

# Intro to R and RStudio

FNCE5352

1/19/2022

# Matt McDonald - CV

## Work:



## Education:



MS Statistics



University of  
Connecticut

MBA Finance



Colgate University  
BA Mathematics

# Career Path



# Class Syllabus

# Install R & RStudio

- <https://stat545.com/install.html>

# R vs Rstudio

R: Engine



RStudio: Interface



# R Packages

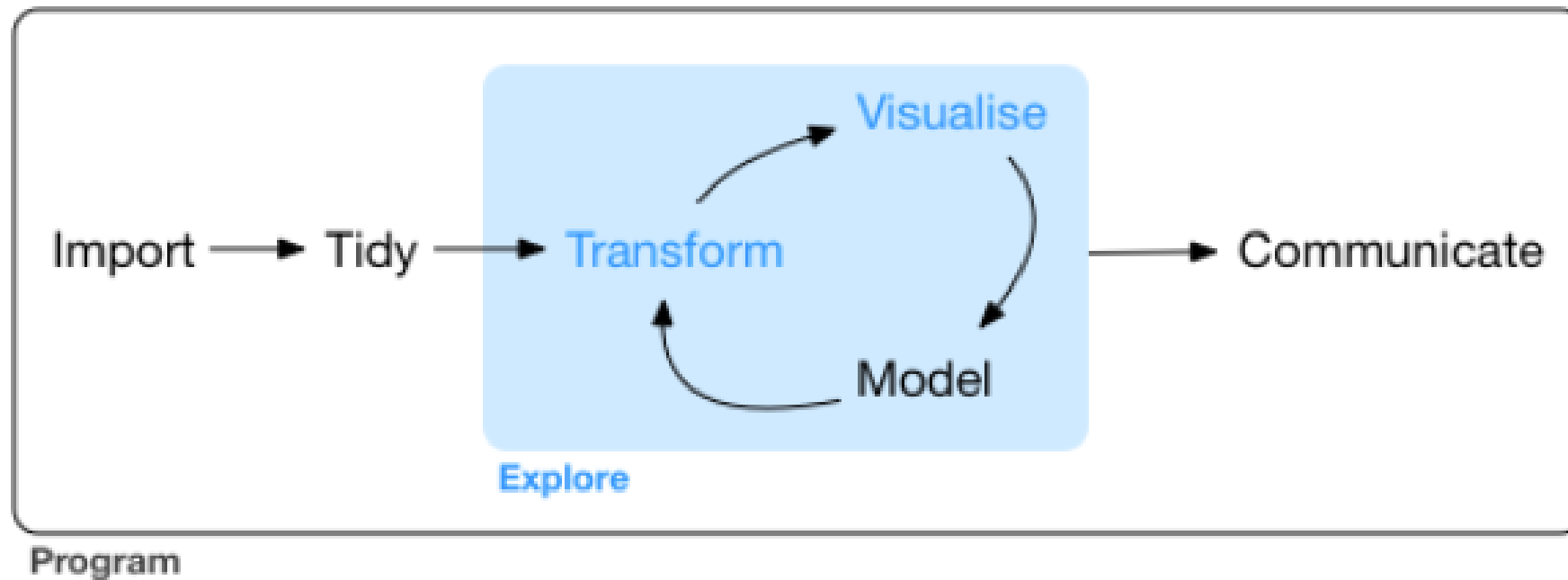
R: New phone



R Packages:  
Apps you can download



# Modeling Workflow





# Project Based Workflow

## Why?

- work on more than 1 thing at a time
- collaborate, communicate, distribute
- start and stop

## How?

- dedicated directory
- RStudio **P**roject
- Git repo, probably syncing to a remote

# Project Based Workflow



smell-test.R  
wrangle.R  
model.R  
make-figs.R  
report.Rmd

>>> everything.R

# Good Enough Practices for Data Science

## Box 1. Summary of practices

### 1. Data management

- a Save the raw data.
- b Ensure that raw data are backed up in more than one location.
- c Create the data you wish to see in the world.
- d Create analysis-friendly data.
- e Record all the steps used to process data.
- f Anticipate the need to use multiple tables, and use a unique identifier for every record.
- g Submit data to a reputable DOI-issuing repository so that others can access and cite it.

