Computing in the Statistics and Data Science Curriculum

Mine Çetinkaya-Rundel

Alex Reinhart

CAUSE/Journal of Statistics and Data Science Education webinar series

Upcoming webinars:

- Teacher education curriculum materials that develop statistical knowledge for teaching (Tuesday, February 9th, 2:00-3:00pm EST)
- Bayesian methods in the statistics curriculum (Tuesday, February 23rd, 4:00-5:00pm EST)
- “Playing the whole game” and “Data scraping for fun and profit” (Tuesday, March 23rd, 4:00-5:00pm EST)
- Signup at https://www.causeweb.org/cause/webinars

Consortium for the Advancement of Undergraduate Statistics Education

https://www.causeweb.org/cause

Breakout deadlines February 1, 2021

Mine Çetinkaya-Rundel

Senior Lecturer
School of Mathematics
University of Edinburgh

Data Scientist and Professional Educator
RStudio

Associate Professor of the Practice
Department of Statistical Science
Duke University

Alex Reinhart

Assistant Teaching Professor
Statistics & Data Science
Carnegie Mellon University

But first, a few words from the special issue co-guest editor

Johanna Hardin
Department of Mathematics & Statistics
Pomona College

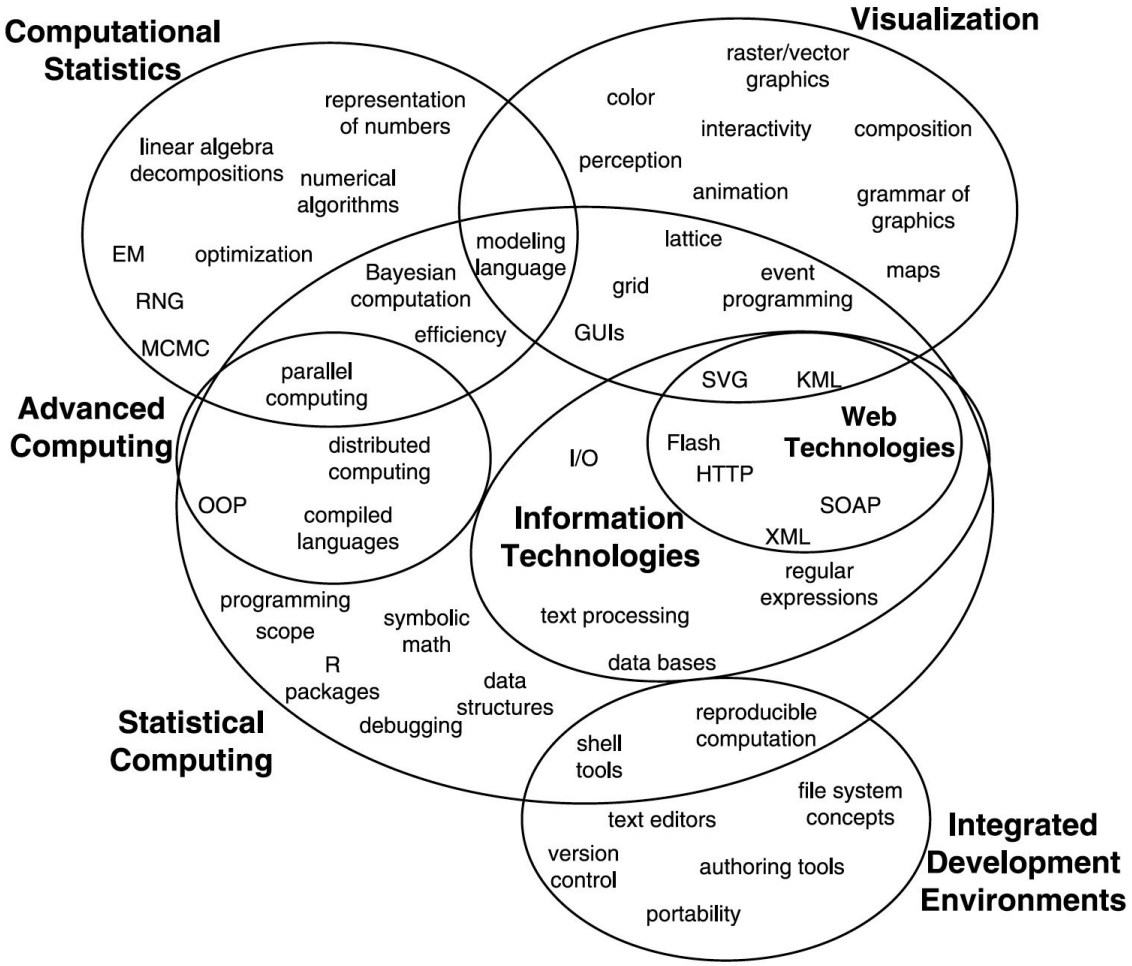
Computing in the Curriculum circa 2010

Nolan and Temple Lang posed the following questions in their paper "Computing in the Statistics Curriculum" (TAS, 2010):

- Computational literacy and programming are as **fundamental to statistical practice** and research as mathematics.
