

Data science vs. statistics: two cultures?

Iain Carmichael & J. S. Marron

**Japanese Journal of Statistics and
Data Science**

ISSN 2520-8756

Jpn J Stat Data Sci
DOI 10.1007/s42081-018-0009-3



Your article is protected by copyright and all rights are held exclusively by Japanese Federation of Statistical Science Associations. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at link.springer.com".

Data science vs. statistics: two cultures?

Iain Carmichael¹  · J. S. Marron²

Received: 4 January 2018 / Accepted: 21 April 2018
© Japanese Federation of Statistical Science Associations 2018

Abstract Data science is the business of learning from data, which is traditionally the business of statistics. Data science, however, is often understood as a broader, task-driven and computationally-oriented version of statistics. Both the term data science and the broader idea it conveys have origins in statistics and are a reaction to a narrower view of data analysis. Expanding upon the views of a number of statisticians, this paper encourages a big-tent view of data analysis. We examine how evolving approaches to modern data analysis relate to the existing discipline of statistics (e.g. exploratory analysis, machine learning, reproducibility, computation, communication and the role of theory). Finally, we discuss what these trends mean for the future of statistics by highlighting promising directions for communication, education and research.

Keywords Computation · Literate programming · Machine learning · Reproducibility · Robustness

✉ Iain Carmichael
iain@unc.edu
J. S. Marron
marron@unc.edu

¹ University of North Carolina at Chapel Hill, B30 Hanes Hall, Chapel Hill, USA

² University of North Carolina at Chapel Hill, 352 Hanes Hall, Chapel Hill, USA

1 Introduction

A simple definition of a data scientist is someone who uses data to solve problems. In the past few years this term has caused a lot of buzz¹ in industry, questions in science² and consternation in statistics.³ One might argue that *data science* is simply a rebranding of *statistics* (e.g. “data science is statistics on a Mac”⁴), but this comment misses the point.

While data analysis has been around for a long time, its economics (costs of and value derived) have changed primarily due to technology driving the availability of data, computational capabilities and ease of communication. The most obvious advances are in computer hardware (e.g. faster CPUs, smaller microchips, GPUs, distributed computing). Similarly, algorithmic advances play a big role in making computation faster and cheaper (e.g. optimization and computational linear algebra). There are many new/improving technologies which allow us to gather data in new, faster and cheaper ways, including: drones, medical imaging, sensors (e.g. Lidar), better robotics, Amazon’s Mechanical Turk, wearables, etc. Finally, improved software (e.g. see Sect. 4.2.1 below) makes it faster, cheaper and easier to communicate the results of data analysis, distribute software, and publish research/educational resources.

These changes mean that more people get more value out of data analysis. One would be hard pressed to find an industry or academic discipline not impacted by data analysis. These developments also mean that data analysis has become increasingly multidisciplinary and collaborative. The field of data analysis has broadened in that more people want to analyze data and data analysis draws on more disciplines. Areas of statistics previously considered specialized (e.g. statistical software, exploratory analysis, data visualization, high dimensional analysis, complex data objects and the use of optimization methods) have become dramatically more valuable.

Various viewpoints have been expressed on the current state/future of statistics and the fashionable topic of *data science* (Tukey 1962; Wu 1998; Cleveland 2001; Breiman et al. 2001; Hand et al. 2006; Yu 2014; Wasserman 2014; Efron and Hastie 2016; Donoho 2017; Hooker and Hooker 2017; Bühlmann and Stuart 2016; Blei and Smyth 2017; Barocas et al. 2017; Bühlmann and van de Geer 2018; Reid 2018). Views have also been expressed in blogs: Simply Statistics,⁵ Statistical Modeling, Causal Inference, and Social Science,⁶ and Normal Deviate.⁷

This paper offers additional opinions and perspectives and we strive for a unique viewpoint through combining both new and old ways of thinking. The first author is

¹ <https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century>.

² <https://www.wired.com/2008/06/pb-theory/>.

³ <https://simplystatistics.org/2015/10/29/the-statistics-identity-crisis-am-i-really-a-data-scientist/>.

⁴ <https://twitter.com/cdixon/status/428914681911070720>.

⁵ <https://simplystatistics.org/>.

⁶ <http://andrewgelman.com/>.

⁷ <https://normaldeviate.wordpress.com/>.

a computationally oriented PhD student in a statistics, who has developed and taught an entirely new undergraduate course on data science⁸ for his department. The second author has 35 years of academic experience in statistics including: research, collaborative applications and teaching at all levels. He has offered previous opinions on *big data*, in particular on the topic of *robustness against heterogeneity* in Marron (2017).

In Sect. 2, we discuss the definition of *data science* and how it relates to the discipline of *statistics*. In Sect. 3, we discuss a few *principal components* of data analysis which give insights into aspects of modern data analysis (e.g. why the rise of computation is tied to the rise of exploratory and predictive analysis). Finally, in Sect. 4, we discuss future directions of statistics research, communication and education including: complex data, robustness, machine learning and data processing, and literate programming.

2 What is data science?

To a general audience, data science is often defined as the intersection⁹ of three areas: math/statistics, computation and a particular domain (e.g. biology) (Conway 2010; Yu 2014; Blei and Smyth 2017). Implicit in this definition is the focus on solving specific problems (in contrast with the type of deep understanding that is typical in academic statistics¹⁰). The focus on problem solving is important because it explains differing judgements to be found on how to value contributions to the field. As stated in Cleveland (2001)

[results in] data science should be judged by the extent to which they enable the analyst to learn from data. . . Tools that are used by the data analyst are of direct benefit. Theories that serve as a basis for developing tools are of indirect benefit.

For the purpose of this article we define data science as the union of six areas of *greater data science* which are borrowed from David Donoho's article titled "50 Years of Data Science" (Donoho 2017). We will refer to these as GDS 1–6 and more details can be found in Section 8 from Donoho's article.¹¹

1. Data gathering, preparation, and exploration.
2. Data representation and transformation.¹²
3. Computing with data.
4. Data modeling.
5. Data visualization and presentation.

⁸ <https://idc9.github.io/stor390/>.

⁹ This idea is usually communicated through a venn diagram, e.g. <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>.

¹⁰ <https://simplystatistics.org/2014/07/25/academic-statisticians-there-is-no-shame-in-developing-statistical-solutions-that-solve-just-one-problem/>.

¹¹ <http://www.tandfonline.com/doi/pdf/10.1080/10618600.2017.1384734>.

¹² This includes both databases and mathematical representations of data.

6. Science about data science.¹³

The purpose of defining data science in this way is to (A) better capture where people who work with data spend their time/effort and (B) put more focus on the value of each tool for providing insights. This definition is given in contrast to the current, perceived state of statistics. A number of other statisticians have proposed similar definitions using different terminologies and ontologies (Tukey 1962; Chambers 1993; Cleveland 2001; Yu 2014). People might reasonably tweak the above definition.

