

Creating optimal conditions for reproducible data analysis in R with 'fertile'

Audrey Bertin

Smith College, '21

Introduction

fertile:

- **What?** Software package written in R
- **Goal:** Improve scientific reproducibility of R projects
- **My Role:** Leading code development for the last ~2 years!



Overview

1. What is reproducibility and why is it lacking?
2. Other attempts to address reproducibility and their shortcomings
3. Why `fertile` is different
4. Potential applications
5. Conclusion
6. Accessing project materials

What is reproducibility?

In data science, research is considered fully **reproducible** when the requisite code and data files produce identical results when run by another analyst.

VS

Replicability: The ability of a researcher to duplicate the results of a study when following the original procedure but collecting new data

The benefits of reproducibility

1. Trusting findings
2. Receiving feedback
3. Extending ideas

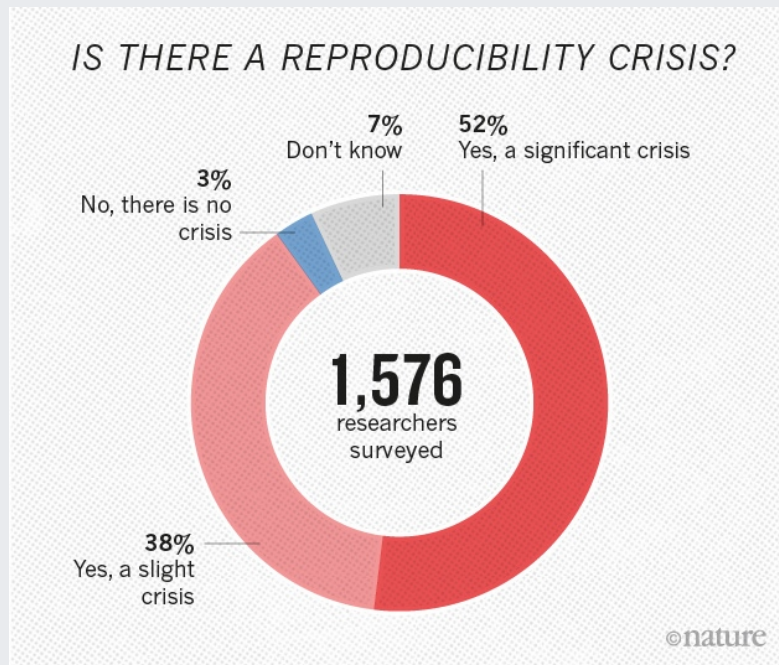
The reproducibility crisis

Nature (2016): 52% respondents claim "crisis"

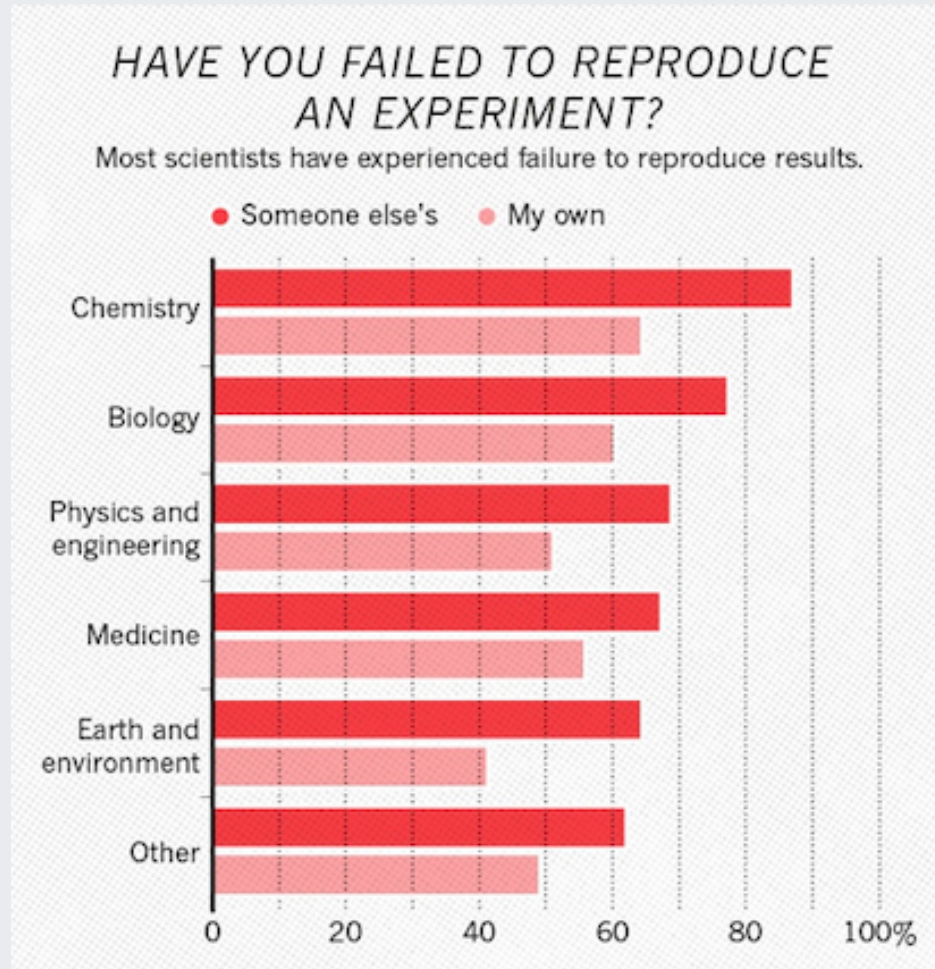
Vast majority cannot reproduce others' work.

