

Making research reproducible: git, R and org-mode

Philipp Homan, MD, PhD

phoman1@northwell.edu

<http://github.com/philiphoman/mrr>



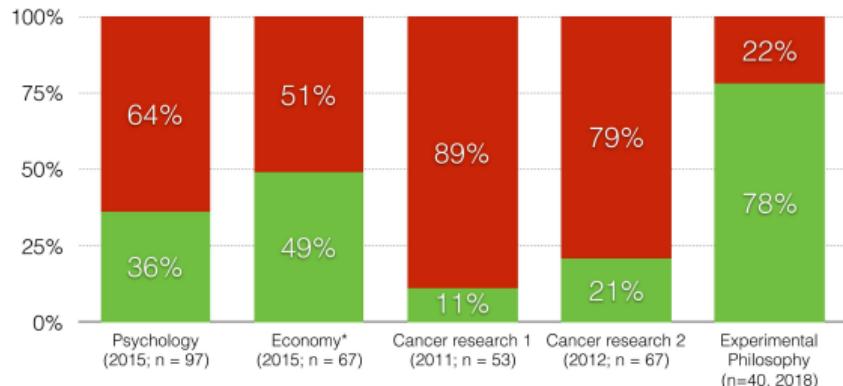
Why should we even care?

It's not reproducible
if it only runs on your laptop!

<http://www.jonzelner.net/docker/reproducibility/>

We are faced with a replication crisis

Which part of published findings can be independently replicated?



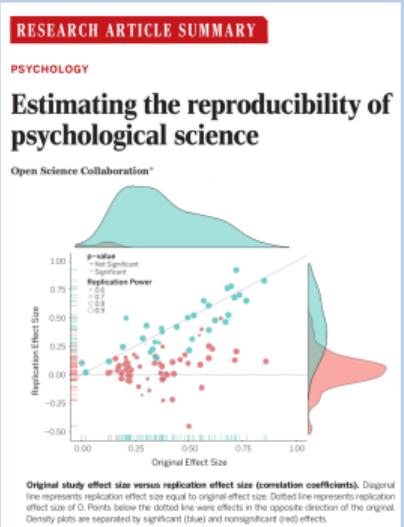
* The data on economics is about *reproducibility*; i.e. the attempt to get the same results if you apply the original data analysis on the original data set.

Open Science Collaboration (2015); Chang & Li (2015); Begley, C. G., & Ellis, L. M. (2012). Prinz, F., Schlange, T., & Asadullah, K. (2011); Cova et al. (2018)

12

Credit: Felix Schoenbrodt 2018 (@nicebread303)

We are faced with a replication crisis



- Ongoing methodological crisis in science
- Results of many scientific studies difficult to replicate
- Involves diverse fields (from psychology to cancer research)

Ioannidis 2005, PLOS Med

Open Science Collaboration 2015, Science



DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

As if this wasn't bad enough



As if this wasn't bad enough

We are probably also faced
with a reproducibility crisis

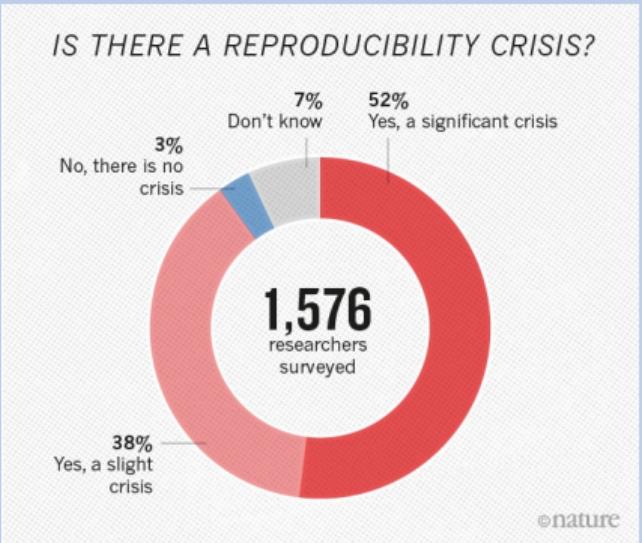


Replicating vs. reproducing

- Replicating: People going out and collecting new data
- Reproducing: People analyzing the same data
- What cannot be replicated often difficult to reproduce

jblevins.org/log/rep

We are probably also faced with a reproducibility crisis



Baker 2016, Nature



DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

Why is it so hard to reproduce our own work?



