is an example for a novel patient engagement effort. Organized as a citizen-owned not-for-profit cooperative, it allows patients and citizens to securely store and manage personal data. This leads to new forms of participation in research (by patient empowerment) besides formal enrolment in clinical trials or

epidemiological studies. In addition to that, much more data can be considered by the switch from institution-controlled data to citizen-controlled data, leading to new opportunities for precision medicine.

The following lessons have been learned since the MIDATA platform has been operational (2016):

- The widespread smartphone use in the population is a major opportunity for the collection of new kind of data and new business models.
- Many citizens are willing to contribute to science, especially when being affected by diseases or having relatives affected.
- Researchers are more and more using real data such as provided by MIDATA because of the need for a better understanding of all determinants of health.
- Not only should the data be at the center of business models, but also applications using such data.
- Structured data are often lacking; hence, the need for more NLP and text mining applications for unstructured data.
- Interoperability is still an issue due to the lack of agreement on semantics. We suggest FHIR as a promising collection of standards in the health domain.
- Data security such as cryptographic technologies is increasingly important when citizens store an increasing amount of data on a platform.
- Transparency concerning the MIDATA platform and its governance structure is highly important, both for those storing data and those using such data.

References

Appelboom, G., LoPresti, M., Reginster, J.-Y., Sander Connolly, E., & Dumont, E. P. L. (2014). The quantified patient: A patient participatory culture. *Current Medical Research and Opinion, 30*(12), 2585–2587. https://doi.org/10.1185/03007995.2014.954032.

Bonney, R., Cooper, C. B., Dickinson, J., Kelling, S., Phillips, T., Rosenberg, K. V., et al. (2009). Citizen science: A developing tool for expanding science knowledge and scientific literacy. *Bioscience, 59*(11), 977–984. https://doi.org/10.1525/bio.2009.59.11.9.

Datenschutzbeauftragte, daschug G., externe. (2018). *Inhalte und Hinweise zur DSGVO/EU-Datenschutz-Grundverordnung*. Retrieved February 26, 2018, from https://www.datenschutz-grundverordnung.eu/

Derder Fathi. (2015). *Recht auf Nutzung der persönlichen Daten. Recht auf Kopie*. Retrieved February 26, 2018, from https://www.parlament.ch/de/ratsbetrieb/suche-curia-vista/geschaeft?AffairId=20154045

Dhar, V. (2013). Data science and prediction. *Communications of the ACM, 56*(12), 64–73. https://doi.org/10.1145/2500499.

FHIR v3.0.1. (2018). Retrieved February 26, 2018, from https://www.hl7.org/fhir/

Hafen, E., Kossmann, D., & Brand, A. (2014). Health data cooperatives - citizen empowerment. *Methods of Information in Medicine, 53*(2), 82–86. https://doi.org/10.3414/ME13-02-0051.

Health Record Banking Alliance. (2018). *HRBA overview*. Retrieved February 26, 2018, from http://www.healthbanking.org/hrba-overview.html

Hürlimann, D., & Zech, H. (2016). Rechte an Daten. *sui generis*. Retrieved from http://sui-generis.ch/article/view/sg.27

Irwin, A. (2001). Constructing the scientific citizen: Science and democracy in the biosciences. *Public Understanding of Science, 10*(1), 1–18. https://doi.org/10.3109/a036852.

Jeff Leek. (2013). *The key word in "Data Science" is not data, it is science*. Retrieved February 26, 2018, from https://simplystatistics.org/2013/12/12/the-key-word-in-data-science-is-not-data-it-is-science/

Kaye, J., Heeney, C., Hawkins, N., de Vries, J., & Boddington, P. (2009). Data sharing in genomics – Re-shaping scientific practice. *Nature Reviews. Genetics, 10*(5), 331–335. https://doi.org/10.1038/nrg2573.

LOINC. (2018). *The freely available standard for identifying health measurements, observations, and documents*. Retrieved February 26, 2018, from https://loinc.org/

Mellström, C., & Johannesson, M. (2008). Crowding out in blood donation: Was Titmuss right? *Journal of the European Economic Association, 6*(4), 845–863. https://doi.org/10.1162/JEEA.2008.6.4.845.

National Research Council (US) Committee on A Framework for Developing a New Taxonomy of Disease. (2011). *Toward precision medicine: Building a knowledge network for biomedical research and a new taxonomy of disease*. Washington, DC: National Academies Press. Retrieved from http://www.ncbi.nlm.nih.gov/books/NBK91503/

Olsson, T., Barcellos, L. F., & Alfredsson, L. (2017). Interactions between genetic, lifestyle and environmental risk factors for multiple sclerosis. *Nature Reviews. Neurology, 13*(1), 25–36. https://doi.org/10.1038/nrneurol.2016.187.

Roehrs, A., da Costa, C. A., Righi, R. d. R., & de Oliveira, K. S. F. (2017). Personal health records: A systematic literature review. *Journal of Medical Internet Research, 19*(1), e13. https://doi.org/10.2196/jmir.5876.

Savova, G. K., Masanz, J. J., Ogren, P. V., Zheng, J., Sohn, S., Kipper-Schuler, K. C., et al. (2010). Mayo clinical text analysis and knowledge extraction system (cTAKES): Architecture, component evaluation and applications. *Journal of the American Medical Informatics Association: JAMIA, 17*(5), 507–513. https://doi.org/10.1136/jamia.2009.001560.

SNOMED International. (2018). Retrieved February 26, 2018, from https://www.snomed.org/

Sospedra, M., & Martin, R. (2005). Immunology of multiple sclerosis. *Annual Review of Immunology, 23*, 683–747. https://doi.org/10.1146/annurev.immunol.23.021704.115707.

Swan, M. (2012). Health 2050: The realization of personalized medicine through crowdsourcing, the quantified self, and the participatory biocitizen. *Journal of Personalized Medicine, 2*(3), 93–118. https://doi.org/10.3390/jpm2030093.

Takacs, J., Pollock, C. L., Guenther, J. R., Bahar, M., Napier, C., & Hunt, M. A. (2014). Validation of the Fitbit one activity monitor device during treadmill walking. *Journal of Science and Medicine in Sport, 17*(5), 496–500. https://doi.org/10.1016/j.jsams.2013.10.241.

Töpel, T., Kormeier, B., Klassen, A., & Hofestädt, R. (2008). BioDWH: A data warehouse kit for life science data integration. *Journal of Integrative Bioinformatics, 5*(2), 49–57. https://doi.org/10.2390/biecoll-jib-2008-93.

Uzuner, O., Solti, I., & Cadag, E. (2010). Extracting medication information from clinical text. *Journal of the American Medical Informatics Association, 17*(5), 514–518. https://doi.org/10.1136/jamia.2010.003947.

Woolley, J. P., McGowan, M. L., Teare, H. J. A., Coathup, V., Fishman, J. R., Settersten, R. A., & Juengst, E. T. (2016). Citizen science or scientific citizenship? Disentangling the uses of public engagement rhetoric in national research initiatives. *BMC Medical Ethics, 17*, 33. https://doi.org/10.1186/s12910-016-0117-1.

Yasnoff, W. A., & Shortliffe, E. H. (2014). Lessons learned from a health record bank start-up. *Methods of Information in Medicine, 53*(2), 66–72. https://doi.org/10.3414/ME13-02-0030.

Chapter 23
Image Analysis at Scale for Finding the Links Between Structure and Biology

Kevin Mader

Abstract Image data is growing at a rapid rate, whether from the continuous uploads on video portals, photo-sharing platforms, new satellites, or even medical data. The volumes have grown from tens of gigabytes to exabytes per year in less than a decade. Deeply embedded inside these datasets is detailed information on fashion trends, natural disasters, agricultural output, or looming health risks. The large majority of statistical analysis and data science is performed on numbers either as individuals or sequences. Images, however, do not neatly fit into the standard paradigms and have resulted in "graveyards" of large stagnant image storage systems completely independent of the other standard information collected. In this chapter, we will introduce the basic concepts of quantitative image analysis and show how such work can be used in the biomedical context to link hereditary information (genomic sequences) to the health or quality of bone. Since inheritance studies are much easier to perform if you are able to control breeding, the studies are performed in mice where in-breeding and cross-breeding are possible. Additionally, mice and humans share a large number of genetic and biomechanical similarities, so many of the results are transferable (Ackert-Bicknell et al. Mouse BMD quantitative trait loci show improved concordance with human genome-wide association loci when recalculated on a new, common mouse genetic map. Journal of Bone and Mineral Research 25(8):1808–1820, 2010).

1 Introduction

Image analysis is a complex process involving many steps (see Fig. 23.1) that are rarely explained sufficiently. Their relevance is continually increasing in both the scientific (microscopy, satellite images, tomography) and commercial (cellphone cameras, YouTube videos, etc.) domains. The processing of such images has drastic

K. Mader (✉)
4Quant Ltd, Zürich, Switzerland

ETH Zurich, Zürich, Switzerland
e-mail: mader@biomed.ee.ethz.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_23

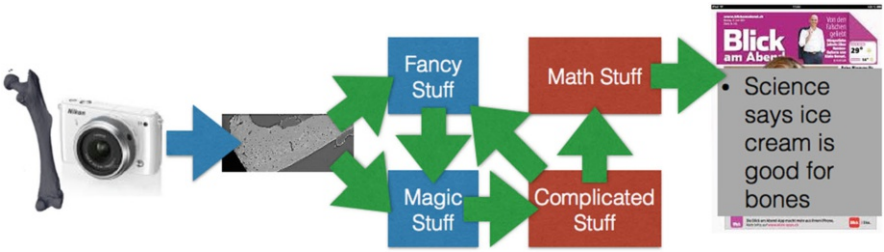

Fig. 23.1 The figure shows the progression from an experiment to a publicized final result and highlights the logical leaps that are taken between the actual measurement and interpretation

consequences on the final results and their reproducibility, accuracy, and statistical significance. Therefore, the steps taken form a critical component of the analysis pipeline.

2 Background: Where Do Images Come From?

Imaging systems follow a standard pattern to create images of objects (see Fig. 23.2). The initial stage is the impulse or excitation (some modalities like bioluminescence have no impulse but most do). The impulse interacts with the characteristic of interest in the sample to produce a response. Finally, there is a detection system, which records this signal to form the image. This flow is quite abstract and to make it more concrete, there are several common imaging modalities listed below with typical results (see Table 23.1).