The term *data science*, both the literal string and the broader idea that it conveys, originates from statisticians (see Sect. 2.1). In a 1993 essay titled “Greater or Lesser Statistics: a Choice for Future Research” statistician John Chambers wrote (Chambers 1993)

Greater statistics can be defined simply, if loosely, as everything related to learning from data, from the first planning or collection to the last presentation or report. Lesser statistics is the body of specifically statistical methodology that has evolved within the profession—roughly, statistics as defined by texts, journals, and doctoral dissertations. Greater statistics tend to be inclusive, eclectic with respect to methodology, closely associated with other disciplines, and practiced by many outside of academia and often outside professional statistics. Lesser statistics tends to be exclusive, oriented to mathematical techniques, less frequently collaborative with other disciplines, and primarily practiced by members of university departments of statistics.

We take the position that data science is a reaction to the narrow understanding of *lesser statistics*; simply put, data science has come to mean a broader view of statistics.

It's worth noting that our discussion is about data science from the perspective of statistics. Since data science is a multidisciplinary field, other disciplines, such as computer science, might see data science in a different way. For example, computer science might focus more on: machine learning, large scale computation and data storage/retrieval.

2.1 What's in a name?

The term data science has caused excitement, confusion and controversy. Some of the confusion is from the lack of a consistent definition.¹⁴ There is an ecosystem of related terms (e.g. *analytics*, *business intelligence*, *big data*, *data mining*). Many companies/people/organizations have their own internal definition for each of these terms, e.g. one company's data scientist is another company's business analyst. The lack of a consistent term makes discussion challenging.

These terms are contentious partially because of the buzz associated with them and because of arguments about academic discipline subsetting, e.g. some argue

¹³ For example, reproducible research would fall under this category and point 5.

¹⁴ We take the position that data science is the practice of broader statistics.

data science is a subset of statistics¹⁵ while others argue statistics is a subset of data science.¹⁶ Academic funding (and egos) play a non-trivial role in the controversy.

Even the origin of the term is up for debate. We point to a few sources for developing¹⁷ both the literal string and the broader idea, “data science”

- Tukey (1962): the idea, not the literal string
- Naur (1974): the literal string, not the idea¹⁸
- Chambers (1993): the idea, not the literal string
- Wu (1998): both the idea and literal string
- Cleveland (2001): both the idea and literal string

Others are credited with bringing the term/idea to industry, e.g. Patil (2011). For more on the development of the term see Donoho (2017) and the two articles linked to below.¹⁹ Both the idea (as defined in Sect. 2) and the literal string (A) have been around for a while and (B) have origins in statistics.

The term data science has broken free from academic statistics into industry and other academic fields. In some cases, data science marginalizes the discipline of statistics which is a detriment to both statistics and anyone who analyzes data. Acknowledging the history of data science and statistics we hope will garner more respect for data science within statistics and for statistics from the broader community of data practitioners.

In the next few section we discuss the origins of data science, critiques of statistics and the broader notion of reproducibility.

2.2 Critiques of statistics

To first order, we summarize the critiques of statistics as: too much theory, not enough computation. We believe theory is important, however, too much theory at the expense of other things is a detriment to the field. See Sect. 4 for a more optimistic discussion of theory.

We thank Hongtu Zhu for pointing out two quotes from the Priceconomics website (Bhardwaj 2017),²⁰ which provides some interesting discussion about data science and statistics.

Statistics was primarily developed to help people deal with pre-computer data problems like testing the impact of fertilizer in agriculture, or figuring out the accuracy of an estimate from a small sample. Data science emphasizes the data problems of the 21st Century, like accessing information from large databases, writing code to manipulate data, and visualizing data.

¹⁵ “A data scientist is a statistician who lives in San Francisco” (Bhardwaj 2017).

¹⁶ <http://andrewgelman.com/2013/11/14/statistics-least-important-part-data-science/>.

¹⁷ We do not claim this list is exhaustive.

¹⁸ Peter Naur uses the term “data science” but in a narrower sense, focusing more on computation.

¹⁹ <http://bulletin.imstat.org/2014/10/ims-presidential-address-let-us-own-data-science/> and <https://www.forbes.com/sites/gilpress/2013/05/28/a-very-short-history-of-data-science/>.

²⁰ <https://priceconomics.com/whats-the-difference-between-data-science-and/>.

This quote is echoed by statisticians such as Hadley Wickham who lament the lack of priority academic statistics has given to such areas.²¹ This point is well taken, e.g. what proportion of statistics undergraduates are competent in R or Python? However, statistics is being sold short on its contributions to computation (Donoho 2017). For example, the R-Project (Members 2017) is a context where many members of the statistical community are directly engaged in writing and sharing code. Furthermore, there is a rather large area called *statistical computing*, with quite a long history, see, e.g. the American Statistical Association's Section,²² which has been in continuous operation since 1972.

Another example is visualizing data. We acknowledge that a large fraction of the statistical community has created a culture of not devoting enough energy in the direction of looking at data. However, again there is a relatively small but very active community devoted to visualization, including the American Statistical Association's Section on Graphics,²³ formed in 1985.

Another popular, but incorrect belief is that statistics is not concerned with *big data* (or phrased sans buzz words: statisticians do not care about computing things efficiently). As Donoho (2017) points out, statisticians have in fact always been interested in large data computation. For example, the word *statistics* came about from work on census data which have been around for centuries and are large even by today's standards. The principle of *sufficiency* is of course a mechanism to deal with large data sets efficiently. The point here is that these pursuits are a part of statistics, but are perhaps considered specialized as opposed to mainstream (e.g. in terms of publications in flagship journals, undergrad/graduate education, etc).

A second quote from Priceonomics [(also echoed by Chambers (1993) and Donoho (2017)]

Statistics, on the other hand, has not changed significantly in response to new technology. The field continues to emphasize theory, and introductory statistics courses focus more on hypothesis testing than statistical computing. . . For the most part, statisticians chose not take on the data problems of the computer age.

This point is on target in a number of ways, but we take a different viewpoint on a few issues.

The first is on the value of statistical theory. Far from viewing it as something old fashioned and hence useless, we argue that the need for theoretical thinking is greater than ever in the data science age (see Sect. 4). There are few calls yet for *data science theory*, although the US National Science Foundation's TRIPODS program²⁴ is an important exception. However we predict that as the field evolves there will be growing realization of the importance of that line of thought, as more

²¹ e.g. see <https://priceonomics.com/hadley-wickham-the-man-who-revolutionized-r/> and the quote about "The fact that data science exists as a field is a colossal failure of statistics".

²² <http://stat-computing.org/computing/>.

²³ <http://stat-graphics.org/graphics/>.

²⁴ <https://www.nsf.gov/funding/pgmsumm.jsp/pimsid=505347>.

and more algorithms become available with very little meaningful basis for making the critical choice among them.

