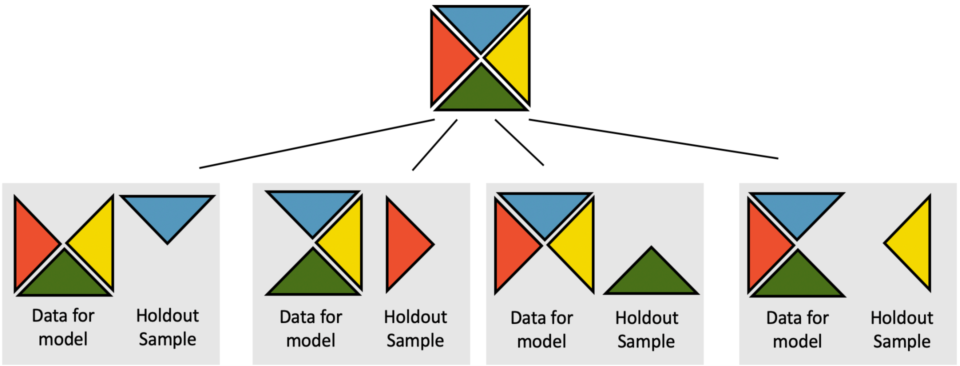
the value of computational thinking in statistics education

Jo Hardin
Pomona College



@jo_hardin47
Github: hardin47
jo.hardin@pomona.edu

Professional Guidelines

- GAISE - Guidelines for Assessment and Instruction in Statistics Education (2016)
    - It is important to view the use of technology **not just as a way to generate statistical output but as a way to explore conceptual ideas** and enhance student learning.
    - Technology tools should also be used to help students visualize concepts and **develop an understanding of abstract ideas** by simulations.
- ASA Curriculum Guidelines for Undergraduate Programs in Statistical Science (2014)
    - They should be able to program in a higher-level language, **to think algorithmically**, to use simulation-based statistical techniques, and to undertake simulation studies.
    - This capacity includes the ability to write functions and **use control flow** in a variety of languages.
    - The capacity to undertake and interpret simulation studies as a way to **complement analytic understanding** and/or check results will be increasingly useful in the workplace.

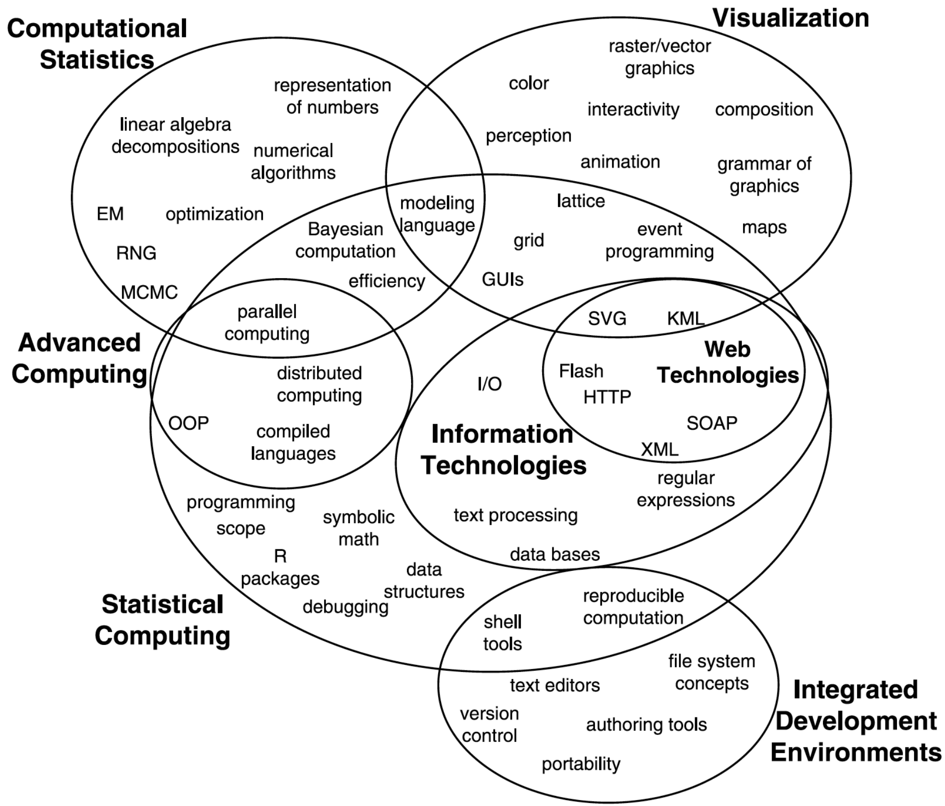
Computing in the Statistics Curricula

Nolan & Temple Lang (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 Statistics Curricula

Nolan & Temple Lang (2010)



What now / next?

