Teaching Data Science: Constructing Pillars in a Fluid Field

Sven Helmer

Abstract Due to the popularity of data science and an increasing demand for training and degree courses, many institutions have put programs in place for teaching data science. However, since data science is not yet a well-established discipline, two general questions remain largely unanswered: What to teach and how to teach. I scrutinize the challenges faced by educators and trainers when developing data science courses, work out what we actually agree on when talking about data science, and sketch potential solutions.

Sven Helmer
University of Zurich, Department of Informatics, Zurich, Switzerland
✉ helmer@ifi.uzh.ch

Archives of Data Science, Series A
(Online First)
KIT Scientific Publishing
Vol. 8, No. 2, 2022

DOI: 10.5445/IR/1000150240
ISSN 2363-9881

1 Introduction

Data science is far from an established field (see e.g. Brodie (2019); Meng (2019)), which makes teaching it very challenging. Generally, as unified and agreed-upon principles, methods, and processes are largely missing, it is unclear

1. what we should teach and
2. how we should teach.

The former is due to the wide range of topics in data science: It includes many different areas of expertise and very rarely does a single individual master all of them (see Irizarry (2020)). O'Neil and Schutt (2014) argue that this is also the reason why in data science projects team members with different backgrounds and skills contribute to the outcome. Consequently, the goal cannot be to make someone an expert in all of these different areas, which would be way too ambitious.

The latter question, how to teach data science, is heavily influenced by the debate on training versus education. Burrus (2015) states that training is a skills-oriented way of teaching to increase proficiency, whereas education is mainly concept-based, providing an understanding of why and how things work.

In my experience, data science courses currently come in two flavors. On the one hand, they tend to lean towards training, which means that the acquired skills become obsolete quite quickly, due to the rapidly changing landscape of tools employed in data science. On the other hand, if they try to be more educational, they focus on fundamental concepts taken from neighboring disciplines, such as computer science or statistics (due to a lack of fundamental concepts in data science). For some concepts this may make sense, but the transfer of concepts should not be applied broadly and blindly. In the recent past, a lot of disciplines probably started like this, e.g. computer science as a blend between electrical engineering and mathematics. Stodden (2020) argues that implementing data science from a point of view that is very close to a particular discipline risks missing the opportunity to develop data science into its own field. For instance, mathematics is an auxiliary science for physics, but that does not mean that physicists should work like mathematicians. Over time, physicists have developed their own methodologies. Before transferring concepts, we

should become aware of our own biases due to our specific backgrounds and double-check whether such a transfer makes sense or not.

Trying to find an answer to the fundamental questions of what and how to teach is very challenging and I do not claim to have the answers, Nevertheless, in the following I delve deeper into the topic of teaching data science by talking about particular challenges faced by teachers and lecturers, outlining what basic concepts of data science people agree on (which can be taken as a basis for a more educational teaching approach), and sketching possible approaches to follow when teaching data science.

2 Particular Challenges

In addition to the two general questions raised in the introduction, there are more particular challenges when it comes to teaching data science. In the following I summarize findings by Brunner and Kim (2016) and Hicks and Irizarry (2018), many of which match my own experiences. First, the background of the students is usually very diverse. Due to the wide range of topics covered by data science, it appeals to people working on data analysis in many different areas and this has a direct impact on the technical depth of a course. For example, not all of the participating students may be able to write complex programs. On the other hand, the goal may not be to train professional software developers, but to show how to use existing platforms and packages and how to integrate code into software stacks.

Second, there is the difficulty in setting up the computing environment. Ideally, we want students to set up an environment on their own machines and to use professional tools. However, setting up these environments on several different operating systems can be a nightmare and some tools easily overwhelm beginners. Another option is to run the environment on a server (e.g. JupyterHub, Renku, or Google Colab). While this provides a smoother experience, some students might be in for a bit of a shock when faced with an actual working environment later. I have run courses with local deployments of the software and the server option as a backup solution, but this duplicates some of the preparation work. The crucial question here is: How realistic do we want to make the experience and how much do we abstract away?

Third, choosing use cases and finding relevant datasets is far from trivial. Even though there is no shortage of public-domain datasets, almost none of

them have been built with the intention of using them for teaching. Ideally, for teaching such a dataset does not require a lot of effort to understand the application domain, but still covers many interesting use cases.