The second is the statement that “statistics courses focus more on hypothesis testing”. This makes the statistical outsider’s mistake of thinking that statistics is a set of recipes for doing data analysis. It misses the deeper truth understood by people who practice data analysis: when properly taught, statistics courses teach an important way of thinking called the *scientific method*. The main idea is that to be really sure of making actual discoveries (as opposed to finding spurious and ungeneralizable sampling artifacts) scientists should first formulate a hypothesis, then collect data and finally analyze.

One can be forgiven, however, for mistaking statistics as a set of recipes. Too many people interact with statistics exclusively via a standard Statistics 101 type class which may in fact treat statistics as a handful of formulas to memorize and steps to follow. While we believe the material taught in these courses is vital to doing science, it is perhaps time to rethink such introductory classes and teach data before (or concurrently with) teaching statistics. See Sect. 4.3 and the references therein.

2.3 Reproducibility and communication

The idea of *reproducibility* has become an important topic in science (Peng 2011; Stodden 2012; Kiar et al. 2017). A narrow sense of reproducibility is *the ability to recompute results*, i.e. that someone else can easily obtain the data and run the code used in the original experiment (Leek and Peng 2015; Patil et al. 2016). Reproducibility, in the broader sense, is about three things: scientific validity, communication and methodological development.

A number of factors go into assessing the scientific validity of a study. The gold standard of scientific validity is *independent replication*. Replicability is about the ability for someone else to rerun the same experiment and obtain the same results.²⁵ Reproducibility is about checking the details of the scientific argument made in a paper (much in the way a mathematician would check the details of a proof of a theorem). The requirement that the code which produced the results can be rerun at a later time and will create the exact same figures/numbers is an important condition for assessing the correctness of the results. It is also a software engineering issue; the scientist has to write code which is understandable, well documented, publicly available, and persists across time and computers.²⁶

The analysis code is a key part of communicating the analysis itself, citing Jon Claerbout (Buckheit and Donoho 1995)

[a]n article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual

²⁵ The literature is not consistent about the definitions of reproducibility and replicability. In this paper we use the definitions given here.

²⁶ Writing code that continues to work overtime is non-trivial; it involves maintaining the same computing environment and managing dependencies correctly, e.g. the software packages the code uses change over time, version 1.1.1 might work the same as version 2.1.1.

scholarship is the complete software development environment and the complete set of instructions which generated the figures.

The text summary of an analysis in the paper may not include all the details of a complex analysis (e.g. how were hyper-parameters tuned, how were the data preprocessed, etc). Moreover the description of the analysis may be incorrect: numbers can be misreported by accident or intentionally, the code may not work the way the analyst thought it did,²⁷ there may be bugs in the code, etc.

Computational methodologies are often reused and built upon. If the code for a methodology is not available then the next person who wants to use/iterate on that methodology has to re-write the code²⁸ which is both an inefficient use of time/resources and can lead to errors.

Three of the main barriers to reproducibility are: culture, computational tools and education (Peng 2011; Stodden 2012; Leek and Peng 2015). Even if the cultural and educational problems are solved, there are still technical challenges which discourage reproducibility in practice. These issues are primarily about software engineering; while reproducibility is technically achievable, it is often too burdensome in practice.²⁹ For example, “in a recent survey of the machine learning community—the single biggest barrier to sharing code and data was the time it takes to clean up and document the work to prepare it for release and reuse” (Stodden 2012). Other issues include: data sharing, data privacy, software sharing, software verification (e.g. unit testing), writing readable code, version control, and legality.

Section 4.2 discusses how the rise of *literate programming* in data science has improved our ability to do and communicate reproducible science. It also discusses how the sharing of code and educational resources is a boon to the field.

3 Some principal components of data science

In this section we discuss different approaches to data analysis and how they relate to broader trends in data science. We call these *principal components* in the sense that they explain variation in the practice of data analysis. Each of the modes discussed below (e.g. predictive vs. inferential analysis) represents a spectrum between two methodologies, one more associated with data science and the other with classical statistics. These modes help explain why different communities seem to talk past each other and why some techniques (e.g. computation) have become more popular in recent times.

²⁷ Understanding the nitty-gritty details of how statistical software works is not trivial: how does the optimization routine determine convergence? Are the data mean centered by default? There is a lack in uniformity in how statistical software is written; we believe this is exacerbated by the lack of of statisticians writing statistical software.

²⁸ Even if the code for a study is available, someone may still want to rewrite the code say in another language. In this case have the original code available to base the new code on is helpful.

²⁹ Publishing messy code is still beneficial and certainly better than not publishing code (Barnes 2010).

3.1 Prediction vs. inference (do vs. understand)

A computer scientist might pejoratively describe a linear or logistic regression as shallow and quaint.³⁰ A statistician might express bewilderment at the buzz around deep learning and question why more principled, interpretable statistical models won't do the trick. The point here is that these two imaginary, curmudgeonly academics are thinking about problems with different goals. The computer scientist is trying to build a system to accomplish some task; the statistician is typically trying to learn something about how the world works.

Prediction vs. inference is a spectrum. Many complicated problems have well defined subproblems which are closer to one end or the other end. The distinction we are trying to make here is maybe better described as engineering vs. science (or at least broad generalizations thereof). Engineering is the business of creating a thing that *does* something.³¹ Science is the business of *understanding* how something works. Of course engineers use science and scientists use engineering. But if we focus on the end goal of a particular problem we can probably, reasonably classify that problem as either science or engineering.³²

Breiman et al. (2001) discusses many of the differences between predictive and inferential modeling. Predictive modeling often uses more sophisticated, computationally intensive models. This often comes with a loss in interpretability and general understanding about how the model works and what the data look like (Freitas 2014). Predictive modeling also places less emphasis on theory/assumptions because there are fairly good, external metrics to tell the analyst how well they are doing (e.g. test set error).

Predictive modeling is one of the main drivers of *artificial intelligence* (AI). The fact that data can be used to help computers automate things is perhaps one of the most impactful innovations of recent decades. Early attempts at AI type applications involved primarily *rules based systems* which did not use a lot of data (Russell and Norvig 2009); modern AI systems are typically based on *deep learning* and are extremely data hungry (Goodfellow et al. 2016). Section 4.1.4 discusses some of the ways statisticians can make important contributions to AI/predictive modeling. Section 4.1.5 discuss some of the ways statisticians/scientists may benefit from and reasons to be aware of machine learning.

3.2 Empirically vs. theoretically driven

Data science is exploratory data analysis gone mad.—Neil Lawrence³³

³⁰ The use of shallow means we can view a generalized linear model as a neural network with 0 layers. The more layers a network has, the more complex of a pattern it can find (Goodfellow et al. 2016).

³¹ i.e. the output of a predictive model may be interesting insofar as it helps us do something.

³² In other words, in many cases understanding is primarily a means to and ends for predictive problems and visa versa.

³³ From Talking Machines season 3, episode 5. <https://www.thetalkingmachines.com/>.

Most quantitative fields of study do both theoretical and empirical work³⁴, e.g. theoretical vs. experimental physics. Within statistics, we might contrast *exploratory data analysis* (EDA) vs. *confirmatory analysis*, i.e. searching for hypotheses vs. attempting to confirm a hypothesis (in this section *theory* refers to the scientific theory being tested).