- Special issue of Journal of Statistics Education on:

  Computing in the Statistics and Data Science Curriculum
  (call in early 2019)

- Name of ASA Section (2019):
  Statistics Education ->
  Statistics and Data Science Education

- Name of JSE (2021):
  JSE ->
  Journal of Statistics and Data Science Education

Journal of Statistics and Data Science Education

Special Issue on Integrating computing in the statistics and data science curriculum

January 2021

editors: Jo Hardin & Nick Horton

- Creative teaching structures
- Novel and technical data science skills and habits
- Teaching computational thinking

Creative teaching structures

Easy-to-Use Cloud Computing for Teaching Data Science

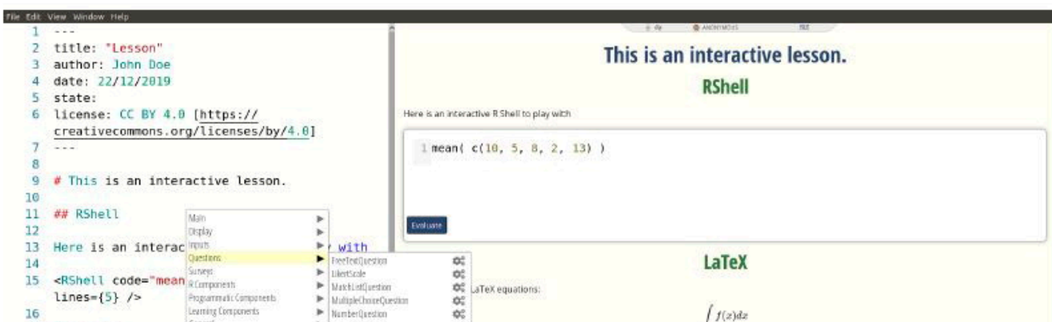
Kim & Henke

	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.

Table 1: A summary of the tools and their uses for creating and delivering executable Jupyter notebooks.

