

# Reproducible (and collaborative) science through RStudio

A whirlwind tour with R, RMarkdown, Python, LaTeX, and more

Jenny Rieck & Derek Beaton

May 21 2019

# The big outline

- ▶ Part 0: Background
- ▶ Part 1: A bit about R
- ▶ Part 2: RStudio & Project setup
- ▶ Part 3: R, RMarkdown, & more
- ▶ Part 4: Advanced, beyond, & our favorites

## Part 0: Background

To dive right in

If you want to skip over the background & RStudio, go straight to  
**Part 2: RStudio & Project setup**

# Background

- ▶ This is a taste and to bring you into a bigger world
  - ▶ R, Python, SQL, and JavaScript are critical data science tools/languages
- ▶ R (language and community) strongly emphasizes
  - ▶ Centralization & standards
  - ▶ Rigor & reproducibility (packages, RMarkdown)
- ▶ An interesting language
  - ▶ Functional
  - ▶ With a sublanguage (or dialect?): the tidyverse

## R is a community (actually many communities!)

- ▶ Help and resources
- ▶ Package development and distribution
- ▶ An ideal example
  - ▶ Not quite always that way
  - ▶ Strong communal presence

## R: Help!

- ▶ So many websites e.g., <https://www.statmethods.net/>
- ▶ Online forums (Stack Exchange, r-lists)
- ▶ SpringerLink
  - ▶ All R books for free (pdf format) or for minimal cost (printed)
- ▶ Vignettes
  - ▶ step-by-step instruction guides for packages
- ▶ Git
  - ▶ With open books (via bookdown)
- ▶ Twitter #rstats
- ▶ RStudio (website)
  - ▶ Videos, cheat sheets

# R Packages

- ▶ Packages are bundles of code made by someone (or many people) for everyone to use
  - ▶ There are packages for everything
  - ▶ We'll cover some of the diversity throughout
- ▶ Comprehensive & Reproducible
- ▶ Available primarily on CRAN
  - ▶ But also github (less so: r-forge)

## Part 1: A bit about R

## R Background

- ▶ Created in 1992 by Gentleman & Ihaka

*[we] considered the problem of obtaining decent statistical software for our undergraduate Macintosh lab. After considering the options, we decided that the most satisfactory alternative was to write our own. [...] Finally we added some syntactic sugar to make it look somewhat like S. We call the result “R”.*

# What is R?

- ▶ R is general purpose programming
  - ▶ Design around & for statistics
  - ▶ “for and by statisticians”
- ▶ R is a collection of tools
  - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
  - ▶ No cost, no restrictions
  - ▶ E.g., Microsoft (nee Revolution) R
- ▶ R is a functional language
  - ▶ Turing complete
  - ▶ Mathematical functions
  - ▶ Pass expressions and functions to and from functions

## R: Data types

- ▶ Stored as *vectors*
  - ▶ see `class()`
- ▶ numeric
  - ▶ real or decimal
  - ▶ Includes `NaN`, `Inf`, `-Inf`
- ▶ integer
- ▶ complex
- ▶ character
- ▶ logical
  - ▶ includes `NA`, `TRUE`, `FALSE`
- ▶ factor
  - ▶ factors are usually not your friends
  - ▶ generally: `stringsAsFactors = F` or convert these

## R: factor disasters

```
a_numeric_vector <- c(3, 0, 1, -2, 2, 5,
                     5, 2, 1)
(a_numeric_vector + 1)

## [1] 4 1 2 -1 3 6 6 3 2

(a_numeric2factor_vector <- as.factor(a_numeric_vector))

## [1] 3 0 1 -2 2 5 5 2 1
## Levels: -2 0 1 2 3 5

(as.numeric(a_numeric2factor_vector))

## [1] 5 2 3 1 4 6 6 4 3

(as.numeric(as.character(a_numeric2factor_vector)))

## [1] 3 0 1 -2 2 5 5 2 1
```

## R: Data structures

- ▶ Starts counting from 1
  - ▶ Not 0
- ▶ vector[1]
- ▶ matrix[1,1]
- ▶ array[1,1,1]
- ▶ list[[1]]
  - ▶ Can contain mixtures of types
  - ▶ or list\$name
- ▶ data.frame:
  - ▶ Is technically a list but access in three ways
  - ▶ data.frame[[1]][1]
  - ▶ data.frame[1,1]
  - ▶ data.frame\$name
  - ▶ tibbles: tidyverse data.frames

## R: Some syntax

```
# allowed but not preferred
a_variable = 10 + 1

# preferred
a_variable <- 10 + 1

# a bonus
10 + 1 -> a_variable
```

## R: Some syntax

```
# allowed but not preferred
a.variable = 10 + 1
## dots have 2 meanings in R, with a 3rd
## in the tidyverse

# preferred
a_variable <- 10 + 1
```

## R: Some reservations

- ▶ c, q, t, C, D, F, I, and T (via [https://www.johndcook.com/blog/r\\_language\\_for\\_programmers/](https://www.johndcook.com/blog/r_language_for_programmers/))
- ▶ Except that these can be redefined
  - ▶ With great power comes great responsibility (and danger)

# Tidyverse

- ▶ R is “base R”
- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
  - ▶ Started(?) with/because of ggplot2
  - ▶ A sublanguage or dialect
  - ▶ Can do so because R is functional language
- ▶ Strongly built around a style:
  - ▶ objects are nouns
  - ▶ functions are verbs
- ▶ Learn it!
  - ▶ But don't learn *only* the tidyverse, you'll be lost in base R

## R cheat sheets

- ▶ R to Matlab:  
<https://cran.r-project.org/doc/contrib/Hiebeler-matlabR.pdf>
- ▶ R for programmers: [https://www.johndcook.com/blog/r\\_language\\_for\\_programmers/](https://www.johndcook.com/blog/r_language_for_programmers/)
- ▶ RStudio's giant list:  
<https://www.rstudio.com/resources/cheatsheets/>

