

# *Data Analysis Pt I*

## *What We Wish We Had Known*

Brett Bessen

Sarah Brown

Damon C. Roberts

October 16, 2021

# *The Replication Crisis and why we should care about Open Science*

An alarming number of scientific papers contain Excel errors

**Science has been in a “replication crisis” for a decade. Have we learned anything?**

Bad papers are still published. But some other things might be getting better.

By Kelsey Piper | Oct 14, 2020, 12:20pm EDT

*Figure:* Headlines of Articles in the news. Right: Washington Post, Left: Vox

# *How do we deal with this?*

- Open Science!
  - Political Science and the social sciences are moving towards this
  - It is a concerted effort to put everything we do out in the open. We share all of our code, we pre-register experiments, we do not perform one undocumented step
  - Why? Because we don't like our credibility tarnished. Especially given where society is.

# *The Open Scientists Mindset*

If you do not have code for it, then you did not do it.

- What this means:
  - You should have code for everything you do. Data cleaning, making graphs, and analyses
  - Do not click on things
  - Do not save files manually. Everything should be automated by code. Graphs are not saved by clicking in RStudio, use *ggsave()*. Tables are not copied and pasted nor do you manually enter numbers, you write the code in R and save the output.

# *How do we do open science?*

1. Good **Project Management & Workflow**
2. Careful and thorough **Data Cleaning**

# *Project Management*

1. Workflow
2. Communication with coauthors
3. IDE's and Text Editors
4. File Organization
5. File Storage

# Workflow

# *Communication with coauthors*

- Email
- Slack



# *IDE's and Text Editors*

- Integrated Developer's Environment (IDE) - Code and building happens here
  - RStudio
  - R Console
  - Visual Studio Code
- Text Editor - Code is written but sent somewhere else
  - ESS/EMACS
  - Sublime Text

# *File Organization*

- Working Directories and R Projects/*here* package
- `snake_case_file_names`
- Splitting scripts, data, figures, memos, drafts files in subfolders

# *File Storage*

- No matter what: encryption and security!
- Dropbox
  - For file storage and collaboration with coauthors
- Github
  - For version control
- Computer Backups
  - Encrypted External Hard Drives

# *Data Cleaning*

1. Where to clean your data
2. R Script Conventions
3. Modular versus Lazy Loading
4. Tidyverse vs. Base R

## *Where to clean your data*

- Not in Excel!!
- In whatever statistical software package you use
- There needs to be code for every step you take for recoding

# *R Script conventions*

- snake\_case for variable and object names
- Sections and Subsections
- Commenting

# *Modular versus Lazy Loading*

Lazy Loading

```
library(dplyr)
```

Modular loading

```
box::use(  
  dplyr = dplyr[mutate]  
)
```

# *Tidyverse versus Base R*

- Tidyverse Benefits
  - Consistency
  - Spans across workflow
  - What is a pipe operator?



# Resources

- **Project Management:** [Github](#)
- **Data Cleaning:** [Github](#)