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

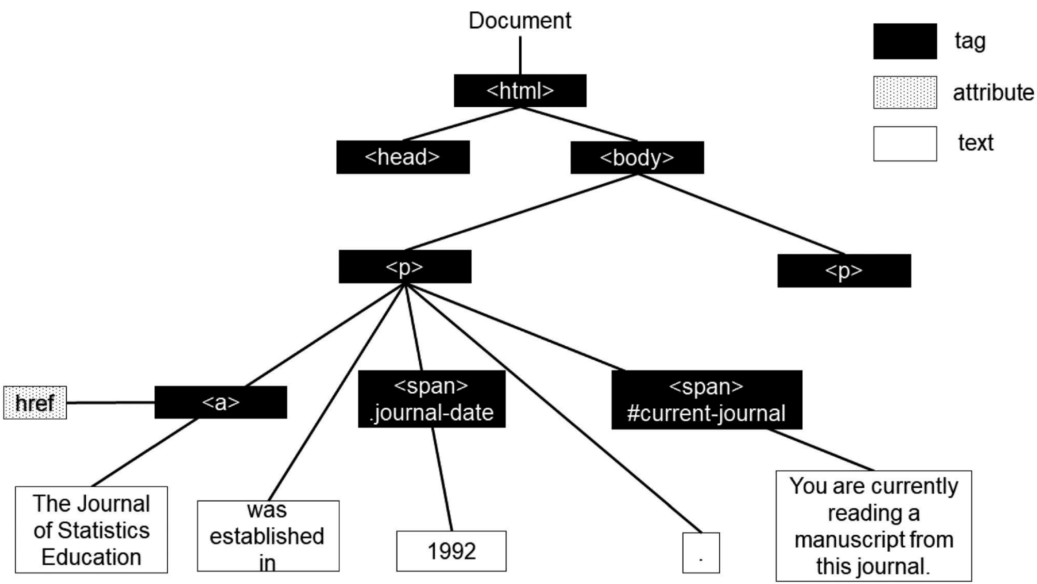
Burckhardt, Nugent, & Genovese



Novel and technical data science skills and habits

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

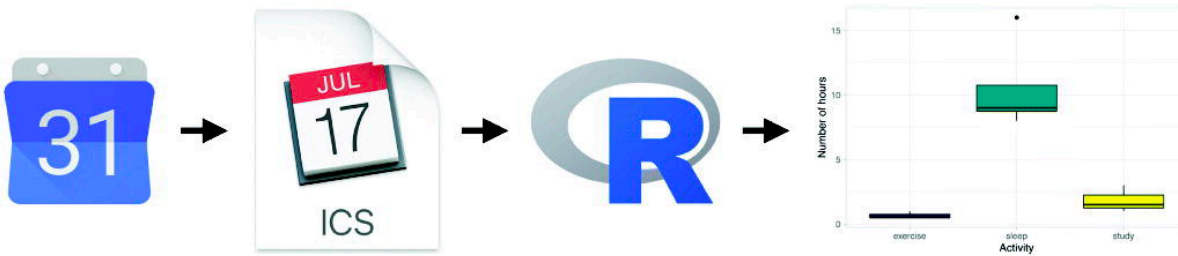
Dogucu & Çentinkaya-Rundel



Novel and technical data science skills and habits

Kim & Hardin

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



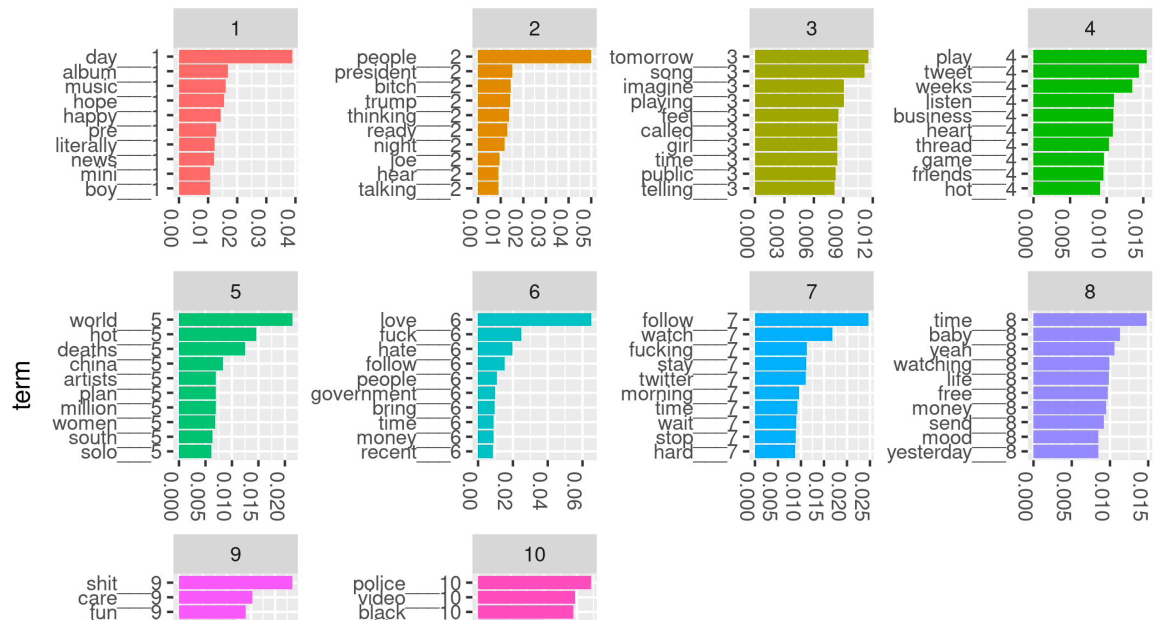
1. **Log activities in Google Calendar**
2. **Export to `.ics` file format**
3. **Import to R using `ical` package**
4. **Analyze**

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

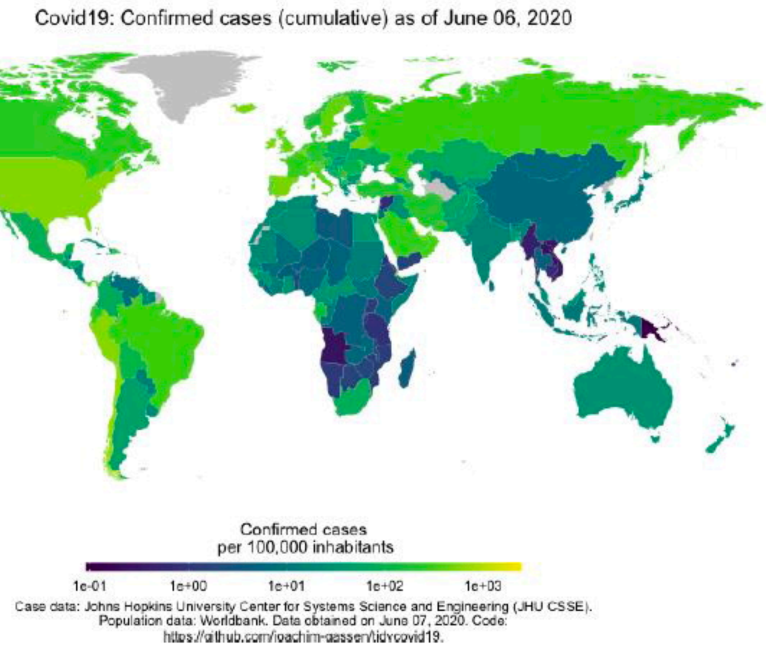


Novel and technical data science skills and habits

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.”



Changing the way we think:

Computer science's contribution to biology goes beyond the ability to search through vast amounts of sequence data looking for patterns. The hope is that data structures and algorithms—our <u>computational abstractions</u> and methods—<u>can represent the structure of proteins in ways that elucidate their function</u>. Computational biology is changing the way biologists think.

