Intro to Reproducibility
&&
Working Reproducibly with R

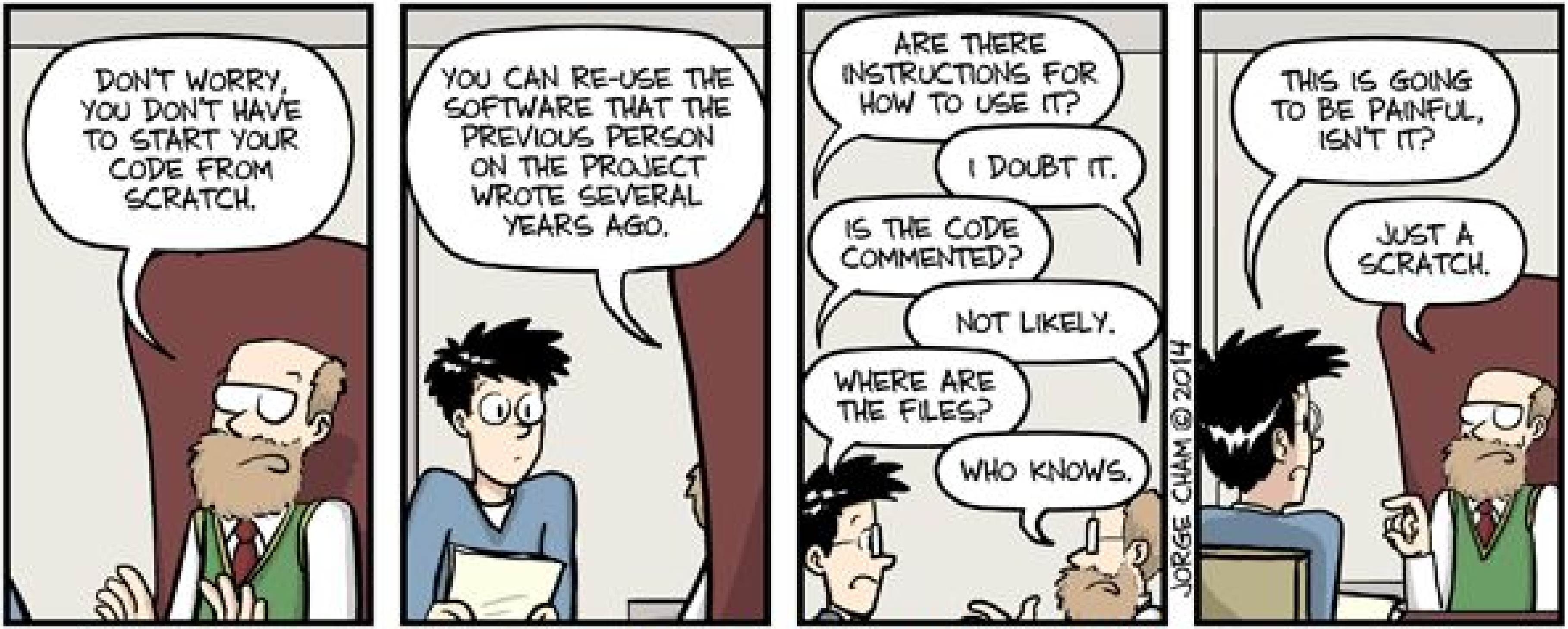
Slides: https://goo.gl/KdNfmP

Vicky Steeves | vicky.steeves@nyu.edu
Librarian for Research Data Management & Reproducibility
NYU Division of Libraries & Center for Data Science

Intro to Reproducibility

Obligatory (but relevant) PhD comic strip...