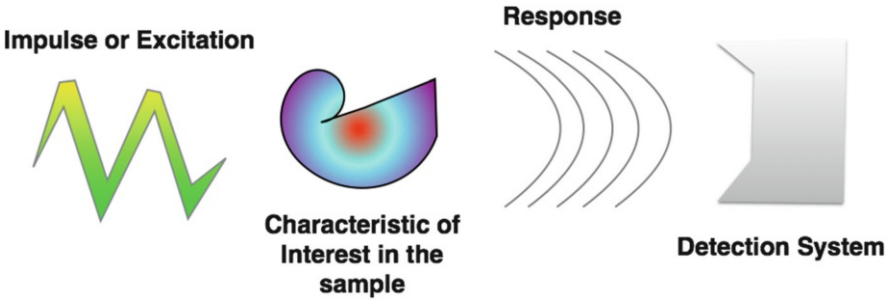

Fig. 23.2 The creation of images shown as a generic pathway from an original impulse to the sample and then a response that is measured by the detection system

Table 23.1 Image creation process for several different standard modalities and how they fit in the model shown in Fig. 23.2

Modality	Impulse	Characteristic	Response	Detection by
Light microscopy	White light	Electronic interactions	Absorption	Film, camera
Phase contrast	Coherent light	Electron density (index of refraction)	Phase shift	Phase stepping, holography, Zernike
Confocal microscopy	Laser light	Electronic transition in fluorescence molecule	Absorption and reemission	Pinhole in focal plane and scanning detection
X-ray radiography	X-ray light	Photo effect and Compton scattering	Absorption and scattering	Scintillator, microscope, camera
Ultrasound	High-frequency sound waves	Molecular mobility	Reflection and scattering	Transducer
MRI	Radio-frequency EM	Unmatched hydrogen spins	Absorption and reemission	RF coils to detect
Atomic force microscopy	Sharp point	Surface contact	Contact, repulsion	Deflection of a tiny mirror

3 How Is an Image Represented?

An image is a pairing between spatial information (position) and some other type of information (value). Typically, an image is represented as a grid or array of such pairings as shown in Fig. 23.3. The values are often represented as colors, intensities, or transparencies to make visualizing the data easier.

4 Use Case: How to Look at Femur Fracture?

Fractures in the femur neck are one of the most debilitating diseases when surgically treated in the elderly. They have a mortality rate of 1 in 4 within a year, and require (for nearly 1 in 2 cases) additional surgery within 2 years. The current way for assessing risk is the standard clinical measurement of bone mineral density (BMD), which quantifies the amount of bone present (Blomfeldt et al. 2005). While such metrics provide some insight into the risk profile, only a small proportion of such fractures can be explained or predicted by this metric. To understand the risk better, more specific information is needed about not just the quantity of the BMD but also the quality.

Imaging can be used to see more detailed views of the bone and assess quality through structure and organization. As bones are not transparent, standard optical

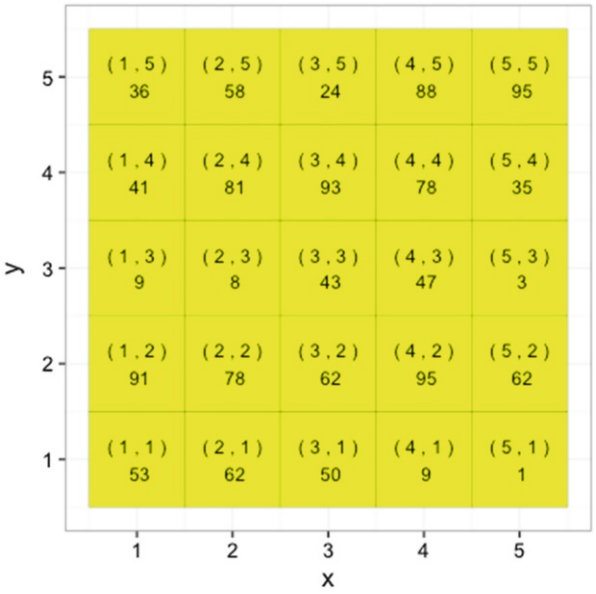

Fig. 23.3 A simple representation of an image where each box has the spatial position shown in the upper half in parentheses and the value shown in the lower half as a number

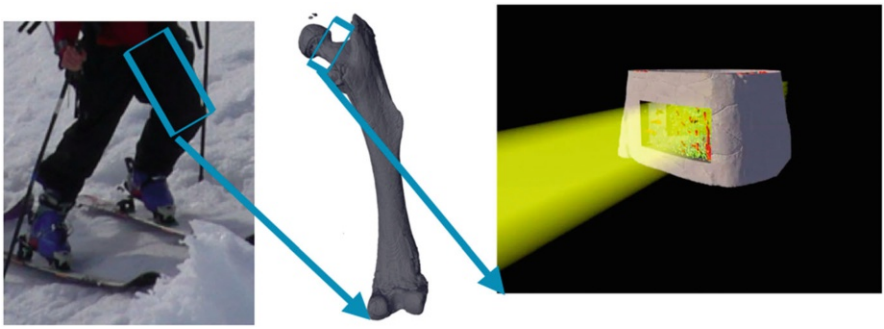

Fig. 23.4 The femur neck shown on a multiscale graphic from body level (left) to cellular level (right)

microscopy approaches work poorly and X-ray based methods are preferred for looking at high-resolution structural differences.

As quality itself could be difficult to quantify in a clinical setting (expensive high-resolution scans with prohibitively high doses of X-ray), we aim to determine the heritable attributes as these can be easily and cheaply examined with low-cost genetic tests (Jansen and Stam 1994). Here we show a study looking at the multiple-scales (see Fig. 23.4) involved in femur fracture and a brief analysis of how the genetic basis of it can be assessed.

5 Study Design

In this section, we address the design of a study to evaluate this idea. The tools of genetics, image analysis, and statistics can then be used to start to break down a complicated process like fracture risk into quantitative assessment of bone structure and specific regions of the genome.

Genomics is the study of genes and their influence on a biological organism. Like many systems in biology, the relationship between genes, anatomy, and physiology is complicated and multifaceted. Genomics has undergone a series of major breakthroughs made possible by improved sequencing techniques (Reuter et al. 2015). It is now possible to quickly and reliably sequence the full genetic makeup of individuals. Some insight can be gained by analyzing the sequences themselves, and a number of Big Data projects are tackling this exact problem (Reuter et al. 2015). However, the insight gained from such analyses is limited without a concrete connection to biology. There are several reasons for the large gaps between genetic makeup and biology: many of the genes are not used or expressed, internal cellular factors can regulate the behavior of genes independently up and down, and many other issues grouped coarsely into the field of epigenetics (Jansen and Stam 1994).

Phenomics is the study of specific visible, expressed traits and their relationship to the genetic makeup. Rather than focusing on genetic material, it takes the opposite approach and focuses on visible, physical traits and works backward to the relevant factors (Jansen and Stam 1994). The easiest approach to studying Phenomics involves making changes to single regions of the genome and seeing what effect this has on the animal (Jansen and Stam 1994). This approach is normally impractical since there are so many different regions that can be changed and so many different effects a single gene can have.

5.1 Image Acquisition

Given the opacity of bones and importance of three-dimensional (3D) structure, visible light is poorly suited for imaging. We therefore measure the femur bones using X-ray tomography (Feldkamp et al. 1989), a technique that uses high-energy X-rays to penetrate entirely through the sample, creating a shadow (radiograph) of low and highly X-ray absorbing regions. From these shadows, 3D volumes can be reconstructed using computational techniques like filtered back-projection (Feldkamp et al. 1989). Structure is an inherently complex idea and does not immediately lend itself to quantification. Structure is a very visual characteristic typically described using qualitative terms like smooth, round, or thick applied to regions of an image. Such assessments, while valuable for gathering insight into complicated samples, cannot be reproducibly applied across the thousands of measurements used in a study. In order to perform reproducible studies, we need a clearly defined process for extracting and summarizing values from the sample. For

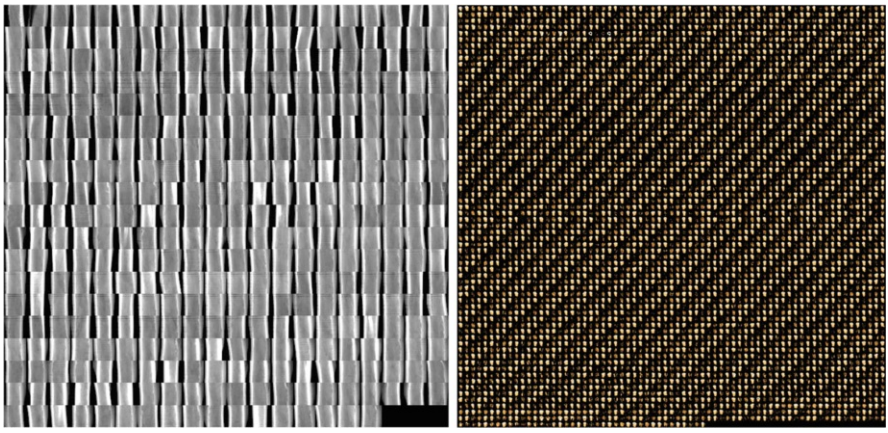

Fig. 23.5 The left panel shows a display of a subset of 377 randomly selected raw tomographic datasets of complete bones. The right panel shows 1300 samples with full 3D renderings based on the known material properties of bone. The graphic illustrates the scale data needed and the requirement for consistency in measurement

each extracted value, a single metric must be defined that measures a given characteristic of the sample. For standard numeric measurements like height and weight, this is an easy task, which involves recording a number and can be easily summarized by examining the distribution, means, and moments. For images this presents a problem, since they are not single metrics nor are they trivial to summarize into such a number (the average image is usually neither meaningful nor easily to calculate).

The images of a subset of the data are shown in Fig. 23.5. Each image of a femur contains over 8 billion voxels (a 3D pixel) of data detailing the X-ray absorption (Fig. 23.5, left panel) in an approximately 2 mm $\times$ 2 mm $\times$ 2 mm cube. Since the femur bone samples are primarily made from three very different materials: calcified bone, cellular soft-tissue, and air, we can apply a threshold approach for classifying each voxel. We thus classify each voxel using upper and lower cutoffs for the expected absorption coefficient of given material (Mader et al. 2013). The classification of the image into different phases results in a number of subimages created for each (one for bone, one for cells, and one for vessels). With each voxel classified into one of these categories, the data has been strongly reduced but there is still far too much information for any further analysis (Fig. 23.5, right panel). To quickly review, we have converted grayscale images with intensity representing physical properties of the sample into binary images representing the class each of those pixels is most likely to be in. The binary images still represent too much data to analyze as they just contain the x, y, z coordinates for bone, cell, and vessels and do not yet quantify any anatomical features in a meaningful way.

5.2 *Image Analysis*

For this study, we focus on the structure and morphology of the bone and its subcellular structures. This means we are interested in how the groups of voxels are organized with respect to each other to form higher structures with anatomical significance like cortical bone, vascular structures, and cells. Several different approaches exist for segmenting and quantifying these structures (Schwarz and Exner 1983) and these tools can be easily combined to create a series of biologically relevant metrics. We thus use a list of established, validated metrics (Mader et al. 2013) to condense the 3D image data into single metrics such as average bone thickness, cell count, cell alignment, and cellular density. From a high-level perspective, the problem seems to be solved; however, the analysis is an interesting case study for scalable image analysis because the size of the images is very large (8 billion voxels per image, i.e., the size of each image is 16 GB) and there are many images in the study (>1000), resulting in a total size of about 16 TB. The large size of each sample is a problem that is ill-suited for database or standard statistics tools, while the large number of samples makes it poorly suited for one-off 3D visualization and analysis tools that are frequently interactive and offer limited "batch-analysis" functionality.

The situation is thus primed for a new paradigm of tools, commonly referred to as Big Data, which offer the raw computational power of high-performance computing paired with a design that is suited to large number of fault-tolerant, reproducible analyses (Mader and Stampanoni 2016). The basic approach involves applying the MapReduce paradigm (Dean and Ghemawat 2008) to break up the entire analysis into small, independent components (map-steps and reduce-steps). These independent components are then run on one or more machines in a parallel, distributed manner.