It used to be that statistics and science were primarily theory driven. A scientist has a model of the world; they design and conduct an experiment to assess this model; then use hypothesis testing to confirm the results of the experiment. With changes in data availability and computing, the value of exploratory analysis, data mining, and using data to generate hypotheses has increased dramatically (Fayyad et al. 1996; Blei and Smyth 2017). EDA often prioritizes the ability to rapidly experiment which means computation can dominate the analysis.

EDA can become problematic when the analyst puts too much faith in its results, i.e. when the analyst mistakes hypothesis generation for hypothesis confirmation. Every statistician can list problems with simply *correlation mining* a data set (e.g. false discovery, sampling bias, etc). These problems do not mean the results are wrong, but rather they mean exploratory analysis provides much weaker evidence for a hypothesis than confirmatory analysis. The amount this matters is context dependent.

One problematic idea is that EDA can solve every problem. For example, in the controversially titled article, “The End of Theory: The Data Deluge Makes the Scientific Method Obsolete” it was argued that EDA will replace the scientific method (Anderson 2008). We disagree. This article is an extreme example of the broader attitude that correlation, and fancy models applied to large data sets, can replace causal inference and the careful, time intensive scientific method. The point that EDA can contribute to scientific applications is well taken, however, and this in fact is becoming more common (Blei and Smyth 2017). These applications likely raise many interesting and impactful research questions arising from the increased value in EDA (in all of GDS 1-6).

3.3 Problem first vs. hammer looking for a nail

Some researchers take a *hammer looking for a nail* approach; the researcher has developed/studied a statistical procedure and then looks for problems where it might be applicable. Other researchers aim to solve some particular problem from a domain. Note that the former approach is strongly correlated with, but not equivalent to theoretical research (same for the latter and applied research).

Both research approaches are valid and productive, however the balance in academic statistics may have shifted too far to the former (hammer) approach. For example, Rafael Irizarry has some interesting further commentary³⁵ which is echoed

³⁴ This statement probably applies to non-quantitative fields. For example, some academics in comparative literature are more “empirical” in the sense they examine a particular body of work, draw conclusions and possibly generalize/relate their conclusions to other bodies of work. Other people in comparative literature apply “theoretical methods”.

³⁵ <https://simplystatistics.org/2014/07/25/academic-statisticians-there-is-no-shame-in-developing-statistical-solutions-that-solve-just-one-problem/>.

by a number of others (Tukey 1962; Breiman et al. 2001; Wasserman 2014; Donoho 2017). We include this section because data science is focused on problem solving and it is this problem solving which makes data analysis useful to other disciplines.

3.4 The 80/20 rule

The 80/20 rule of data analysis is:³⁶

One of the under appreciated rules in statistical methods development is what I call the 80/20 rule (maybe could even be the 90/10 rule). The basic idea is that the first reasonable thing you can do to a set of data often is 80% of the way to the optimal solution. Everything after that is working on getting the last 20%.

Applying basic models to a data set often provides the most value (and/or solves the problem of interest much of the time). The 80/20 rule explains part of why a number of techniques have become more valuable than in the past and why the six areas of GDS emphasize previously undervalued areas. These include: data visualization, exploratory data analysis, data mining, programming, data storage/processing, computation with large datasets and communication.

4 Going forward

In this section we discuss a number of areas we believe are particularly promising for statistics research, communication and education. A pervasive theme in this section is the role of computation, defined broadly as in Nolan and Temple Lang (2010).

4.1 Research

As data analysis becomes more valuable, existing statistical theory and methodology also become more valuable. Criticisms of statistical theory (e.g. Sect. 2.2 and Donoho 2017) are largely about ignoring other, less mathematically glamorous areas of statistics.

In this section we highlight a number of (potentially) promising areas of statistics research. There are many ways in which one can do *good* statistical research, including all of those discussed in Tao (2007). The list below is biased in favor of our tastes and areas which involve both theory and aspects of greater data science (from Sect. 2). Rather than diminishing the role of mathematical statistics, we emphasize that many of these areas will require novel contributions from mathematical statistics which itself should be broadened.

³⁶ <https://simplystatistics.org/2014/03/20/the-8020-rule-of-statistical-methods-development/>.

4.1.1 Complex data and representation

The recently fashionable area of Big Data has received a large amount of well deserved attention and has led to many serious analytic challenges. Currently less well understood is that a possibly bigger challenge is non-standard or *complex data*. In particular, many modern data analytic scenarios involve non-standard data such as: networks, text, sound, images, movies, shapes, very high dimensional data, data living on a manifold, etc. In response to this challenge, (Wang and Marron 2007; Marron and Alonso 2014) have proposed *Object Oriented Data Analysis* (OODA).

A fundamental concept of OODA is that in complex scientific contexts, it is often not even clear what should be the atoms of the statistical part of the analysis, i.e. the experimental units. OODA provides terminology for pivotal team discussions on this topic between domain scientists and statisticians which lead to an effective final analysis, via resolving *what should be the data objects*? An interesting example of this in an image analysis context can be found in Lu et al. (2014).

The OODA discussion goes beyond choice of experimental units also to include the key technical issue of *data representation*. This includes both fairly standard statistical techniques such as data transformation, but also mathematically deeper issues such as the appropriate data space, as discussed for example in Pizer and Marron (2017). Much of the success of deep learning may come from its ability to automatically discover useful data representation (Bengio et al. 2013); connections between OODA and *representation learning* are unexplored and potentially fruitful directions for future research.

Complex data and OODA present many research opportunities such as: invention of powerful new tools for data practitioners, computational challenges, methodological developments, and developments in statistical theory. OODA often involves bringing in a number of mathematical disciplines such as differential geometry, topology, optimization, probability, etc, providing a sense in which data science should become a truly interdisciplinary endeavor. It is also an opportunity to greatly extend usage of the term *mathematical statistics* to include many more mathematical areas beyond just the conventional probability theory.

4.1.2 Robustness to unknown heterogeneity

An even less widely acknowledged and studied challenge in data science is *data heterogeneity*. This topic is very current to many modern sciences, where there has been a growing realization that complex data collection by a single lab tends to result in sample sizes that are inadequate to address many modern scientific issues of interest.

The poster child for this problem may be cancer research, where the very diversity of the disease requires very large sample sizes in a context where the needed measurements are very labor, time and cost intensive. Such challenges have led to the formation of large multi-lab research consortiums. In the cancer world, a well known effort of this type is The Cancer Genome Atlas (TCGA) (Network et al. 2012). While great care is taken in such efforts to try to standardize laboratory protocols and many other aspects of the data collection, it is well known among all

data collectors that there are always impossible to control biases that creep into the data set when data from different labs are combined.