- Our field needs to **define statistical computing more broadly** to include advancements in modern computing, beyond traditional numerical algorithms.
- Information technologies are increasingly important and should be added to the curriculum, as should the ability to **reason about computational resources**, work with large datasets, and perform computationally intensive tasks.

Computing in the Curriculum circa 2010

Nolan and Temple Lang posed the following questions in their paper "Computing in the Statistics Curriculum" (TAS, 2010):



Special issue on Computing in the Curriculum

- Spurred by 10 year anniversary of the publication of Nolan and Temple Lang's paper
- Call for papers in 2019
- Reviews and revisions in 2020
- Publication scheduled for early 2021 (pandemic delays)

Papers in the special issue

- Editorial
- Commentary from Nolan and Temple Lang
- 14 papers on a variety of topics:
  - Creative teaching structures
  - Novel skills and habits
  - Computational thinking
- https://www.tandfonline.com/doi/full/10.1080/10691898.2020.1870416 for the editorial and link to individual papers

Creative Teaching Structures

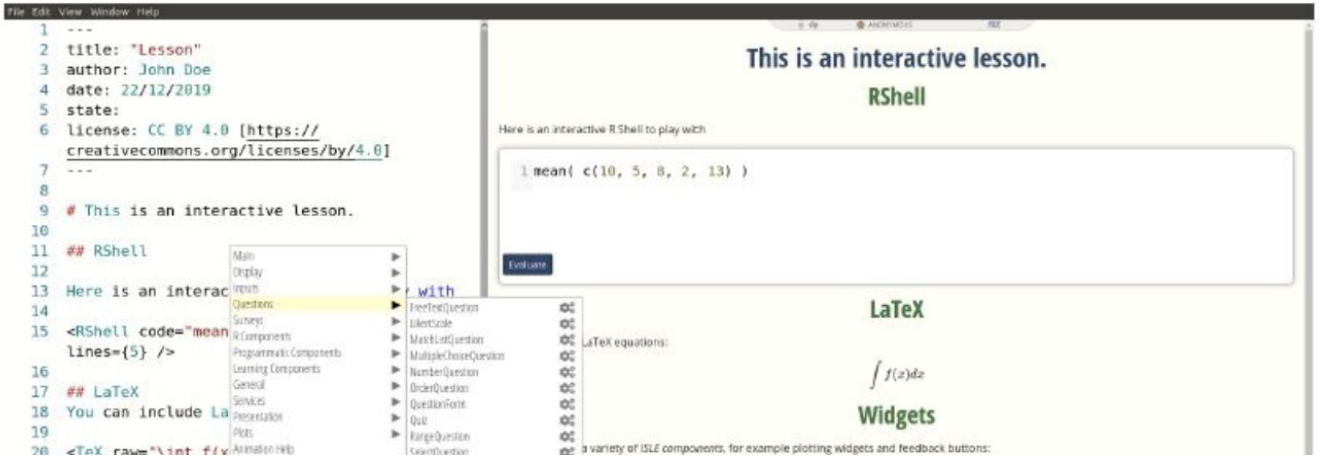
Easy-to-Use Cloud Computing for Teaching Data Science

Kim & Henke

	Tool	Function	Details
Step 1	Jupyter Notebooks	Document	Build teaching material.
Step 2	GitHub	Online Repository	Store notebooks online.
Step 3	Binder	Cloud Service	Deliver in the cloud.

