Meet the toolkit

Intro to Data Science

Shawn Santo

Today's agenda

- R and RStudio
- R Markdown
- Git and GitHub
- Reproducible data analysis

R and RStudio

What are R and RStudio?

- R is a statistical programming language.
- RStudio is a convenient interface for R (an integrated development environment, IDE).
- At its simplest:
  - R is like a car's engine
  - RStudio is like a car's dashboard

R: Engine

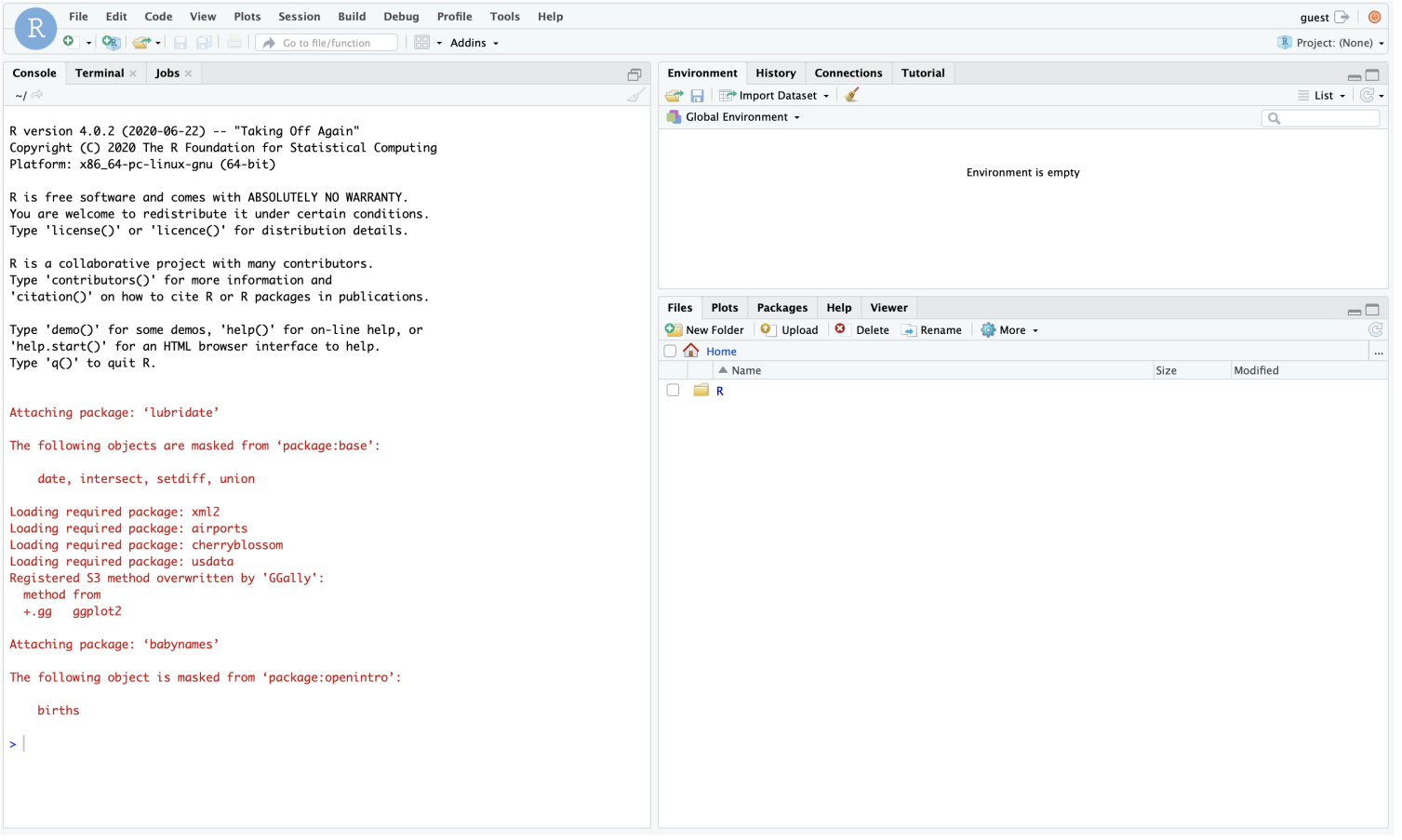
RStudio: Dashboard

The RStudio interface makes working with R much easier.