Dealing with this issue is the main challenge of data heterogeneity. It is clear that the standard statistical Gaussian model for noisy data is no longer appropriate. When data from different sources are combined, a much more appropriate statistical model is a Gaussian mixture, but this presents a major challenge to classical statistical approaches such as the likelihood principle.

Much more research is strongly needed on all aspects of data analysis for heterogeneous data, including methods, computation and theory. It is also clear that real progress is going to be made by approaches which truly integrate all three of those. More discussion of important early work in that direction can be found in Bühlmann and Meinshausen (2016), Bühlmann and Stuart (2016) and Marron (2017).

4.1.3 Scalable and robust models

Robustness issues are important for modern, large data applications (see Huber 2011; Hampel et al. 2011; Staudte and Sheather 2011; Maronna et al. 2006 for an overview of this area). For the gains to be realized from robust statistical procedures for large data sets, these procedures need to be computationally efficient (or *scalable*). We want models which are both scalable and robust. We point to Bottou et al. (2016) for an overview of important computational methods in machine learning.

Robustness presents a challenge for developing scalable models for large data because often robust methods are harder to compute.³⁷ Models which are both scalable and robust present an opportunity for research developments drawing on all of statistics, optimization and computer science. The work of Aravkin and Davis (2016) is a recent example in this vein. We point this area out because it requires joint reasoning based on both computation and statistical theory.

4.1.4 Automation and interpretability

One of the biggest ways data is impacting society is by powering automation through machine learning. While data driven automation has a lot of potential to do good, recent years have brought new attention to its negative consequences (Zarsky 2016; O'Neil 2017; Doshi-Velez and Kim 2017; Crawford 2017). For example, Bolukbasi et al. (2016) demonstrate that natural language processing algorithms trained on google news text data learned offensive gender stereotypes. In a book titled "Weapons of Math Destruction" data scientist Cathy O'Neil provides many examples of how automation can perpetuate and even reinforce existing societal inequalities (O'Neil 2017).

Statistics is the discipline historically most concerned with the myriad of ways in which data can be misleading. The rise of automation presents new opportunities to the discipline. First, the decades of statistics' hard won knowledge about dealing with data is salient to many applications of data driven automation. Second,

³⁷ e.g. L2 regularized (ridge) linear regression has a closed form while L1 regularization (LASSO) does not.

automation presents new technical challenges to statistics since machine learning often involves applying sophisticated modeling techniques to large, complex datasets. One of the primary technical challenges is interpretability.

Interpretability, as discussed in Lipton (2016) and Doshi-Velez and Kim (2017), is desirable for a number of reasons including: trust, causality, transferability, informativeness, fair and ethical decision-making, legality, and safety. Additionally, interpretability can mean different things, e.g. a visualization, a verbal explanation, understanding the model, etc. The canonical examples of interpretable models are generalized linear models and decision trees. The canonical example of an *uninterpretable* model is a deep neural network. One interesting approach to interpretability is discussed in Ribeiro et al. (2016), called *local interpretable, model-agnostic explanations* (LIME), which proposes the use of simple models to help explain predictions made by more sophisticated models.

The pursuit of understanding machine learning models to make them more interpretable is a wide open area for statisticians. These questions may involve narrowing the gap between predictive and inferential modeling discussed in Sect. 3.1.

4.1.5 Machine learning and data processing

Complex data preprocessing pipelines are familiar to statisticians working in many areas including genomics and neuroscience (Gentleman et al. 2006; Kiar et al. 2017). The raw data go through a number of processing steps before reaching the analyst as a .csv file. These steps may include: simple transformations, complex algorithms, output of statistical models, etc. Knowledge of the processing pipelines can be important for the analyst because the pipelines can have errors and introduce systemic biases in the data.

Machine learning, particularly deep learning, may cause complex data processing pipelines to be more frequently used in modern data analysis. A lot of work in machine learning is about measurement: image recognition, image captioning, speech recognition, machine translation, syntactic parsing, sentiment analysis, etc (Goodfellow et al. 2016). These examples somewhat blur the line between data processing and data gathering. We suspect that it will be more common for data analysts to have one or more variables in their datasets which are the output of a deep learning model. While machine learning can provide the analyst with rich, new variables, it should also give the statistician some pause.

There are at least two major issues with using a deep learning model to gather data: it puts black box models into the data processing pipeline and introduces an external data set (i.e. the one used to train the deep learning model). Neither of these are new issues, but they will become more prominent. The black box model is problematic because it means the analyst cannot fully understand where the raw data came from. External data is problematic because it can infect new datasets it comes into contact with via the deep learning model on which it was trained.³⁸ Both

³⁸ For example, if google's algorithms mistakenly think someone is dead, then likely the rest of the world will too <https://www.nytimes.com/2017/12/16/business/google-thinks-im-dead.html>.

of these issues are likely to create systemic biases which are challenging to detect and appropriately handle.

We suspect that machine learning will create a lot of value in data processing in many domains. However there are important theoretical and methodological questions which will arise when using deep learning models to gather data for inferential, scientific applications.

4.2 Computation and communication

In this section we discuss a number of ways in which computation can improve communication in data analysis.³⁹ Section 4.2.1 below discusses how *literate programming* helps solve many of the issues in reproducibility discussed in Sect. 2.3 above.

4.2.1 Literate programming

Literate programming is a concept developed by computer scientist Donald Knuth in the 1980s (Knuth 1984).

Literate programming is a methodology that combines a programming language with a documentation language, thereby making programs more robust, more portable, more easily maintained, and arguably more fun to write than programs that are written only in a high-level language. The main idea is to treat a program as a piece of literature, addressed to human beings rather than to a computer.

Literate programming is about both cultural and technical developments. Cultural in the sense that it emphasizes the programmer's responsibility to communicate with humans. Lessons learned in English courses about organization, clarity, prose, etc are relevant in computer science. Technical in the sense that programming languages need to be altered or created to enable the programmer to write documents which communicate effectively (Knuth 1984). Literate programming is important to data analysis because data analysis is done with computer code; explaining the results and details of an analysis involves communicating the details of a computer program.

Traditional data analysis code is not written for human consumption (e.g. little documentation, not made public); the analyst communicates the results through prose and figures which summarize the execution of the code. For some applications this is acceptable in the sense that it accomplishes what is needed. Problems with this style of programming include: it's not reproducible, it's not generalizable, it doesn't communicate the details of the analysis, it can make it harder to find errors with the analysis code, etc. As data analysis becomes more used, more multidisciplinary, and more complex these issues only become more important.

³⁹ Our argument is that computation can help communication. Others have taken this idea further and use computation, specifically information theory, as a metaphor for communication, e.g. Doumont (2009).

The core idea of literate programming—writing code whose target audience includes humans—has a lot to contribute to data analysis.

Communicating data analysis through the medium of code is challenging. Best practices have not yet been established (e.g. how should the code be commented, how to evaluate trade-offs between code clarity and simplicity/efficiency). Software engineers have built up best practices for programming (Wilson et al. 2014, 2017), which can be helpful for data analysts to learn, but may need to be adapted in some cases.⁴⁰ In addition to methodology, there are technical developments which make literate programming easier.