Teaching Statistical Concepts and Modern Data Analysis with a Computing-Integrated Learning Environment (ISLE)

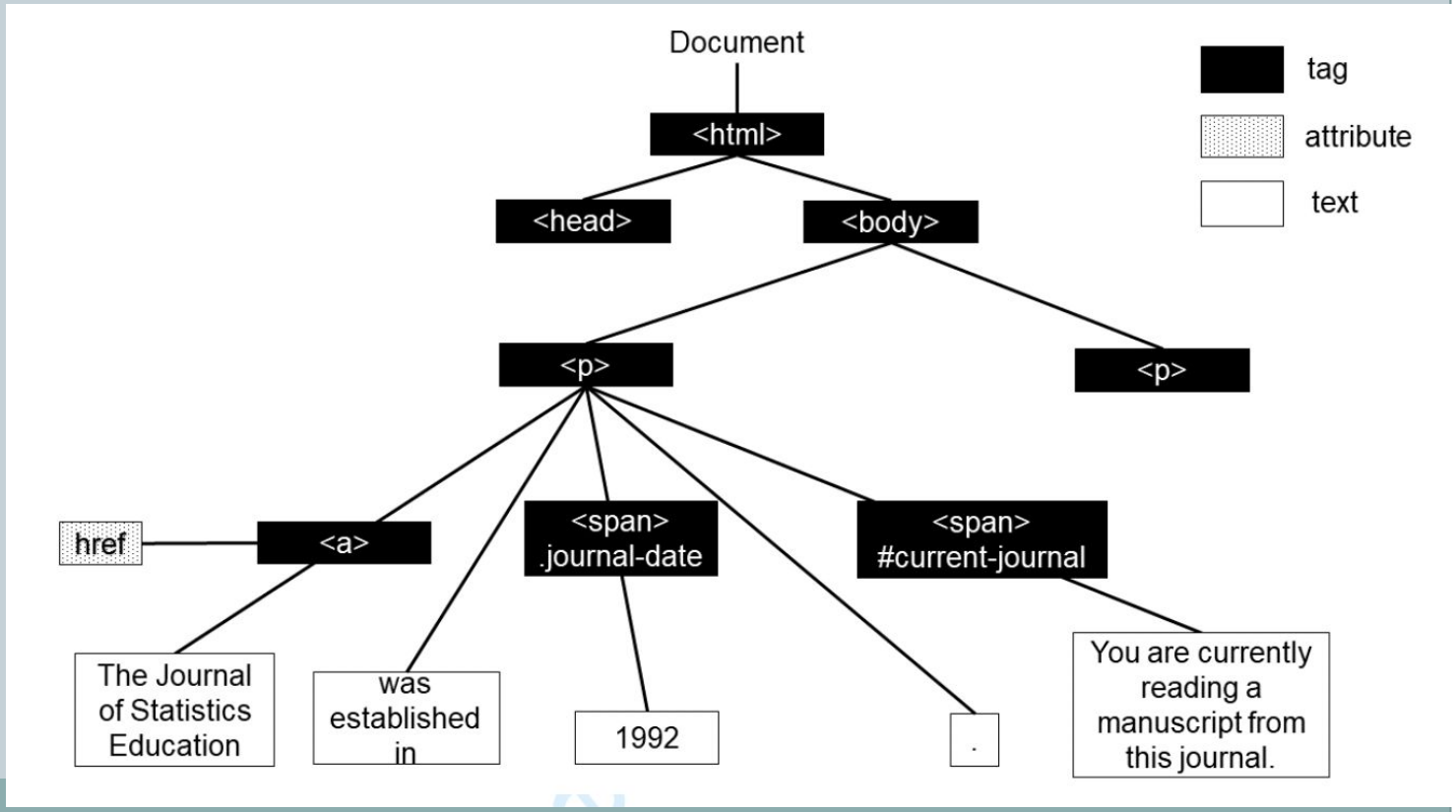
Burckhardt, Nugent, & Genovese



Novel and technical data science skills and habits

Web Scraping in the Statistics and Data Science Curriculum: Challenges and Opportunities