## Part 2: RStudio & Project setup

# RStudio

- ▶ IDE: Integrated development environment
- ▶ RStudio: Does so much
  - ▶ We scratch the surface here
- ▶ Quick walk through
- ▶ Followed by specific set up
  - ▶ Generally, but
  - ▶ Also for this workshop

# RStudio Setup

- ▶ Download R and Rstudio
- ▶ Strongly recommend Microsoft R  
(<https://mran.microsoft.com/open>)
  - ▶ Comes with Intel MKL
- ▶ Plain R is fine (<https://cran.r-project.org/>)
  - ▶ Can relink to faster libraries
- ▶ Download RStudio (<https://www.rstudio.com/>)

# RStudio Environment

The screenshot shows the RStudio interface with the following components:

- Code Editor:** Displays R code for creating an ADNI data subset. The code includes library imports, data loading, merging, and variable class modification.
- Console:** Shows statistical summaries for various variables like APOE4, FDG, AV45, CDRSB, ADAS13, and MOCA across different brain regions.
- Environment:** Shows the global environment with objects like `merge_subset`, `ids`, and `MOCA`.
- Files:** Shows the project structure with files like `Renviron`, `2019_Rstudio_Magic.Rproj`, and `README.md`.

```
## ~/workshops/2019_Rstudio_Magic - master - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
+ Go to file/function
+ Run Source All
+ Replace All
Data PTRACCAT Source on Save Find Prev All Replace
+ Match case Whole word Wrap
1 library(ADNImerGE)
2
3 #####
4 #### Load and clean data
5 #####
6
7 ## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA=1
8 admn.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTETHCAT", "PTRACCAT", "APOE4", "FDG",
9 admn.rows <- cadminmerge$VISCODE=="b1" & admnmerge$MOCA==16)
10 merge_subset <- admnmerge[admn.rows, admn.cols]
11
12 ##### remove participants with missing data
13 merge_subset <- merge_subset[complete.cases(merge_subset),]
14
15 ## 0.2 Bring in modified hachkins
16 merge_subset$MSMSCORE <- modhach$MSMSCORE[match(merge_subset$RID, modhach$RID)]
17
18 ## 0.3 Manually change variable classes (remove class 'labelled')
19 +
```

	APOE4	FDG	AV45	CDRSB	ADAS13	MOCA
Min.	:0.0000	Min. :0.6983	Min. :0.8385	Min. :0.0000	Min. :0.0	Min. :16.00
1st Qu.	:0.0000	1st Qu.:1.1100	1st Qu.:1.1100	1st Qu.:0.0000	1st Qu.:8.0	1st Qu.:22.00
Median	:0.0000	Median :1.2802	Median :1.1105	Median :1.0000	Median :10.0	Median :23.00
Mean	:0.5248	Mean :1.2682	Mean :1.1989	Mean :1.202	Mean :11.8	Mean :23.89
3rd Qu.	:1.0000	3rd Qu.:1.3620	3rd Qu.:1.3714	3rd Qu.:2.0000	3rd Qu.:18.0	3rd Qu.:26.00
Max.	:2.0000	Max. :1.7013	Max. :2.0256	Max. :5.500	Max. :46.0	Max. :30.00

```
> view(merge_subset)
> |
```

# RStudio Environment

The screenshot shows the RStudio interface with the following components:

- Script Editor:** Displays a script named `create_ANOVA_data.R` containing R code for data manipulation and analysis.
- Console:** Shows the output of the R code, including statistical summaries for variables like APOE4, FDG, AV45, CDRSB, ADAS13, and MOCA across different brain regions.
- File Browser:** Shows the project structure under `workshops/2019_Rstudio_Magic`, including files like `README.md`, `script.R`, and `plots`.

```
library(ADNImerge)
#####
## Load and clean data
#####
## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA>=1
admin.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTRECAT", "APOE4", "FDG", "AV45", "CDRSB", "ADAS13", "MOCA")
admin.rows <- c(adminmerge$VISCODE=="b1" & adminmerge$MOCA>=16)
admin_subset <- adminmerge[admin.rows,admin.cols]
#####
## remove participants with missing data
admin_subset <- admin_subset[complete.cases(admin_subset),]
#####
## 0.2 Bring in modified hachinski
admin_subset$HMSCORE <- modhach$HMSCORE[match(admin_subset$RID, modhach$RID)]
#####
## 0.3 Manually change variable classes (remove class 'labelled')
admin_subset <- as.data.frame(admin_subset)
```

Variable	APOE4	FDG	AV45	CDRSB	ADAS13	MOCA
Min.	:0.0000	Min. :0.6983	Min. :0.8385	Min. :0.0000	Min. :0.0	Min. :16.00
1st Qu.	:0.0000	1st Qu.:1.1860	1st Qu.:1.1860	1st Qu.:0.0000	1st Qu.:8.0	1st Qu.:22.00
Median	:0.0000	Median :1.2802	Median :1.1105	Median :0.0000	Median :10.0	Median :25.00
Mean	:0.5248	Mean :1.2682	Mean :1.1989	Mean :1.202	Mean :11.8	Mean :23.89
3rd Qu.	:1.0000	3rd Qu.:1.3620	3rd Qu.:1.3714	3rd Qu.:1.2000	3rd Qu.:18.0	3rd Qu.:26.00
Max.	:2.0000	Max. :1.7013	Max. :2.0256	Max. :15.500	Max. :46.0	Max. :30.00