3 What Do We Agree On?

On a very abstract level, data science is about unearthing knowledge from data in a systematic approach. Many definitions of data science go in this direction. For instance, Dhar (2013) defines data science as the "study of the generalizable extraction of knowledge from data." Stodden (2020) mentions conversations she had with colleagues developing the following definition: "Data Science is the science of (collaboratively) generating, acquiring, managing, analyzing, carrying out inference, and reporting on data." While these definitions are correct, they are very vague and do not help in distinguishing data science from other disciplines. After all, the above definitions could also be used to broadly describe the work of statisticians, data analysts, or researchers.

A more concrete approach is to define data science by using the general workflow followed by data scientists as a framework. Brodie (2019) argues that the workflow or pipeline of predefined steps for knowledge discovery is roughly based on the scientific method. It may not be as rigorous and some of the concrete tasks within a step may be domain-specific and hard to generalize, but according to Brodie, "the central organizing principle of a data science activity is its workflow or pipeline and its life cycle management." Wing (2019) defines data science as an umbrella term encompassing the complex and multi-step processes used to extract value from data. Like Brodie, she identifies different steps in a workflow. O'Neil and Schutt (2014) call this the *data science process* and again we find a framework consisting of different steps.

Stodden (2020) uses the data life cycle as a starting point to define her data science life cycle. She notes that there is no single fixed definition of a data life cycle, but typical phases involve acquiring, cleaning, using, publishing, and in the end archiving or destroying the data. The data life cycle is also embedded into an environment referring to aspects such as ethics, policy, stewardship, and platforms.

The data science life cycle focuses not only on the dataset, but also includes all the artifacts used to create and process the dataset, such as code, workflow, and

computational environment information. The data science life cycle is divided into different layers:

- The *application layer* is very similar to the descriptions used by Brodie (2019), Wing (2019), and O'Neil and Schutt (2014). It includes typical steps of the data science workflow, such as data collection, exploratory data analysis, data cleaning and processing, model building, and communicating the results.
- Each step in the application layer has a counterpart in the *infrastructure layer*, which describes the computational infrastructure that enables and supports the application layer. For instance, for the data collection step this includes database structures, for the model building step, scripts and notebooks, and for the visualization step, visualization software.
- Underlying the application and infrastructure layer and running across the entire data science life cycle is the *system layer*. This layer includes the computing infrastructure, cloud computing systems, data structures, and storage systems. In a nutshell, this is the hardware and other technology on which data science tasks are carried out.
- Last, but not least, there is an overarching *"The Science of Data Science" layer*, which concerns itself with the meta-level and includes, for example, research ethics, reproducibility of results, artifact reuse, metadata creation and documentation, governance, curation, and regulatory and legal considerations.

Stodden (2020) interprets the term *life cycle* to mean that along with the published findings the data, software, and associated artifacts need to be made available to the research community, so that other researchers are able to continue the effort. She sees the data science life cycle as an effort "to support the development of a scientific discipline, enabling progress toward fulfilling Tukey's three criteria for a science. The criteria set out by Tukey (1962) are: (1) intellectual content, (2) organization into an understandable form, and (3) reliance upon the test of experience as the ultimate standard of validity.

There also seems to be a consensus that students should apply the newly acquired knowledge in capstone projects, ideally using real-world datasets (see Brunner and Kim (2016); Hicks and Irizarry (2018); Kross and Guo

(2019)). However, as Wild and Pfannkuch (1999) point out, just adding projects to a curriculum is not enough: "The usual panacea for 'teaching' students . . . is, with apologies to Marie-Antoinette, 'let them do projects'. Although this enables students to experience more of the breadth of ... [an] activity, experience is not enough. The cornerstone of teaching in any area is the development of a theoretical structure with which to make sense of experience, to learn from it and transfer insights to others."

4 Possible Approaches

In the following, I would like to sketch some approaches to teaching data science. This is roughly organized along the lines of the recommendations made by Irizarry (2020).