Piled Higher and Deeper ***by Jorge Cham***


DON'T WORRY, YOU DON'T HAVE TO START YOUR CODE FROM SCRATCH.
YOU CAN RE-USE THE SOFTWARE THAT THE PREVIOUS PERSON ON THE PROJECT WROTE SEVERAL YEARS AGO.
ARE THERE INSTRUCTIONS FOR HOW TO USE IT?
I DOUBT IT.
IS THE CODE COMMENTED?
NOT LIKELY.
WHERE ARE THE FILES?
WHO KNOWS.
JORGE CHAM © 2014
THIS IS GOING TO BE PAINFUL, ISN'T IT?
JUST A SCRATCH.

title: "Scratch" - originally published 3/12/2014 WWW.PHDCOMICS.COM

Why Reproducibility?

> "If I have seen further, it is by standing on the shoulders of giants." - Sir Isaac Newton

To build on top of previous work – science is incremental!

To verify the correctness of results

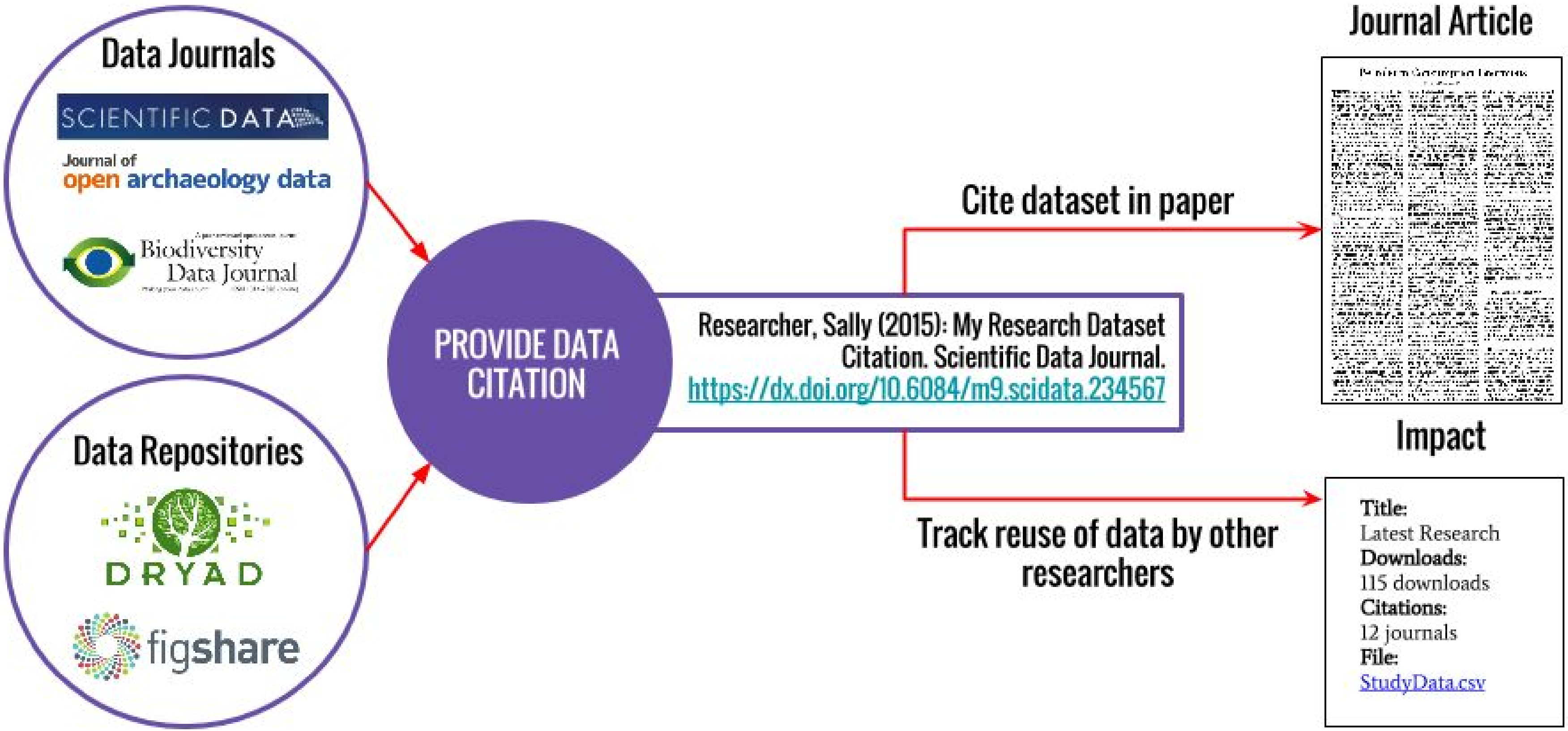
To defeat self-deception[1]

