Harvard Data Science Review • Issue 3.2, Spring 2021

A Decision Tree for Introductory Data Science

Allen B. Downey^{1,2}

¹Department of Computer Science, John A. Paulson School of Engineering and Applied Sciences,
Harvard University, Cambridge, Massachusetts, United States of America,

²Olin College of Engineering, Needham, Massachusetts, United States of America

Published on: Jun 07, 2021

DOI: <https://doi.org/10.1162/99608f92.811df726>

License: [Creative Commons Attribution 4.0 International License \(CC-BY 4.0\)](https://creativecommons.org/licenses/by/4.0/)

Three Decisions

If you want to teach data science to undergraduates, you have some decisions to make. I will present some of these decisions, explain how my collaborators and I navigated them, and compare our choices to the design of Data 8 as described by its designers (Adhikari et al., 2021).

The first decision is whether to use statistical software—like Tableau, SPSS, or SAS—or a general-purpose programming language—like Python or R. The advantage of statistical software is that students can do more, sooner, and with less frustration. But it takes time to learn any tool, and students might end up locked into a technology that is in decline (Muenchen, 2021). With general purpose programming languages, the skills students develop are more broadly useful. Also, programming makes it easier to work with messy data and work on open-ended projects that engage with real-world problems.

If you decide to use a general-purpose programming language, the next decision is which one. Popular choices include R, Python, MATLAB, and Java. All but Java are good choices for beginners, with simple syntax and good interactive development environments (IDEs). All include libraries with enough statistical and visualization tools for a variety of courses. And they are all widely used in academics and industry. R and Python are free software; MATLAB is proprietary, but Octave is a free alternative that is compatible enough for an introductory class and more; the status of Java is complicated, but for classroom use it is effectively free software.

A third decision is what prerequisites to require in mathematics, statistics, and programming. In one model, students learn foundational skills within academic disciplines and then apply them in advanced, interdisciplinary classes like data science. At the other extreme is a data-first approach that starts with real-world applications and develops skills on demand. Somewhere in the middle is the mash-up, which combines elements from existing computer science and statistics classes.

These are the decisions I considered in 2014 when I started teaching data science at Olin College and in 2019 when I collaborated with faculty at Harvard University to design an introductory data science class called DS10.

Elements of Data Science

Like the designers of Data 8, my collaborators and I rejected statistical software in favor of general-purpose programming languages in order to "empower [students] to ask and answer their own questions" (Adhikari et al., 2021). We chose Python over R on the theory that it will be more broadly useful to students in the future, especially students in majors outside statistics. And we chose to minimize the prerequisites: at Harvard, DS10 has no prerequisites at all; at Olin, data science has a programming prerequisite that will be dropped for the

next offering. Because great minds think alike, or possibly because fools seldom differ, Data 8 and our classes ended up in the same octant of the design space.

One place where we differ is in the use of custom libraries. Data 8 uses a Python module to "ensure that students can carry out a wide variety of table manipulation and data visualization operations using only a core set of the most fundamental programming language concepts" (Adhikari et al., 2021). We agree that it is valuable to give students a small set of versatile features, but based on student feedback, we (mostly) avoid custom libraries in favor of the standard SciPy stack, including a curated set of functions from NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, and Statsmodels. These libraries can be overwhelming at first, but students tell us they value exposure to tools they will use in the future.

Like Data 8, we use Jupyter notebooks extensively, although we do not run our own JupyterHub. Instead, our notebooks are designed to run on Google Colab, which is a free Jupyter environment that runs on Google servers. Students can copy notebooks from our repositories on GitHub (among other sources), run the code, work on exercises, and save the modified notebooks in their Google Drive (among other destinations). Instructors can read and run student notebooks, modify them, and add comments.

The curriculum we have developed is available as a collection of modules. The core of the curriculum is the online text *Elements of Data Science* (Downey, 2021c), which includes these three modules:

1. From Python to Pandas: Six chapters that introduce lists, dictionaries, NumPy arrays, and Pandas DataFrames; data types including integers, floating-point numbers, times and dates, latitude and longitude, strings, and text files; and line and bar plots.
2. Exploratory Data Analysis: Four chapters that introduce data cleaning and validation; visualization of distributions with histograms, discrete PMFs and CDFs, and kernel density estimation (KDE); visualization of relationships using scatter plots, box plots, and violin plots; and quantification of relationships using correlation and regression (simple, multiple, and logistic). Examples use data from a variety of sources, including the National Survey of Family Growth (NSFG) and the Behavior Risk Factor Surveillance Survey (BRFSS).
3. Computational Inference: Three chapters that use resampling, bootstrapping, and permutation to introduce statistical inference. Examples use data from the NSFG and BRFSS again, and from the General Social Survey (GSS).