The most-common "Hello World" example for Big Data Processing is counting words in a number of different documents (Dean and Ghemawat 2008). For MapReduce, the task is divided into map steps that breakdown every document into a bunch of individual words and the reduce step, which groups by word and then counts the occurrences. For images to benefit from such a degree of parallelism, they need to be subdivided into smaller blocks. These blocks can thus be independently processed and brought into the final result.

For a simple operation like estimating the volume fraction of the different phases in bone, a pipeline for the necessary tasks is shown in Fig. 23.6. This is the image analysis equivalent of a word-count where, instead of unique words, the idea is to count the number of pixels in two different phases in the image. We will now explain these steps in more detail.

- The first two steps of the pipeline are IO-related: dividing the image into blocks and loading the data for each block in.
- The third step is applying the thresholding function (*threshold_image*) to classify the pixels into bone and air. For the example of bone, this function would be to take the gray value of the image and threshold it above and below 600 Hounsfield

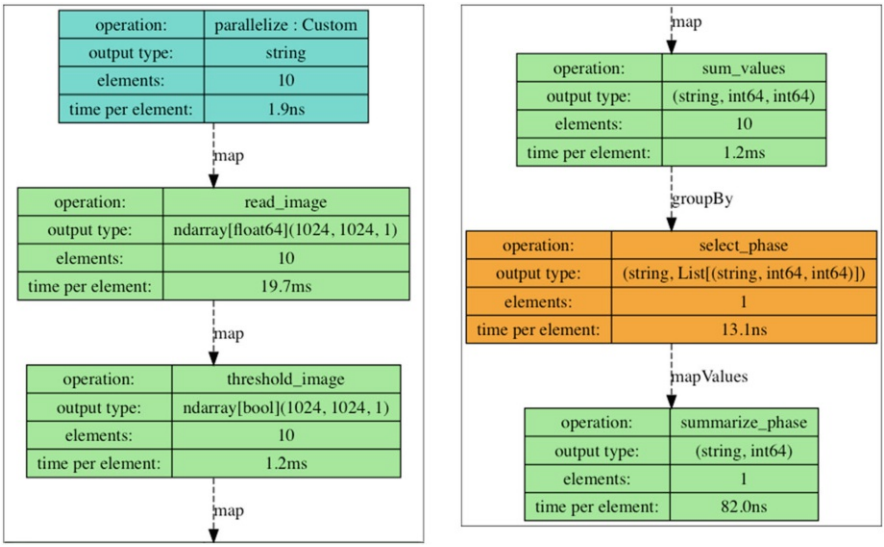

Fig. 23.6 A display of the directed acyclic graph (DAG) of the MapReduce-style workflow behind the volume-fraction analysis. The types here are pseudo-code based on Python where ndarray is an n-dimensional array with the dimensionality for the given problem listed afterward

units, the standard absorption value for calcified tissues (Mader et al. 2013). Everything above 600 would be considered bone and everything below would be soft-tissues, air, and water.

- The fourth step is taking the bone and air post-threshold images and counting the number of pixels for each block.
- The last two steps take these labels (bone, air) and pixel counts and group them by the label name and aggregate the pixel counts by adding the values together.

The workflow covers data loading, thresholding, summation, and summarization. The level of parallelism until the very last step is equal to the number of blocks being processed (ten in the example) and thus can be executed on a number of machines independently – what is commonly called embarrassingly parallel (Dean and Ghemawat 2008). We show the performance of the pipeline shown in Fig. 23.6 in Fig. 23.7. The graph shows how well a small task can be distributed among multiple cores and multiple machines. The graph is not meant to be perfectly quantitative or entirely reproducible (cluster-load, network traffic, core utilization, garbage collection, and dozens of other factors make true benchmarking very difficult), but the results represent averages over ten runs. The image sizes were varied from 30 MB to 463 GB on a cluster of machines using from 1 to 360 cores. A key factor for performance is the number of blocks used. This is important because the blocks make up the units that are processed and have a cost of being transferred across the network and serialized on disk. Therefore, their size must be matched to the network

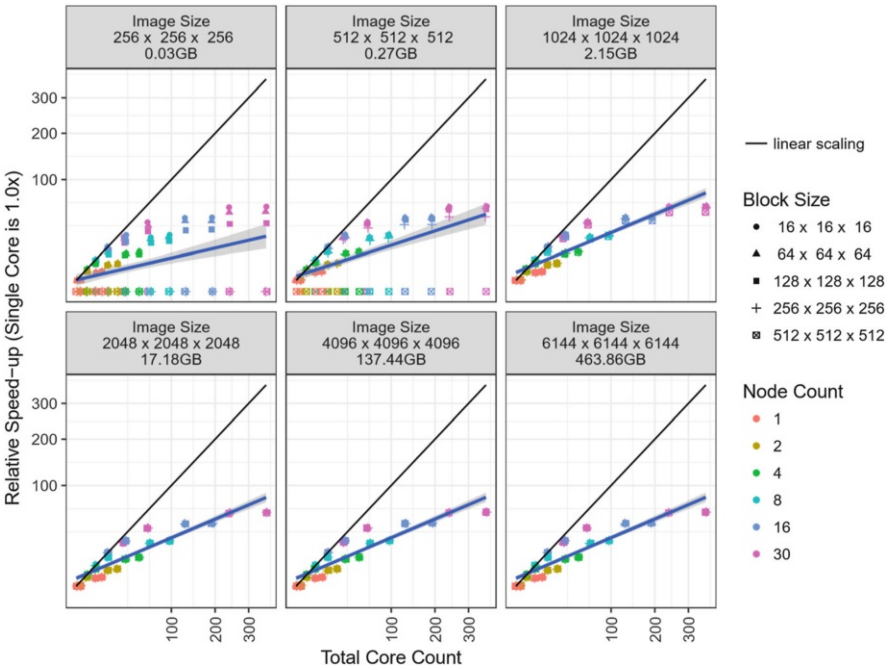

Fig. 23.7 A panel of the figure showing the scaling performance using MapReduce pipelines on 3D images. The title shows the image size, the *x*-axis shows the total core count, and the *y*-axis shows the speed-up. Different colors represent the number of nodes used and different point shapes show different block sizes for computation. The black-line shows perfect linear scaling

performance, disk performance, memory on each machine, and the complexity of the operations performed (Mader and Stampanoni 2016).

The last step of aggregating all of the results together cannot be done as efficiently as it involves communication between nodes. Fortunately, the size of the data at this step is in this case (and many others) much smaller (Mader and Stampanoni 2016).

While the workflow is very similar to the MapReduce word-count example, it is very different from standard image analysis pipelines done in tools on high-performance cluster computers using OpenMPI (Almeer 2012). These standard OpenMPI workflows typically operate on images as large contiguous memory blocks in shared-memory on one machine that multiple threads are able to scan through in parallel. While the standard workflows do provide high performance on datasets up to tens of gigabytes in size, they struggle to transition to a large distributed and larger than memory datasets required for the processing of terabytes of data (Almeer 2012).

5.3 Genetic Cross-Studies

Once the image data have been processed, it is possible to begin to link the results to the biology. Since each of the samples measured and analyzed above has a slightly different genetic makeup, we want to link the genetic makeup to its manifestation in structural properties. In an ideal world, rather than starting with a random, diverse group, we would take identical specimens and make single point mutations to each one and measure the effect. Unfortunately, due to the complexity of such mutations and the sheer number that would need to be made (>50,000 genes in a mouse), we make use of cross-studies. Cross-studies look at a huge number of different regions of the genome at the same time. The standard process involves taking two pure strains of mice (inbred over many generations) that are homozygous (the same genes from both parents) at every position of the genome. As shown in Fig. 23.8, we take two different strains of mice with different known bone mechanical properties. The group on the left has a low bone mass with thin bones (blue), and the group on the right (red) has high bone mass with correspondingly thicker bones. We choose these starting points because we expect to find a number of differences explaining which genes are responsible for the larger bone-mass bones. A single cross (F1) results in a population of identical heterozygous (different genes from each parent) mice. A second cross (F2) results in a population of mixed mice with genetic regions coming from the first strain (shown in blue in Fig. 23.8) and the second strain (shown in red). These mice can then be tagged at a number of different regions throughout the genome to identify which strain that region came from (homozygous blue, homozygous red, or heterozygous) (Silver 1995).

5.4 Messy Data

As is common in biology and medicine, the measured data from such a study is very messy. The confidence of measurements is low, and in many cases so low as to be treated as a missing value. An overview of the genetic data collected in this study is shown in Fig. 23.9 and the white regions represent missing data. Each row represents an individual mouse and each column is a position along the genome (marker). The markers are regions of the genome that can be easily, uniquely identified and thus mapped for large genome-scale studies. The markers correspond to a sampling of the genome information, since the number of base-pairs would be 2.7 billion (Silver 1995). For each marker, the color shows in the figure from where the genetic sequence at that position for that mouse came.

Our next goal is to compare the values at each position to the computed metrics on the X-ray image data to find out which positions in the genome were most responsible for changing the computed metrics, ultimately giving us more insight into fracture risk. Specifically, these metrics need to be correlated on a sample-by-sample basis to positions. This task is further complicated by missing

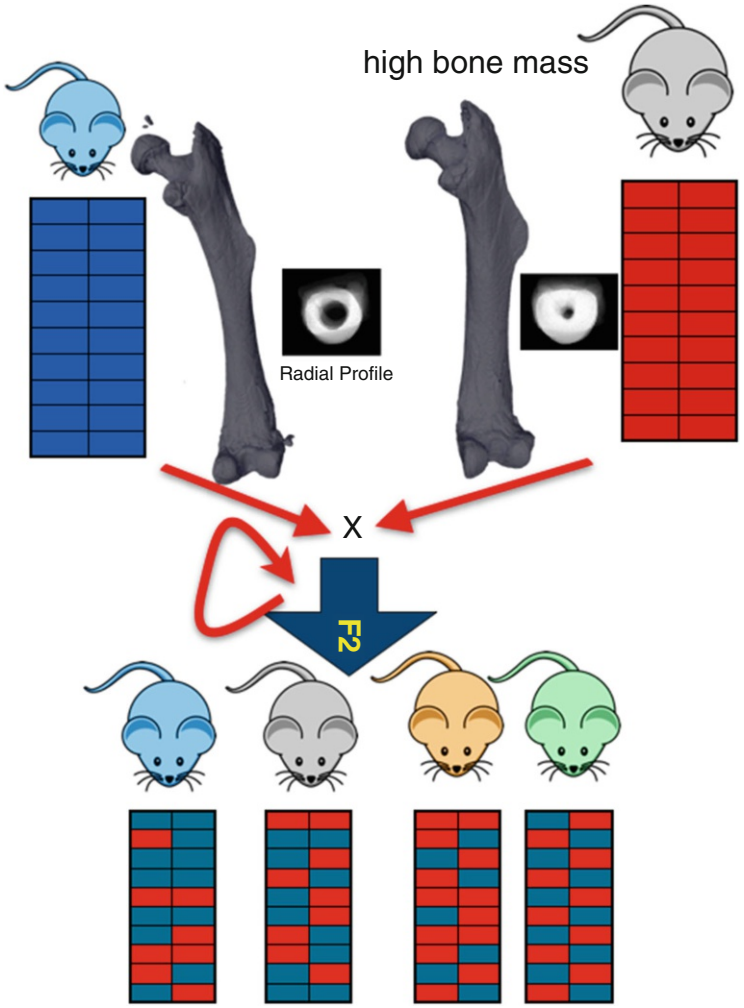

Fig. 23.8 The progression from parental strains (above) to the F2 cross-strain (below). The two strains are low bone mass animals (left) and high bone mass animals (right). The figure also shows average radial profile for each strain

and possibly erroneous genetic markers. Furthermore, while 1300 samples are a large number for an imaging study, it is very small compared to the number of variables (90 markers) and the resulting degrees of freedom. Using classical statistical methods like student's *t*-test and ANOVA are poorly suited for such problems and would be unlikely to deliver significant or meaningful results. They are poorly suited for two keys reasons: assumed independence of markers and multiple testing correction. The basic models of student's *t*-test assume each test that is performed is independent from the previous; in genetic systems, we know the marker distribution