We highlight *knitr* (Xie 2015) and R Markdown⁴¹ as examples of research into GDS 6, “science about data science.” As discussed in Donoho (2017),

This helps data analysts authoring source documents that blend running R code together with text, and then compiling those documents by running the R code, extracting results from the live computation and inserting them in a high-quality PDF file, HTML web page, or other output product. In effect, the entire workflow of a data analysis is intertwined with the interpretation of the results, saving a huge amount of error-prone manual cut and paste moving computational outputs and their place in the document.

There are other examples of technological developments in literate programming for data analysis (e.g. Jupyter notebooks Perez and Granger 2015). Others⁴² developing tools to solve technical issues⁴³ in reproducible research include (Sandve et al. 2013; Kiar et al. 2017). Literate programming is also impactful in statistics education,⁴⁴ particularly for demonstrating programming examples.⁴⁵

4.2.2 Open source

As discussed in Sect. 2.3 reproducibility has a number of technical challenges and literate programming goes a long way to address these issues. Reproducibility crucially demands that the analysis code and data be publically released. This is probably more of a cultural issue than a technical issue (though it’s worth pointing out that technologies such as GitHub make sharing code much easier). Some journals (e.g. Biostatistics) encourage the authors to release their code; we hope that more journals will follow this example.

Releasing analysis code makes the original analysis more impactful. A statistics paper is useful primarily to statisticians; an R package is useful to anyone who

⁴⁰ For example, it is often suggested that code comments should describe *why* the code was written the way it was, not *what* the code is doing. For data analysis, where the target audience is probably less experienced with programming, describing the *what* may also be useful.

⁴¹ For more information and examples see <http://rmarkdown.rstudio.com/>.

⁴² e.g. see the list of people discussed in <https://simplystatistics.org/2015/12/11/instead-of-research-on-reproducibility-just-do-reproducible-research/>.

⁴³ The complexity and time costs to making research reproducible is, in part, technical issue.

⁴⁴ <https://simplystatistics.org/2017/06/13/the-future-of-education-is-plain-text/>.

⁴⁵ e.g. see each of the notes from <https://idc9.github.io/stor390/>.

knows R. If a future analyst has existing code to work with then their life will often become easier. This can take a number of different forms depending on the project.

For some projects, writing an R script which carries out the analysis will allow a future analyst to copy, paste and modify the original script. In other cases, if a researcher develops a new algorithm or complex methodology then releasing a software package may be most appropriate. Resources such as Wickham (2015) make developing software packages fairly straightforward. Finally, many research advancements build upon existing algorithms/methodologies. In this case, enhancing an existing software package (as opposed to developing a new one) can be most cost effective. The open source software community has built up best practices for developing and maintaining open source software projects.⁴⁶

It's worth noting there is a large amount of open source statistical software already available, e.g. CRAN, bioconductor, scipy, sklearn. However, we suspect the majority of modern statistics research does not make it into good, publicly available software packages.

Open source extends beyond analysis/research code to educational resources. There are now many technologies which make it easy to publish educational resources for free through a variety of mediums: books, blogs, massive open online courses (MOOCs⁴⁷), websites, etc. As the first author can attest, data science has been particularly good at making resources available for free online. This is an important trend for a number of reasons; it saves students money, provides them with more resources to learn from and it democratizes the subject in the sense that anyone can learn the basic skills to get started in the subject.

The technology is largely available to make most data analyses, statistical algorithms and educational resources freely accessible. Two of the biggest barriers to releasing these materials are education and incentives. Education in the sense that many data analysts do not currently know these technologies which causes the release of code to be too burdensome. Incentives in the sense that there are few of them to encourage releasing these materials. How would a statistics department's hiring committee value a software package vs. a paper? Surely they are both intellectually challenging to produce and valuable, but we suspect the latter would typically receive much more weight than the former. There are some examples of people promoting the value of research software (Sonnenburg et al. 2007) and we hope there will be more in the future.

While open source has lot of potential value, it also raises some questions. For example, not all data can be shared due to privacy/confidentiality reasons; what steps should researchers take in this case to share their analysis code/data? One research avenue worth noting in this regard is *differential privacy* (Smith et al. 2016). As noted in Graves et al. (2000) and Eick et al. (2001) software decays if it is not actively maintained. This raises the question, for how long should research be reproducible?⁴⁸

⁴⁶ e.g. see <https://github.com/scipy/scipy/blob/master/HACKING.rst.txt>.

⁴⁷ <http://mooc.org/>.

⁴⁸ <https://simplystatistics.org/2017/02/01/reproducible-research-limits/>.

4.3 Education

A number of people have written about updating the statistics curriculum in ways which better reflect the broader definition of data science and the skills required for doing data analysis (Nolan and Temple Lang 2010; Association et al. 2014; De Veaux et al. 2017; Hardin et al. 2015; Hicks and Irizarry 2017). A number of programs which have embraced these recommendations have proven to be successful such as the Johns Hopkins Data Science Specialization on Coursera⁴⁹ (Kross et al. 2017) and Berkeley's Data8 program⁵⁰ (Alivisatos 2017). We observe three takeaways from this literature (and our own experiences): more computation, more data analysis and the use of open source material.

Communication is another area worth highlighting in this education section because of its importance and ubiquity. The modern data analyst is expected to communicate across a number of different media: written papers/reports, static/dynamic visualizations, online via creating a website/blog, through code, etc. Communication is already under represented in STEM education, in spite of the demand from employers (academic and industrial alike) (Felder and Brent 2016). Both authors have made effort⁵¹ to include communication in statistics education (Marron 1999).

4.3.1 More computation

Computation comes into data analysis in a number of ways, from processing data to fitting statistical models to communicating the results. For example, as discussed in Sect. 2.3 many of the issues in reproducibility can be partially addressed by teaching more software engineering to scientists. Many new areas of statistics research involve tackling both statistical and computational issues, e.g. see Sect. 4.1.3 on scalable, robust estimators or Efron and Hastie (2016).

With the large number of technologies involved with data analysis (programming languages, visualization software, algorithms, etc) one often feels a bit overwhelmed at what one might be expected to know. It is infeasible to know everything. This is where updating the statistics education curriculum is critical. There is probably some rough core set of computational knowledge every trained statistician should have. Once the fixed cost of learning the core computational curriculum is paid, the marginal cost of learning additional computational skills will go down.

⁴⁹ <https://www.coursera.org/specializations/jhu-data-science>.

⁵⁰ <http://data8.org/>.

⁵¹ e.g. including a lecture on communication in an undergraduate data science course: <https://ide9.github.io/stor390/notes/communication/communication.html>.

4.3.2 Pedagogy

The references cited above in Sect. 4.3 make a number of good recommendations for improving statistics education. We highlight a couple of points here.

