

R-reproducible Psychological Science

Rick Gilmore

2017-08-17 05:38:17

Contents

Themes	1
Is there a reproducibility crisis?	1
Not just in psychology	2
If so, why?	2
(Munafò et al. 2017) manifesto	2
What am I trying to reproduce?	2
Reproducible workflows	2
Using R for reproducible workflows	2
Example 1	3
Make script that calls sequence of R commands or functions	3
Strengths & Weaknesses	3
Example 2 - R Markdown	4
Structure of an R Markdown file	4
Markdown	4
Code “chunks”	4
Inline code	4
One R to rule them all and in the console bind them...	4
Your turn	4
Things to try if you like	5
Big idea	5
Smaller idea in service of bigger	5
Let’s try it with some data	5
One file, many output options	5
Scripting the pipeline	6
Key points	7
Toward a reproducible psychological science...	7
Advanced topics	7
My GitHub workflow	7
Learn from my mistakes	7
References	8

Themes

1. Why worry about reproducible science?
2. What is reproducible psychological science?
3. How can R make my science more transparent, open, and reproducible?

Is there a reproducibility crisis?

- Yes
- A slight crisis
- No crisis

- Don't know
-

Baker 2016

Not just in psychology

If so, why?

Baker 2016

Here are the data from the Nature survey.

(Munafò et al. 2017) manifesto

This recent manifesto from Nature Human Behavior describes the risks to reproducible science at every step of the process. I urge you to read it.

What am I trying to reproduce?

- My own workflow
 - Data collection
 - Cleaning
 - Visualization
 - Analysis
- “Hit by a truck” scenario

But today I want us to think more parochially about our own workflows. How can using R make our own data collection, cleaning, visualization, and analysis workflows more reproducible. Ask yourself this: Can you pick up where you left off on a project you were working on yesterday? Last week? Last month? Six months ago? Put it this way: If you were hit by a truck tomorrow, could your adviser and collaborators pick up where you left off?

Reproducible workflows

- Scripted, automated = minimize human-dependent steps.
- Well-documented
- Be kind to your future (forgetful) self
- Transparent to me & colleagues == transparent to others

Reproducible workflows are scripted. They minimize human contact with your data files. They are well-documented. And it turns out that workflows that are transparent to you and your colleagues are transparent to others. This makes them easy to share.

Using R for reproducible workflows

- **Option 1:** All commands in an R script: e.g., `project_analysis.R`
- **Option 2:** Mix R code, output, comments in an R Markdown document
 - R Markdown files = text files

- One input file, multiple outputs to
- PDF, Word (.docx)
- HTML for notebooks, web pages, slides

We’ve already shown you in this bootcamp how writing R scripts and functions can let you import, clean, munge, reorganize, plot, and analyze data. We’ve already seen how commenting code fragments makes it easier to read and understand. An extension to R called R Markdown lets us mix R code, analyses, text, tables, and other formatting to make all sorts of products. R Markdown files are just text files. But with this one text file, it’s easy to produce multiple output types: PDF or Word formatted documents; HTML for blogs, web sites, or even slide presentation.

Example 1

```
# Import data

# Clean data

# Visualize data

# Analyze data

# Report findings
```

```
# Import data
my_data <- read.csv("path/2/data_file.csv")

# Clean data
my_data$gender <- tolower(my_data$gender) # make lower case
...
```

Make script that calls sequence of R commands or functions

```
# Import data
source("R/Import_data.R") # source() runs scripts, loads functions

# Clean data
source("R/Clean_data.R")

# Visualize data
source("R/Visualize_data.R")
...
```

Strengths & Weaknesses

- R commands in files that can be re-run
- Separate pieces of workflow kept separate
- “Master” script that can be run to regenerate full sequence of results
 - Error in raw data file?
 - No problem; fix and re-run “Master”
- How to save results or share with collaborators?

Example 2 - R Markdown

- James' R commands from Day 1: Raw R script (.R)
- Converted to R Markdown
- Output as | HTML notebook | HTML Slides | PDF | DOCX |

Just to show you how easy this is, let's look at the R syntax James used yesterday. I'm going to show you how adding just a tiny bit of text to that file transforms it. Here is the original R script. Here is the transformed file with a .Rmd extension.

Structure of an R Markdown file

- header info in YAML Ain't Markup Language (YAML) format
- Markdown for formatting text (headers, **boldface**/*italics*, `code`, bulleted or numbered lists, web links, etc.

Markdown

- Human-readable formatting
- Converts to HTML
- Used in blogging engines the world over
- Markdown + R bits -> R Markdown

Code “chunks”

- Surround code with triple back-ticks and {r}

Inline code

```
quest <- "the holy grail"  
color <- "blue"
```

My quest is the holy grail and my favorite color is blue.

One R to rule them all and in the console bind them...

- One file, many possible outputs
 - pdf_document, word_document, or github_document
 - ioslides_presentation for HTML slide show
 - Cool interactive web-app like Dan's tutorial
 - Web sites like the one for this bootcamp, blogs, even books

Your turn

1. Open “File/New File/R Notebook”
2. Change title: "R Notebook" to something else, like title: "Rick's R Notebook"
3. Save the file (default name is Untitled) with an .Rmd extension.
4. Look at the *.Rmd code.
5. Look at the *.nb.html file in a browser.