Half cannot even reproduce their own!

Some fields have reproducibility rate <10%.



The reproducibility crisis





















6 components of reproducibility

- 1) Basic files made accessible
- 2) Organized file structure
- 3) Good documentation
- 4) File paths
- 5) Randomness
- 6) Style

Attempts to address reproducibility: educational programs

- Replication studies at Graduate level (Cambridge, Harvard, UNC, Penn State, Berkeley)
- At undergraduate level, requirement for work to be completed in .Rmd (Smith, Duke)

Attempts to address reproducibility: journals

Journal Name	Code Sharing Required?	Data Sharing Required?	Other Components Required?
Journal of the American Statistical Association			
Journal of Statistical Software			
Journal of Computational and Graphical Statistics			
The R Journal			
The American Statistician			
The Annals of Statistics			

★ Component recommended, but not required.

◆ Component required, but exceptions granted.

Attempts to address reproducibility: software

- `rrtools`: create basic package structure, Docker dependency management
- `orderly`: automation of projects
- `workflowr`: version control
- `checkers`: custom checks to assess reproducibility
- `renv` (formerly `packrat`): dependency management
- `drake`: makefiles
- `reproducible`: caching to speed up analysis
- Continuous integration tools: `wercker`, GitHub Actions, Travis CI, Circle CI.



Travis CI



Shortcomings of current attempts

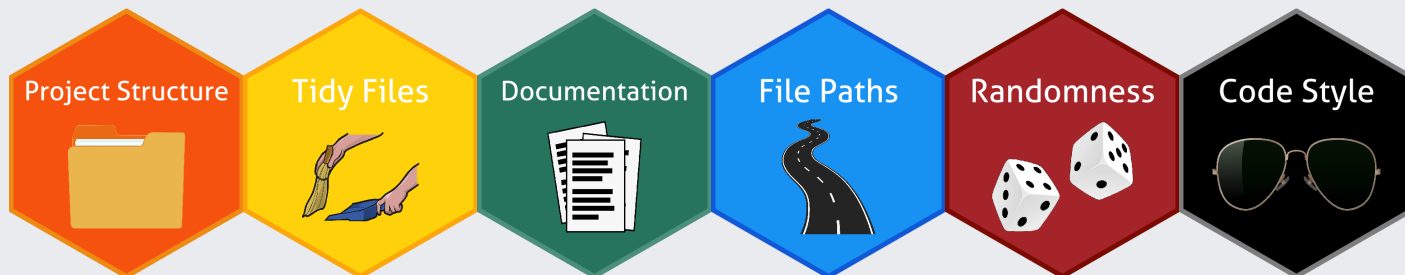
- Education:
 1. Workshops often optional
 2. Only really at graduate level
 3. Takes time away from other important class topics
 4. Takes a lot of effort for professor to set up
- Journals:
 1. Authors lack knowledge and/or time to make changes
 2. Reproducibility review is time/cost intensive for journals
- Software:
 1. Packages narrow in scope
 2. Complex functions, bad for new users
 3. Cumbersome, with steep learning curve
 4. CI tools: lack software-specific tools