4.1 Structuring Data Science

First, since data science is such a wide field (and is even considered an umbrella term by, for instance, Meng (2019) and Wing (2019)), it makes sense to distinguish different subtopics. Irizarry suggests differentiating between *backend data science* and *frontend data science*. The backend part focuses on data engineering aspects, such as hardware, storage and processing infrastructure, and efficient computing. Essentially, this is what Stodden (2020) classifies as the system layer and, to a certain extent, the infrastructure layer. The frontend part covers data analysis tasks, such as data exploration, data cleaning, and fitting models to data. This includes the application layer in the data science life cycle and knowledge about how to use the tools provided by the infrastructure layer. (Irizzary also mentions that machine learning could be split off from frontend data science as its own subtopic.) In my opinion, distinguishing between different subtopics makes a lot of sense, as trying to tackle all the different facets of data science in their entirety in a single study program is too ambitious. Focusing on a subtopic helps in determining and structuring the content of a data science course or program. It also supports students in identifying the background they need to be able to follow a course and what they will get out of it. Stodden (2020) also discusses how to tackle the broadness of the

data science field from a different point of view. According to her, the major challenge faced by data science on its way towards becoming an independent discipline is acquiring a coherent scope while at the same time being highly interdisciplinary. Essentially, both Stodden and Irizarry argue that achieving this coherence without a framework to structure the various topics found in data science would be very hard. The verdict on what the framework ultimately should be is still out, but currently many people use a data science workflow or pipeline as such a framework.

4.2 Technical Depth of Teaching

The second point that Irizarry (2020) makes is that students need to acquire the skills to build and maintain data processing pipelines and need to learn the appropriate programming languages (together with their ecosystems) to be able to do this. While I agree in principle, the crucial question is how deep to go into technical details. When teaching backend data science, letting students implement algorithms or (parts of) systems from scratch to obtain a deeper understanding (for an example, see Scherzinger (2019) on building a SQL-on-Hadoop query engine) is definitely within scope. On the other hand, when teaching frontend data science, the participants will likely have a different background. However, students still need to understand how their work is integrated into the data science workflow. Code needs to be written in a modular, reliable, and reusable way (e.g. Jupyter notebooks are not the right choice when it comes to deploying code in a production environment). Lau et al. (2022) give an account of an interesting discussion between computer science and statistics instructors on how to best teach gradient descent (and models based on it). The discussion revolved around just using `scikit-learn` packages or letting students implement gradient descent from scratch. In this case, the issue was resolved by teaching the use of `scikit-learn` packages and letting students implement a basic version of gradient descent from scratch. This illustrates that often questions on how to teach data science do not have “either-or” answers, but that a balance has to be struck depending on the needs of the target audience.

4.3 Theory versus Practice

Third, Irizarry argues that applications need to play a much bigger role than theory. This is a point I disagree with. Without a substantial framework grounded in theory, data science would just be a collection of tools, technologies, and best practices and would not deserve to be called a science. I accept that the application domain aspect has to be there, but as already indicated in the previous section I think that adding a framework would make the experience gained from the practical work much more impressive. I have started teaching a new course called "Systems for Data Science" at the University of Zurich in the spring term 2022 and I divided the content of the course into chapters I label as "principles" or "concrete systems". For example, in a chapter called "Principles: Scalability" I show different approaches for scaling distributed systems, such as master-worker and peer-to-peer architectures. In a later chapter called "Concrete Systems: File Systems", I present concrete systems implementing these architectures, such as the Hadoop Distributed File System (HDFS) as an example for a master-worker architecture and GlusterFS as an example for a peer-to-peer one. These file systems are then used by the students in practical exercises. Let me now come back to the role of applications in teaching data science. For teaching purposes, we need high-quality data sets (e.g. by taking datasets from the Swiss Open Data initiative as a basis: Opendata.ch (2011)), ideally originating in the real world and at-scale. However, this does not come for free: Berman et al. (2018) believe that this requires responsible data stewardship, i.e., people who develop, curate, and share the datasets.