Each chapter is a Jupyter Notebook that includes text, executable code, and exercises. Additional homework exercises are available.

This list of topics might sound ambitious for a one-semester class with no prerequisites. It is possible because we focus on what these methods do and how to use libraries that implement them. We postpone theory and proofs, and present most ideas using Python code rather than mathematical notation. We do not require or teach

any calculus or linear algebra. *Elements of Data Science* includes two case studies, although not all offerings of the class use both.

Political Alignment Case Study

Using data from the GSS again, we introduce cross-tabulation, pivot tables, and the Pandas group-by operation. Students use these tools to explore changing opinions on a variety of topics among survey respondents in the United States. This case study provides scaffolding for a project in the first half of the semester. Students choose one of about 120 survey questions and see how responses have changed over time and how these changes relate to political alignment (conservative, moderate, or liberal). In our courses, students write short reports that describe their findings and get feedback on their data visualization and written communication (Downey, 2021d).

Recidivism Case Study

This case study is based on a well-known paper, "Machine Bias" (Angwin et al.), which was published in 2016. It relates to COMPAS, a statistical tool used in the criminal justice system to assess the risk that a defendant will commit another crime if released. The ProPublica article concludes that COMPAS is unfair to Black defendants because they are more likely to be misclassified as high risk. A response article in the *Washington Post* (Corbett-Davies et al., 2016) suggests that "It's actually not that clear." Using the data from the original article, this case study defines the metrics used to evaluate binary classifiers, explains the challenges of defining algorithmic fairness, and provides an engaging starting place for discussions of the context, ethics, and social impact of data science (Downey, 2021e).

In one of our classes, we also include an introduction to probability with a focus on Bayes' Theorem, using *Bite Size Bayes* (Downey, 2021b). Future offerings might include an introduction to SQL using *Astronomical Data in Python* (Downey, 2021a). All of these resources are freely available under a Creative Commons license.

Let me close with two pleas for the future of the data science curriculum.

Push for a Broad Definition

When people hear data science, they often think of big data, machine learning, and applications like advertising, quantitative finance, and sports analytics. But the data science curriculum should not be defined by a particular set of tools or applications. Data science is the use of data to answer questions and guide decision-making; we should design the curriculum accordingly. The broad definition of data science has been a recurring theme in *HDSR*, going back to the first issue (Meng, 2019).

Design for Students

Like the designers of Data 8, we think the data science curriculum should not be "a mash-up of computer science topics and statistics topics," but should "focus on specific problems and bring computation and

statistics to bear jointly in the solution." And we agree that "students come to our curriculum with their own questions and passions" (Adhikari et al., 2021). The curriculum we design should serve their needs and interests, feed their passions, and give them tools and skills to answer their questions.

Disclosure Statement

Allen B. Downey has no financial or non-financial disclosures to share for this article.

References

- Adhikari, A., DeNero, J., & Jordan, M. I. (2021). Interleaving computation and inferential thinking: Data science for undergraduates at Berkeley. *Harvard Data Science Review*, 3(2).
<https://doi.org/10.1162/99608f92.cb0fa8d2>
- Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016, May 23). Machine bias. *ProPublica*.
<https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>
- Corbett-Davies, S., Pierson, E., Feller, A., & Goel, S. (2016, October 17). A computer program used for bail and sentencing decisions was labeled biased against blacks. It's actually not that clear. *Washington Post*.
<https://www.washingtonpost.com/news/monkey-cage/wp/2016/10/17/can-an-algorithm-be-racist-our-analysis-is-more-cautious-than-propublicas/>
- Downey, A.B. (2021a). *Astronomical data in Python*. GitHub. <https://allendowney.github.io/AstronomicalData>
- Downey, A. B. (2021b). *Bite size Bayes*. GitHub. <https://allendowney.github.io/BiteSizeBayes>
- Downey, A. B. (2021c). *Elements of data science*. GitHub.
<https://allendowney.github.io/ElementsOfDataScience>
- Downey, A. B. (2021d). *Political alignment case study science*. GitHub.
<https://allendowney.github.io/PoliticalAlignmentCaseStudy>
- Downey, A. B. (2021e). *Recidivism case study science*. GitHub.
<https://allendowney.github.io/RecidivismCaseStudy>
- Meng, X.-L. (2019). Data science: An artificial ecosystem. *Harvard Data Science Review*, 1(1).
<https://doi.org/10.1162/99608f92.ba20f892>
- Muenchen, R. A. (2021). *The popularity of data science software*. r4stats.com.
<http://r4stats.com/articles/popularity>
-

©2021 Allen B. Downey. This article is licensed under a Creative Commons Attribution (CC BY 4.0) [International license](https://creativecommons.org/licenses/by/4.0/), except where otherwise indicated with respect to particular material included in the article.