Source: Modern Dive

RStudio

Access RStudio at https://vm-manage.oit.duke.edu/containers/rstudio



Package `tidyverse`

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

R essentials (a short overview)

Packages are installed with the `install.packages()` function and loaded with function `library()` once per session.

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

Data frames are rectangular objects that contain data. Each column in a data frame is a **vector**.

```
#>                    mpg cyl    wt
#> Mazda RX4         21.0   6 2.620
#> Mazda RX4 Wag     21.0   6 2.875
#> Datsun 710        22.8   4 2.320
#> Hornet 4 Drive    21.4   6 3.215
#> Hornet Sportabout 18.7   8 3.440
#> Valiant           18.1   6 3.460
```

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 Markdown

R Markdown

- Generate fully reproducible reports - the analysis is run from the beginning each time you knit
- Simple Markdown syntax for text
- Code goes in chunks, defined by three backticks, narrative goes outside of chunks

Sample R Markdown syntax

Header syntax	Example
# Level one	Level one
## Level two	Level two
### Level three	Level three
#### Level four	Level four
##### Level five	Level five
###### Level six	Level six

Syntax	Example
bold text	**bold text**
italicized text	*italicized text*
- one - two - three	• one • two • three
`` `in-line code` ``	`in-line code`

RStudio and R Markdown tour

First, recall "The murky tale of Flint's deceptive water data"

- How many samples were taken from each sampled home?
- What is the EPA action level?

Live demo

- Access RStudio: https://vm-manage.oit.duke.edu/containers/rstudio
- Create your private repository of today's notes: https://classroom.github.com/a/Tpy7C0oC

R Markdown tips

Resources

- R Markdown cheat sheet
- In RStudio, Markdown Quick Reference:
    - `Help -> Markdown Quick Reference`

Remember: The workspace of the R Markdown document is separate from the console.

Workspace vs. console

Run the following in your console.

```
x <- 2
x * 3
```

Then, add the following chunk in your R Markdown document and knit.

```
x * 3
```

What happens? Why the error?

How will we use R Markdown?

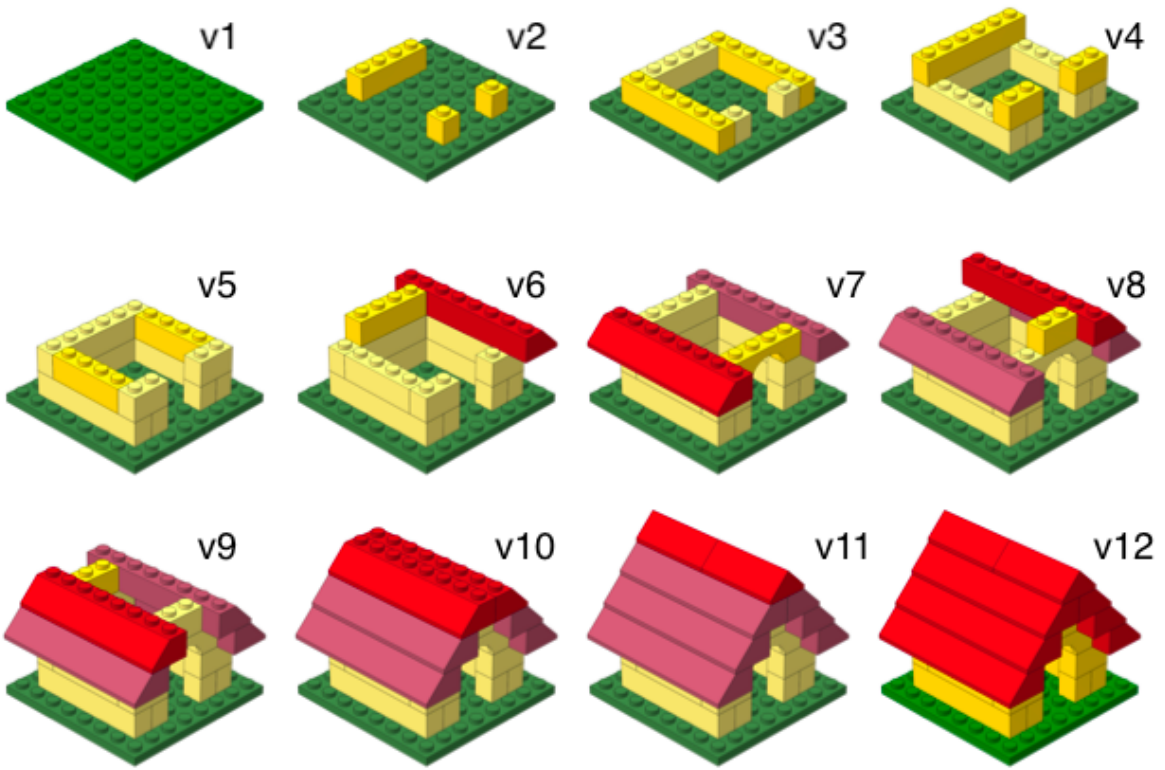
- Every homework, lab, and project will involve an R Markdown document.
- You'll always have a template R Markdown document to start with; however, the amount of scaffolding in the template will decrease over the semester.

