StatCafe

Eli Kravitz

2/28/2019

Introduction

- I will give an **opinionated** way for you to manage your research workflow in R.

    - I've learned this through trial and error... mostly error.

- Take parts of this you like, ignore parts you don't

- Many people are more qualified to talk about this

    - none of them will come here and do it for free. You're stuck with me

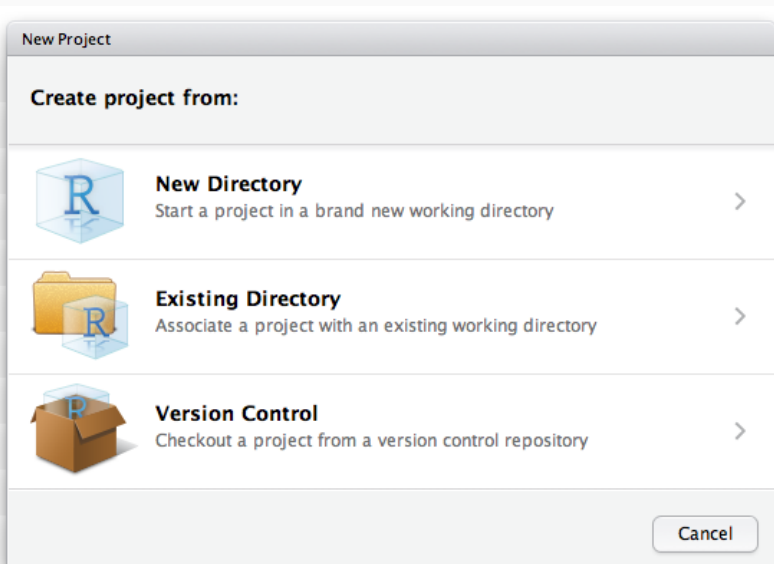## Scenarios (totally not based on personal experience)

- **Scenarios 1**: You work on a project, complete 90% of it, then stop working on it for 2 years.

  - **Nothing works** on your new computer, you can't locate all the data and files you need.

- **Scenarios 2**: You have a simulation working perfectly. You make a lot of changes. **A few days later** you can't remember why you made these changes, and now **your simulation does not work correctly**.

- **Scenarios 3**: You decide to rewrite most of the paper, show it to your advisor, and they tell you it was better before.

  - You only have the old draft of the paper in PDF. You DO NOT want to retype it.

Storing Everything: Rprojects

- I organize all my research projects the same way.

  - Show folder structure

- Make the directory a Rproject.

- R projects let you:

  - Divide your work naturally
  - Store environments for every project
  - "freeze" R in a state that works (advanced, won't cover)

- Integrates nicely other things we'll cover

Making an R project

- Go to File → New Project

- Follow the prompts to make your project.

- You're done. Easy.

- Now do this **every time** you start a new research project.

Relative Filepaths and here()

- Does your R code start with this?

```
setwd("C:\Users\Eli\path\that\only\I\have")
mydata = read.csv(data.csv)
```

- This is **bad**. As soon as you rename or move directories, it breaks.

- What if someone else wants to use your code? What if you want to use a different computer?

Relative Filepaths and here()

- Putting your files in an Rproject already helps. The directory is to to where the .Rproj file is.

```
getwd()
```

```
## [1] "/Users/kravitze/Documents/SafeCafe/Code"
```

- Let's do better with the here package.

Relative Filepaths and here()

- Here looks for a .Rproj file and makes file paths relative

```
here()
```

```
## [1] "/Users/kravitze/Documents/SafeCafe"
```

```
here("Inner_Directory")
```

```
## [1] "/Users/kravitze/Documents/SafeCafe/Inner_Directory"
```

- Code won't need to change between computers

- Example: Load data

```
my_data = here("Data", "my_file.csv")
```

- Now nothing is tied to the filepaths on our computer.

- How can we **back everything up** and **make it available on multiple computers**?

- Github!

Saving Progress with Github

- Setting up Github is simple, but many steps.
    - See links at the end for instructions
- I will go over what it can do for you, and why you should work with it.

Saving Progress with Github

- Github let's you track versions on your file

- Github saves the history of your file

- Made changes to your code $\rightarrow$ you can **revert back to an earlier**. version.

- Works with .tex too!