Region	Hippocampus	Midtemp	nPACCtailB	HMSCORE
WholeBrain	Min. :-14.421	Min. :-12.213	Min. :-18.6883	Min. :0.000
1st Qu.	988410	1st Qu.: 6510	1st Qu.: 5535	1st Qu.: 0.051
Median	1051621	Median : 7223	Median : -2.5250	Median : 1.000
Mean	1057026	Mean : 7150	Mean : -3.6882	Mean : 0.588
3rd Qu.	1120570	3rd Qu.: 7834	3rd Qu.: 22088	3rd Qu.: -0.3482
Max.	1486036	Max. :10602	Max. :32189	Max. : 5.3540

# RStudio Environment

The screenshot shows the RStudio interface with several windows open:

- Script Editor:** Displays the R script `create_ADNI_data.R`. The code performs the following steps:
  - Imports required packages: `tidyverse`, `lapply`, `data.table`, and `stringr`.
  - Creates a function `PTRACCAT` that takes a list of data frames and merges them into a single data frame.
  - Specifies column names for the merged data frame.
  - Loads and cleans data from `ADNI_merge` and `MOCA` datasets.
  - Specifies column names and participants for the baseline visit.
  - Removes participants with missing data.
  - Brings in modified hachimaki functions.
  - Manually changes variable classes (removing class `'labelled'`).
- Console:** Shows statistical summaries for various variables across different datasets (e.g., APOE4, FDG, AV45, CDRSB, ADAS13, MOCA, Hippocampus, Midtemp, nPACCtrailsB, HMSCORE). For example, the APOE4 dataset has the following summary statistics:

	Mean	3rd Qu.	Min.	Max.
APOE4	:71.92	:76.60	:69.60	:83.85
FDG	:71.92	:76.60	:69.60	:83.85
AV45	:71.92	:76.60	:69.60	:83.85
CDRSB	:71.92	:76.60	:69.60	:83.85
ADAS13	:71.92	:76.60	:69.60	:83.85
MOCA	:71.92	:76.60	:69.60	:83.85

- Environment:** Shows the global environment with objects like `merge_subset` (665 obs. of 17 variables), `variable_type_map` (num [1:17, 1:3] 0 1 0 0 0 0 0 0 1 0 ...), and `ids` (chr [1:665] "2002" "2003" "2007" "2010" "2011" "2012").
- Files:** A red box highlights the file browser window, which shows the project structure:

  - Home
  - workshops > 2019\_Rstudio\_Magic
  - Renviron
  - 2019\_Rstudio\_Magic.Rproj
  - external
  - mac
  - output
  - R
  - README.md
  - Rmd

FILES, PLOTS, HELP

# RStudio Environment

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** Shows R code for creating an ADNI data subset. The code includes library imports, data loading, merging, and subset selection. A red box highlights the final command: `> view(merge_subset)`.
- Environment:** Shows the `merge_subset` object, which is a data frame with 665 observations and 17 variables. It lists variables like `ADAS13`, `CDRSB`, `MOCA`, and `PTEDUCAT`.
- File Browser:** Shows the project structure under `workshops > 2019_Rstudio_Magic`. It includes files like `README.md`, `environment.Rproj`, and `output`.
- Text Overlay:** A large red text overlay in the center-right area reads "VARIABLES, HISTORY, VERSION CONTROL".

# RStudio Environment

The screenshot displays the RStudio interface with several panes:

- Code pane:** Shows R code for creating an ADNI data subset. The code includes library imports, data loading, cleaning, and subset selection. It uses functions like `library`, `read.csv`, `subset`, and `complete.cases`.
- Console pane:** Displays statistical summaries for variables like APOE4, FDG, AV45, CDRSB, ADAS13, and MOCA. For example, APOE4 has a mean of 71.92 and a median of 70.00. The FDG variable has a range from 89.60 to 208.00.
- Environment pane:** Shows the global environment with objects like `anmerge\_subset` (665 obs., 17 variables), `variable\_type\_map`, `values`, and `functions` (e.g., `scatterplotter`).
- File browser pane:** Shows the project structure with files like `Renvirn`, `2019\_Rstudio\_Magic.Rproj`, `external`, `mac`, `output`, `R`, and `README.md`.

```
library(ADNImerGE)
#####
## Load and clean data
#####
## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA>=1
admin.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTETHCAT", "PTRACCAT", "APOE4", "FDG", "ADAS13", "CDRSB", "MOCA")
admin.rows <- c(adminmerge$VISCODE=="b1" & adminmerge$MOCA>=16)
anmerge_subset <- adminmerge[admin.rows,admin.cols]
#####
## remove participants with missing data
anmerge_subset <- anmerge_subset[complete.cases(anmerge_subset),]
#####
## 0.2 Bring in modified hachkins
anmerge_subset$MSMSCORE <- modhach$MSMSCORE[match(anmerge_subset$RID, modhach$RID)]
#####
## 0.3 Manually change variable classes (remove class 'labelled')
anmerge_subset$FDG <- as.numeric(as.character(anmerge_subset$FDG))
```

	Min.	Q1	Median	Q3	Max.
APOE4	:0.0000	.06983	.08385	.0000	:0.0
FDG	:0.0000	.06983	.08385	.0000	:208.00
AV45	:0.0000	.06983	.08385	.0000	:20.00
CDRSB	:0.0000	.06983	.08385	.0000	:16.00
ADAS13	:0.0000	.06983	.08385	.0000	:16.00
MOCA	:0.0000	.06983	.08385	.0000	:20.00
Wholebrain	:0.0000	.06983	.08385	.0000	:12213
Hippocampus	:0.0000	.06983	.08385	.0000	:18.6883
Midtemp	:0.0000	.06983	.08385	.0000	:18.6883
nPACCtailslB	:0.0000	.06983	.08385	.0000	:18.6883
MSMSCORE	:0.0000	.06983	.08385	.0000	:100.00

# RStudio Environment

The screenshot displays the RStudio interface with several windows open:

- Data Viewer:** A central window titled "DATA VIEWER" showing a table of 665 observations across 17 variables. The variables include DX, AGE, PTGENDER, PTEDUCAT, PTRECAT, PTRACCAT, APOE4, FDG, AV45, CDRSB, ADAS13, MOCA, WholeBrain, and Hippocampus.
- Global Environment:** A window showing the global environment with objects like anerage\_subset, variable\_type\_map, values, and functions.
- File Browser:** A window showing the file structure under "workshops > 2019\_Rstudio\_Magic".
- Console:** A window showing R code and its output, including descriptive statistics for variables like APOE4, FDG, AV45, CDRSB, ADAS13, MOCA, WholeBrain, Hippocampus, Midtemp, nPACCtrailsB, and HMSCore.