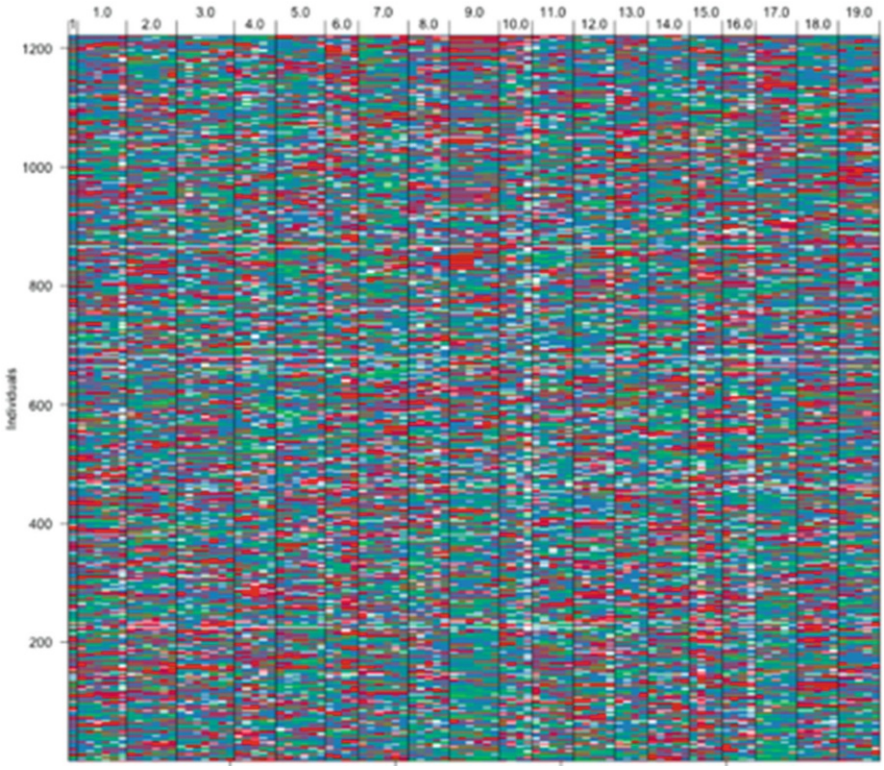

Fig. 23.9 The genetic tagging for each animal is shown. The vertical axis shows each animal as a row and the horizontal axis shows the markers and their position on the genome (chromosome number). The color is either homozygous B6, homozygous C3H, heterozygous, or unknown

is heavily interdependent. Correcting for multiple tests would massively increase (90 different markers to test) the difference required for statistical significance and many possible meaningful results would be excluded.

6 Statistical Methods

In this section, we cover the set of statistical analysis tools that can be applied to large-scale genomics studies. The goal of these tools is to determine which regions of the genome are responsible for which structural properties of the bone structure. From this we can begin to derive the regions that increase or decrease fracture risk. As the previously discussed standard tools like student's *t*-test and ANOVA are poorly suited to these problems, we introduce the approach of Quantitative Trait Loci Analysis (QTL), imputation, and bootstrapping.

6.1 Quantitative Trait Loci Analysis (QTL)

Since we know a great deal about the underlying biological processes, we can take a step beyond looking for single correlations or linear models. Specifically, genetic ideas like inheritance, dominance, and gene–gene interactions are not easily exposed from a correlation coefficient. We use the standard phenotype model [see Eq. (23.1)] to model these effects (Jansen and Stam 1994). Our goal is to perform a curve-fitting of the measured data to this model to estimate the parameters (a, b, m, μ).

Equation 23.1 The basic model for fitting the value of a phenotype to a series of genetic loci (Jansen and Stam 1994). Interactions are not included in this study since their contribution is typically much smaller than the primary additive and dominance effects.

$$y = \sum_{i} (a_i x_i + b_i [x_i \leq 0]) + \sum_{i,j} (m_{i,j} x_i x_j) + \mu$$

Phenotype; Additive Effect; Genotype at Loci i; Dominance Effect; Interaction; Genotype at Loci i; Genotype at Loci j

While the intricacies of the model are best examined in other texts (Jansen and Stam 1994), we can intuitively come to a better understanding of the model by examining the ideas of inheritance and dominance. We define inheritance as the similarity between parent and offspring strains. We then quantify this by looking at measurable traits (phenotypes). To model this, we represent genotype (x) as an indicator variable (AA is −1, AB is 0, and BB is +1). The most basic model for inheritance is an additive effect (see Fig. 23.10 middle panel). These are the product of the effect size (a) and the genotype (x). We can then incorporate dominance effects by including a nonlinear term (see Fig. 23.10 bottom panel). For this we define the dominance effect size (b) by taking the difference of the mean of AB from the average of the means of AA and BB (if this is zero, there is no dominance component).

These models are then initially fit to the phenotype for the additive and dominance term (interaction effects are usually much smaller). The output of a QTL Analysis is a logarithm of the odds (LOD) score for each phenotype at each measured marker. This score indicates the likelihood that this region is important for explaining the variance in the population group. This score itself is a relative measure and only establishes a given region's likelihood of involvement compared to other regions measured (Jansen and Stam 1994). Being a logarithmically scaled number, small differences in value correspond to orders of magnitude differences in the corresponding likelihoods.

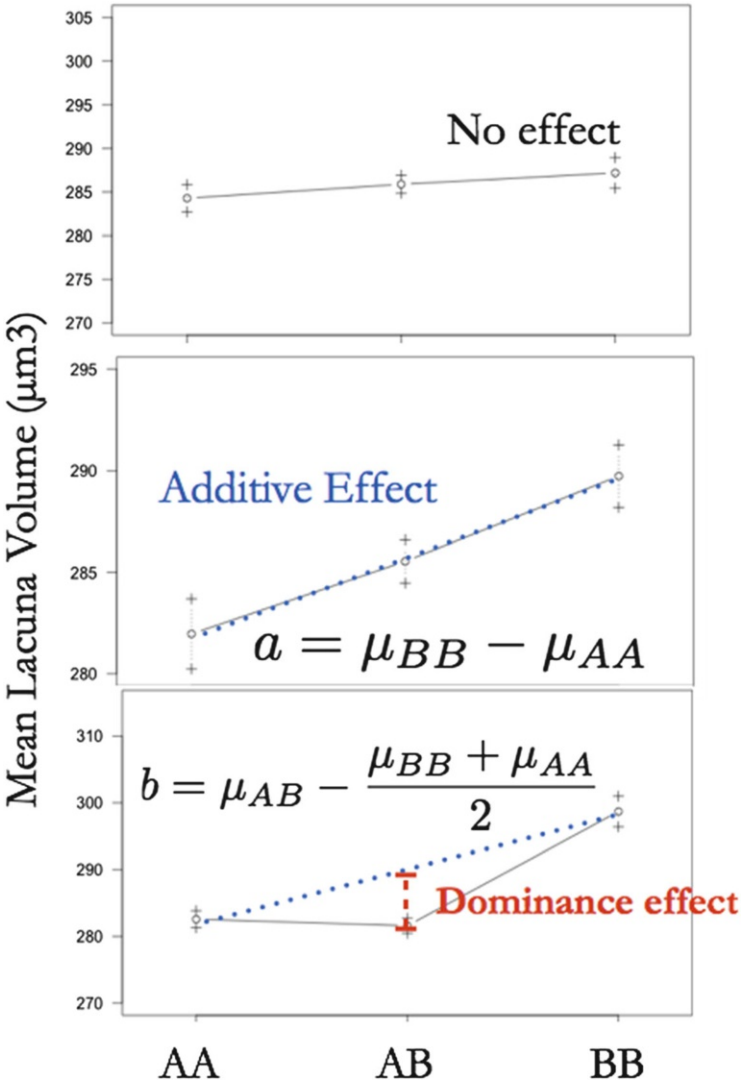

Fig. 23.10 The details of the basic additive-dominance modeling for three different trait loci (panels) compared to the single phenotype of Mean Lacuna Volume (*y*-axis) versus different genotypes (*x*-axis). The AA indicates both came from the A strain, AB is a homozygous combination of the A and B strains, and BB indicates both coming from the B strain

6.2 Imputing

The strategy used for filling the holes in the missing data is called imputing. For completely unknown sequences this would mean including every combination of possible values at every missing point. Since the process of chromosome pairing is

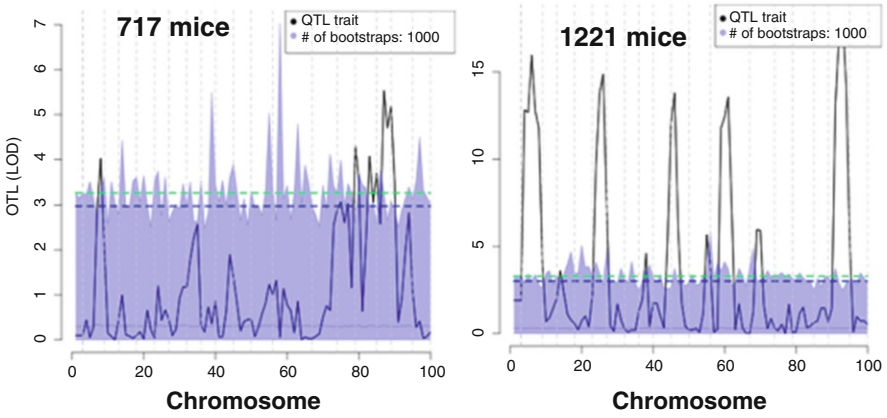

Fig. 23.11 The LOD scores and bootstrapping established baseline for determining significance for 717 and 1221 animals, respectively

better understood and shows several correlations, the data can be more intelligently interpolated at the missing points. The data is thus augmented based on these correlations to a much larger set with no missing points so it can be further processed (Broman and Sen 2009).

6.3 Bootstrapping

Bootstrapping is a commonly used statistical approach that involves reusing existing data in different ways (Broman and Sen 2009). As stated above, the dimension of the data and variable space is too large that simple *t*-tests or ANOVA analyses would be unlikely to show any statistically significant results. For this we use a bootstrapping approach to determine which effects are significant.

We rerun the QTL analysis above many times (in this case 1000) on the original dataset, where we permute the phenotype values and keep the genotype information the same. As the inputs no longer correspond to the outputs, none of the results are expected to be relevant (Silver 1995).

For each of these analyses an average LOD curve is produced. The real analysis is compared to this baseline. The significance of a given peak can thus be assessed by the amount it lies above the baseline curve. Figure 23.11 also shows how important it is to have large numbers of samples (mice) in the study. The difference between 717 and 1221 mice is the difference between finding 1 and 5 significant regions on the genome (Mader et al. 2015). Finding a region here means the black curve (the actual results) is above the randomly occurring background (blue). Specifically finding more regions that are strongly differentiated from random chance enables us to have more potential targets to identify causes for fracture risk.