### 2. Software

- a Place a brief explanatory comment at the start of every program.
- b Decompose programs into functions.
- c Be ruthless about eliminating duplication.
- d Always search for well-maintained software libraries that do what you need.
- e Test libraries before relying on them.
- f Give functions and variables meaningful names.
- g Make dependencies and requirements explicit.
- h Do not comment and uncomment sections of code to control a program's behavior.
- i Provide a simple example or test data set.
- j Submit code to a reputable DOI-issuing repository.

### 3. Collaboration

- a Create an overview of your project.
- b Create a shared "to-do" list for the project.
- c Decide on communication strategies.
- d Make the license explicit.
- e Make the project citable.

### 4. Project organization

- a Put each project in its own directory, which is named after the project.
- b Put text documents associated with the project in the `doc` directory.
- c Put raw data and metadata in a data directory and files generated during cleanup and analysis in a results directory.
- d Put project source code in the `src` directory.
- e Put external scripts or compiled programs in the `bin` directory.

- f Name all files to reflect their content or function.

### 5. Keeping track of changes

- a Back up (almost) everything created by a human being as soon as it is created.
- b Keep changes small.
- c Share changes frequently.
- d Create, maintain, and use a checklist for saving and sharing changes to the project.
- e Store each project in a folder that is mirrored off the researcher's working machine.
- f Add a file called `CHANGELOG.txt` to the project's `docs` subfolder.
- g Copy the entire project whenever a significant change has been made.
- h Use a version control system.

### 6. Manuscripts

- a Write manuscripts using online tools with rich formatting, change tracking, and reference management.
- b Write the manuscript in a plain text format that permits version control.