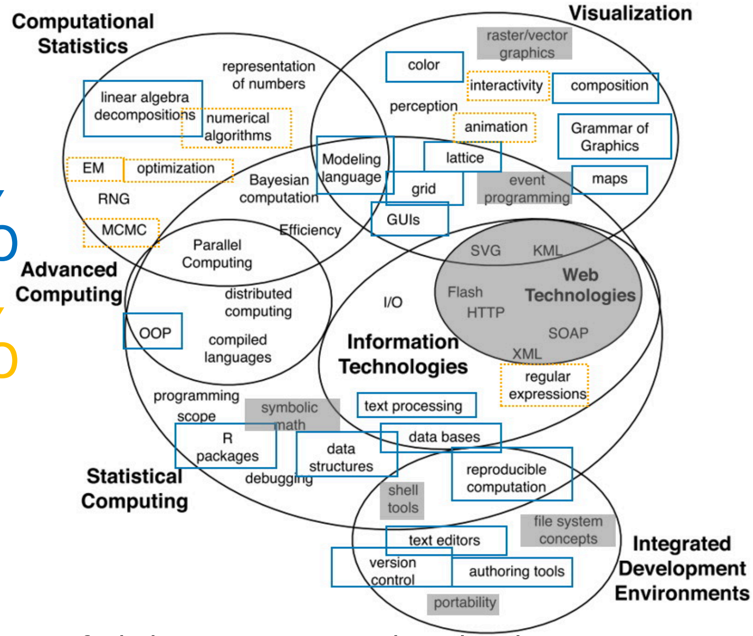
Wing, "Computational Thinking", Communications of the ACM, 2006.

Teaching computational thinking

covered > 75%

covered > 50%

not asked



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

Schwab-McCoy, Baker, & Gasper

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

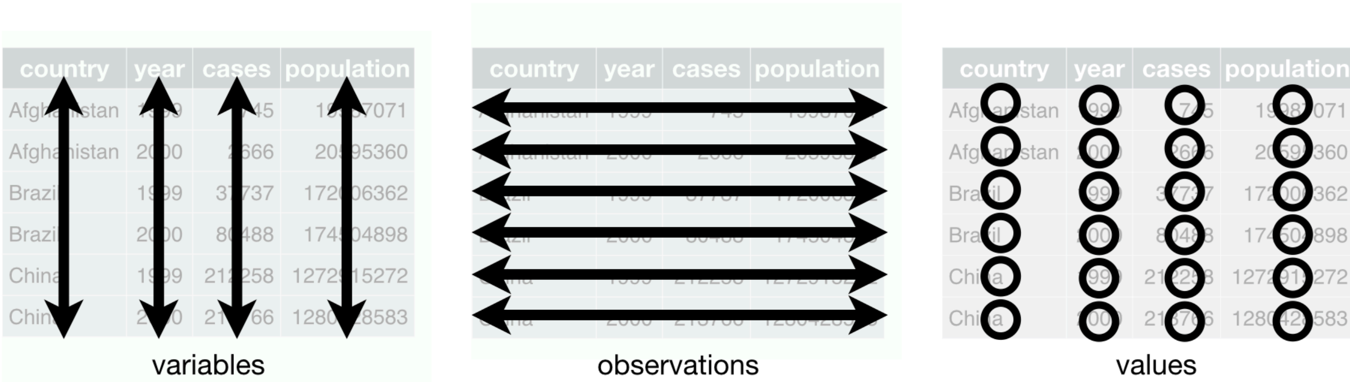
Theobold, Hancock, Mannheimer

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.

Teaching computational thinking in practice

- What does it all mean?
- How do we do this in the classroom?
- What are best practices?

Tidy Data



https://towardsdatascience.com/what-is-tidy-data-d58bb9ad2458

the tidyverse

The tidyverse is an opinionated collection of R packages… [sharing] an underlying design philosophy, grammar, and data structures.

https://www.tidyverse.org/

Tidy Processing

$$\mathrm{f}(x) = log_{10}(\sqrt{x})$$

rounded, to 2 digits

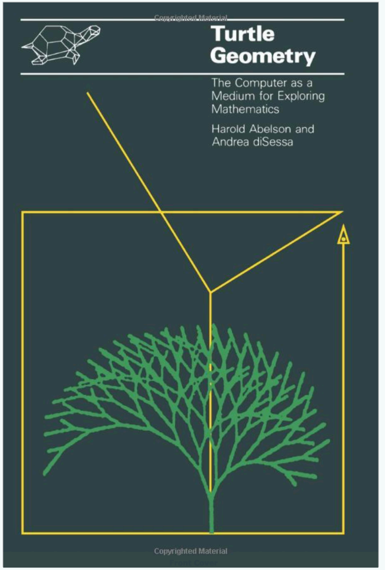
```
> x <- c(1:10)
> x
 [1]  1  2  3  4  5  6  7  8  9 10
>
> round(log(sqrt(x), base = 10), digits = 2)
 [1] 0.00 0.15 0.24 0.30 0.35 0.39 0.42 0.45 0.48 0.50
>
> x %>% sqrt() %>% log(base = 10) %>% round(digits = 2)
 [1] 0.00 0.15 0.24 0.30 0.35 0.39 0.42 0.45 0.48 0.50
```

not tidy

tidy

Computing & Math

- **Turtle Geometry** is a college-level math text … exploring mathematical properties visually via a simple programming language.



Programs must be written for people to read, and only incidentally for machines to execute.