To help newcomers

To increase impact, visibility[2] and research quality[3]

1. http://www.nature.com/news/how-scientists-fool-themselves-and-how-they-can-stop-1.18517
2. http://infoscience.epfl.ch/record/136640/files/VandewalleKV09.pdf
3. http://www.nature.com/nature/journal/v483/n7391/full/483531a.html

Why Reproducibility? Think Selfishly



Data Journals

SCIENTIFIC DATA

Journal of open archaeology data

Biodiversity Data Journal

Data Repositories

DRYAD

figshare

PROVIDE DATA CITATION

Researcher, Sally (2015): My Research Dataset Citation. Scientific Data Journal. https://dx.doi.org/10.6084/m9.scidata.234567

Cite dataset in paper

Journal Article

Track reuse of data by other researchers

Impact

Title:
Latest Research
Downloads:
115 downloads
Citations:
12 journals
File:
StudyData.csv

Why Reproducibility? Think Selflessly

- Others can re-use and extend your work more easily!
  - You can even find interesting collaborations and future research projects out of this.
- YOU can re-use and extend your work more easily! (sort of selfless…)
  - Future you is your greatest collaborator.
- Newbies to the field can more easily learn the methods by reproducing your work!
  - Your reproducible work is their greatest teacher.

Reproducibility? Replication? Huh?

Reproducibility

Independently confirm results with the **same** data and code

Replication

Independently confirm results with **new** data and code

Reproducibility on a spectrum

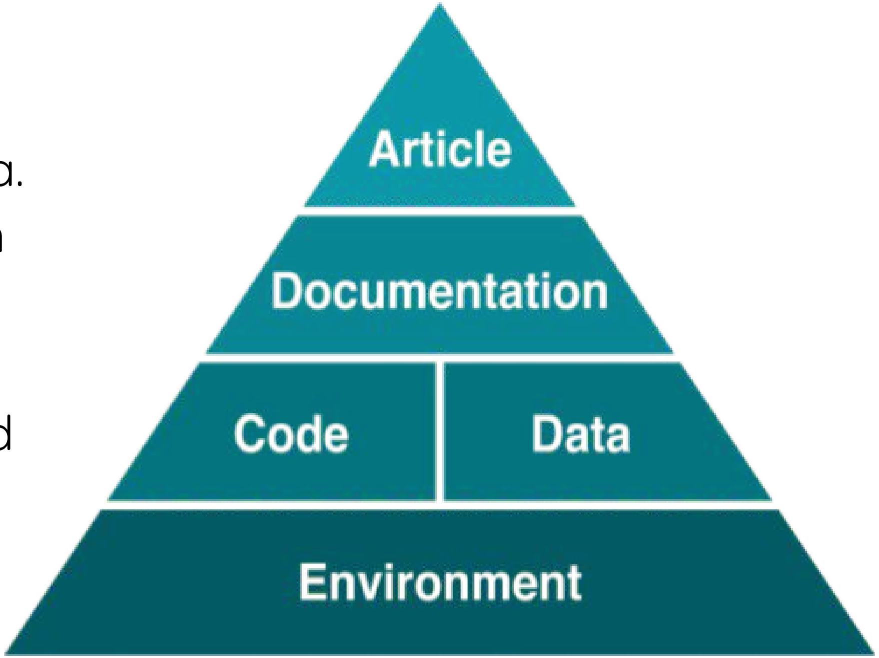
Reviewable Research: Sufficient detail for peer review & assessment.