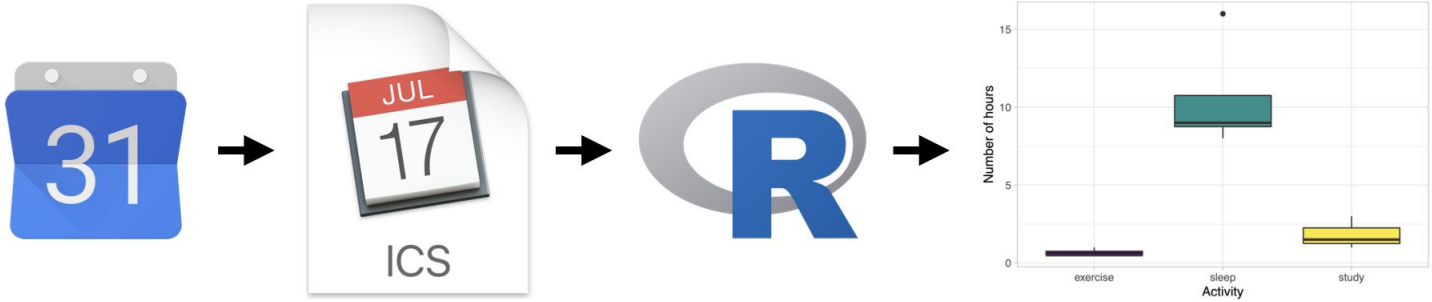
Dogucu & Çentinkaya-Rundel



Novel and technical data science skills and habits

Kim & Hardin

"Playing the whole game":
A data collection & analysis exercise with Google Calendar



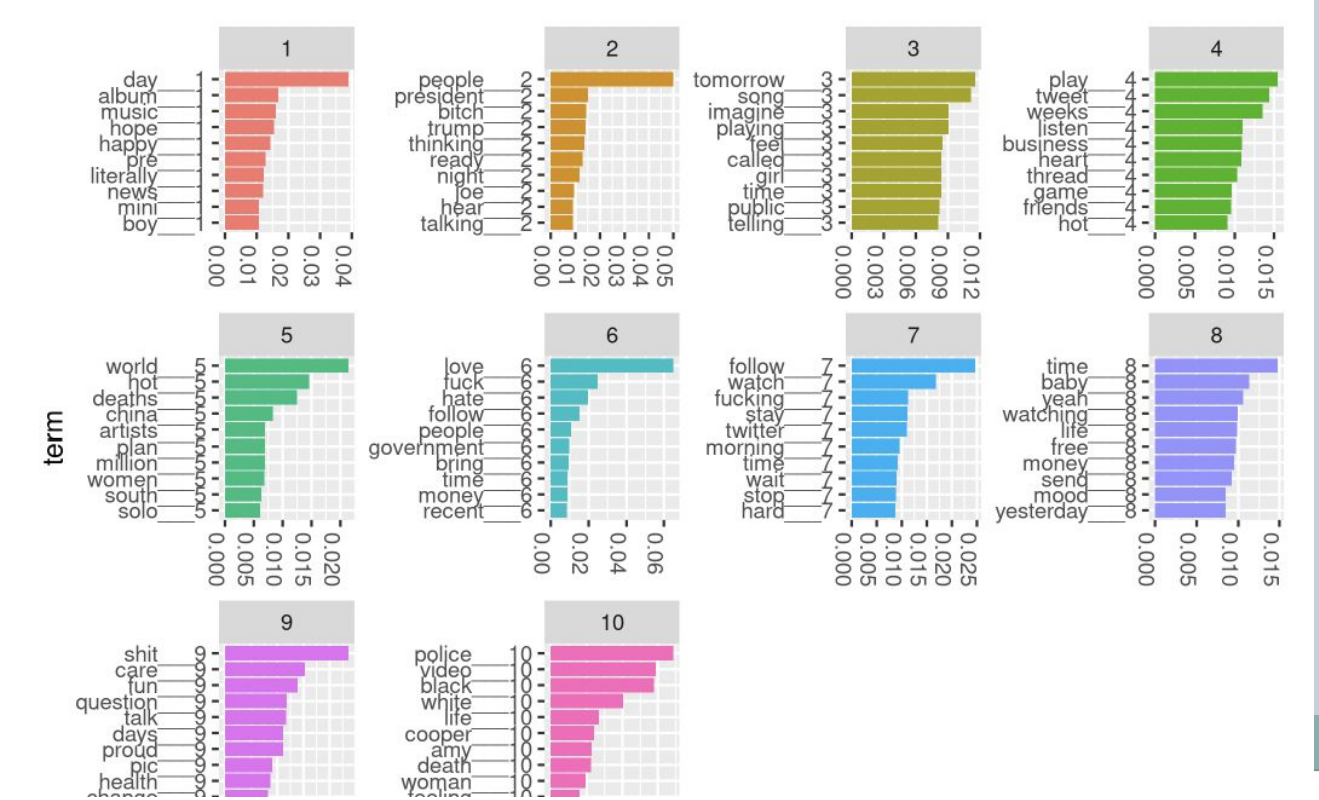
1. **Log activities in Google Calendar**
2. **Export to `.ics` file format**
3. **Import to R using `ical` package**
4. **Analyze**