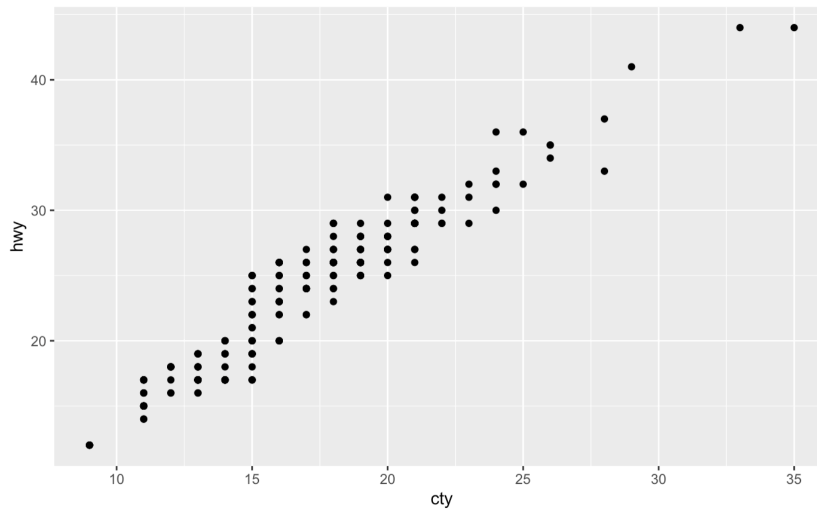
Abelson & Sussman, "Structure and Interpretation of Computer Programs", preface to the first edition

Tidy Plots

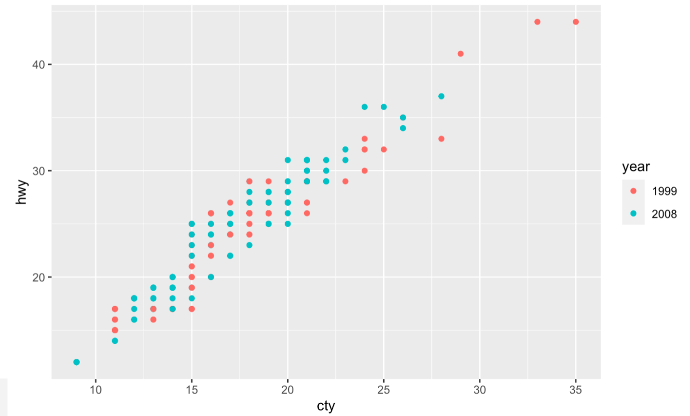
```r
ggplot(data = mpg)
```



```r
ggplot(data = mpg) +
  geom_point(aes(x = cty, y = hwy))
```