As many of the above references discuss, the current statistics curriculum often lacks data analysis (Tukey 1962; Nolan and Temple Lang 2010). Real data analysis makes the discipline more concrete to students. Focus on solving a real problem can be engaging to students who might otherwise find the subject boring. Teaching data analysis is challenging,⁵² but it's challenging in the way that teaching the practice of engineering or using the scientific method is challenging. By not giving students practice doing data analysis for a real problem, the statistics curriculum may encourage students to view statistical methodology as a hammer to be procedurally applied to data. It's well established in engineering and the physical sciences that students should get some practical experience doing the thing during their education: why does the same principle not apply more often statistics at the undergraduate (and graduate) level?

When teaching statistical modeling⁵³ it might be more effective to first introduce the model (e.g. linear/logistic regression) in terms of a predictive context instead of the traditional inferential context. While the math behind linear models may not be particularly sophisticated, the concept of randomness and relating it to the real world through data is non-trivial. Moreover, inferential modeling comes with a lot of baggage. For predictive purposes, models such as linear regression can be introduced as an optimization problem which can be heuristically motivated and analytically solved.⁵⁴ Once students are comfortable with data modeling we can then introduce inferential/confirmatory modeling.

Finally, we suggest teaching data before statistics in introductory statistics courses. In other words, we should teach exploratory analysis before inferential analysis. This would involve teaching programming, data visualization and manipulation before teaching hypothesis testing. Students are more likely to care about hypothesis testing if they have actually worked on a real problem/dataset which motivates it (as opposed to a hypothetical or artificial one). Berkeley's new course Data 8: The Foundations of Data Science appears to take this approach.⁵⁵

5 Conclusion

So far the arguments in this paper have been about providing value to society by broadening the discipline in technical ways. Equally as important is increasing diversity in statistics by encouraging women and underrepresented minorities to join and stay in the discipline. Programs, conferences and other efforts such as: ASA

⁵² <https://simplystatistics.org/2017/12/20/thoughts-on-david-donoho-s-fifty-years-of-data-science/>.

⁵³ e.g. in an upper level undergraduate course such as UNC's STOR 455: Statistical Methods I.

⁵⁴ See for example <https://www.inferentialthinking.com/chapters/13/prediction.html>.

⁵⁵ e.g. see the order of the chapters in the textbook: <https://www.inferentialthinking.com/>.

Committee on Minorities in Statistics,⁵⁶ the Women in Machine Learning conference⁵⁷ and the Women in Statistics and Data Science conference⁵⁸ should be strongly encouraged and expanded upon.

We return to the question of whether data science and statistics are really two different disciplines. If statistics is defined as the narrow discipline described by the quote from John Chambers in Sect. 2 then the answer is yes. However, if statistics embraces the broader idea of greater data science (e.g. by putting more focus on computation in education, research and communication) then we argue the answer is no.

Acknowledgements This research was supported in part by the National Science Foundation under Grant No. 1633074. We would like to thank Deborah Carmichael for editorial comments.

References

- Alivisatos, P. (2017). Stem and computer science education: Preparing the 21st century workforce. Research and Technology Subcommittee House Committee on Science, Space, and Technology.
- Anderson, C. (2008). The end of theory: The data deluge makes the scientific method obsolete. *Wired Magazine*, 16(7), 16–07.
- Aravkin, A., & Davis, D. (2016). A smart stochastic algorithm for nonconvex optimization with applications to robust machine learning. arXiv preprint [arXiv:161001101](https://arxiv.org/abs/161001101).
- Association, A. S., et al. (2014). Curriculum guidelines for undergraduate programs in statistical science. Retrieved March 3, 2009, from <http://www.amstat.org/education/curriculumguidelines.cfm>.
- Barnes, N. (2010). Publish your computer code: It is good enough. *Nature News*, 467(7317), 753–753.
- Barocas, S., Boyd, D., Friedler, S., & Wallach, H. (2017). Social and technical trade-offs in data science.
- 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.
- Bhardwaj, A. (2017). What is the difference between data science and statistics? <https://priceonomics.com/whats-the-difference-between-data-science-and/>.
- Blei, D. M., & Smyth, P. (2017). Science and data science. *Proceedings of the National Academy of Sciences*, 114(33), 8689–8692.
- 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).
- Bottou, L., Curtis, F. E., & Nocedal, J. (2016). Optimization methods for large-scale machine learning. arXiv preprint [arXiv:160604838](https://arxiv.org/abs/160604838).
- Breiman, L., et al. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3), 199–231.
- Buckheit, J. B., & Donoho, D. L. (1995). Wavelab and reproducible research. In: *Wavelets and statistics* (pp. 55–81), Springer.
- Bühlmann, P., & van de Geer, S. (2018). Statistics for big data: A perspective. *Statistics and Probability Letters*.
- Bühlmann, P., & Meinshausen, N. (2016). Maging: maximin aggregation for inhomogeneous large-scale data. *Proceedings of the IEEE*, 104(1), 126–135.
- Bühlmann, P., & Stuart, A. M. (2016). Mathematics, statistics and data science. *EMS Newsletter*, 100, 28–30.
- Chambers, J. M. (1993). Greater or lesser statistics: A choice for future research. *Statistics and Computing*, 3(4), 182–184.

⁵⁶ <http://community.amstat.org/cmis/home>.

⁵⁷ <http://wimlworkshop.org/>.

⁵⁸ <http://ww2.amstat.org/meetings/wds/2018/>.