Iterate as needed!

Novel and technical data science skills and habits

What is happening on Twitter?
A framework for student research projects with tweets

Boehm & Hanlon

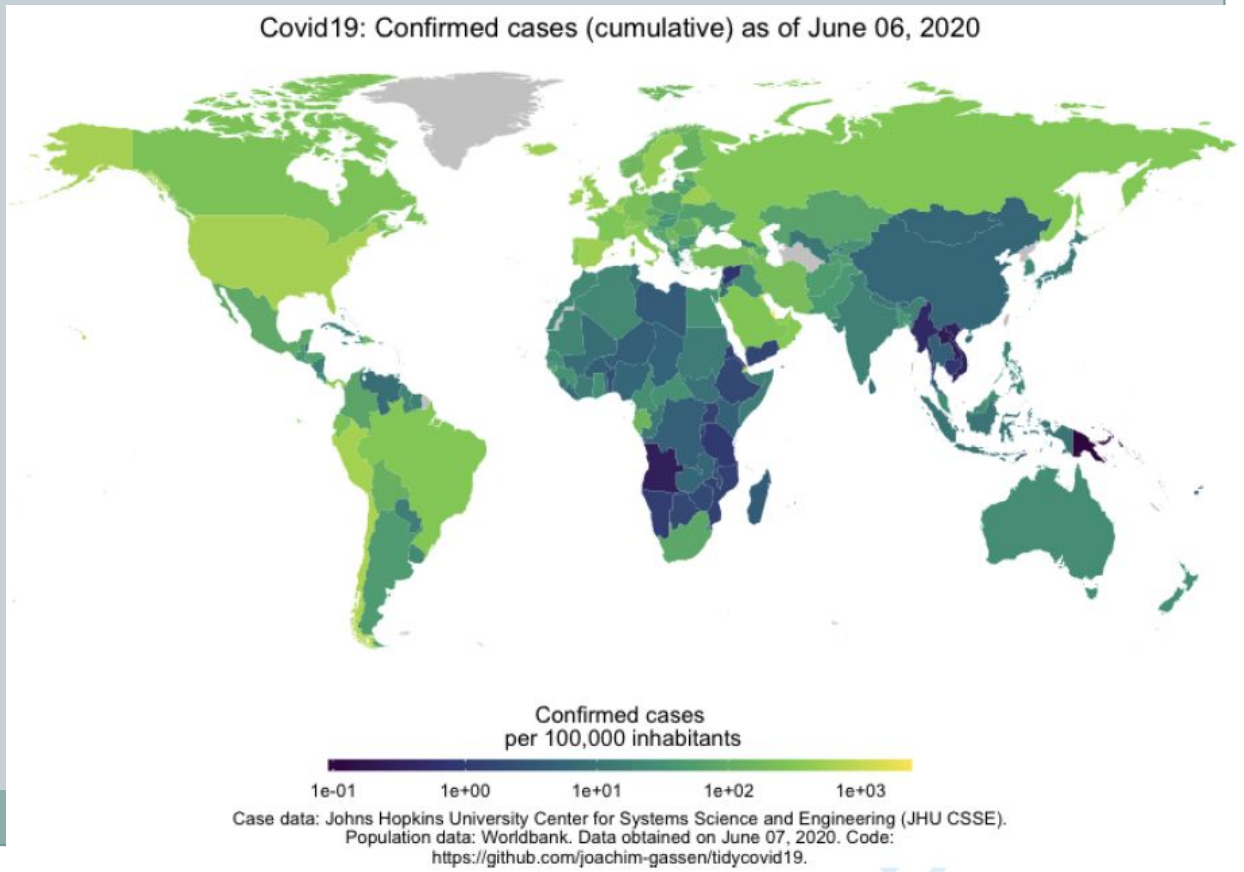


Novel and technical data science skills and habits

Computational Skills for Multivariable Thinking in Introductory Statistics

Adams, Baller, Jonas, Joseph, & Cummiskey

"Proficiency in a statistical programming language facilitates the development of multivariable thinking by giving students tools to investigate complex data on their own."