# Some benefits of RStudio

- ▶ Built-in integration with version control (git or SVN)
- ▶ Package and documentation generation
- ▶ Reproducible science!
  - ▶ R Markdown documents
    - ▶ Save and execute code
    - ▶ Generate high quality reports that can be shared
  - ▶ Create presentations (like this one!)
  - ▶ Even write papers
  - ▶ Python, D3 (JavaScript), SQL, Shiny, LaTeX, Git/SVN, HTML/CSS, and so much more.
- ▶ This workshop
  - ▶ Will walk you through some of this (and more)
  - ▶ See [https://github.com/jennyrieck/workshops/tree/master/2019\\_Rstudio\\_Magic](https://github.com/jennyrieck/workshops/tree/master/2019_Rstudio_Magic)

## RStudio is more

- ▶ Not just an IDE
- ▶ A company
- ▶ A community
- ▶ A conference
- ▶ A centralized resource

# RStudio Resources

The screenshot shows the RStudio website homepage. At the top, there's a navigation bar with links for Products, Resources, Pricing, About Us, Blogs, and a search icon. Below the navigation is a decorative banner featuring a colorful, abstract graphic of overlapping colored bands.

**RStudio**: A screenshot of the RStudio IDE interface, showing the code editor, workspace, and plots.

**Shiny**: An image of a map of the United States with a "ZIP explorer" interface overlaid.

**R Packages**: Icons for several popular R packages: `markdown`, `Shiny`, `tidyverse`, `knitr`, and `ggplot2`.

**RStudio** description: RStudio makes R easier to use. It includes a code editor, debugging & visualization tools.

**Shiny** description: Shiny helps you make interactive web applications for visualizing data. Bring R data analysis to life.

**R Packages** description: Our developers create popular packages to expand the features of R. Includes `ggplot2`, `dplyr`, `R Markdown` & more.

At the bottom, there are download and learn more buttons for each section, and a horizontal orange progress bar.

# RStudio Resources

Online Learning - RStudio

https://www.rstudio.com/online-learning/

R Studio

Products Resources Pricing About Us Blogs

## Online learning

A wealth of tutorials, articles, and examples exist to help you learn R and its extensions. Scroll down or click a link below for a curated guide to learning R and its extensions.

- R Programming
- Shiny
- R Markdown
- Data Science
- Books

R Programming  
Read More >

Shiny  
Read More >

R Markdown  
Read More >

Data Science  
Read More >

# RStudio Resources

Cheatsheets - RStudio x + - □ x

https://www.rstudio.com/resources/cheatsheets/

R Studio Products Resources Pricing About Us Blogs SEARCH

## RStudio Cheat Sheets

The cheat sheets below make it easy to learn about and use some of our favorite packages. From time to time, we will add new cheat sheets to the gallery. If you'd like us to drop you an email when we do, let us know by clicking the button to the right.

SUBSCRIBE TO CHEAT SHEET UPDATES HERE

- RStudio IDE
- R Markdown
- Shiny
- Package Development
- Data Import
- Data Transformation with dplyr
- Data Visualization with ggplot2
- Apply functions with purrr
- Deep Learning with Keras
- Data Science in Spark with Sparklyr
- String manipulation with stringr
- Dates and times with lubridate

### Python with R and Reticulate Cheat Sheet

The reticulate package provides a comprehensive set of tools for interoperability between Python and R. With reticulate, you can call Python from R in a variety of ways including importing Python modules into R scripts, writing R Markdown Python chunks, sourcing Python scripts, and using Python interactively within the RStudio IDE. This cheatsheet will remind you how.  
Updated 4/19.

Use Python with R with reticulate :: CHEAT SHEET

The reticulate package makes it easy to have and use Python in R. It includes functions, imports, and a Python interface.

Python in R Markdown

Object Conversion

Helpers



## Project and Environment Setup

- ▶ NEED CLEAN UP STARTING HERE.
- ▶ Special & hidden files
- ▶ Having a structure

## RStudio Setup

- ▶ See <https://jennybc.github.io/2014-05-12-ubc/r-setup.html> for a detailed guide

## For safety & collaboration

- ▶ Project(s) files
  - ▶ SOMETHING!

# Projects through Git

- ▶ Create a new project File

New Project

Create Project

---

 **New Directory**  
Start a project in a brand new working directory >

---

 **Existing Directory**  
Associate a project with an existing working directory >

---

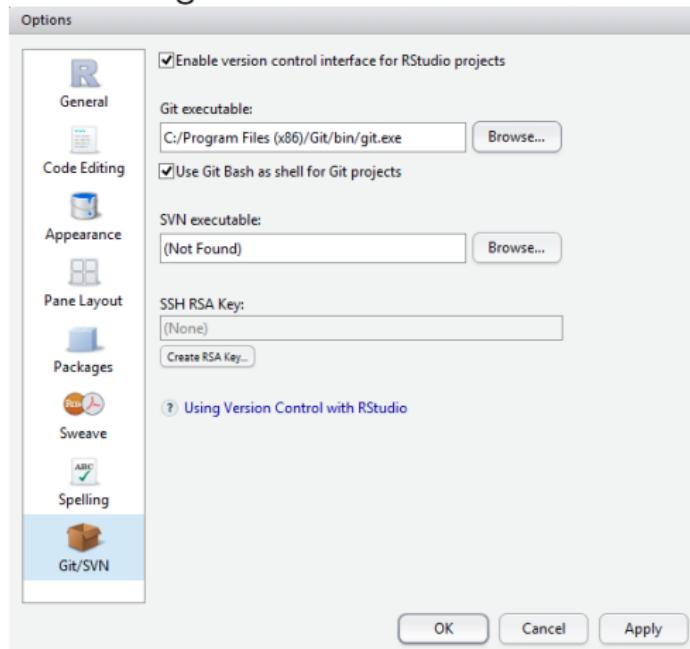
 **Version Control**  
Checkout a project from a version control repository >