Things to try if you like

Big idea

Smaller idea in service of bigger

- Supporting point
- Another supporting point

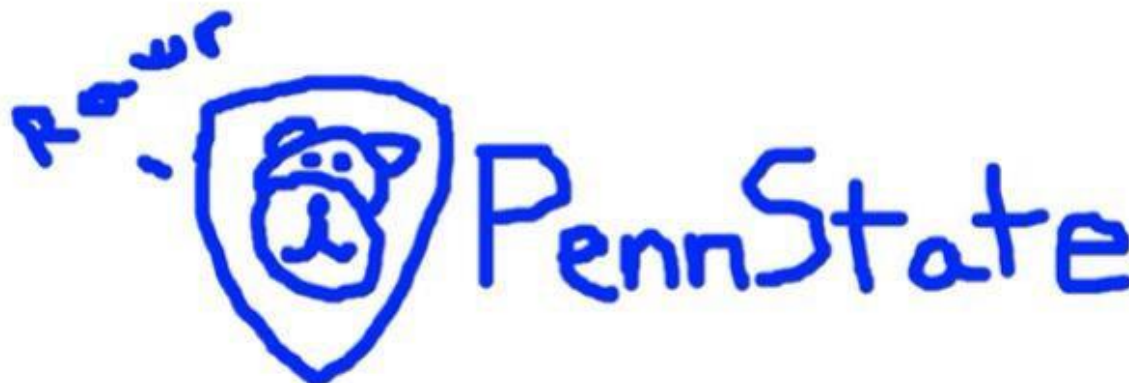
1. an enumerated **bold** point
1. an enumerated *italicized* point

- [html_notebook](http://rmarkdown.rstudio.com/r_notebooks.html) Like [jupyter notebooks](http://jupyter.org)
- a [link](http://psu-psychology.github.io/r-bootcamp) to this bootcamp
- an image: ![rawr](https://www.insidehighered.com/sites/default/server_files/media/PennState2.PNG)

Big idea

Smaller idea in service of bigger

- Supporting point
- Another supporting point
- a **bold** point
- an *italicized* point
- a link to this bootcamp



- an image:

Let's try it with some data

- bootcamp-survey.Rmd
- bootcamp-survey.md

One file, many output options

- 'Default' for the file: `rmarkdown::render("talks/bootcamp-survey.Rmd")`
- **PDF document:** `rmarkdown::render('talks/bootcamp-survey.Rmd', output_format = "pdf_document")`

- **Word document:** `rmarkdown::render('talks/bootcamp-survey.Rmd', output_format = "word_document")`
- **HTML slides:** `rmarkdown::render('talks/bootcamp-survey.Rmd', output_format = "ioslides_presentation")`
- **Multiple outputs:** `rmarkdown::render('talks/bootcamp-survey.Rmd', output_format = c("pdf_document", "word_document", "github_document", "ioslides_presentation"))`

Scripting the pipeline

```
# Get_bootcamp_googlesheet.R
#
# Script to authenticate to Google, extract R bootcamp survey data

library(googlesheets)
library(tidyverse)

survey_url <- "https://docs.google.com/spreadsheets/d/1Ay56u6g4jyEEdlmV2NHxTLBlcjI2gHavta-Ik0kGrpg/edit#gid=111111111"

bootcamp_by_url <- survey_url %>%
  extract_key_from_url() %>%
  gs_key()

bootcamp_sheets <- gs_ws_ls(bootcamp_by_url)

boot_data <- bootcamp_by_url %>%
  gs_read(bootcamp_sheets[1])

names(boot_data) <- c("Timestamp",
                     "R_exp",
                     "GoT",
                     "Age_yrs",
                     "Sleep_hrs",
                     "Fav_date",
                     "Tidy_data")

write_csv(boot_data, path = "data/survey.csv")

# Update_survey.R
#
# Updates Googlesheet survey data and generates new R Markdown report
#

source("R/Get_bootcamp_googlesheet.R")
rmarkdown::render("talks/bootcamp-survey.Rmd",
                  output_format = c("github_document",
                                    "pdf_document",
                                    "word_document",
                                    "ioslides_presentation"))
```

Key points

- Use R scripts to capture & reproduce workflows and/or
- Use R Markdown files for documents, reports, presentations.
 - One or more output formats from the same file.
 - Analysis/lab notebook.
- Use R scripts or functions to automate different pieces of the pipeline.
- Make README files to explain how to put pieces together.

Toward a reproducible psychological science...

- Transparent, reproducible, open workflows pre-publication
- Openly shared materials + data + code
- (Munafò et al. 2017): reproducible practices across the workflow
 - Where to share and when? Lots of options. Let's talk.
- (Gilmore and Adolph 2017): video and reproducibility

Advanced topics

- Write papers in R Markdown using `papaja`
 - Make this from this
- Use R Studio projects
- Version control with git and GitHub
- Web sites, blogs, (even books) with R Markdown
- Scriptable analysis workflows
 - Reports for each participant, e.g. PEEP-II project
 - This bootcamp's `Make_site.R`

My GitHub workflow

1. Create a repo on GitHub
2. Copy repo URL
3. `File/New Project.../`
4. Version Control, Git
5. Paste repo URL
6. Select local name for repo and directory where it lives.
7. Open project within R Studio `File/Open Project...`
8. Commit early & often

Learn from my mistakes

- Script **everything** you possibly can
 - If you have to repeat something, make a function or write a parameterized script
- Document **all the time**
 - Comments in code
 - Update README files
- Don't be afraid to ask
- Don't be afraid to work in the open
- Learn from others
- Just do it!

References

Gilmore, Rick O, and Karen E Adolph. 2017. “Video Can Make Behavioural Science More Reproducible.” *Nature Human Behavior* 1 (12~jun). doi:10.1038/s41562-017-0128.

Munafò, Marcus R, Brian A Nosek, Dorothy V M Bishop, Katherine S Button, Christopher D Chambers, Nathalie Percie du Sert, Uri Simonsohn, Eric-Jan Wagenmakers, Jennifer J Ware, and John P A Ioannidis. 2017. “A Manifesto for Reproducible Science.” *Nature Human Behaviour* 1 (10~jan): 0021. doi:10.1038/s41562-016-0021.