7 Results/Evaluation

The final results of this analysis are models explaining a large percentage of the changes observed in a number of phenotypes. These models can be visually represented as several different regions on a map of the genome. Given the lack of precision of the markers, missing data, and other forms of noise, the regions come with a confidence interval. Figure 23.12 shows the 95% confidence interval for the results. The interesting aspects are the overlaps between the newly identified metrics such as number of cells (Lc.Dn) and cell volume (Lc.V) and known properties like bone mineral density (BMD) or mechanical traits. The most interesting points are the loci identified that do not overlap with BMD but do overlap with mechanical properties as these are regions responsible for bone quality (Mader et al. 2015).

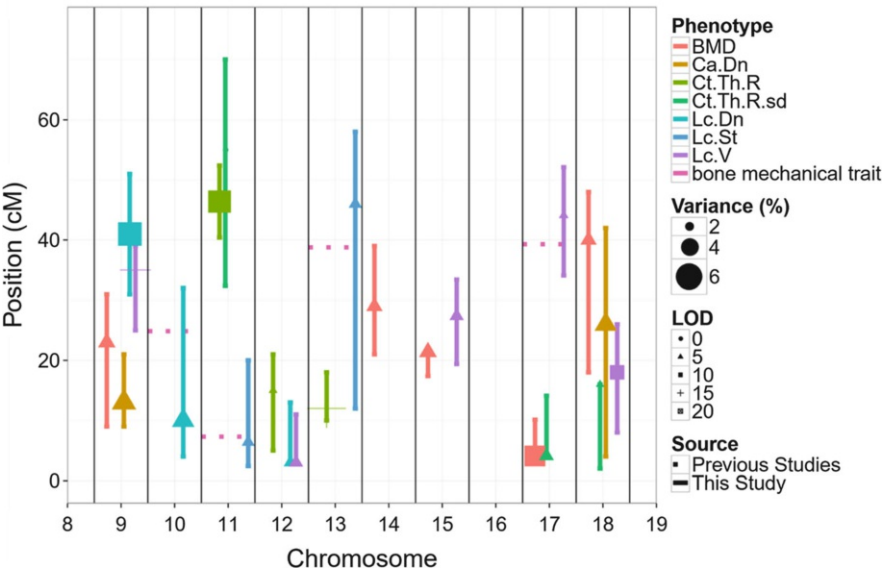

Fig. 23.12 The regions identified in QTL search shown by phenotype (color) and chromosome (*x*-axis) position (*y*-axis). The solid lines vertical represent the findings in this study and the dashed horizontal lines show the results from previous studies (Mader et al. 2015). The vertical lines reflect the 95% confidence intervals from the model output. The glyph represents the most likely position in this band. Finally, the size of the glyph represents the contribution of this region to the overall phenotype value

8 Conclusions

The study shows the required process for taking a large challenging problem like fracture risk and using the tools of image analysis, big data, and statistics to come up with new insights. The most critical factor for such studies is the processing of large amounts of high-quality data. The imaging (data acquisition) and genetic tagging form just a small piece of the entire pipeline. The most critical steps are the proper storage and analysis of the image data. For this study, the development and validation of meaningful image analysis steps took 24 times as long as the measurements (data acquisition) themselves (Mader and Stampanoni 2016). We categorize here post-processing tasks as all of the tasks that take place after image acquisition and storage. The post-processing tasks consist of segmentation, quantification, and statistical analysis. For future studies, more effort should be made to scale and simplify the post-processing aspects of image analysis. Figure 23.13 shows the relative breakdown of imaging studies at a large scientific research facility (the TOMCAT Beamline at the Paul Scherrer Institut). We can see that the majority of the time is spent on the post-processing component and that the measurements themselves will be an almost insignificant portion of the entire pipeline (Mader and Stampanoni 2016).

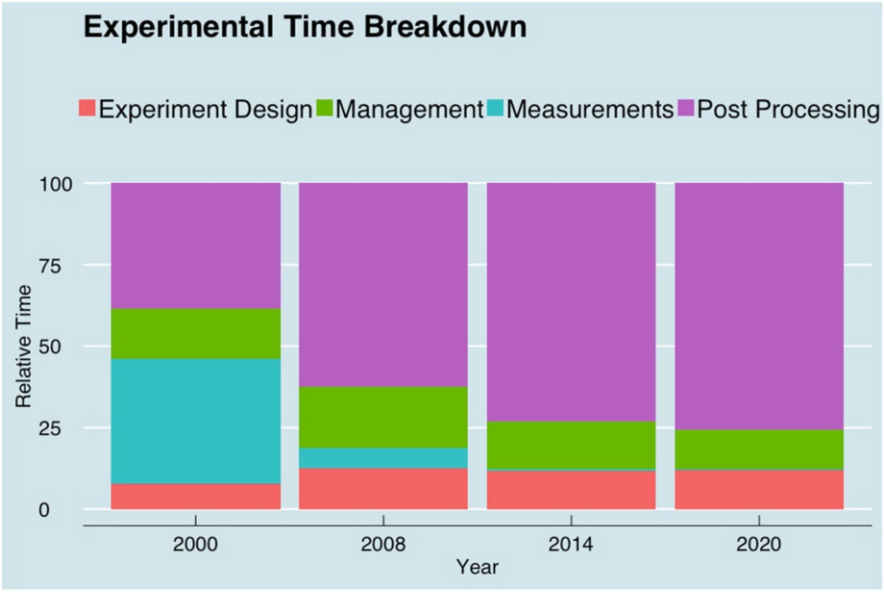

Fig. 23.13 The figure shows the change in time-distribution for experiments (as observed at the TOMCAT Beamline of the Swiss Light Source and linearly extrapolated beyond 2014). In 2000, the division between the four phases was relatively even. As acquisition has become quicker, the measurement phase has shrunk drastically and the post-processing component has grown to dominate the time spent

9 Lessons Learned

1. In order to correlate image data to other data-types, quantitative metrics need to be extracted from them. For this study it was genomic data, but the lesson equally applies to satellite images and weather, YouTube videos and click-through rates, and many other topics.
2. Image data are well-suited for qualitative analysis but require significant processing to be used in quantitative studies.
3. Given the wide variety of preprocessing steps and the effect they have on the results, it is important to have a reproducible way to run analysis quickly and efficiently.
4. Representing long, time-consuming computation in a simple, declarative manner (like Map-Reduce) allows for you to focus on the data science rather than the engineering problem. In particular, since many of the tools and platforms of data science are constantly improving, a less rigid, implementation-focused approach makes transitioning to newer tools easier.
5. Simple *t*-tests are poorly suited for studies with large number of variables and samples.
6. Bad or missing data should be avoided, but by utilizing the tools of imputation, bootstrapping, and incorporating known distributions, the problem can be dealt with much better than removing all samples with missing data points.

References

Ackert-Bicknell, C. L., Karasik, D., Li, Q., Smith, R. V., Hsu, Y.-H., Churchill, G. A., et al. (2010). Mouse BMD quantitative trait loci show improved concordance with human genome-wide association loci when recalculated on a new, common mouse genetic map. *Journal of Bone and Mineral Research: the Official Journal of the American Society for Bone and Mineral Research, 25*(8), 1808–1820. https://doi.org/10.1002/jbmr.72.

Almeer, M. H. (2012). Cloud hadoop map reduce for remote sensing image analysis. *Journal of Emerging Trends in Computing and Information Sciences, 3*(4), 637–644.

Blomfeldt, R., Törnkvist, H., Ponzer, S., Söderqvist, A., & Tidermark, J. (2005). Internal fixation versus hemiarthroplasty for displaced fractures of the femoral neck in elderly patients with severe cognitive impairment. *Journal of Bone and Joint Surgery (British), 87*(4), 523–529. https://doi.org/10.1302/0301-620X.87B4.15764.

Broman, K. W., & Sen, S. (2009). *A guide to QTL mapping with R/qtl* (Statistics for Biology and Health) (412 p). Berlin: Springer.

Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM, 51*(1), 107. https://doi.org/10.1145/1327452.1327492.

Feldkamp, L. A., Goldstein, S. A., Parfitt, M. A., Jesion, G., & Kleerekoper, M. (1989). The direct examination of three-dimensional bone architecture in vitro by computed tomography. *Journal of Bone and Mineral Research, 4*, 3–11. https://doi.org/10.1002/jbmr.5650040103.

Jansen, R. C., & Stam, P. (1994). High resolution of quantitative traits into multiple loci via interval mapping. *Genetics, 136*(4), 1447–1455.

Mader, K., & Stampanoni, M. (2016, January). Moving image analysis to the cloud: A case study with a genome-scale tomographic study. In *AIP Conference Proceedings* (Vol. 1696, No. 1, p. 020045). AIP Publishing.

Mader, K. S., Schneider, P., Müller, R., & Stampanoni, M. (2013). A quantitative framework for the 3D characterization of the osteocyte lacunar system. *Bone, 57*(1), 142–154. https://doi.org/10.1016/j.bone.2013.06.026.

Mader, K. S., Donahue, L. R., Müller, R., et al. (2015). High-throughput phenotyping and genetic linkage of cortical bone microstructure in the mouse. *BMC Genomics, 16*(1), 493. https://doi.org/10.1186/s12864-015-1617-y.

Reuter, J. A., Spacek, D., & Snyder, M. P. (2015). High-throughput sequencing technologies. *Molecular Cell, 58*(4), 586–597. https://doi.org/10.1016/j.molcel.2015.05.004.

Schwarz, H., & Exner, H. E. (1983). The characterization of the arrangement of feature centroids in planes and volumes. *Journal of Microscopy, 129*(2), 155–169. https://doi.org/10.1111/j.1365-2818.1983.tb04170.x.

Silver, L. M. (1995). *Mouse genetics: Concepts and applications*. Oxford: Oxford University Press.

Part III
Lessons Learned and Outlook

Chapter 24
Lessons Learned from Challenging Data Science Case Studies

Kurt Stockinger, Martin Braschler, and Thilo Stadelmann

Abstract In this chapter, we revisit the conclusions and lessons learned of the chapters presented in Part II of this book and analyze them systematically. The goal of the chapter is threefold: firstly, it serves as a directory to the individual chapters, allowing readers to identify which chapters to focus on when they are interested either in a certain stage of the knowledge discovery process or in a certain data science method or application area. Secondly, the chapter serves as a digested, systematic summary of data science lessons that are relevant for data science practitioners. And lastly, we reflect on the perceptions of a broader public toward the methods and tools that we covered in this book and dare to give an outlook toward the future developments that will be influenced by them.

1 Introduction

Part II of this book contains 16 chapters on the nuts and bolts of data science, divisible into fundamental contributions, chapters on methods and tools, and texts that apply the latter while having a specific application domain in focus. Some of these chapters report on several case studies. They have been compiled with the goal to stay relevant for the readership beyond the lifetime of the projects underlying the specific case studies. To establish this book as a useful resource for reference in any data science undertaking, this chapter serves as a key to unlock this treasure.

The chapter is organized as follows: Sect. 2 presents a taxonomy that covers the main dimensions of content in the individual chapters previously presented in Part II. In Sect. 3, we give concise summaries of all chapters and their learnings. On this basis, we then provide an overall aggregation of the lessons learned in Sect. 4, together with more general insights. Final conclusions are drawn in Sect. 5.