Cancel

# Git & Projects

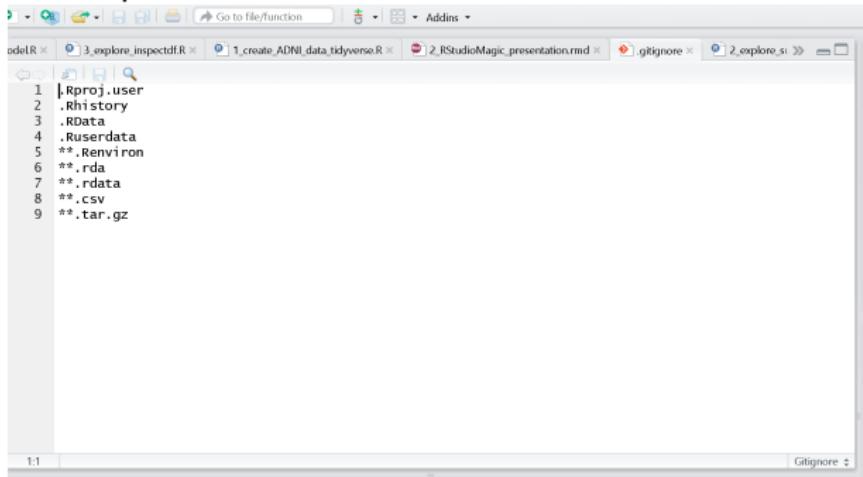
## ► Git

- Download git and link executable within RStudio



# Format .gitignore

- ▶ File types to ignore via version control
  - ▶ \*\* before each extension will match directories anywhere in the repo

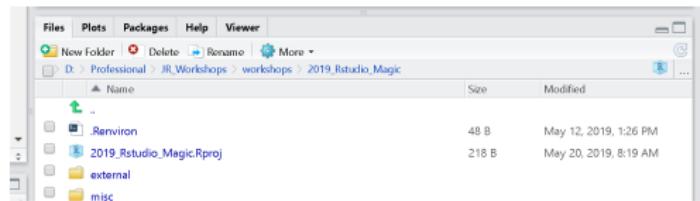


The screenshot shows the RStudio interface with the .gitignore tab selected in the top navigation bar. The main workspace displays the following content in the .gitignore file:

```
1 |Rproj.user
2 .Rhistory
3 .RData
4 .Ruserdata
5 **.Renvironment
6 **.rda
7 **.rdata
8 **.CSV
9 **.tar.gz
```

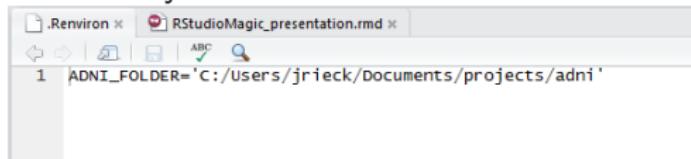
The code editor at the bottom shows the number "1:1" and the word "Gitignore" in the status bar.

# Environment variables



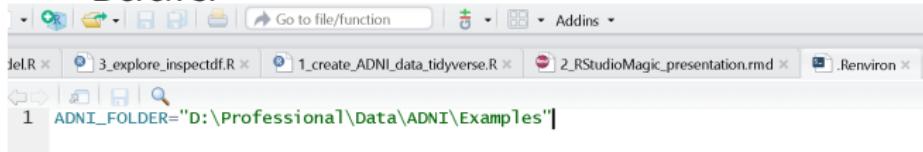
# Format environmental variables

- ▶ Set environmental variables (ie, directory location of data) to make code generalizable across computers
  - ▶ Don't commit or share these
- ▶ In **your** project folder create a `.Renvironment` file and define variables
  - ▶ Jenny's:



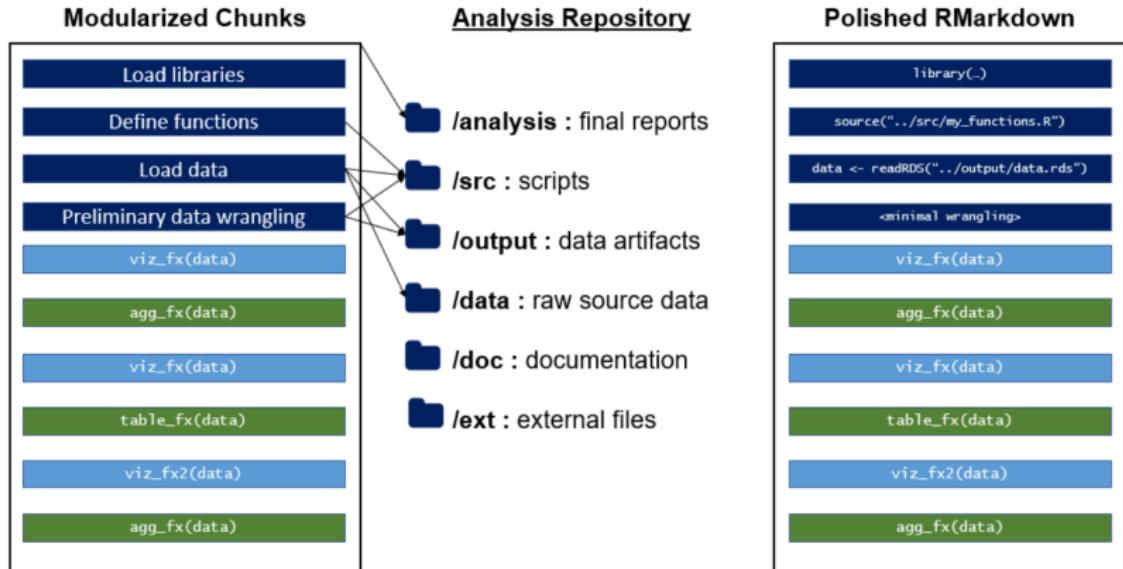
```
1 ADNI_FOLDER='C:/Users/jrieck/Documents/projects/adni'
```