- 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.
- Conway, D. (2010). The data science Venn diagram.
- Crawford, K. (2017). The trouble with bias. Conference on Neural Information Processing Systems, invited speaker.
- De Veaux, R. D., Agarwal, M., Averett, M., Baumer, B. S., Bray, A., Bressoud, T. C., et al. (2017). Curriculum guidelines for undergraduate programs in data science. *Annual Review of Statistics and Its Application*, 4, 15–30.
- Donoho, D. (2017). 50 years of data science. *Journal of Computational and Graphical Statistics*, 26(4), 745–766.
- Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. arXiv preprint [arXiv:170208608](https://arxiv.org/abs/1702.08608).
- Efron, B., & Hastie, T. (2016). *Computer age statistical inference* (vol 5). Cambridge: Cambridge University Press.
- Eick, S. G., Graves, T. L., Karr, A. F., Marron, J., & Mockus, A. (2001). Does code decay? Assessing the evidence from change management data. *IEEE Transactions on Software Engineering*, 27(1), 1–12.
- Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (1996). *Advances in knowledge discovery and data mining* (Vol. 21). Menlo Park: AAAI press.
- Felder, R. M., & Brent, R. (2016). *Teaching and learning STEM: A practical guide*. Hoboken: Wiley.
- Freitas, A. A. (2014). Comprehensible classification models: A position paper. *ACM SIGKDD Explorations Newsletter*, 15(1), 1–10.
- Gentleman, R., Carey, V., Huber, W., Irizarry, R., & Dudoit, S. (2006). *Bioinformatics and computational biology solutions using R and Bioconductor*. Berlin: Springer.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Book in preparation for mit press. <http://www.deeplearningbook.org>.
- Graves, T. L., Karr, A. F., Marron, J., & Siy, H. (2000). Predicting fault incidence using software change history. *IEEE Transactions on Software Engineering*, 26(7), 653–661.
- Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., & Stahel, W. A. (2011). *Robust statistics: the approach based on influence functions* (Vol. 114). Hoboken: Wiley.
- Hand, D. J., et al. (2006). Classifier technology and the illusion of progress. *Statistical Science*, 21(1), 1–14.
- Hardin, J., Hoerl, R., Horton, N. J., Nolan, D., Baumer, B., Hall-Holt, O., et al. (2015). Data science in statistics curricula: Preparing students to “think with data”. *The American Statistician*, 69(4), 343–353.
- Hicks, S. C., & Irizarry, R. A. (2017). A guide to teaching data science. *The American Statistician*(**just-accepted**).
- Hooker, G., & Hooker, C. (2017). Machine learning and the future of realism. arXiv preprint [arXiv:170404688](https://arxiv.org/abs/1704.04688).
- Huber, P. J. (2011). Robust statistics. In: *International Encyclopedia of Statistical Science* (pp. 1248–1251). Springer.
- Jl, Doumont. (2009). *Trees, maps, and theorems*. Brussels: Principia.
- Kiar, G., Bridgeford, E., Chandrashekar, V., Mhembere, D., Burns, & R., Roncal, W. G., et al. (2017). A comprehensive cloud framework for accurate and reliable human connectome estimation and meganalysis. *bioRxiv* p 188706.
- Knuth, D. E. (1984). Literate programming. *The Computer Journal*, 27(2), 97–111.
- Kross, S., Peng, R. D., Caffo, B. S., Gooding, I., & Leek, J. T. (2017). The democratization of data science education. *Peer J (PrePrints)*.
- Leek, J. T., & Peng, R. D. (2015). Opinion: Reproducible research can still be wrong: Adopting a prevention approach. *Proceedings of the National Academy of Sciences*, 112(6), 1645–1646.
- Lipton, Z. C. (2016). The mythos of model interpretability. arXiv preprint [arXiv:160603490](https://arxiv.org/abs/1606.03490).
- Lu, X., Marron, J., & Haaland, P. (2014). Object-oriented data analysis of cell images. *Journal of the American Statistical Association*, 109(506), 548–559.
- Maronna, R., Martin, R. D., & Yohai, V. (2006). *Robust statistics* (Vol. 1). Chichester: Wiley.
- Marron, J. (1999). Effective writing in mathematical statistics. *Statistica Neerlandica*, 53(1), 68–75.
- Marron, J. (2017). Big data in context and robustness against heterogeneity. *Econometrics and Statistics*, 2, 73–80.
- Marron, J., & Alonso, A. M. (2014). Overview of object oriented data analysis. *Biometrical Journal*, 56(5), 732–753.

- Members, R. P. (2017). The r project for statistical computing. <https://www.r-project.org/>.
- Naur, P. (1974). *Concise survey of computer methods*.
- Network, C. G. A., et al. (2012). Comprehensive molecular characterization of human colon and rectal cancer. *Nature*, 487(7407), 330–337.
- Nolan, D., & Temple Lang, D. (2010). Computing in the statistics curricula. *The American Statistician*, 64(2), 97–107.
- O’Neil, C. (2017). *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books.
- Patil, D. (2011). *Building data science teams*. “O’Reilly Media, Inc.”.
- Patil, P., Peng, R. D., & Leek, J. (2016). A statistical definition for reproducibility and replicability. bioRxiv p 066803.
- Peng, R. D. (2011). Reproducible research in computational science. *Science*, 334(6060), 1226–1227.
- Perez, F., & Granger, B. E. (2015). *Project jupyter: Computational narratives as the engine of collaborative data science*. Tech. rep., Technical Report. Technical report, Project Jupyter.
- Pizer, S. M., & Marron, J. (2017). Object statistics on curved manifolds. In *Statistical Shape and Deformation Analysis: Methods, Implementation and Applications* (p. 137).
- Reid, N. (2018). Statistical science in the world of big data. *Statistics and Probability Letters*.
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should i trust you? Explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144). ACM.
- Russell, S., & Norvig, P. (2009). *Artificial intelligence: A modern approach*. Egnlewood Cliffs: Artificial Intelligence Prentice-Hall.
- Sandve, G. K., Nekrutenko, A., Taylor, J., & Hovig, E. (2013). Ten simple rules for reproducible computational research. *PLoS Computational Biology*, 9(10), 285. (e1003) .
- Smith, M. T., Zwiessle, M., & Lawrence, N. D. (2016) Differentially private Gaussian processes. arXiv preprint [arXiv:160600720](https://arxiv.org/abs/1606.00720).
- Sonnenburg, S., Braun, M. L., Ong, C. S., Bengio, S., Bottou, L., Holmes, G., et al. (2007). The need for open source software in machine learning. *Journal of Machine Learning Research*, 8(oct), 2443–2466.
- Staudte, R. G., & Sheather, S. J. (2011). *Robust estimation and testing* (Vol. 918). Hoboken: Wiley.
- Stodden, V. (2012). Reproducible research for scientific computing: Tools and strategies for changing the culture. *Computing in Science and Engineering*, 14(4), 13–17.
- Tao, T. (2007). What is good mathematics? *Bulletin of the American Mathematical Society*, 44(4), 623–634.
- Tukey, J. W. (1962). The future of data analysis. *The Annals of Mathematical Statistics*, 33(1), 1–67.
- Wang, H., & Marron, J. (2007). Object oriented data analysis: Sets of trees. *The Annals of Statistics*, 1849–1873.
- Wasserman, L. (2014). Rise of the machines. In *Past, present, and future of statistical science* (pp. 1–12).
- Wickham, H. (2015). *R packages: Organize, test, document, and share your code*. O’Reilly Media, Inc.
- Wilson, G., Aruliah, D. A., Brown, C. T., Hong, N. P. C., Davis, M., Guy, R. T., et al. (2014). Best practices for scientific computing. *PLoS Biology*, 12(1), 745. (e1001) .
- Wilson, G., Bryan, J., Cranston, K., Kitzes, J., Nederbragt, L., & Teal, T. K. (2017). Good enough practices in scientific computing. *PLoS Computational Biology*, 13(6), 510. (e1005) .
- Wu, C. (1998). Statistics = data science? <http://www2.isye.gatech.edu/~jeffwu/presentations/datascience.pdf>.
- Xie, Y. (2015). *Dynamic Documents with R and knitr* (Vol. 29). Boca Raton: CRC Press.
- Yu, B. (2014). Ims presidential address: Let us own data science. <http://bulletin.imstat.org/2014/10/ims-presidential-address-let-us-own-data-science/>.
- Zarsky, T. (2016). The trouble with algorithmic decisions: An analytic road map to examine efficiency and fairness in automated and opaque decision making. *Science, Technology, and Human Values*, 41(1), 118–132.