Why is it so hard to reproduce our own work?

We don't use
standardized work flows



What does that mean?

- We use intuition when we organize data, manuscripts, code
- The same goes for analyzing data
- After a couple of months (sometimes weeks) it is hard to remember:



What does that mean?

- We use intuition when we organize data, manuscripts, code
- The same goes for analyzing data
- After a couple of months (sometimes weeks) it is hard to remember:
 1. What we did



What does that mean?

- We use intuition when we organize data, manuscripts, code
- The same goes for analyzing data
- After a couple of months (sometimes weeks) it is hard to remember:
 1. What we did
 2. Why we did it



What does that mean?

- We use intuition when we organize data, manuscripts, code
- The same goes for analyzing data
- After a couple of months (sometimes weeks) it is hard to remember:
 1. What we did
 2. Why we did it
 3. How we did it



What can we do about it?

Three simple rules:



What can we do about it?

Three simple rules:

1. Separate data from analysis



What can we do about it?

Three simple rules:

1. Separate data from analysis
2. Use version control



What can we do about it?

Three simple rules:

1. Separate data from analysis
2. Use version control
3. Use code to analyze data (not GUIs)

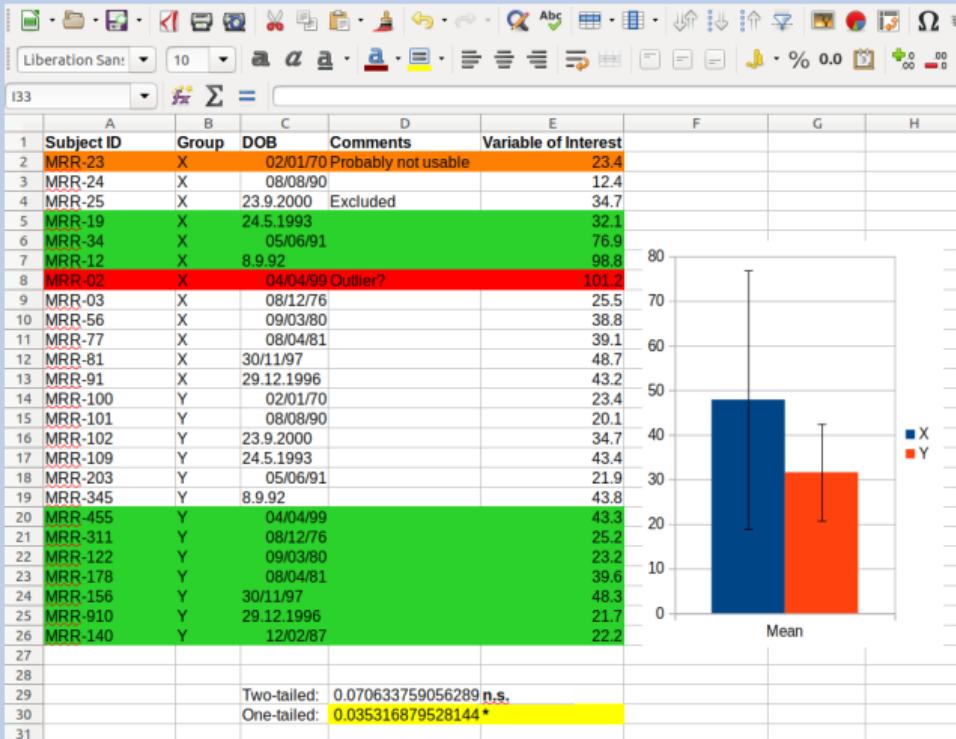


What can we do about it?