- ▶ Derek's:



```
1 ADNI_FOLDER="D:\Professional\Data\ADNI\Examples"
```

# Organize your project folders and markdown



<https://emilyriederer.netlify.com/post/rmarkdown-driven-development/>

## Organize your project folders and markdown

- ▶ What works for you?
- ▶ What works for your organization or team?
- ▶ Maximize utility, minimize complexity

# This works for us

 [jennyrieck / workshops](#)

 Watch ▾ 1     Star 0     Fork 0

 Code     Issues 0     Pull requests 0     Projects 0     Wiki     Insights     Settings

Branch: master ▾ [workshops / 2019\\_Rstudio\\_Magic /](#)

 Create new file     Upload files     Find file     History

 jennyrieck added our favoRite things ...    Latest commit d818f26 6 hours ago

..

	R	more updates to manuscript example!	23 hours ago
	Rmd	added our favoRite things	6 hours ago
	external/images	reorganizing pngs	6 hours ago
	misc	reorganizing pngs	6 hours ago
	2019_Rstudio_Magic.Rproj	initial folder structure	5 days ago
	README.md	create readme	5 days ago
 README.md			
Rstudio magic for BrainHack Toronto 2019			

# This works for us

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights

Branch: master workshops / 2019\_Rstudio\_Magic / R / Create new file Upload files Find file History

derekbeaton almost done now we hope Latest commit e87b65d 16 hours ago

..

<a href="#">0_create_ADNI_data_base.R</a>	fixed rownames/race coding	8 days ago
<a href="#">1_create_ADNI_data_tidyverse.R</a>	fixed rownames/race coding	8 days ago
<a href="#">2_explore_summarytools.R</a>	almost done now we hope	16 hours ago
<a href="#">3_explore_inspectdf.R</a>	almost done now we hope	16 hours ago
<a href="#">4_explore_DataExplorer_one_liner.R</a>	almost done now we hope	16 hours ago
<a href="#">5_linear_model.R</a>	almost done now we hope	16 hours ago
<a href="#">6_covstatis_example.R</a>	please don't collide.	2 days ago

# This works for us

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights

Branch: master workshops / 2019\_Rstudio\_Magic / Rmd / Create new file Upload files Find file History

derekbeaton small update Latest commit 74c5384 14 hours ago

1\_a\_Simple\_RMarkdown\_PDF\_files/figure-latex more updates to manuscript example! 3 days ago

3\_RMarkdown APA Manuscript\_files updated numbers & structures 16 hours ago

1\_a\_Simple\_RMarkdown\_PDF.Rmd almost done now we hope 16 hours ago

1\_a\_Simple\_RMarkdown\_PDF.log whatever 2 days ago

1\_a\_Simple\_RMarkdown\_PDF.pdf more updates to manuscript example! 3 days ago

1\_a\_Simple\_RMarkdown\_PDF.tex tons of bells-and-whistles via the manuscript. 4 days ago

2\_RStudioMagic\_presentation.pdf small update 14 hours ago

2\_RStudioMagic\_presentation.rmd small update 14 hours ago

2\_RStudioMagic\_presentation.tex small update 14 hours ago

3\_RMarkdown APA Manuscript.Rmd updated numbers & structures 16 hours ago

3\_RMarkdown APA Manuscript.docx updated numbers & structures 16 hours ago

3\_RMarkdown APA Manuscript.pdf updated numbers & structures 16 hours ago

3\_RMarkdown APA Manuscript.tex updated numbers & structures 16 hours ago

r-references.bib updated numbers & structures 16 hours ago

## Get the packages you need

```
# to install from CRAN
install.packages("devtools", dependencies = TRUE)

# to install from a git (requires the
# devtools package)
dev.tools::install_github(Gibbsdavidl/CatterPlots)

# to install from a file
install.packages("/mypath/to/package/ADNIMERGE.tar.gz",
                 type = "source", repos = NULL)
```

## Read in and create your dataframe

- ▶ ADNI Dataset adnimerge package
  - ▶ Reduce full dataset to only those participants (rows) and variables (columns) you're interested in
- ▶ Two methods to create your dataframe
  - ▶ using base R functions: 0\_create\_ADNI\_data\_base.R
  - ▶ Using tidyverse functions:  
`1_create_ADNI_data_tidyverse.R`

# ADNI & ADNIMERGE

## Cleaning: Base

```
library(ADNIMERGE)
adni.cols <- c('RID', 'VISCODE', 'DX', 'AGE', 'PTGENDER',
adni.rows <- c(adnimerge$VISCODE=='bl' & adnimerge$MOCA>=10)
amerge_subset <- adnimerge[adni.rows,adni.cols]
amerge_subset <- amerge_subset[complete.cases(amerge_subset)]
amerge_subset$HMSCORE <- modhach$HMSCORE[match(amerge_subset$VISCODE, modhach$VISCODE)]
```