4.4 Level of Teaching

Finally, Irizarry advocates that data science programs should be run at the graduate level (i.e., on the Master level), which I fully agree with. Due to the complexity and vagueness of the topic, it would be very difficult to teach data science with as much coherence and structure as other undergraduate courses. In my own experience, I found that undergraduate students prefer a standard textbook as the main reading material for a course. The lack of such a book usually results in a longish list of reading material, the individual items of which are not necessarily consistent with each other. This is fine for

a course on a graduate level, e.g., for my Master-level course "Systems for Data Science", I have a reading list consisting of almost thirty items, including various book chapters, research papers, technical reports, white papers, and web pages. While I expect a Master student to be able to read, interpret, and put into context all this material, an undergraduate student, who still needs to acquire fundamental knowledge in a field, will likely struggle with these tasks, especially the contextualizing. Clearly, there is always the option that the lecturer of a course provides a (detailed) manuscript (or even writes his/her own textbook), but we cannot realistically expect every lecturer to do so.

5 Conclusions and Outlook

Currently, when preparing data science courses it is difficult to decide what to teach and how to teach. While I cannot provide a final answer to these questions, I identify major challenges faced by data science lecturers today. As already indicated earlier, I think that potential solutions are not extreme either-or decisions, but compromises along a whole spectrum. In the following, I present what I believe are the most important criteria that have to be balanced when teaching data science.

First, we have theory versus practice. Clearly, teaching data science from a purely theoretical viewpoint would be far from ideal, especially given the current state of data science. At the moment, major parts of data science revolve around analyzing real-world datasets with a plethora of different tools, systems, and platforms, so the whole field is driven by very practical considerations. However, I believe that a purely practical approach would also not work. If researchers and analysts only learn how to use certain tools without a deeper understanding of the underlying principles, we end up with a very mechanical way of conducting data science and the whole field will have a hard time establishing itself as a discipline. In the context of teacher training, a quote by Immanuel Kant on the issue of balancing has been paraphrased in an interesting way (see Brandl (2012)): "Theory without practice is empty, practice without theory is blind." Finding a compromise between theory and practice is hardly a new thing. For instance, Etzkowitz and Peters (1991) illustrate how the conflict between theory and practice played out over time on a more general level in the academic landscape of United States. While I am unsure what a good balance between

theory and practice should be for data science, in my opinion, the pendulum has swung too far in the direction of practice.
Second, there is the technical depth of the teaching or, phrased differently, frontend versus backend data science. While it is possible to implement everything from scratch (e.g., see Grus (2019)), this means that in the end the students will have detailed knowledge about specific topics but run the risk of losing sight of the big picture. Data science is such a wide field, making it unrealistic to cover everything at a very detailed level (even the scope of Grus (2019) is rather limited). On the other hand, not implementing anything and not knowing anything about the internals of tools, systems, and platforms makes it difficult to use them to their full potential and to combine them effectively. I think what the compromise will look like in the end depends a lot on the target audience.

Third, there is interdisciplinarity versus focus. If we overemphasize the interdisciplinarity of data science and just see it as an overlapping of many different fields, it will be hard for it to establish itself as an independent discipline. However, if we narrow the scope too far, we may lose important defining characteristics of data science and end up with a subtopic within an existing discipline. At the moment, I find this the most difficult dimension to balance. As mentioned in a special issue of Nature (2015) (and also from my own personal experience), one of the greatest challenges in making interdisciplinary research work is to find a common language among all the different participants. This takes time and cannot be rushed. It seems that we are in the midst of the process of finding a language to define data science.

In summary, I believe there is no simple recipe for answering the questions of what to teach in data science and how to teach it. The competing forces described in the previous paragraphs need to be balanced depending on the needs of the specific target audience. However, this can be done in many different ways. For instance, illustrating the interdisciplinarity of data science could be achieved by inviting selected guest lecturers. In order to increase the technical depth, existing online tutorials for introducing basic programming skills could be run in a flipped classroom setting. Still, developing data science courses is not an easy feat, but I think the effort is worth it, as it helps someone working as a data scientist in defining the field more clearly. It forces a teacher to think deeper about the topic and its underlying structure while organizing the material. Like the editorial in Nature (2015), Berman et al. (2018) warn about rushing things by

standardizing the field too quickly. Instead, they suggest taking our time to "gain critical experience in how to best educate new generations of data scientists."