Three simple rules:

- 1. Separate data from analysis**
- 2. Use version control**
- 3. Use code to analyze data (not GUIs)**

Separating data from analysis





DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

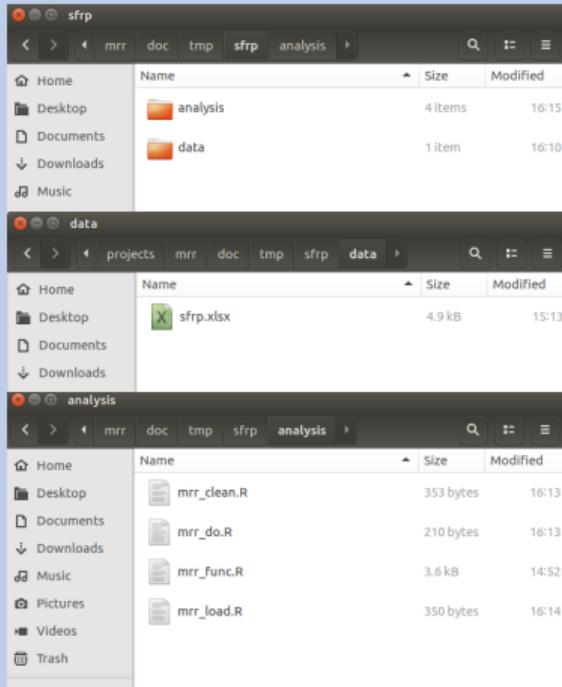
Was this done here?



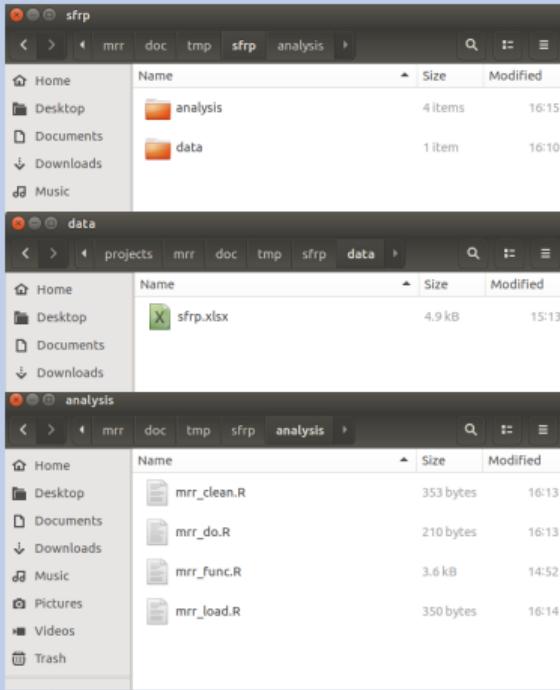
DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

Was this done here?

Separating data from analysis



Separating data from analysis



- We want one and only one data set to work with
- Once finalized (cleaned etc.), it is never touched again
- Any analysis reads from but never writes to this data set

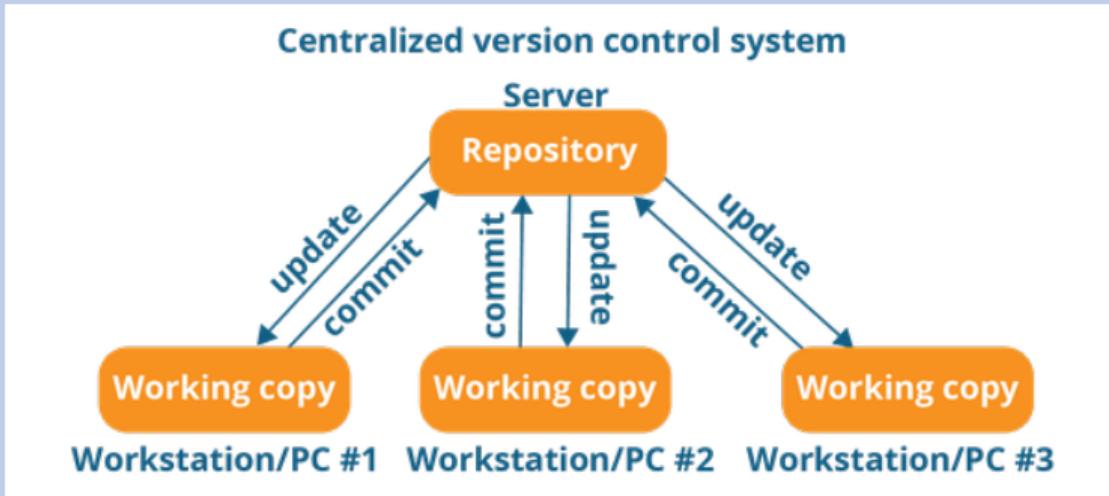


What can we do about it?