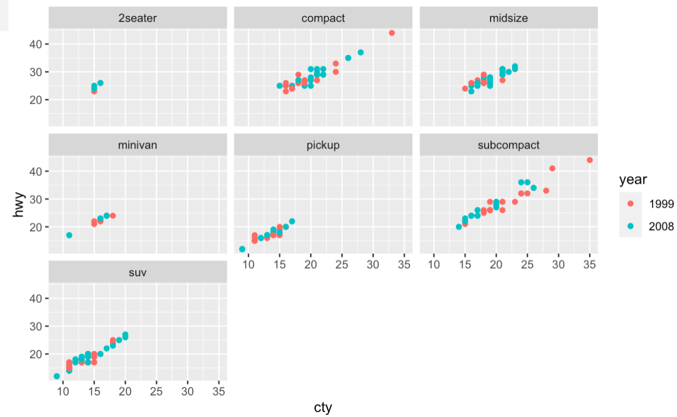


Tidy Plots



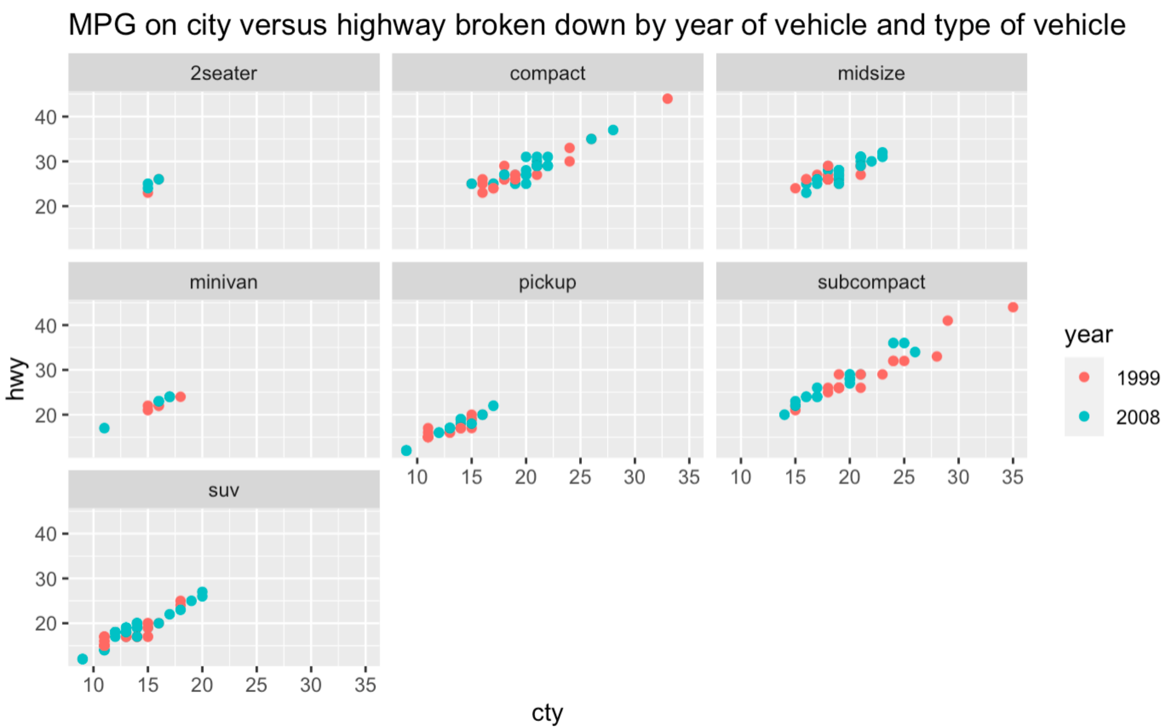

```r
ggplot(data = mpg) +
  geom_point(aes(x = cty, y = hwy, color = year))
```

```r
ggplot(data = mpg) +
  geom_point(aes(x = cty, y = hwy, color = year)) +
  facet_wrap(~class)
```

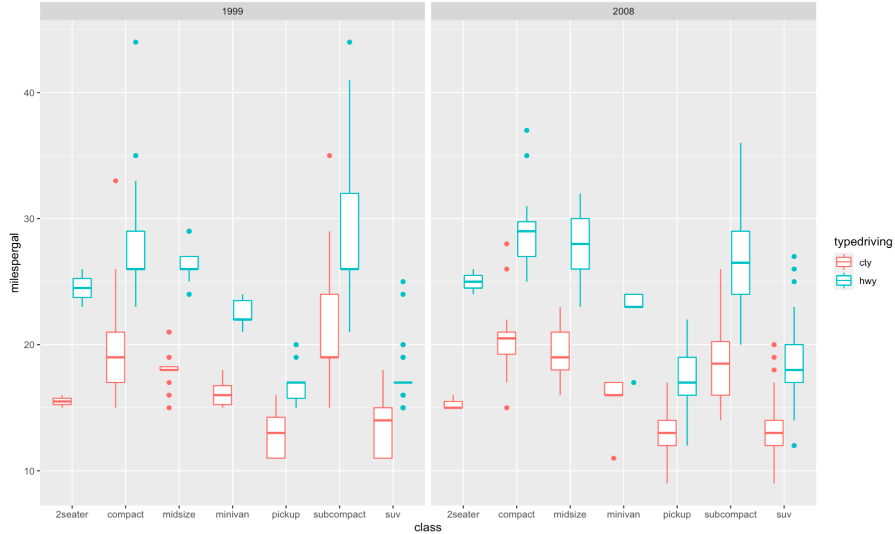


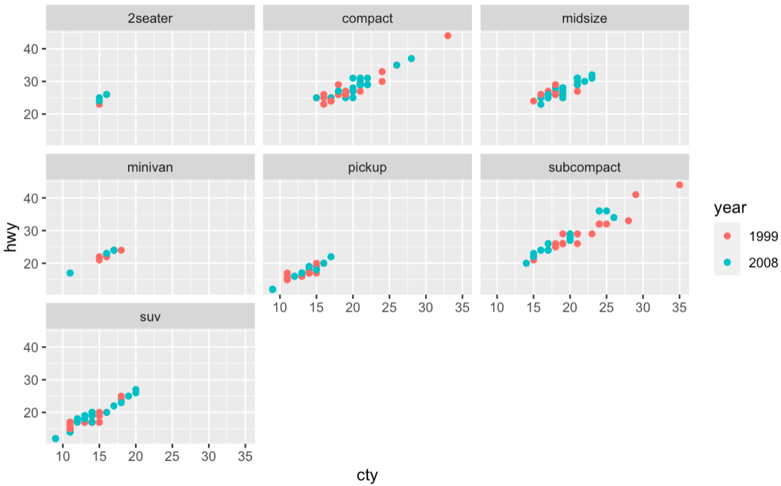
Tidy Plots

```r
ggplot(data = mpg) +
  geom_point(aes(x = cty, y = hwy, color = year)) +
  facet_wrap(~class) +
  ggtitle("MPG on city versus highway broken down by year of vehicle and type of vehicle")
```