K. Stockinger (✉) · M. Braschler · T. Stadelmann
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland
e-mail: stog@zhaw.ch

M. Braschler et al. (eds.), *Applied Data Science*,
https://doi.org/10.1007/978-3-030-11821-1_24

2 Taxonomy

Table 24.1 provides a taxonomy covering the content of the case studies described in Part II. The taxonomy highlights the main items of the individual chapters and serves as a structured index for the reader to navigate Part II.

3 Concise Reference of Individual Lessons Learned

In this section, we provide a reference to the distilled lessons learned of each chapter of Part II. The section can thus serve the readers to assess their level of data science knowledge and pick out the most pertinent areas for further study.

Chapter 8: What Is Data Science?
A treatise of the fundamentals of data science and data science research from a senior researcher's perspective.

Lessons Learned:

- Data science is an emerging paradigm for accelerated discovery in any field of human endeavor based on the automated analyses of all possible correlations. It has no tools to establish causality between the observed relationships.
- Maturity of data science as a discipline is approximately a decade ahead and will depend on (a) general principles applicable equally to all domains; and (b) collaboration of experts across previous disciplinary silos (which needs a "chief scientific officer" role).
- Based on the analysis of 150 use cases, a generic ten-step data science workflow (in extension of the knowledge discovery process from Chap. 2) is presented and exemplified based on three major scientific projects.

Chapter 9: On Developing Data Science
Suggests the twentieth-century hardware–software virtuous innovation cycle as a role model for how data science projects and the discipline itself should be furthered.

Lessons Learned:

- Data science is inherently an applied science that needs to be connected to real-world use cases: "necessity is the mother of invention," and data scientists even in research profit from solving pressing problems of businesses.
- Still, data science is more than doing data science projects, and data science research units need to be more than the sum of their parts, contributing to data science "per se" by developing software platforms and generally applicable methodology across domains.

Table 24.1 Taxonomy of the case studies described in Part II

Taxonomy	Discussed in chapters															
	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Main focus																
Fundamentals of data science	x	x	x													
Methodology or algorithm				x	x	x	x	x	x	x			x			x
Tool							x		x						x	
Application		x	x							x	x	x	x	x	x	x
Survey or tutorial				x	x			x			x					
Stages in knowledge discovery process[a]																
Data recording	x							x	x	x				x	x	x
Data wrangling	x				x			x			x	x			x	
Data analysis	x			x	x	x	x	x	x	x	x	x	x	x		x
Data visualization and/or interpretation	x		x	x			x				x	x				x
Decision making	x		x				x			x			x	x		
Competence area[b]																
Technology								x	x					x	x	x
Analytics				x	x	x			x	x	x	x	x	x		x
Data management						x	x		x		x	x		x	x	x
Entrepreneurship		x	x					x								
Communication							x					x				
Subdisciplines																
Simulation										x				x		
Data modeling				x							x					
Data warehousing											x			x	x	
Big data technology[c]									x					x		x
Information Retrieval						x										

(continued)

Table 24.1 (continued)

Taxonomy	Discussed in chapters															
	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Clustering					x						x	x				
Classification[d]					x			x				x				x
Regression				x						x						x
Anomaly detection					x			x	x							
Forecasting				x									x	x		
Visualization				x			x					x				
Security								x								
Governance															x	
Ethics			x													
Tools																
IT infrastructure[e]	x					x		x	x		x			x	x	x
Scripted analytics				x	x	x		x		x	x	x	x			x
Open source software				x	x		x		x	x	x	x		x		x
Commercial software							x					x				
Off-the-shelf visualization solution							x					x				
Scripted visualization				x	x		x				x	x				
Data modalities																
Numerical data	x		x	x	x		x	x	x	x	x	x	x	x		
Text						x	x	x			x	x				
Images	x				x											x
Audio[f]					x											
Time series	x			x	x		x					x				
Transactional data								x			x	x	x			
Open data							x									

Table 24.1 (continued)

Taxonomy	Discussed in chapters															
	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Application domain																
Research	x			x	x	x	x		x						x	
Business			x			x	x		x	x				x		
Biology					x											x
Health	x			x			x					x			x	x
e-Commerce and retail			x					x			x		x			
Finance			x											x		
IT								x								
Industry and manufacturing					x					x						
Services	x		x		x						x					

[a]See Chap. 2, Sect. 3.6
[b]See Chap. 3, Fig. 3.1
[c]For example, parallel processing, stream processing
[d]For example, deep neural networks, SVMs
[e]For example, databases, middleware
[f]For example, speech, music

- Several common misunderstandings regarding the adoption of data science in businesses are addressed, including "data science is expensive" or "it is all about AI."

Chapter 10: The Ethics of Big Data Applications in the Consumer Sector
An introduction to and guidelines for ethical considerations in data science applications is given, helping with questions like "to whom does the data belong," or "how is (and should) autonomy, privacy, and solidarity (be) affected."

Lessons Learned:

- A practical guideline regarding unwanted ethical effects is this: would customers still use the product or provide the data if they knew what their data is used for? What could incentivize them to continue doing it if they knew?
- Trust and acceptance of data science applications can be created by informing the customers transparently, and by always providing an option to choose.
- Based on five case studies, a practical weighing of the core values of autonomy, equality, fairness, freedom, privacy, property-rights, solidarity, and transparency that can be adopted in a cookbook fashion.

Chapter 11: Statistical Modeling
A plea for the use of relatively simple, traditional statistical modeling methods (also in contrast to "modern black box approaches"). How to maximize insight into model mechanics, and how to account for human interventions in the modeling process?

Lessons Learned:

- Descriptive analysis requires explicit statistical models. This includes concrete knowledge of the model formulation, variable transformations, and the error structure.
- Statistical models can and should be verified: check if the fit is in line with the model requirements and the subject matter knowledge.
- To obtain sound results and reliable interpretations, the data-generating mechanism within the model developing process and during model assessment has to be considered.

Chapter 12: Beyond ImageNet: Deep Learning in Industrial Practice
An introduction to various case studies on deep learning beyond classifying images: segmentation, clustering, anomaly detection on documents, audio and vibration sensor signals.

Lessons Learned:

- For designing a deep neural network, start with a simple architecture and increase the complexity when more insights into the data and model performance are gained. Generally, if a human expert sees the pattern in the data, a deep net can learn it, too.

- There are many options to deal with limited resources, especially limited training data: transfer learning, data augmentation, adaptable model architectures, or semi-supervised learning. Applying deep learning does not need gigabytes of data.
- Deep models are complex, but far from being black boxes: in order to understand the model performance and the learning process, "debugging" methods such as visualizing the learned weights or inspecting loss values are very helpful.

Chapter 13: The Beauty of Small Data: An Information Retrieval Perspective
Discussion and case studies that show the different challenges between leveraging small and big data.

Lessons Learned:

- Finding patterns in small data is often more difficult than in big data due to the lack of data redundancy.
- Use stemming to increase the occurrences of terms in small document collections and hence increase the potential redundancy to find patterns.
- Enrich data with additional information from external resources and synthesize new, additional keywords for query processing based on relevance feedback.

Chapter 14: Narrative Information Visualization of Open Data
Overview of open data portals of the USA, the EU, and Switzerland. Description of visualization applications on top of open data that enable narrative visualization: a new form of web-based, interactive visualization.

Lessons Learned:

- Data preparation: The most time-consuming aspect of information visualization. Data needs to be manually transformed, harmonized, cleaned, and brought into a common data model that allows easy visualization.
- Visualization technology: High-level visualization frameworks that enable quick prototyping often cannot be used out of the box. In order to get full visualization flexibility, interactive information visualization, and especially narrative visualization often require a development path from rapid prototyping using "out-of-the-box" data graphics toward "customized" visualizations that require some design and coding efforts.

Chapter 15: Security of Data Science and Data Science for Security
A survey on the aspect of computer security in data science (vulnerability of data science methods to attacks; attacks enabled by data science), and on the use of data science for computer security.

Lessons Learned:

- Protect your information systems with suitable security controls by rigorously changing the standard privacy configurations, and using a secure software development life cycle (SSDLC) for all own developments.

- Guidelines are given in the "CIS top twenty security controls," and current security issues are posted, for example, in the "OWASP top 10" for web applications.
- Also secure your models: anonymization is not perfect, analysis on encrypted or anonymized data is still under research, and attackers might try to exploit data-driven applications by data poisoning, model extraction, etc.

Chapter 16: Online Anomaly Detection over Big Data Streams
Various anomaly detection strategies for processing streams of data in an Apache Spark Big Data architecture.

Lessons Learned:

- Make sure that data processing is performed efficiently since data can be lost in case the stream processing buffers fill up.
- Pearson correlation and event counting work well for detecting anomalies with abrupt data changes. For detecting anomalies based on gradually occurring changes, use relative entropy measures.
- Use resampling techniques to determine statistical significance of the anomaly measure. When annotated ground truth data is available, use supervised machine learning techniques to automatically predict the anomaly type.

Chapter 17: Unsupervised Learning and Simulation for Complexity Management in Business Operations
A study on developing a purely data-driven complexity measure for industrial products in order to reduce unnecessary drivers of complexity, made difficult by the unavailability of data.

Lessons Learned:

- In cases where low-level data is unavailable, available high-level data can be turned into a simulation model that produces finer-grained synthetic data in arbitrary quantity, which in turn can be used to train a machine-learning model with the ability to generalize beyond the simulation's discontinuities.
- Complexity of industrial product architectures and process topologies can be measured based on the minimum dimensionality of the bottleneck layer of a suitably trained autoencoder.
- Data-driven complexity measurement can be an alternative to highly qualified business consultants, measuring complexity in a fundamentally different but result-wise comparable way.

Chapter 18: Data Warehousing and Exploratory Analysis for Market Monitoring
An introduction to data warehouse design, exemplified by a case study for an end-to-end design and implementation of a data warehouse and clustering-based data analysis for e-commerce data.

Lessons Learned:

- Data warehouse design and implementation easily take 80% of the time in a combined data preparation and analysis project, as efficiently managing a database with dozens of tables of more than 10^7 records requires careful database tuning and query optimization.
- Data from anonymous e-commerce users can be enriched using Google Analytics as a source; however, the data quality of this source is not easily accessible, making results based on this source to be best considered as estimates.
- When using clustering as an instance of unsupervised machine learning, the necessary human analysis of the results due to the unavailability of labels can be eased using sampling: verify a clustering by analyzing some well-known clusters manually in detail.

Chapter 19: Mining Person-Centric Datasets for Insight, Prediction, and Public Health Planning
A data mining case study demonstrating how latent geographical movement patterns can be extracted from mobile phone call records, turned into population models, and utilized for computational epidemiology.

Lessons Learned:

- Data processing for millions of individuals and billions of records require parallel processing toolkits (e.g., Spark); still, the data needed to be stored and processed in aggregated form at the expense of more difficult and expressive analysis.
- It is important to select the right clustering algorithm for the task (e.g., DBSCAN for a task where clusters are expressed in different densities of the data points, and K-means where clusters are defined by distances), and to deal with noise in the measurements.
- Visualization plays a major role in data analysis: to validate code, methods, results; to generate models; and to find and leverage to wealth of unexpected, latent information and patterns in human-centric datasets.