## Cleaning: tidyverse

```
library(ADNIMERGE)
adnimerge %>%
  dplyr::select(RID, VISCODE, DX, AGE, PTGENDER, PTEDUCAT,
  filter(VISCODE == "bl") %>%
  filter(MOCA >= 16) %>%
  drop_na() -> amerge_subset

amerge_subset %<>% inner_join(modhach[,c("RID", "HMSCORE")])
```

## Exploring your data

- ▶ Many packages to help explore and describe your data:
  - ▶ `summarytools`: `2_explore_summarytools.R`
  - ▶ `inspectdf`: `3_explore_inspectdf.R`
  - ▶ `DataExplorer`: `4_explore_DataExplorer_one_liner.R`

Code w/ eval=F

## Hard Break

- ▶ DataExplorer is dangerous
- ▶ Blind analyses can be *criminal*
  - ▶ de Leeuw paper quote
  - ▶ DEREK RANTS, PER USUAL.

## Analyze your data

- ▶ Linear models: 5\_linear\_model.R

## Screenshots / Code w/ eval=F

## Get experimental

- ▶ Explain motivation, not method
- ▶ covSTATIS: `6_covstatis_example.R`

## Part 3: R, RMarkdown, & more

# RMarkdown

- ▶ What it is /why to use it
- ▶ A short deviation for LaTeX, and new helpers: kable & kableExtra
  - ▶ A taxonomy and how to approach this *Tying it all together through here* 1: simple RMD Plot-based visuals
    - ▶ Base, gt, ggplot, grobTable()/grid/gridExtra
    - ▶ 2: Slides (these ones here)
    - ▶ 3: Manuscripts!!
- ▶ Reporting/presentin

## RMarkdown Don(u)'ts

- ▶ Don't hardcode values
- ▶ Don't hardcode absolute file paths
- ▶ Don't do complicated database queries
- ▶ Don't litter
  - ▶ avoid eval=FALSE
  - ▶ reduce repeated code by making functions
- ▶ Don't load unnecessary libraries
- ▶ More at: <https://emilyriederer.netlify.com/post/rmarkdown-driven-development/>

## Part 4: Advanced R

## Some advanced/other things we're not covering

- ▶ package development
- ▶ Shiny
- ▶ SQL
- ▶ C/C++
- ▶ R2D3

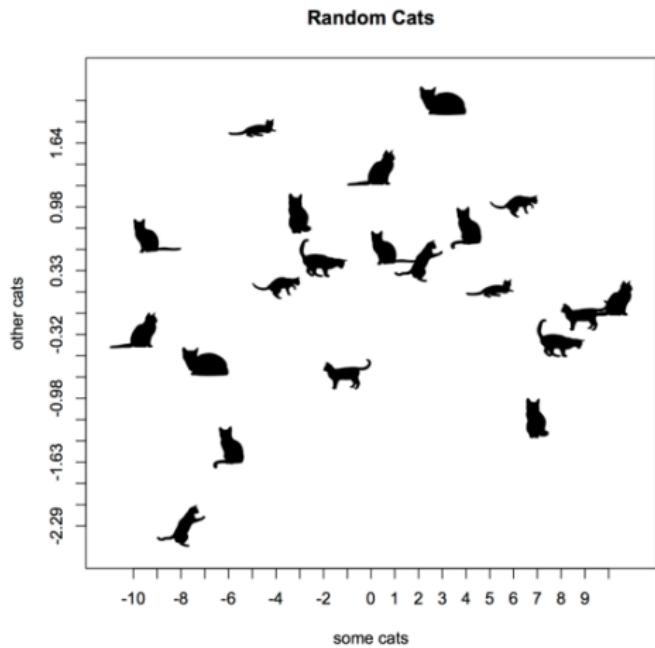
## A few of our favorite things

- ▶ Fun R do-dads

# CatterPlot for feline based graphics:

► <https://github.com/Gibbsdavidl/CatterPlots>

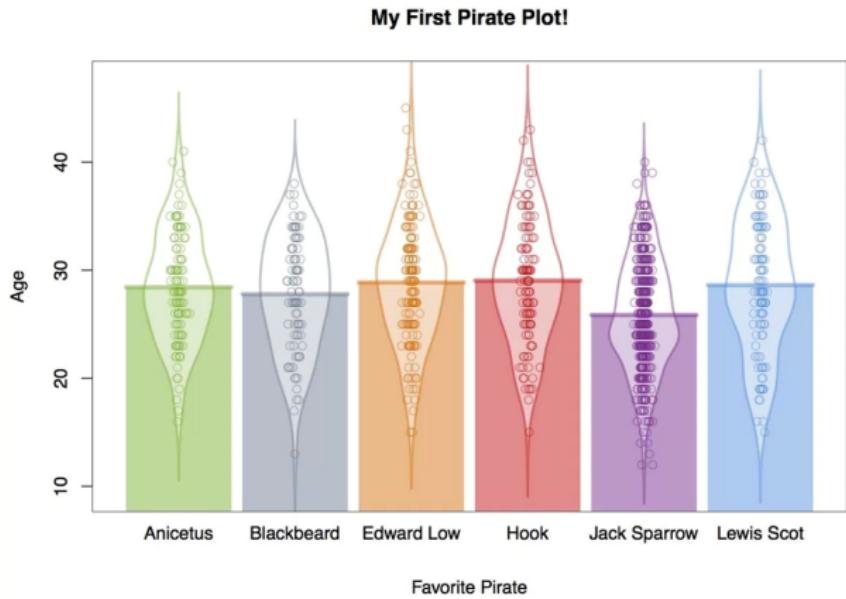
```
dev.tools::install_github(Gibbsdavidl/CatterPlots)
```



# What's a pirate's favorite programming language?

► <https://cran.r-project.org/web/packages/yarr/vignettes/pirateplot.html>

```
install.packages('yarr')
```