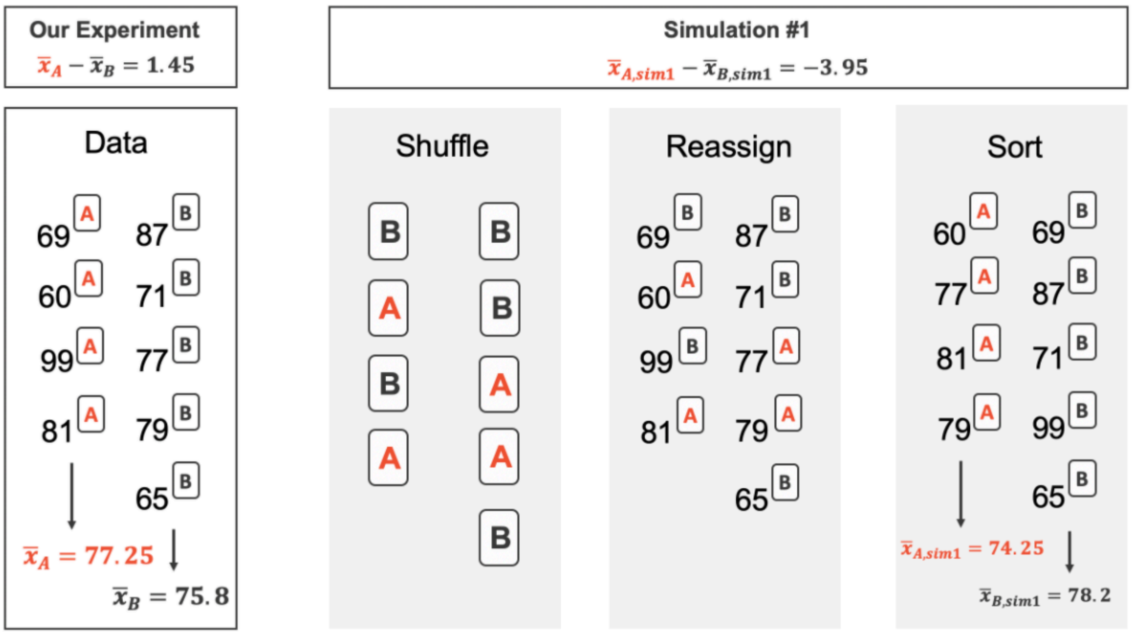
Currently:
we want a viz,
the computer creates it



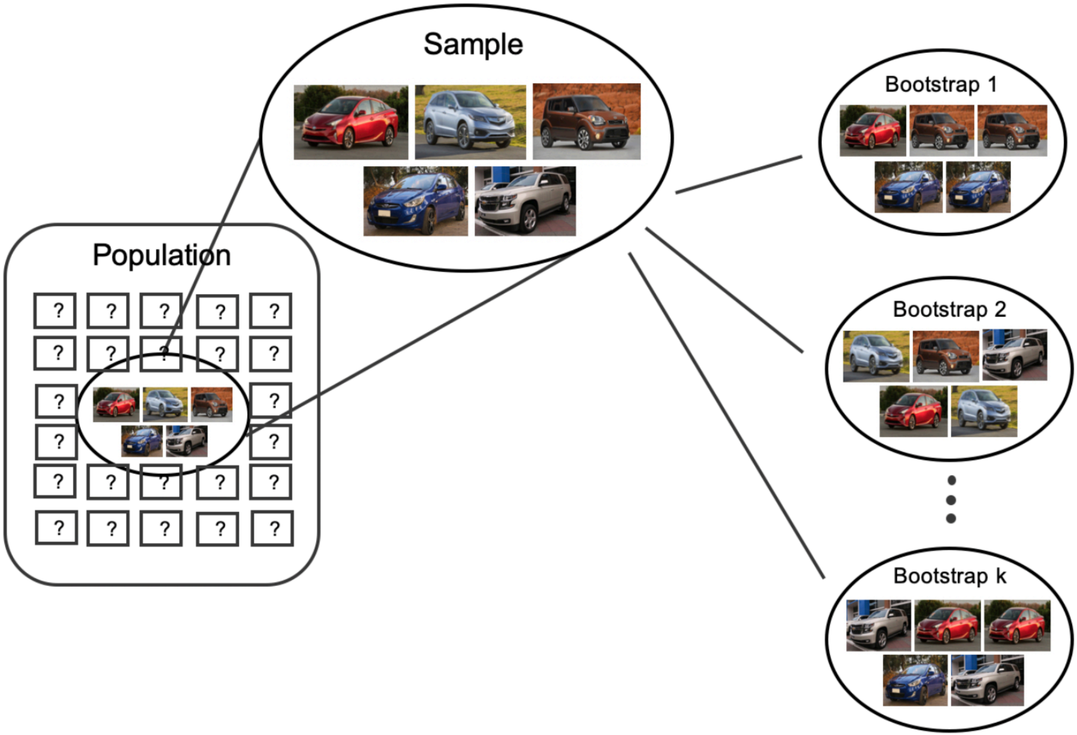


Instead:
the computer creates,
the viz elucidates
→ what model?