Teaching Computational Thinking

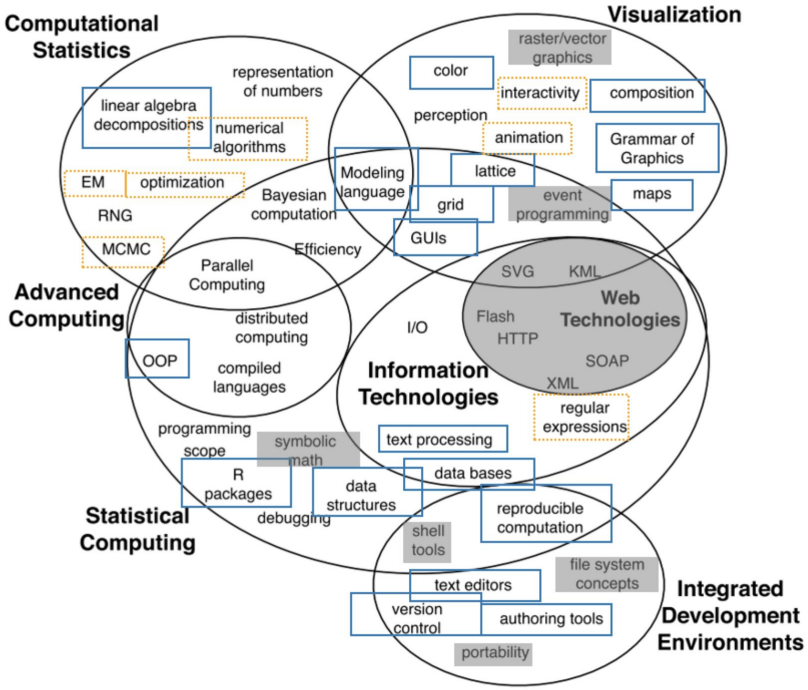
Data Science in 2020: Computing, Curricula, and Challenges for the Next 10 Years

Schwab-McCoy, Baker, & Gasper

covered > 75%

covered > 50%

not asked



The nature of doing computation in the classroom requires students to be familiar with concepts like debugging, code formatting, and reproducible programming. However, **are we truly developing students who understand** how R, Python, or any of the other computing languages used to teach data science "think"?