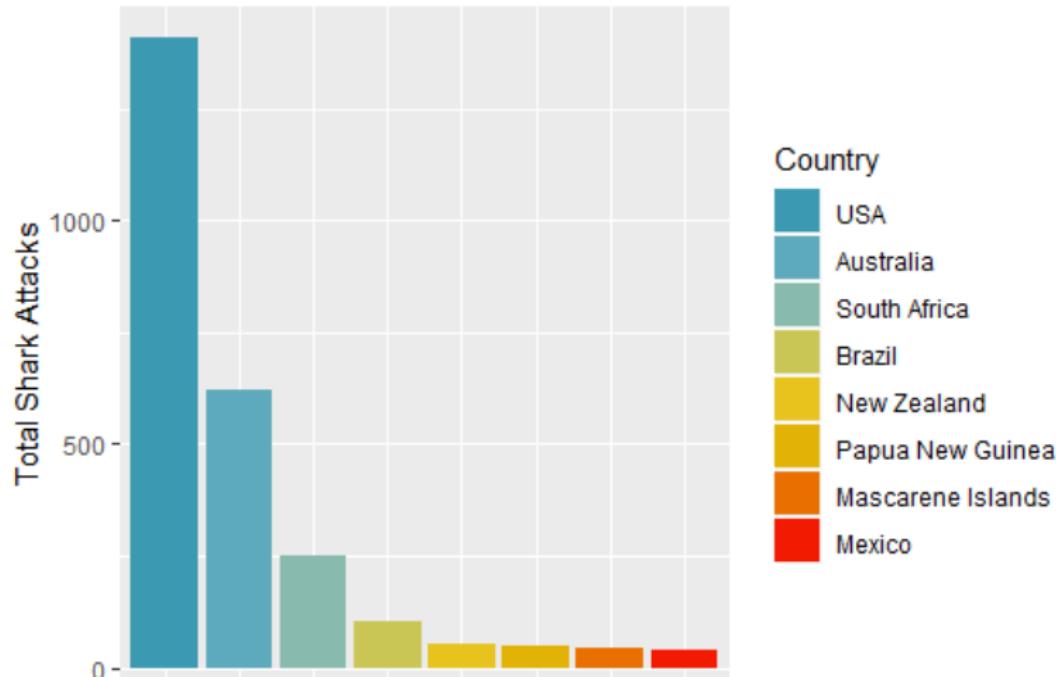


# Color palettes to fit your mood

► <https://github.com/karthik/wesanderson>

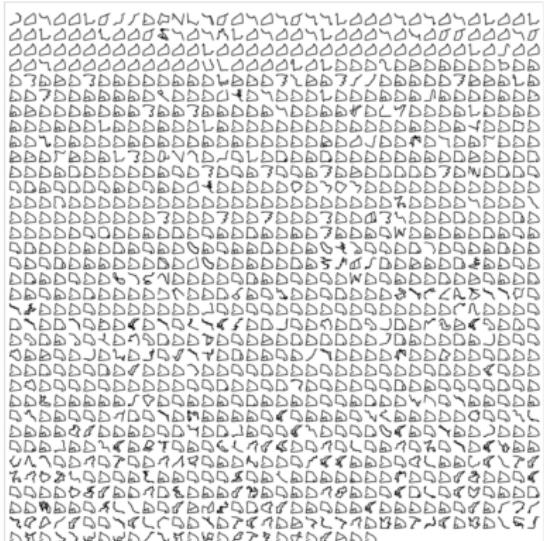
```
dev.tools::install_github(karthik/wesanderson)
```

Top countries with shark attacks  
(Esteban was eaten)



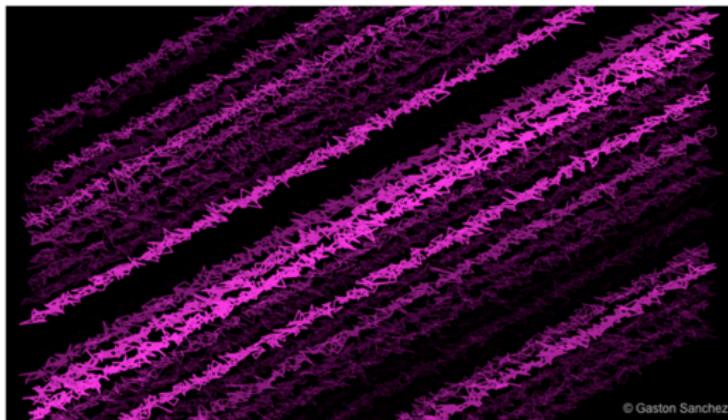
# Mapping your Strava routes

- ▶ <https://www.r-bloggers.com/strava-rides-map-in-r/>
- ▶ ALSO <https://marcusvolz.com/?p=4068>
  - ▶ `dev.tools::install_github(marcusvolz/strava)`



# Make aRt!

- ▶ R Graph Gallery
  - ▶ <http://www.r-graph-gallery.com/>
- ▶ Rtist: Gaston Sanchez
  - ▶ <http://gastonsanchez.com/Rtist/>



```
# -----
# Pink Barbs
# -----
# generate points x-y values
x <- seq(0, 100, length = 1000)
y <- x + rnorm(1000)

# -----
# Pink Barbs
# -----
# see graphical parameters
op <- par(bg = "black", mar = rep(0, 4))
# plot
plot(x, y, type = "n")
for (i in seq(-80, 70, by = 5))
{
  lines(x + rnorm(1000), x + i + rnorm(1000, 2), pch = 19,
        lwd = rnorm(2.4), lty = i, runif(1000),
        lwd = sample(seq(0.1, 2, length = 20), 1))
}
# signature
legend("bottomright", legend = "@ Gaston Sanchez", bty = "n",
       text.col = "gray77")
# reset par
par(op)
dev.off()
```