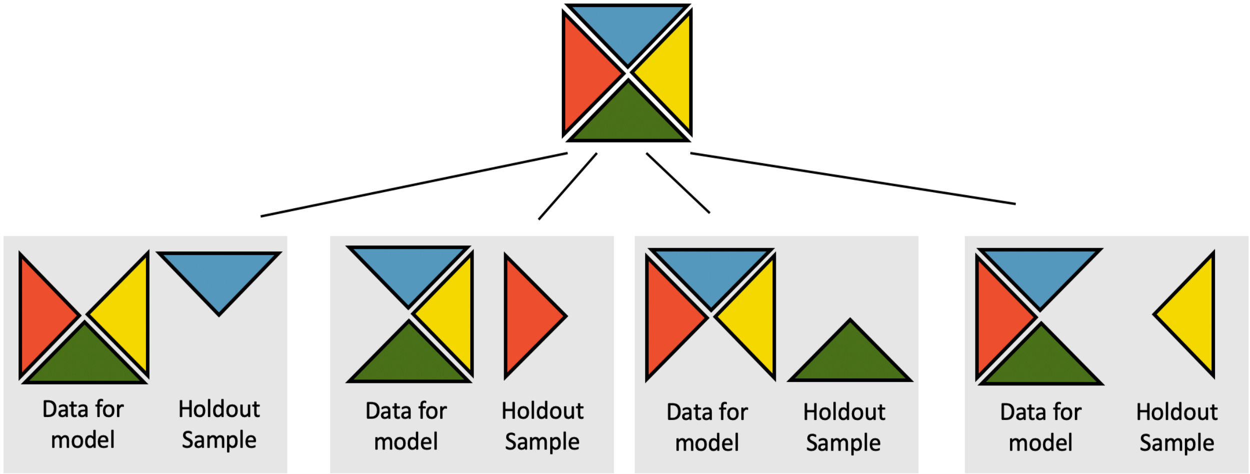
Randomization Tests

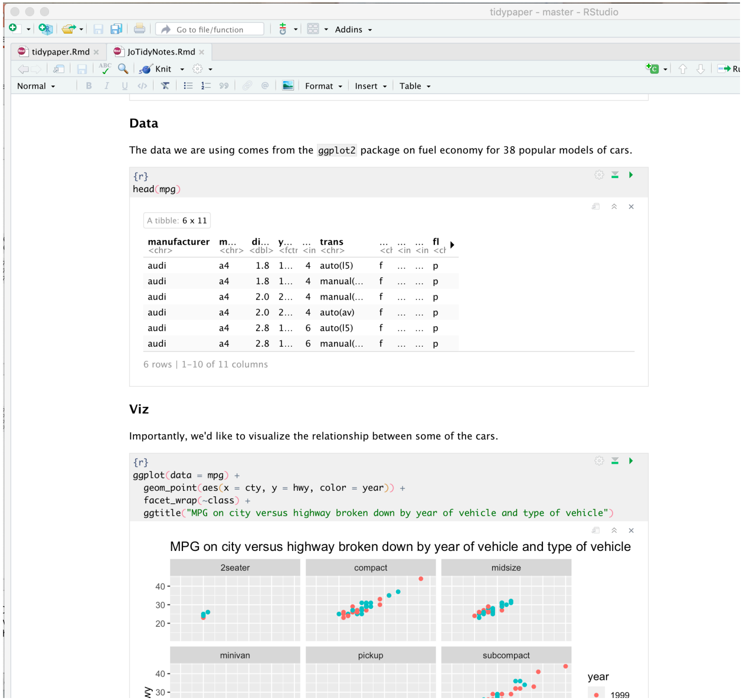


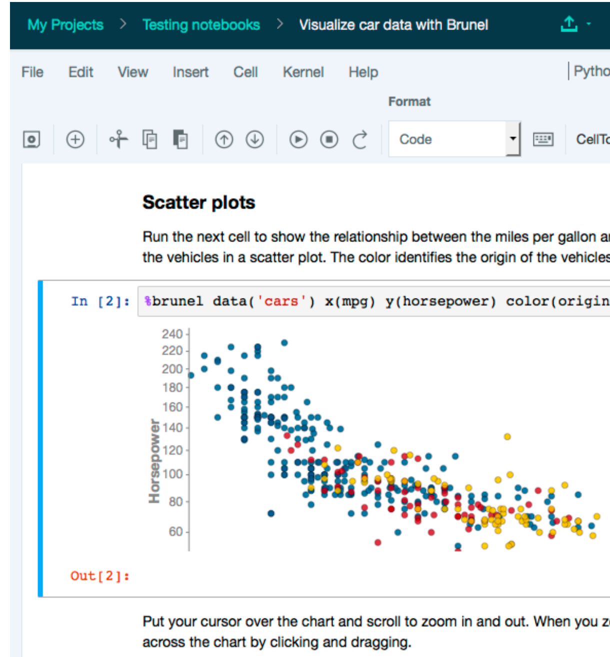
Bootstrapping



Cross Validation







R Markdown

Jupyter Notebook

technology & communication are *intimately* related

Image credit: Pomona College

Thank you

the value of computational thinking
in statistics education

Jo Hardin

jo.hardin@pomona.edu

@jo_hardin47

https://github.com/hardin47

http://research.pomona.edu/johardin/