Teaching Computational Thinking

Teaching Creative and Practical Data Science at Scale

Donoghue, Voytek, & Ellis

Key skills for the budding data scientist include how to explore and **debug** both code and data issues, and how to decide on a path forward when what to do next is unclear... We seek to **explicitly instruct students on the data-centric debugging strategies** employed when analyzing data by running sessions on debugging and how to proceed if one's code is not working.

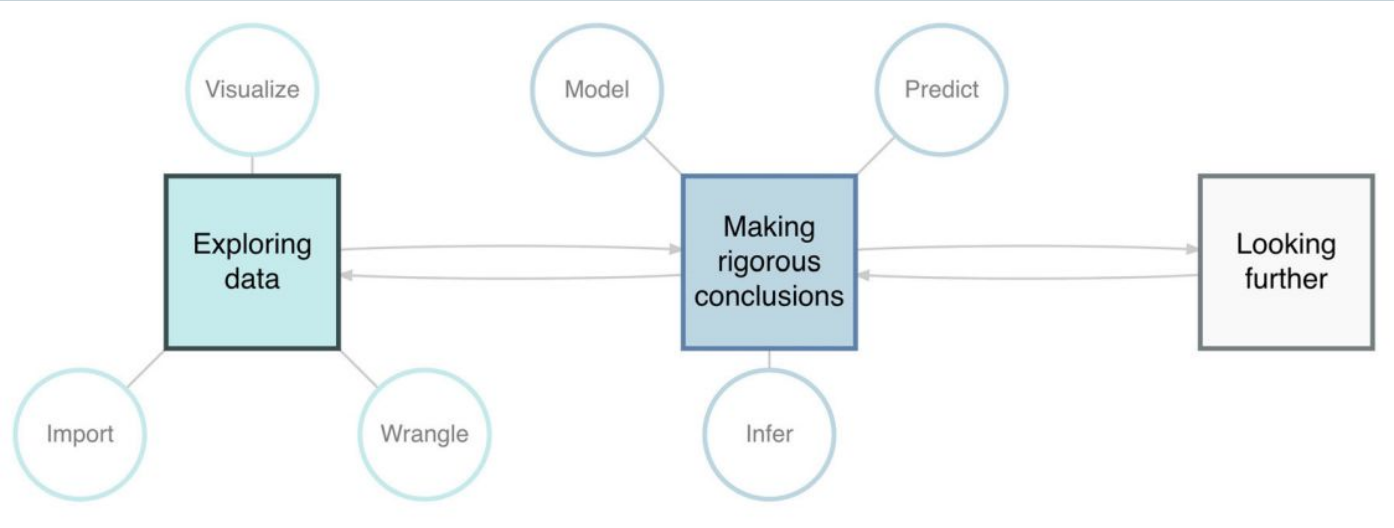
Designing Data Science Workshops for Data-Intensive Environmental Science Research

Theobold, Hancock, Mannheimer

The skills necessary for students to engage in [the data analysis] cycle may include general programming concepts such as **looping, user-defined functions, or conditional statements**.

A fresh look at introductory data science

Case study of an introductory data science course, designed for undergraduates



Mine Çetinkaya-Rundel & Victoria Ellison (2020) A Fresh Look at Introductory Data Science, Journal of Statistics Education, DOI: 10.1080/10691898.2020.1804497

A fresh look at introductory data science

We tackled the “what is data science?” question empirically by surveying contents of data science courses

Table 1. Summary of programming languages used in each course and the estimated breakdown of percent of class time spent on various course components.