# Getting to Know GIT

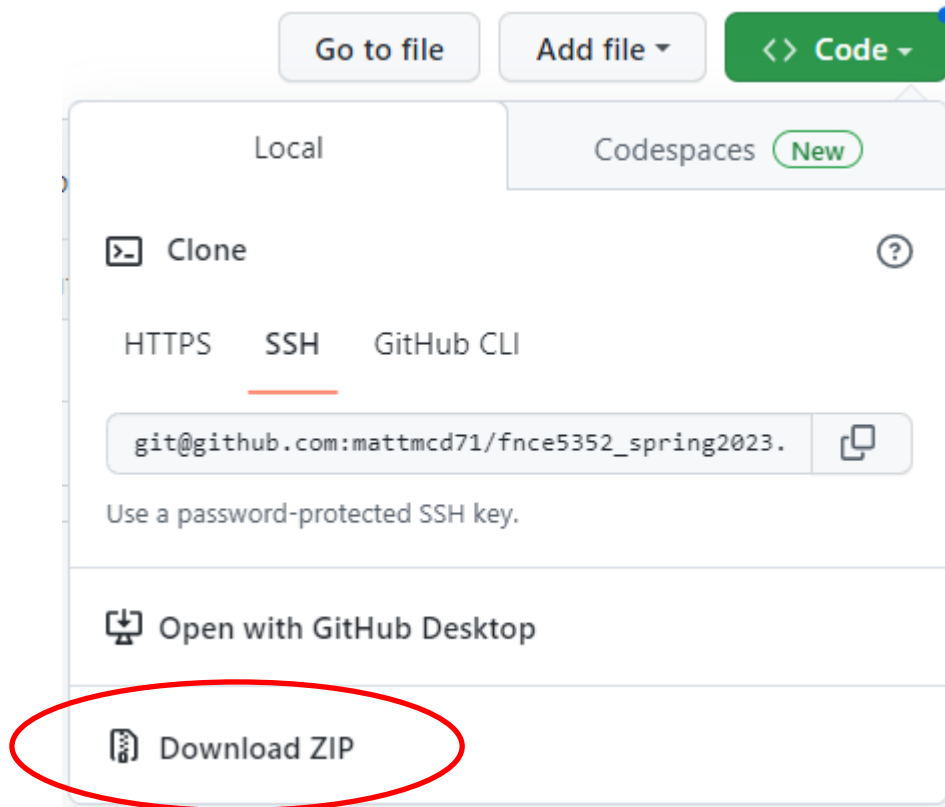
- <https://happygitwithr.com/>

## Class Repo:

- [https://github.com/mattmcd71/fnce5352\\_spring2023](https://github.com/mattmcd71/fnce5352_spring2023)

# How to Access the Class GIT Repo

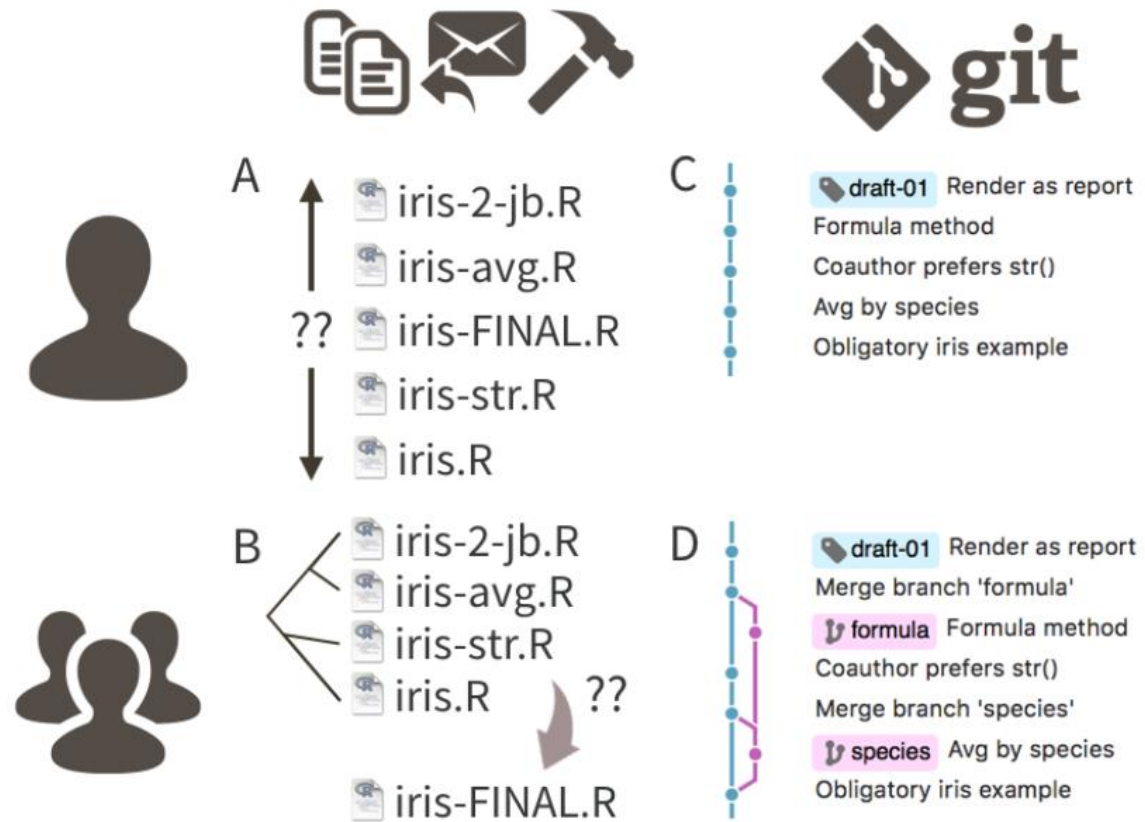
- Download ZIP



- Clone

- HTTPS or SSH
- SSH will require a setup of your SSH keys
  - See happygitwithr page for complete instructions on that

# Why GIT?



# GIT vs GitHub