References

Berman F, Rutenbar R, Hailpern B, Christensen H, Davidson S, Estrin D, Franklin M, Martonosi M, Raghavan P, Stodden V, Szalay AS (2018) Realizing the Potential of Data Science. Communications of the ACM 61(4):67–72. ISSN: 0001-0782, DOI: 10.1145/3188721.

Brandl W (2012) Kant reloaded: Es mag ja in der Theorie richtig sein, taugt aber nicht für die Praxis. HiBiFo–Haushalt in Bildung und Forschung 1(4):5–6, Verlag Barbara Budrich.

Brodie ML (2019) What is Data Science? In: Braschler M, Stadelmann T, Stockinger K (eds.), Applied Data Science. Springer, pp. 101–131. DOI: 10.1007/978-3-030-11821-1.

Brunner RJ, Kim EJ (2016) Teaching Data Science. Procedia Computer Science 80:1947–1956, Elsevier. DOI: 10.1016/j.procs.2016.05.513.

Burrus D (2015) Teach a Man to Fish: Training vs. Education. The Huffington Post. URL: `https://www.huffpost.com/entry/teach-a-man-to-fish-training-vs-education_b_7553264`. [Online; accessed April 2021].

Dhar V (2013) Data Science and Prediction. Communications of the ACM 56(12):64–73, Association for Computing Machinery, New York, NY, USA. ISSN: 0001-0782, DOI: 10.1145/2500499.

Etzkowitz H, Peters LS (1991) Profiting from Knowledge: Organisational Innovations and the Evolution of Academic Norms. Minerva 29(2):133–166, Springer. ISSN: 002-6-4695, -15-7, DOI: 10.1007/BF01096406.

Grus J (2019) Data Science from Scratch. O'Reilly, Sebastopol, CA. ISBN: 978-1-491901-42-7.

Hicks SC, Irizarry RA (2018) A Guide to Teaching Data Science. The American Statistician 72(4):382–391, Taylor & Francis. DOI: 10.1080/00031305.2017.1356747.

Irizarry RA (2020) The Role of Academia in Data Science Education. Harvard Data Science Review 2(1). DOI: 10.1162/99608f92.dd363929.

Kross S, Guo PJ (2019) Practitioners Teaching Data Science in Industry and Academia: Expectations, Workflows, and Challenges. In: Proc. of the 2019 CHI Conf. on Human Factors in Computing Systems (CHI'19), pp. 1–14. DOI: 10.1145/3290605.3300493.

Lau S, Nolan D, Gonzalez J, Guo PJ (2022) How Computer Science and Statistics Instructors Approach Data Science Pedagogy Differently: Three Case Studies. In: Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 1, SIGCSE 2022, pp. 29–35. ISBN: 978-1-450390-70-5, DOI: 10.1145/3478431.3499384.

Meng XL (2019) Data Science: An Artificial Ecosystem. Harvard Data Science Review 1(1). DOI: 10.1162/99608f92.ba20f892.

Nature (2015) Mind Meld. Nature 525(1569):289–290. DOI: 10.1038/525289b.

O'Neil C, Schutt R (2014) Doing Data Science. O'Reilly, Sebastopol, California.

Opendata.ch (2011) Open Knowledge Switzerland. Published via: `https://opendata.ch/`. [Online; accessed April 2021].

Scherzinger S (2019) Build your own SQL-on-Hadoop Query Engine: A Report on a Term Project in a Master-level Database Course. SIGMOD Rec. 48(2):33–38. DOI: 10.1145/3377330.3377336.

Stodden V (2020) The Data Science Life Cycle: A Disciplined Approach to Advancing Data Science as a Science. Communications of the ACM 63(7):58–66. ISSN: 0001-0782, DOI: 10.1145/3360646.

Tukey J (1962) The Future of Data Analysis. Annals of Mathematical Statistics 33:1–67. DOI: 10.1214/aoms/1177704711.

Wild CJ, Pfannkuch M (1999) Statistical Thinking in Empirical Enquiry. International Statistical Review 67(3):223–248. DOI: 10.1111/j.1751-5823.1999.tb00442.x

Wing JM (2019) The Data Life Cycle. Harvard Data Science Review 1(1). DOI: 10.1162/99608f92.e26845b4.