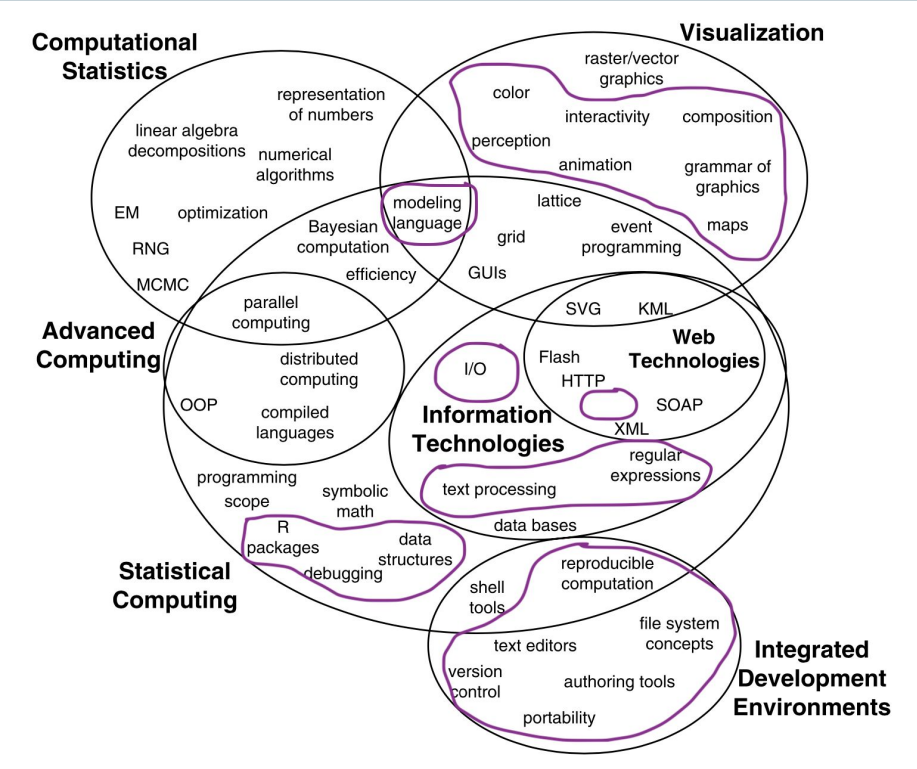
	Duke	Berkeley	Cambridge	Smith	Stanford
Programming language	R	Python	Pseudocode	R, SQL	R
Data visualization	15%	5%	0%	32%	10%
Data wrangling	10%	15%	0%	36%	0%
Other EDA	10%	5%	0%	12%	10%
Inference	20%	30%	25%	0%	50%
Modeling	25%	20%	35%	0%	20%
Programming principles	10%	10%	0%	5%	0%
Mathematical foundations/theory	5%	5%	35%	0%	0%
Communication	5%	5%	0%	10%	10%
Ethics	0%	5%	5%	5%	0%

A fresh look at introductory data science

For each unit:

- Learning goals
  - + justification, based on literature, for why we chose those particular learning goals
- Case study examples
- Pacing of topics

All materials are open source (datasciencebox.org)



A fresh look at introductory data science

We also describe

- Pedagogical choices
- Computing infrastructure
- Computing competencies students acquire
- Assessment
- Impact of course on undergraduate curriculum

A fresh look at introductory data science

Why “fresh”?

- Introduction to programming via visualisation
- Modern computing in R (tidyverse and more)
- Reproducible workflows from day one
- Focus on skills around the data science life cycle
- Modeling over inference
- Built-in flexibility for evolution of topics
- Emphasis on collaborative work

Computing in the graduate curriculum

- What should computing look like in the *graduate* statistics curriculum?
- We (Reinhart & Genovese) argue that statisticians are often called on to deliver statistical *products*, not analyses
- This requires a mastery of software engineering principles, not just the syntax of R or Python
- Topics, notes at https://36-750.github.io/

Computing in the graduate curriculum

- We describe a computing course for first-semester PhD and MS students in statistics & data science
- Goal: Give them experience designing and maintaining complex software
- Feedback through revision and mastery grading process
- Content covers design, unit testing, object-oriented and functional programming, databases…

Back to the Nolan and Temple Lang questions

1. When they graduate, what ought our students be able to do computationally, and are we preparing them adequately in this regard?
2. Do we provide students the essential skills needed to engage in statistical problem solving and keep abreast of new technologies as they evolve?
3. Do our students build the confidence needed to overcome computational challenges to, for example, reliably design and run a synthetic experiment or carry out a comprehensive data analysis?
4. Overall, are we doing a good job preparing students who are ready to engage in and succeed at statistical inquiry?