Replicable Research: Tools are available to duplicate the author’s results using their data.

Confirmable Research: Main conclusions can be attained independently without author’s software.

Auditable Research: Process & tools archived such that it can be defended later if necessary.

Open/Reproducible Research: Auditable research made openly available.



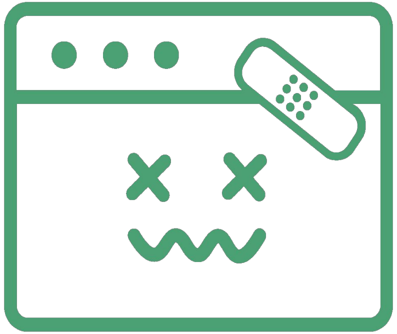
Stodden et al ICERM report (2013)

Even if runnable, results may differ...

The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements

We investigated the effects of data processing variables such as FreeSurfer version (v4.3.1, v4.5.0, and v5.0.0), workstation (Macintosh and Hewlett-Packard), and Macintosh operating system version (OSX 10.5 and OSX 10.6). **Significant differences** were revealed between **FreeSurfer version v5.0.0 and the two earlier versions**. [...] About a factor two smaller differences were detected between**Macintosh and Hewlett-Packard workstations** and between **OSX 10.5 and OSX 10.6**

Challenges in Reproducibility



- People make mistakes--and it impacts their research
- It’s good to have other people check out your data and analyses--it’s like having a copy editor for your data!

- It’s *hard* to keep track of what version of what was used
- Software get updates, and these changes can disrupt reproducibility

So, what do we need?

Tool that can automatically capture all the dependencies in the original environment and automatically set them up in another environment.

- Packrat (R library)
- ReproZip
- Docker

Tool to seamlessly review whole research projects without the reviewer having to manually debug all dependencies, code, and data.

- Packrat (R library)
- ReproZip

Working Reproducibly with R

Prerequisites

Please make sure you have the following installed:

- R & RStudio
- R Markdown -- `install.packages("rmarkdown")`
- knitr -- `install.packages("knitr")`
- packrat -- `install.packages("packrat")`

Let’s take 5 minutes to make sure everyone has these things installed.

You can't have any sort of reproducibility without good data and project management.

Basic Project Structure

Put each project in its own directory, which is named after the project.

Put text documents associated with the project in the `doc` folder.

Put raw data and metadata in the `data` folder, and files generated during cleanup and analysis in a `results` folder.

Put source for the project’s scripts and programs in the `src` folder.

Name all files to reflect their content or function, with NO special characters (!@#$%^*) or spaces! Use underscores or dashes, A-Z, and numbers!

Basic Project Etiquette

If you’re coding, use **version control**. When you create a new project in R, it lets you instantiate a **git repo** at the same time. If you click that checkbox, you can later push your R project to GitHub or GitLab easily!

No matter what you are doing, **document** it.

All this is for nothing if your work isn’t **backed up**.

Anonymizing Data

Anonymizing data:

- Direct identifiers (name, DOB, SSN, address, id numbers, etc.)
- Indirect identifiers (variables in combination that enable identification
    - In 2000, researchers showed that 87% of the US population could be uniquely identified using ZIP code, birthdate, and sex.

Solutions:

- Removal of identifying variables
- Binning values/top coding (i.e. hide unique outlier values or aggregate values)
- Disturbing (add random values to encoded value, retaining integrity of statistical accuracy)