Chapter 20: Economic Measures of Forecast Accuracy for Demand Planning: A Case-Based Discussion
Methods for evaluating the forecast accuracy to estimate the demand of food products.

Lessons Learned:

- Error metrics are used to evaluate and compare the performance of different forecasting models. However, common error metrics such as root mean square error or relative mean absolute error can lead to bad model decisions for demand forecasting.
- The choice of the best forecasting model depends on the ratio of oversupply costs and stock-out costs. In particular, a baseline model should be preferred over a

peak model if the oversupply costs are much higher than the stock-out costs and vice versa.
- Choosing the optimal observation time window is key for good quality forecasts. A too small observation window results in random deviations without yielding significant insights. A too large observation window might cause poor performance of short-term forecasts.

Chapter 21: Large-Scale Data-Driven Financial Risk Assessment
Study of an approach to standardize the modeling of financial contracts in view of financial analysis, discussing the scalability using Big Data technologies on real data.

Lessons Learned:

- Computational resources nowadays allow solutions in finance, and in particular in financial risk analysis, that can be based on the finest level of granularity possible. Analytical shortcuts that operate on higher levels of granularity are no longer necessary.
- Financial (risk) analysis is possible at the contract level. The analysis can be parallelized and distributed among multiple computing units, showing linear scalability.
- Modern Big Data technologies allow the storage of the entire raw data, without pre-filtering. Thus, special purpose analytical results can be created quickly on demand (with linear computational complexity).
- Frequent risk assessment of financial institutions and ultimately the whole financial system is finally possible on a level potentially on par with that of other fields such as modern weather forecasts.

Chapter 22: Governance and IT Architecture
Governance model and IT architecture for sharing personalized health data.

Lessons Learned:

- Citizens are willing to contribute their health data for scientific analysis if they or family members are affected by diseases.
- Data platforms that manage health data need to have highly transparent governance structures, strong data security standards, data fusion, and natural language processing technologies.
- Citizens need to be able to decide by themselves for which purpose and with whom they share their data.

Chapter 23: Image Analysis at Scale for Finding the Links between Structure and Biology
End-to-end image analysis based on big data technology to better understand bone fractures.

Lessons Learned:

- Image data are well-suited for qualitative analysis but require significant processing to be used in quantitative studies.
- Domain-specific quantitative metrics such as average bone thickness, cell count, or cellular density need to be extracted from images before they can be correlated to images and other data modalities.
- Rather than removing data samples with missing values, data quality issues can be handled by imputation, bootstrapping, and incorporating known distributions.

4 Aggregated Insights

On the basis of the individual lessons learned that we described in the previous section, we will now provide an overall condensation of the lessons learned. We feel that these points are highly relevant and that they form a concise set of "best practices" that can gainfully be referenced in almost every data science project.

- Data science is an inherently interdisciplinary endeavor and needs close collaboration between academia and business. To be successful in a wide range of domains, close *collaboration and knowledge exchange* between domain experts and data scientists with various backgrounds are essential.
- Building a *trust relationship* with customers early on by providing transparent information about the data usage along with rigorous data security practices is key to guarantee wide adoption of data products. Let the customers choose which data they want to share with whom. Part of building trust is also to care for potential *security* issues in and through data analysis right from the start.
- *Data wrangling*, which includes transforming, harmonizing, and cleaning data, is not only a vital prerequisite for machine learning but also for visualization and should thus be a key effort of each data science project. Ideally, data wrangling should be automated using machine learning techniques to ease the burden of manual data preparation.
- Leverage existing *stream processing frameworks* for enabling data wrangling and analysis in real time.
- When choosing a machine learning model to solve a specific problem, start with *simple algorithms* where only a small number of hyperparameters need to be tuned and a simple model results. Increase the complexity of the algorithms and models if necessary and as more insights into the data and model performance are gained.
- Use *visualization* to gain insights into data, track data quality issues, convey results, and even understand the behavior of machine learning models (see also below).
- Modern *big data technology* allows storing, processing, and analyzing vast amounts of (raw) data—often with linear scalability. Restricting models to representative data samples for the sake of reducing data volumes is not strictly necessary any more.

- Leveraging *small data* with low redundancy requires different and maybe more sophisticated approaches than leveraging *big data* with high redundancy.

In condensing the lessons learned to best practices that are generalizable, there is a danger of losing the surprising, inspiring insights that only more detailed looks at specific contexts can bring. By necessity, it is impossible to exhaustively compile such "inspiration" in a list. However, we very much think that much of this inspiration can be found between the covers of this book. In reflecting on the journey of the book's creation, on our own experiences with data science projects over the years, and on the collaboration with the excellent colleagues that have contributed to this volume, we want to emphasize some of these "highlights" that we found:

Data science education has to be interdisciplinary and above Bachelor level to ensure the necessary skills also for societal integration. What are useful outcome competencies for data scientists? The answer to this question differs for data scientists focusing on the engineering aspect compared to those specializing in business aspects or communication or any application domain. But they all will have the following in common: an understanding of the core aspects and prospects of the main methods (e.g., machine learning), tools (e.g., stream processing systems), and domains (e.g, statistics) as well as experience in hands-on projects (in whatever role in an interdisciplinary team). This, combined with the maturity that comes with completed discipline-specific studies during one's Bachelor years, enables a data scientist to ponder and weigh the societal aspects of work in a responsible and educated manner.

Data-driven innovation is becoming increasingly fast, yet not all innovation is research-based; that is why networks of experts are becoming more important to find the right ideas and skills for any planned project. In the area of pattern recognition, for example, we see a usual turnover time from published research result at a scientific conference to application in an industrial context of about 3 months. Many of the results there are driven by deep learning technology, and the lines between fundamental and applied research have become reasonably blurred in recent years [with companies producing lots of fundamental results, and universities engaging in many different application areas, compare e.g. Stadelmann et al. (2018)]. This speaks strongly for collaborations between scientists and engineers from different organizations and units that complement each other's knowledge and skills, for example, from the academic and industrial sector. Simultaneity in working on the fundamental aspects of methods (e.g., furthering deep learning per se) and making it work for a given problem by skillful engineering (e.g., by clever problem-dependent data augmentation and a scalable hardware setup) seems to be key.

On the other hand, only one-third of data-driven innovation needs novel research to happen in order to take place—two-thirds are implementable based on existing technology and tools once the party in need of the innovation gets to know the availability or feasibility of the endeavor, given that resources are available (Swiss Alliance for Data-Intensive Services 2018). If two-thirds of the innovation potential in a country like Switzerland are achievable by education (informing stakeholders

about possibilities) and consulting (bringing in expert knowledge on how to approach the sought innovation), this is a strong argument for every interested party to team up with like-minded organizations and individuals, again to complement each other's skills and know-how to "*together move faster.*"[1]

The paradigm of data parallelism that is enabled by state-of-the-art big data technology makes designing parallel programs relatively easy. However, fully understanding their performance remains hard. Writing scalable, parallel, or distributed programs has generally been considered hard, especially when data is not read-only but can be updated. The main challenge is how to solve the "critical section" (Quinn 2003), that is, how to avoid that two program threads update a specific data item at the same time and thus result in data inconsistency. Different communities use different approaches to tackle this problem. One of the lowest level concepts for parallel programming is to use multithreading, which requires explicit handling of the "critical section" via semaphores (Kleiman et al. 1996). The high-performance community typically uses a higher level of abstraction based on "message passing" where parallel processes communicate via explicit messages (Gropp et al. 1999). Both approaches require highly skilled people to write efficient programs that scale and do not result in deadlocks. The paradigm of data parallelism deployed by state-of-the-art big data technology such as Apache Spark enables implicit parallelism (Zaharia et al. 2016). By design, the core data structures such as Resilient Distributed Datasets or Dataframes enable parallel processing based on the MapReduce paradigm where the programmer has only little design choices to influence the program execution. This implicit parallelism has the great advantage that even people without deep knowledge of parallel programming can write programs that scale well over tens or hundreds of compute nodes. However, the implicit parallelism also comes with a big disadvantage, namely, the illusion that programs scale "by default" and that "parallel programming becomes easy." The hard part of writing good parallel programs with novel big data technology is to fully understand the complex software stack of a distributed system, the various levels of distributed memory management and the impact of data distribution on the runtime of SQL queries or machine learning algorithms. Hence, detailed performance analyses of the workloads and manual optimization techniques such as task repartitioning based on workload characteristics is often the best solution to overcome potential performance problems. The important takeaway message is that understanding and tuning the performance of big data applications can easily take a factor of 10 more time than writing a program that leverages big data technology.

Let machine learning and simulation complement each other. The traditional scientific approach is often based on experimentation and simulation (Winsberg 2010). Experiments are carefully designed based on a specific model. Once data is available or produced by (physical) experiments, the certain phenomena of interest can be evaluated empirically. In addition, simulation is used to complement

[1] See https://data-service-alliance.ch/ for an example of implementing this principle in a way the three authors of this chapter are involved in.

experimentation. Hence, simulation can be used to verify experiments, and experiments can be used to adapt the simulation model. By comparing experimental outcomes with those from simulation, the degree of current understanding of the observed phenomenon (as encoded in the simulation) can be assessed. However, the disadvantage of this approach is that building experiments can be very time-consuming and costly. For instance, building a high-energy physics experiment end-to-end can take more than 10 years (Brumfiel 2011). Moreover, there might not be enough data available to run statistically significant experiments. Finally, building simulation models might become extremely complex, in particular, when some physical, chemical, or biological processes are not fully understood yet.

Hence, machine learning can be applied as an additional pillar. In traditional experimental science, machine learning can be used to *learn* a model from both the experimental and simulated data. The resulting model has the potential to generalize beyond the discontinuities of the simulation model, thus relieving one from making the simulation overly complex. This is not to replace experimentation and simulation, but in addition. On the other hand, in other fields of data science, simulation can serve as a means to data synthesis, thus enhancing the available training data for machine learning approaches. This is heavily used under the umbrella term of "data augmentation," for example, in the field of deep learning.

Models learned from data need to be robust and interpretable to facilitate "debugging" and make them acceptable to humans. Statistical or machine learning models are usually subject to a comprehensive empirical evaluation prior to deployment; the results of these experiments have the power to both show the respective strengths and weaknesses of the model as well as to demonstrate their reliability and generalization capabilities to a critical reviewer (e.g., a business owner, customer, or human subject to a machine-supported decision). Yet, we as humans feel generally uncomfortable when we are subject to processes that we cannot fully grasp and at the mercy of which we feel we are (Lipton 2018); and as developers, having no insight into complex processes like machine learning pipelines and training processes hinders debugging and effective optimization of the model (Stadelmann et al. 2010).

Recent research and development into model interpretability (see, e.g., Ng 2016, Shwartz-Ziv and Tishby 2017, or Olah et al. 2017) not only allows the statement that even the most seemingly opaque machine learning models like deep neural networks can be comprehended to a large degree by humans. The respective work also opens up many more possible developments in research (through a better understanding of what goes wrong) and specific high-risk application domains like automated driving or clinical health (due to the ability to fulfill regulations and bring about necessary performance gains). Thus, trust can be built in applications that directly face a human customer; and better understanding by developers also brings about more robust models with less peculiar behavior (compare Szegedy et al. 2013 with Amirian et al. 2018). Moreover, the understanding possible through introspection into models enables data scientists that are mere users of machine learning to select the best fitting approach to model the data at hand—a task that otherwise needs intimate

knowledge of the inductive biases (Mitchell 1997, Chap. 2) of many potential methods as well as of the structure of the given data.