Git and GitHub

Version control

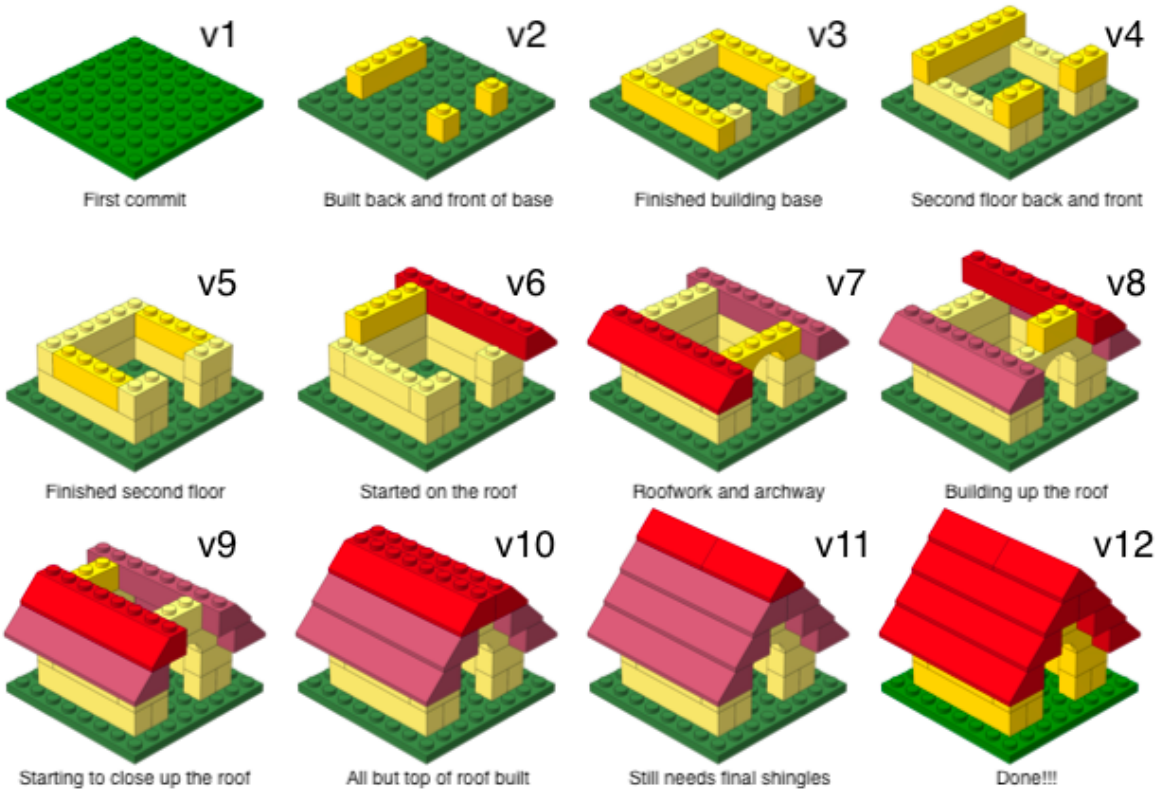
- We introduced GitHub as a platform for collaboration
- But it's much more than that...
- It's actually designed for version control

Versioning



Versioning

We can add human readable messages.