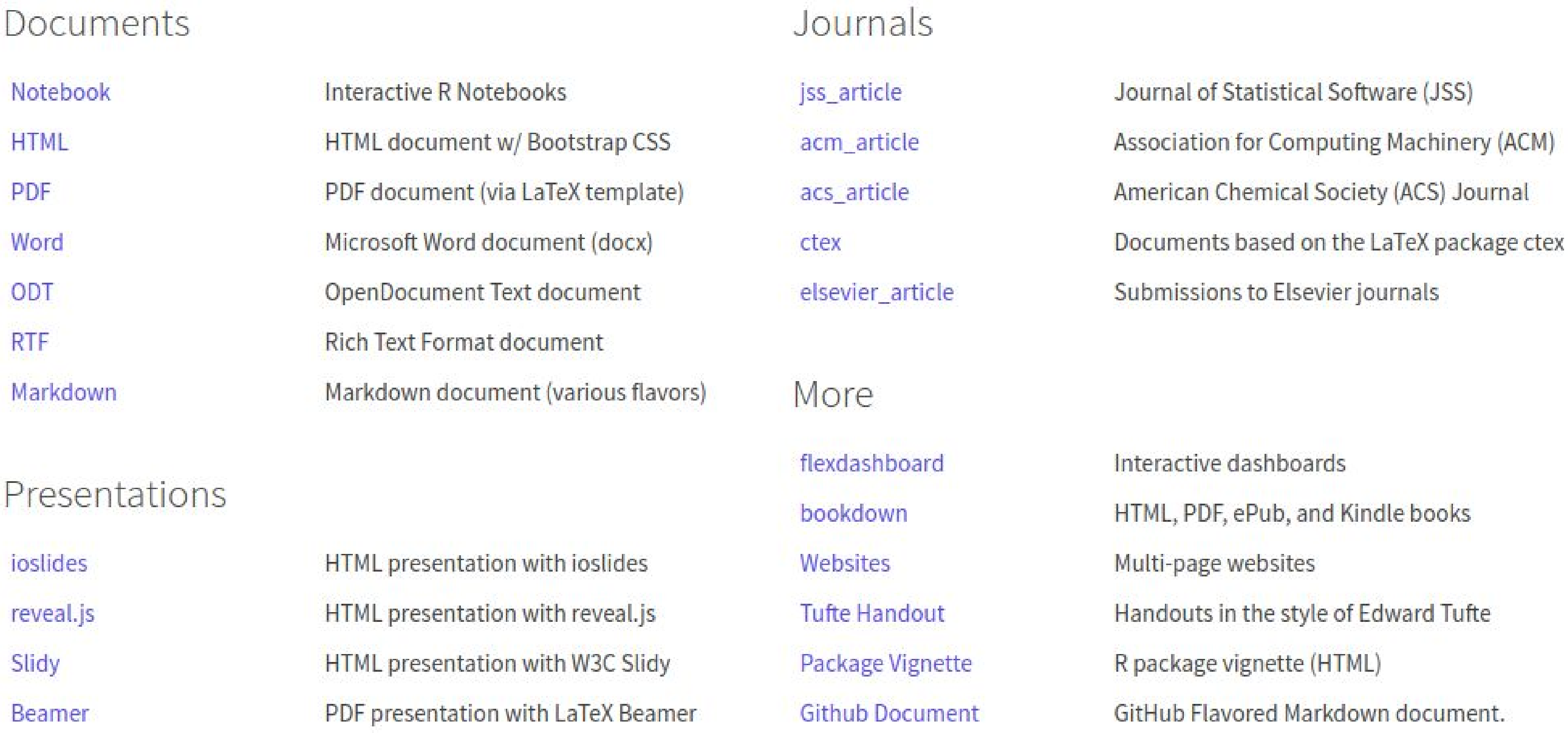
R Markdown (.Rmd)

R Markdown is an extension of Markdown. It works sort of like an executable paper -- it mixes documentation & code. And not just R! You can insert code snippets from other languages (SQL, bash, Python)

This allow you write documents which integrate results from your analysis. Incorporating R results directly into your documents is an important step in reproducible research. Any changes that occur in either your data set or the analysis are automatically updated in your document the next time the document is created.