5 Conclusions

Data science is a highly interesting endeavor, breaking new ground in many ways. Due to the young age and the wide range of the discipline, a number of myths have already taken deep hold, most prominently those that lead to exasperated outbursts along the lines of "no one knows how these algorithms work" or "no one can understand why the output looks like this." We claim that this is plainly untrue, and the various case studies covered in Part II of this book are an excellent testament to this: there is a wide range of scientific literature, and an abundance of tools and methods available to data science practitioners today; there is a wealth of well-founded best practices on how to use them, and there are numerous lessons learned waiting to be studied and heeded.

5.1 Deconstructing Myths by the Example of Recommender Services

If we look at the disruptive players in the information space and their platforms, such as Facebook, Google, Amazon, and others, they also very much rely on these tools and methods to drive their services. Many of the phenomena that, for example, recommender services exhibit in their selection of items are indeed fairly easily and conclusively interpretable by those that have studied the relevant, well-documented algorithms.

It follows that discussions about whether such machine learning components exhibit unwanted biases are certainly very pertinent, but oftentimes not led in the most effective manner [see, e.g., the discussion on biases in word embeddings by Bolukbasi et al. (2016)]. The rapidly increasing use of recommenders based on machine learning to support many knowledge-intensive processes such as media consumption, hiring, shopping, etc., is observed with anxiety by some of those that used to enjoy influence in these fields. Unfortunately, however, these discussions on the merits of machine-generated recommendations are many times led under the wrong pretext. Often the starting point is whether the operators of the recommender service follow a sinister agenda, for example, feeding consumers a steady diet of questionable information of very little variety ["filter bubble"—see Pariser (2011)]. In this view, compounding the sinister agenda of the operator is, again, the fact that "nobody knows what they are doing and how they do it." Scenarios such as "artificial intelligence is already making hiring decisions and your every blink is going to influence your chances" are talked up.

Looking at the situation more soberly, and abstracting from the source of a decision—be it human or machine—the question should be: What do we really want as the output? And does a human (as the chief alternative to the AI-based recommender system) deliver it better and with less bias? In a sense, algorithms can exhibit traits that are very human: if the data used for training exhibits unwanted biases, so will the output of the recommender. A widely reported instance of this was the Microsoft chatbot "Tay" that quickly learned abusive and racist language from Twitter feeds (Hunt 2016).

Reflecting on the filter bubble, the narrow focus of the information stream delivered to some consumers can easily be an expression of overfitting—of the hard problem to generalize to things unseen in prior training, and in incorporating aspects beyond mere item similarity, such as novelty, diversity, etc., into the selection mechanism.

Which closes the circle and brings us back to the all-important question: What do we want from our data? Do we want a "superhuman result"—insight that a human could not have gleaned from the data, or behavior that a human would not exhibit? Or do we want to emulate the (competent) human, producing the same decision a human expert would have arrived at, potentially faster or at lower cost? Are we open to new insights, and can machine-generated recommendations augment human decision making by delivering complementary information, being able to leverage (volumes of) information that humans cannot process? Can it even help to overcome human bias?

5.2 Outlook to a Data-Driven Society

In an abstract perspective, a recommendation—be it made by a human or a computer—is the output of a function of the case-specific inputs plus a number of parameters inherent to the instance making the recommendation, such as preferences and previous history. Two human experts will produce different recommendations given the same inputs. Analogously, the output of an algorithm will change as we change the parametrization. Human decision makers are often bound by rules and regulations in their freedom to make decisions. In the course of the evolution of civilization, there has been constant debate on how to shape these rules and regulations, whom to grant the power to define them, and who to task with enforcing them. Unsurprisingly, we are not at the end of this road. We see no fundamental reason why similar rules and regulations cannot influence the parametrization, and thus the operation of, for example, recommender services.

Data science in general has not only the ability to automate or support decision processes previously reserved to capable humans only, at scale; it also has the potential to alter the ways our societies work in disruptive ways. Brooks (2017) skillfully disarms unsubstantiated fears of mass unemployment in the next decade, and multitudes of humanoid robots or the rise of human-level artificial intelligence are nowhere to be seen. But the current technological possibilities paired with

contemporary economic incentives make it quite clear that society will be impacted on a fundamental level: How can this debate be held in a constructive way in the face of the opinion economy on social media? How to distribute work when repetitive jobs (e.g., medical diagnose, legal case research, or university-level teaching) get digitized to some degree? How to fill one's time in a meaningful way and distribute the gain from increased economic efficiency fairly if it is generated by algorithms in large corporations?

With these exemplary questions above we do not foremost promote to engage in research on "data science for the common good" (see, e.g., Emrouznejad and Charles 2018), although this is important. We rather suggest that much more than thinking about rules of how humans and technology can get along and interact in the future, the possibilities presented to us through a wider deployment of data science will bring us to deal with an age-old topic: How do we want to get along with our fellow human beings? It is a question of society, not technology, to decide on how we share the time and other resources made available to us through the value generated from data; whom we let participate (education), profit (economy), and decide (politics). A big challenge lies ahead in having such a meaningful dialog between technological innovators (and chiefly among them, data scientists), and stakeholders from government and society.

As it is hard not only to predict, but also to imagine a future that deviates largely from a simple extrapolation of today, it is very helpful to recall some of the scenarios that researchers and thinkers have created. Not because they are necessarily likely or desirable, but because seeing a vivid mental picture of them could help in deciding if these scenarios are what we want—and then take respective action. There is Kurzweil's (2010) vision of a superhuman artificial intelligence that controls everything top-down. It can be contrasted with the bottom-up scenario of digitally enabled self-organization suggested by Helbing (2015) that is based on today's technology. Pearl and Mackenzie's (2018) observe as well that current artificial intelligence is limited as long as it cannot use causation (and thus cannot imagine new scenarios), thus outruling superintelligence in the medium term. Harari (2016) puts future influences of massively applied data science on the job market in the center, exploring the possibilities of how humans augment (instead of supersede) themselves with biotechnology, robotics, and AI, but creating a new class of unemployables. Future "class" differences are also a major outcome of the data-driven analyses of Piketty (2014). Precht's (2018) utopia finally reestablishes the humanitarian ideal of working just to better ourselves and the rest of humanity, funded by the profit generated by increasing automatization. We encourage the reader to dive into the original sources of these heavily abbreviated scenario descriptions to see potential consequences of today's developments in pure (and thus often extreme, thus unrealistic) form.

In the end, these sophisticated scenarios may suggest the following prime challenges of society when dealing with the opportunities and risks of data science applied largely and at scale: the "shaping of the future" is not a technical-scientific undertaking, but takes larger efforts (foremost politically, to change regulatory frameworks that still work but are unfit for changed circumstances as are likely to

happen). Change could be driven by a societal consensus on how collaboration in the future should function (the digital technology works as a means to this collaboration), when we overcome the urge to let short-time gains in convenience take us down a path of advancement to an unimagined end. Opportunities, both for individual stakeholders in businesses and industry as well as for societies, are large. Risks exist, mitigations likewise. We suggest taking the lessons learned so far, some of them collected in this volume, and creating places—at work, at home, on earth—worthy of living in and working for.

References

Amirian, M., Schwenker, F., & Stadelmann, T. (2018). Trace and detect adversarial attacks on CNNs using feature response maps. In: *Proceedings of the 8th IAPR TC3 Workshop on Artificial Neural Networks in Pattern Recognition (ANNPR)*, Siena, September 19–21, IAPR.

Bolukbasi, T., Chang, K. W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In: *Advances in Neural Information Processing Systems* (pp. 4349–4357).

Brooks, R. (2017). The seven deadly sins of AI predictions. *MIT Technology Review*. Retrieved March 28, 2018, from https://www.technologyreview.com/s/609048/the-seven-deadly-sins-of-ai-predictions/

Brumfiel, G. (2011). High-energy physics: Down the petabyte highway. *Nature News, 469*(7330), 282–283.

Emrouznejad, A., & Charles, V. (2018). *Big data for the greater good*. Berlin: Springer.

Gropp, W. D., Gropp, W., Lusk, E., & Skjellum, A. (1999). *Using MPI: Portable parallel programming with the message-passing interface* (Vol. 1). Cambridge, MA: MIT Press.

Harari, Y. N. (2016). *Homo Deus: A brief history of tomorrow*. New York, NY: Random House.

Helbing, D. (2015). *Thinking ahead-essays on big data, digital revolution, and participatory market society*. Berlin: Springer.

Hunt, E. (2016). Tay, Microsoft's AI chatbot, gets a crash course in racism from Twitter. *The Guardian*, 24.

Kleiman, S., Shah, D., & Smaalders, B. (1996). *Programming with threads* (p. 48). Mountain View: Sun Soft Press.

Kurzweil, R. (2010). *The singularity is near*. Milan: Gerald Duckworth.

Lipton, Z. C. (2018). The mythos of model interpretability. *Queue, 16*(3), 31–57.

Mitchell, T. M. (1997). *Machine learning*. New York, NY: McGraw Hill.

Ng, A. (2016). Nuts and bolts of building AI applications using deep learning. *NIPS Keynote Talk*. Retrieved July 26, 2018, from https://media.nips.cc/Conferences/2016/Slides/6203-Slides.pdf

Olah, C., Mordvintsev, A., & Schubert, L. (2017). Feature visualization. *Distill*, *2*(11), e7. Retrieved July 26, 2018, from https://distill.pub/2017/feature-visualization/

Pariser, E. (2011). *The filter bubble: What the internet is hiding from you*. London: Penguin Books.

Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect*. New York, NY: Basic Books.

Piketty, T. (2014). *Capital in the 21st century*. Cambridge, MA: Harvard University Press.

Precht, R. D. (2018). *Hunters, Herdsmen, critics. A utopia for digital society*. Munich: Goldmann.

Quinn, M. J. (2003). Parallel programming. *TMH CSE, 526*.

Shwartz-Ziv, R., & Tishby, N. (2017). Opening the black box of deep neural networks via information. *arXiv*:1703.00810.

Stadelmann, T., Wang, Y., Smith, M., Ewerth, R., & Freisleben, B. (2010). Rethinking algorithm design and development in speech processing. In: *Proceedings of the 20th IAPR International Conference on Pattern Recognition (ICPR'10)*, Istanbul, Turkey, August 23–26.

Stadelmann, T., Amirian, M., Arabaci, I., Arnold, M., Duivesteijn, F. F., Elezi, I., et al. (2018). Deep learning in the wild. In: *Proceedings of the 8th IAPR TC 3 Workshop on Artificial Neural Networks for Pattern Recognition (ANNPR'18)*, Siena, September 19–21.

Swiss Alliance for Data-Intensive Services. (2018, January 16). Digitization & innovation through cooperation. Glimpses from the digitization & innovation workshop at "Konferenz Digitale Schweiz". *Blog Post*. Retrieved July 26, 2018, from https://www.data-service-alliance.ch/blog/blog/digitization-innovation-through-cooperation-glimpses-from-the-digitization-innovation-workshop

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., et al. (2013). Intriguing properties of neural networks. *arXiv*:1312.6199.

Winsberg, E. (2010). *Science in the age of computer simulation*. Chicago, IL: University of Chicago Press.

Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., et al. (2016). Apache spark: A unified engine for big data processing. *Communications of the ACM, 59*(11), 56–65.