Three simple rules:

1. Separate data from analysis
2. **Use version control**
3. Use code to analyze data (not GUIs)

Use a version control system (= use git)



<https://www.quora.com/>

What is git and why should I use it?

- Version control system for source code management
- Tracks every file in a project
- Keeps track of any change to any file
- Is relatively easy to use
- Downside: it works best with text

```
Head:      master added panels to git-slide
Merge:    origin/master added panels to git-slide
Untracked files (1)
Makefile-
Instaged changes (1)
modified   src/mrr_presentation.org
@@ -285,6 +285,7 @@ https://www.quora.com/
- Version control system for source code management
- Tracks every file in a project
- Keeps track of any change to any file
+- Is relatively easy to use
** Right

7 UR *magit: mrr All <E> M
```



DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

Example: git



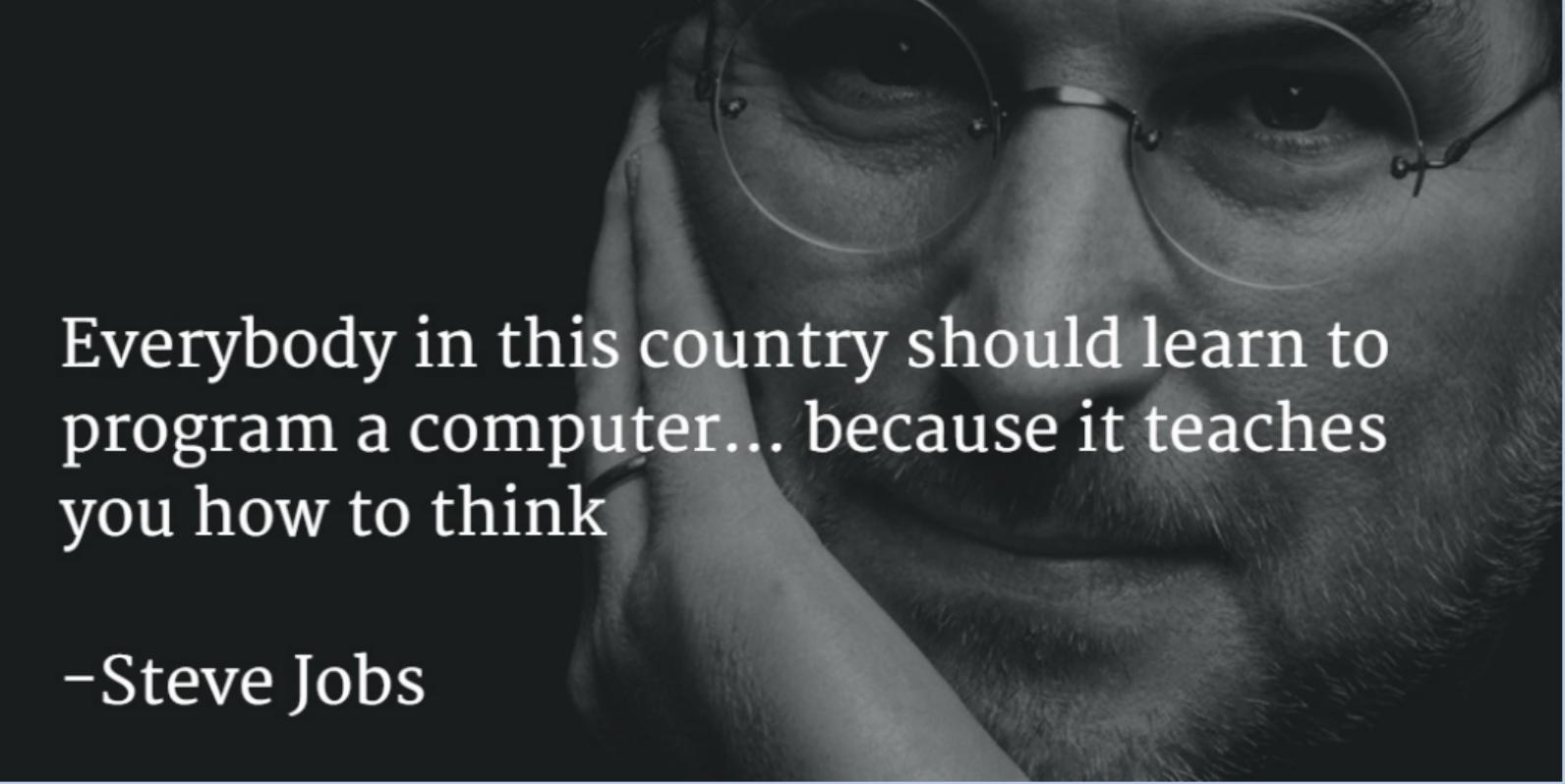
What can we do about it?