Git and GitHub tips

- **Git** is a version control system, similar to “Track Changes” features from Microsoft Word.
- **GitHub** is the home for your Git-based projects on the internet (like DropBox but much better).
- There are a lot of Git commands and very few people know them all; most of the time you will use

```
git add
git commit
git push
git pull
```

  All of these commands can be executed through RStudio's Git tab.

Git and GitHub tips

- We will be using git and interfacing with GitHub through RStudio
    - If you Google for help you might come across methods for doing these things in the command line -- skip that and move on to the next resource unless you feel comfortable trying it out.

- There is a great resource for working with git and R: happygitwithr.com.
    - Some of the content in there is beyond the scope of this course, but it's a good place to look for help.

Git and GitHub live demo

- Concepts introduced:
    - Connect an R project to GitHub repository
    - Working with a local and remote repository
    - Making a change locally, committing, and pushing

- In Lab 01 you will go through the full version control cycle. As the semester progresses we will guide you in using Git/GitHub in a team-based environment.

Reproducible data analysis

Reproducibility checklist

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

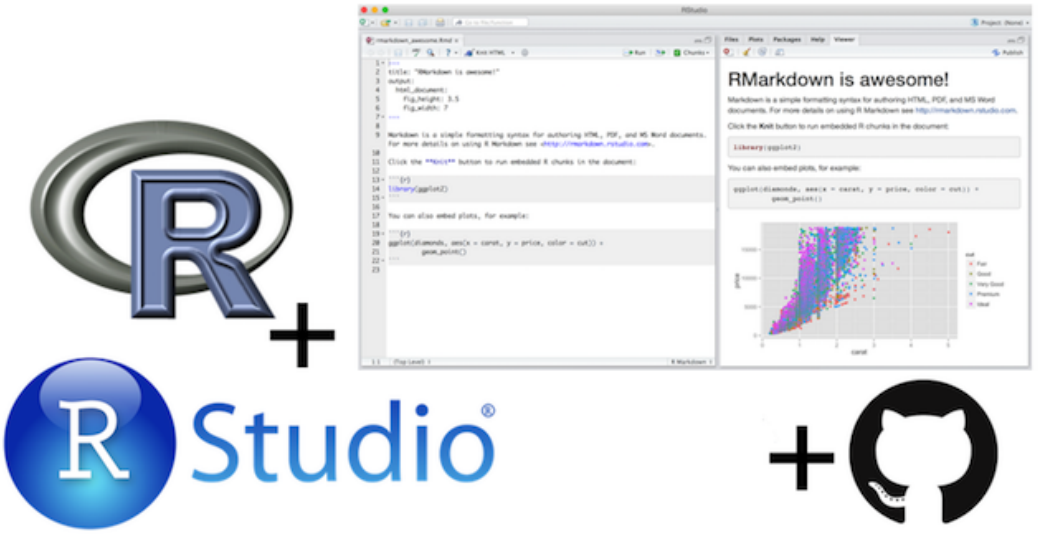
Near-term goals:

- Are tables and figures reproducible from the code and data?
- Does 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 updates to the current data?
- Can the code be used for other data?
- Can you extend the code to do other things?

Toolkit



- Scriptability $\rightarrow$ R
- Literate programming (code, narrative, output in one place) $\rightarrow$ R Markdown
- Version control $\rightarrow$ Git / GitHub

Recap

Can you answer these questions?

- What is a reproducible data analysis, and why is it important?
- What is version control, and why is it important?
- What is R vs. RStudio?
- What is git vs. GitHub?

Concepts introduced:

- Cloning a project from GitHub to RStudio
- Knitting documents
- R Markdown and (some) R syntax
- Console
- Using R as a calculator
- Environment
- Loading and viewing a data frame
- Accessing a variable in a data frame
- R functions

References

1. McConville, C. (2019). Statistical Inference via Data Science. Moderndive.com. Retrieved from https://moderndive.com/