You can export from RMD files to a variety of different formats --

Export Formats for R Markdown (.Rmd)



Documents

Notebook	Interactive R Notebooks
HTML	HTML document w/ Bootstrap CSS
PDF	PDF document (via LaTeX template)
Word	Microsoft Word document (docx)
ODT	OpenDocument Text document
RTF	Rich Text Format document
Markdown	Markdown document (various flavors)

Presentations

ioslides	HTML presentation with ioslides
reveal.js	HTML presentation with reveal.js
Slidy	HTML presentation with W3C Slidy
Beamer	PDF presentation with LaTeX Beamer

Journals

jss_article	Journal of Statistical Software (JSS)
acm_article	Association for Computing Machinery (ACM)
acs_article	American Chemical Society (ACS) Journal
ctex	Documents based on the LaTeX package ctex
elsevier_article	Submissions to Elsevier journals

More

flexdashboard	Interactive dashboards
bookdown	HTML, PDF, ePub, and Kindle books
Websites	Multi-page websites
Tufte Handout	Handouts in the style of Edward Tufte
Package Vignette	R package vignette (HTML)
Github Document	GitHub Flavored Markdown document.

http://rmarkdown.rstudio.com/formats.html

`Packrat` -- package management for R

`Packrat` stores package dependencies **inside a project**, rather than relying on the personal R library that is shared across all R sessions. Why I like `Packrat`:

- **Isolated:** Gives each project its own private package library.
- **Portable:** Easily transport projects from one computer to another, even across different platforms. `Packrat` makes it easy to install the packages your project depends on.
- **Reproducible:** `Packrat` records the exact package versions code depends on, and ensures those exact versions are the ones that get installed.

Here's a good overview:
https://www.r-bloggers.com/creating-reproducible-software-environments-with-packrat/

Getting Started

1. Click the “File” menu button, then “New Project”.
2. Click “New Directory”.
3. Click “Empty Project”.
4. Type in the name of the directory to store your project, e.g. “my_project”.
5. Select the checkboxes for “Create a git repository” and “Use packrat with this project”
6. Click the “Create Project” button.
7. Work as you’d normally do -- install packages, write scripts, etc.

Let’s get coding!

Please download this dataset (https://goo.gl/RsB2se) & this dataset (https://goo.gl/G1Ue6N)

Using `Packrat` with our work

1. `packrat::snapshot()`
    a. This saves new libraries and dependencies in the packrat/src project subdirectory. It also records metadata about each package.
2. `packrat::bundle():` when you're finished and want to share your work.
    a. This creates a `.tar.gz` file with the source code of all the R dependencies needed to rerun your scripts.
3. `packrat::unbundle()`: Unbundle a packrat project, generating a project directory with libraries restored
    a. `packrat::unbundle(bundle="testproject-2014-07-15.tar.gz", where="/home/bob/projects")`