Three simple rules:

1. Separate data from analysis
2. Use version control
3. **Use code to analyze data (not GUIs)**



Why code?



Everybody in this country should learn to program a computer... because it teaches you how to think

-Steve Jobs

Why code?

- To keep track of the workflow
- To make the analysis transparent
- To improve your skills and get more efficient as you code⁶

```
1 parse_msd <- function(m, sd) {  
2   #  
3   # this function will  
4   # produce a nicely formatted string of  
5   # mean and sd to be used inline in text  
6   #  
7   print(paste("M = ", round(m, 2),  
8             ", SD = ", round(sd, 2),  
9             sep=""))  
10 }
```



Without code your analysis won't be reproducible

Options:

- R or RStudio (it's free!), ideally also Python (it's free!)
- Alternatively, Matlab (great, but commercial)
- SAS (has been the market leader in commercial analytics, and it does include a free University Edition now)



Without code your analysis won't be reproducible

Options:

- **R or R studio (it's free!), ideally also Python (it's free!)**
- Alternatively, Matlab (great, but commercial)
- SAS (has been the market leader in commercial analytics, and it does include a free University Edition now)

Example: R

```
1 #-----  
2 # This is a simple R program  
3 # 9/18/18, PH  
4 #-----  
5 #  
6 # 1. Load and visualize data  
7 #-----  
8 dat <- read.csv("../data/mrr.csv")  
9  
10 # Histograms  
11 hist(dat$y[dat$group=="X"], col="blue")  
12 hist(dat$y[dat$group=="Y"], col="blue")  
13  
14 # 2. Compute linear model, adjusted for age  
15 #-----  
16 lmfit <- lm(y ~ group + age, data=dat)  
17  
18 # 3. Visualize residuals to check model assumptions  
19 #-----  
20 plot(density(resid(lmfit)))  
21  
22 # 4. Print coefficients  
23 #-----  
24 summary(lmfit)
```



Coding: the good news

- It is easier than you think
- It actually teaches you how to think
- It will be (fairly) easy to learn another one



Summary: How to make research reproducible

Essential:

1. Separate data and analysis
2. Use git to keep track of changes
3. Use R to keep track of your workflow

Optional:

1. **Combine coding and writing to produce manuscripts**
2. Use Make to build your project

Combining coding and writing

Several Options:

- knitr (RStudio)
- **org-mode**
- sweave





DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

Example: org-mode



Summary: How to make research reproducible

Essential:

1. Separate data and analysis
2. Use git to keep track of changes
3. Use R to keep track of your workflow

Optional:

1. Combine coding and writing to produce manuscripts
2. **Use Make to build your project**



DONALD AND BARBARA
ZUCKER SCHOOL OF MEDICINE
AT HOFSTRA/NORTHWELL

Example: Makefile



Conclusion

- We need transparent and reproducible workflows
- Efficient way to improve analyses and writing
- Sharing data, code, workflows may become a requirement

Acknowledgments

- Joe Zellner (jonzellner.net/docker/reproducibility/)
- Andrew Gelman (andrewgelman.com)
- Papaja package in R ([crsh.github.io/papaja_{man}/](http://crsh.github.io/papaja/man/))

