

Reproducible toolbox

Reproducible research toolkit



Reproducibility checklist

What does it mean for a data analysis to be "reproducible"?

Reproducibility checklist

What does it mean for a data analysis to be "reproducible"?

Near-term goals:

- Are the tables and figures reproducible from the code and data?
- Does the code actually do what you think it does?
- In addition to what was done, is it clear **why** it was done? (e.g., how were parameter settings chosen?)

Long-term goals:

- Can the code be used for other data?
- Can you extend the code to do other things?

Meet the toolkit

Primary tool: R



```
side_one <- 3
side_two <- 4
hypotenuse <- sqrt(side_one^2 + side_two^2)
result <- paste(
  "A triangle with sides of length",
  side_one,
  "and length",
  side_two,
  "has a hypotenuse of length",
  hypotenuse
)
print(result)
```

```
## [1] A triangle with sides of
## length 3 and length 4 has a
## hypotenuse of length 5
```

Meet the toolkit

Utility belt: tidyverse packages



R packages for data science

The tidyverse is an opinionated **collection of R packages** designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

Install the complete tidyverse with:

```
install.packages("tidyverse")
```

Meet the toolkit

Workshop: RStudio Server

A screenshot of the RStudio Server interface accessed via a web browser. The browser address bar shows 'https://rstudio.cos.gmu.edu'. The RStudio interface includes a menu bar (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help), a toolbar, and a main workspace. The workspace is divided into several panes: a Console/Terminal pane on the left showing R version information and help text; an Environment pane on the top right; a Files pane on the bottom right showing a file explorer view of the current project directory. The file explorer shows files like .gitignore, .Rhistory, _output.yaml, DESCRIPTION, Makefile, README.md, and rstudio-server-tutorial.Rproj. The terminal pane contains the following text:

```
~/rstudio-server-tutorial/ ↵  
  
R version 3.5.0 (2018-04-23) -- "Joy in Playing"  
Copyright (C) 2018 The R Foundation for Statistical Computing  
Platform: x86_64-redhat-linux-gnu (64-bit)  
  
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.  
  
Natural language support but running in an English locale  
  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> |
```

Meet the toolkit

Storage room: GitHub

A screenshot of the GitHub website's sign-up page. The browser's address bar shows "https://github.com". The navigation bar includes links for "Features", "Business", "Explore", "Marketplace", and "Pricing", along with a search bar and "Sign in" or "Sign up" options. The main content area has a dark background with a circuit-like pattern. On the left, the text reads "Built for developers" followed by a paragraph: "GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside millions of other developers." On the right, a white sign-up form is displayed with fields for "Username" (placeholder: "Pick a username"), "Email" (placeholder: "you@example.com"), and "Password" (placeholder: "Create a password"). Below the password field is a note: "Use at least one letter, one numeral, and seven characters." A green "Sign up for GitHub" button is at the bottom of the form, with a small disclaimer below it: "By clicking 'Sign up for GitHub', you agree to our [terms of service](#) and [privacy statement](#). We'll occasionally send you account related emails."

Some R history

- The first stable version of R, v1.0.0, was released on February 29, 2000.
- R itself is an implementation of the **S programming language**, which was designed at Bell Laboratories in the mid-1970s.
- *Base R* was built for statisticians and for doing data analysis, but not necessarily for modern Data Science
- It's age and legacy brings along old implementations of data structures and abbreviated function (commands) names

Source: David Smith, *Over 16 years of R project history*, *Revolutions blog*, last updated on March 4, 2016, accessed September 20, 2017, <http://blog.revolutionanalytics.com/2016/03/16-years-of-r-history.html>

Modernizing R with tidyverse

Over the last 3 years, chief scientist at RStudio, Hadley Wickham, has brought R into the modern era with the `tidyverse`.

The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying philosophy and common APIs.

– [Front page of the Tidyverse website](#)

In practice, this meant reducing everything to a small, core set of commands that all behave in a similar way.

R essentials

A short list (for now):

R essentials

A short list (for now):

- Functions are (most often) verbs, followed by what they will be applied to in parentheses:

```
do_this(to_this)
do_that(to_this, to_that, with_those)
```

R essentials

A short list (for now):

- Functions are (most often) verbs, followed by what they will be applied to in parentheses:

```
do_this(to_this)
do_that(to_this, to_that, with_those)
```

- Packages are installed with the `install.packages` function and loaded with the `library` function, once per session:

```
install.packages("package_name")
library(package_name)
```

R essentials

A short list (for now):

- Functions are (most often) verbs, followed by what they will be applied to in parentheses:

```
do_this(to_this)
do_that(to_this, to_that, with_those)
```

- Packages are installed with the `install.packages` function and loaded with the `library` function, once per session:

```
install.packages("package_name")
library(package_name)
```

- Packages bring in additional functions for us to use!

R Markdown

```
1- ---
2- title: "Hello R Markdown!"
3- author: "CDS 101"
4- output: html_document
5- ---
6-
7- ```{r setup, include=FALSE}
8- knitr::opts_chunk$set(echo = TRUE)
9- ```
10-
11- ## R Markdown
12-
13- This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on
14- using R Markdown see <http://rmarkdown.rstudio.com>.
15-
16- When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code
17- chunks within the document. You can embed an R code chunk like this:
18-
19- ```{r cars}
20- summary(cars)
21- ```
22-
23- ## Including Plots
24-
25- You can also embed plots, for example:
26-
27- ```{r pressure, echo=FALSE}
28- plot(pressure)
29- ```
30-
31- Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.
```

- Fully reproducible reports – each time you knit the analysis is ran from the beginning
- Simple markdown syntax for text
- Code goes in chunks, defined by three back-ticks, narrative goes outside of chunks

R Markdown

```
1- ---
2 title: "Hello R Markdown!"
3 author: "CDS 101"
4 output: html_document
5 ---
6
7- ```{r setup, include=FALSE}
8 knitr::opts_chunk$set(echo = TRUE)
9- ```
10
11- ## R Markdown
12
13 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on
using R Markdown see <http://rmarkdown.rstudio.com>.
14
15 When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code
chunks within the document. You can embed an R code chunk like this:
16
17- ```{r cars}
18 summary(cars)
19- ```
20
21- ## Including Plots
22
23 You can also embed plots, for example:
24
25- ```{r pressure, echo=FALSE}
26 plot(pressure)
27- ```
28
29 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.
30
```

If it works for you, then it works for me!

How will we use R Markdown?

- You will submit all your homework assignments and the final group project as an R Markdown document
- You will be shown examples in the form of "real-time" demos inside an R Markdown document. While watching, you should create your own document and follow along!
- For assignments, you'll always have a template R Markdown document to start with

Credits

License

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International

Acknowledgments

Content adapted from the [Meet the toolkit slides](#) by Mine Çetinkaya-Rundel and made available under the [CC BY 4.0 license](#).