BONUS: ReproZip

Documentation

reprozip.readthedocs.org

Source Code

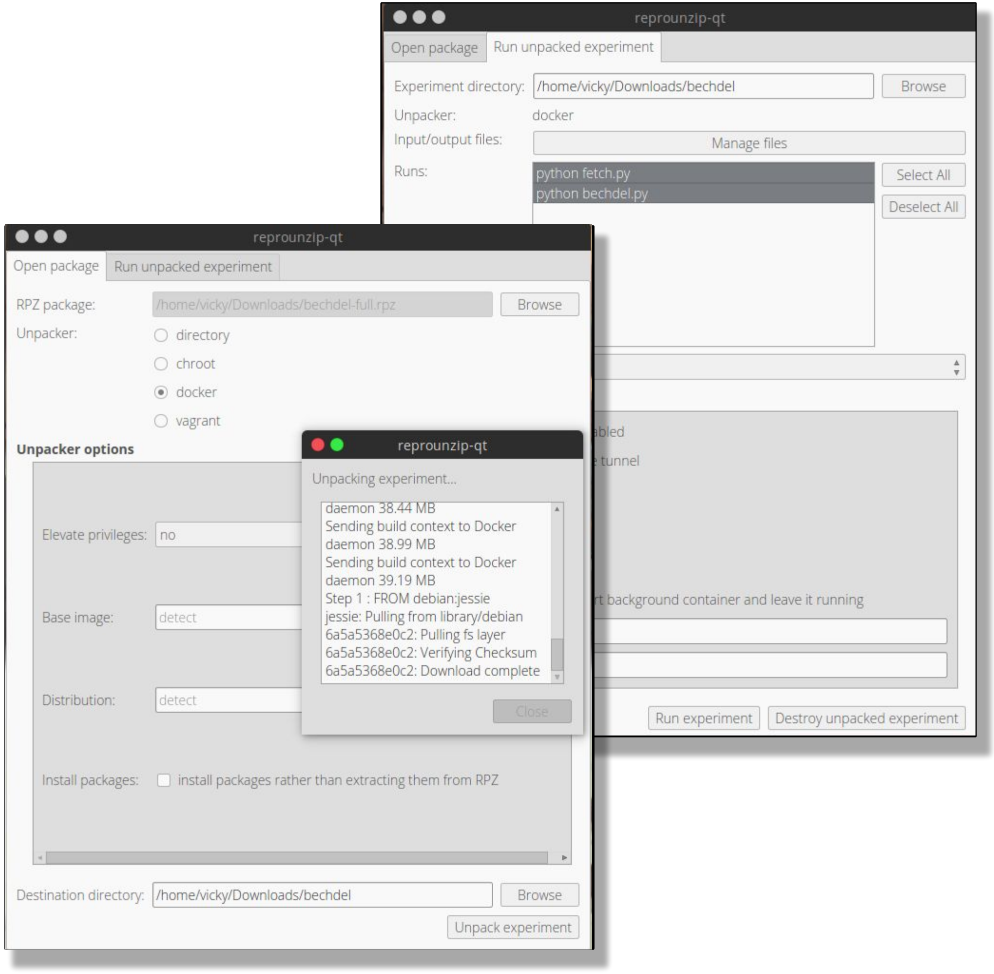
https://github.com/ViDA-NYU/reprozip

Case Studies

https://github.com/ViDA-NYU/reprozip-examples

Video Demo

https://www.youtube.com/watch?v=-zLPuwCHXo0



Experiment directory: /home/vicky/Downloads/bechdel

Unpacker: docker

Input/output files:

Runs: python fetch.py, python bechdel.py

...abled

...tunnel

...t background container and leave it running

RPZ package: /home/vicky/Downloads/bechdel-full.rpz

Unpacker: ○ directory ○ chroot ● docker ○ vagrant

Elevate privileges: no

Base image: detect

Distribution: detect

Install packages: □ install packages rather than extracting them from RPZ

Destination directory: /home/vicky/Downloads/bechdel

Unpacking experiment...

```
daemon 38.44 MB
Sending build context to Docker
daemon 38.99 MB
Sending build context to Docker
daemon 39.19 MB
Step 1 : FROM debian:jessie
jessie: Pulling from library/debian
6a5a5368e0c2: Pulling fs layer
6a5a5368e0c2: Verifying Checksum
6a5a5368e0c2: Download complete
```

Some light reading --

- Barba-group reproducibility syllabus: https://hackernoon.com/barba-group-reproducibility-syllabus-e3757ee635cf
- Ten Simple Rules for Reproducible Computational Research: http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003285
- Five selfish reasons to work reproducibly: https://genomebiology.biomedcentral.com/articles/10.1186/s13059-015-0850-7
- How scientists fool themselves – and how they can stop: https://www.nature.com/news/how-scientists-fool-themselves-and-how-they-can-stop-1.18517
- Gitlab-CI for R packages: https://gitlab.com/jangorecki/r.gitlab.ci && https://bertelsen.ca/example-gitlab-ci-yml-for-r-projects/

Questions?

Email me: vicky.steeves@nyu.edu
Get this presentation:
github.com/VickySteeves/Repro-R