Setting **fertile** apart: package goals

- 1) Simple and straightforward
- 2) Accessible to variety of users
- 3) Addresses many aspects of reproducibility
- 4) R-specific features
- 5) Customizable
- 6) Educational
- 7) Applicable in many domains

Overview functions

- `proj_check()`: run a bunch of different tests on various parts of reproducibility (files, paths, documentation, etc.)
- `proj_analyze()`: package dependencies, files, file move suggestions, paths
- `proj_badges()`: earn/display badges for different components, summary of project generation info



Overview functions

FilesPlotsPackagesHelpViewer

Project: project_miceps

fertile reproducibility report

2020-10-29 14:00:49

Badges Awarded:

Project Structure

File Paths

Badges Failed:

Tidy Files

Documentation

Randomness

Code Style

Reasons for Failure:

Tidy Files

```
## # A tibble: 3 x 1
##   check_name
##   <chr>
## 1 has tidy images
```

Overview functions

Code Style

```
## # A tibble: 1 x 1
##   check_name
##   <chr>
## 1 has_no_lint
```

Output Generation Details:

This project summary was generated on 2020-10-29 at 14:00:51 (America/New_York) by a user with the following information:

- Full name: Audrey Bertin
- Username: audreybertin
- Email: N/A
- GitHub Username: N/A

The computer used to generate this file was running R version 4.0.2 (2020-06-22) on the x86_64-apple-darwin17.0 (64-bit) platform and the macOS Catalina 10.15.5 operating system.

The files analyzed in the creation of this summary, as well as their last-modified timestamp, are provided below:

```
## # A tibble: 9 x 2
##   file_name                last_edited
##   <chr>                  <dtm>
## 1 Blot_data_updated.csv  2020-10-12 14:25:17
## 2 CS_data_redone.csv     2020-10-12 14:25:22
## 3 Estrogen_Receptors.docx 2020-10-12 14:25:27
## 4 README.md              2019-01-25 14:19:39
## 5 analysis.Rmd           2020-10-12 14:25:32
## 6 citrate_v_time.png      2020-10-29 14:00:40
## 7 mice.csv               2020-10-12 14:29:31
## 8 miceps.Rproj           2019-01-25 14:19:39
## 9 proteins_v_time.png     2020-10-29 14:00:39
```


Educational features

- Interactive path warning system
- Checks provide informative messages
 - Explain problem
 - Provide solution

```
read.csv("~/Desktop/my_data.csv")
```

```
## Error: Detected absolute paths. Absolute paths are not reproducible  
## and will likely only work on your computer. If you would like  
## to continue anyway, please execute the following command:  
## utils::read.csv('~/.Desktop/my_data.csv')
```

Customizability

- `proj_check_some()`: run subset of checks
- `proj_check_badges()`: run each of the badges from `proj_badges()` individually
- Controlling which functions throw warnings about paths:
 - Some built in, but users can add/edit others:
 - `add_shim()`: add a function to the warning list
 - `edit_shims()`: edit warning list
 - `load_shims()`: activate warning system
 - `unload_shims()`: deactivate warning system

Potential applications: teaching reproducibility

- Introduce reproducibility in undergrad classrooms
- Limited barriers to entry:
 1. R and RStudio installed on their computer
 2. Knowledge of how to install a package from GitHub and load it into their environment
 3. Knowledge of how to create an R project
 4. Knowledge of how to run basic functions and input simple file paths

Potential applications: miscellaneous

- Private companies: increasing transparency w/ clients, building trust
- Conferences: reproducibility standards as requirement for acceptance
- Informal analysis: more reproducible work for events like tidy tuesday --> share knowledge!

Conclusion

- There is currently a reproducibility crisis
- Existing solutions are lacking for a variety of reasons
- `fertile` addresses these all in one!
 - Customizable
 - Easy to use
 - Educational
 - R specific features
 - Addresses multiple aspects of reproducibility
 - Applicable to many domains
- Potential uses:
 - Classroom
 - Journals
 - Offices
 - Conferences
 - Informal analysis

To learn more

- GitHub repository for `fertile`: <https://github.com/baumer-lab/fertile>
- My repository for `fertile`, to track my changes:
<https://github.com/ambertin/fertile>
- Currently writing a thesis. The library will have a copy sometime in the near future!
- `fertile` article in Stat journal: <https://doi.org/10.1002/sta4.332>

Questions?