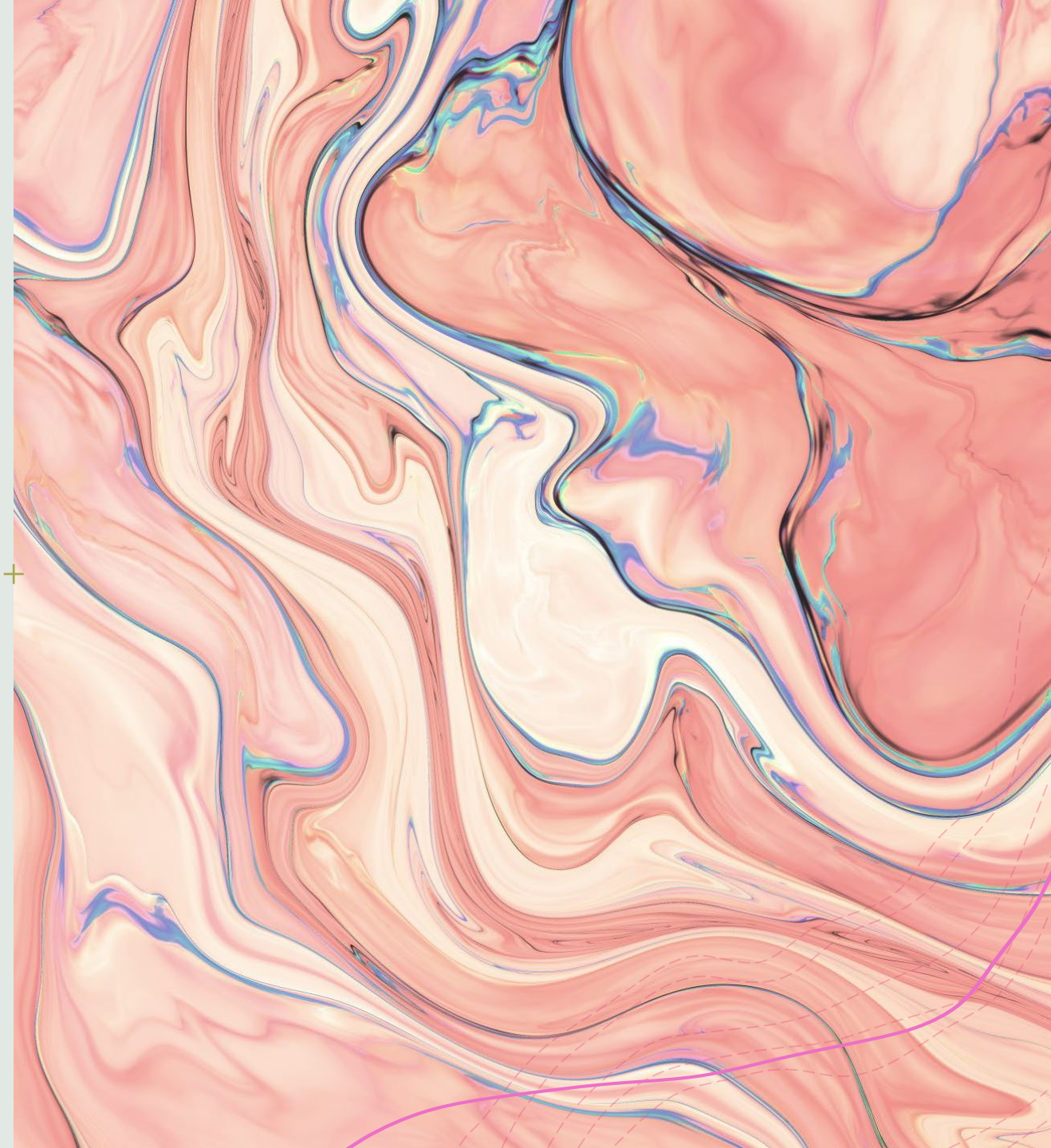
Using RStudio and GitHub for Reproducible Science



AGENDA

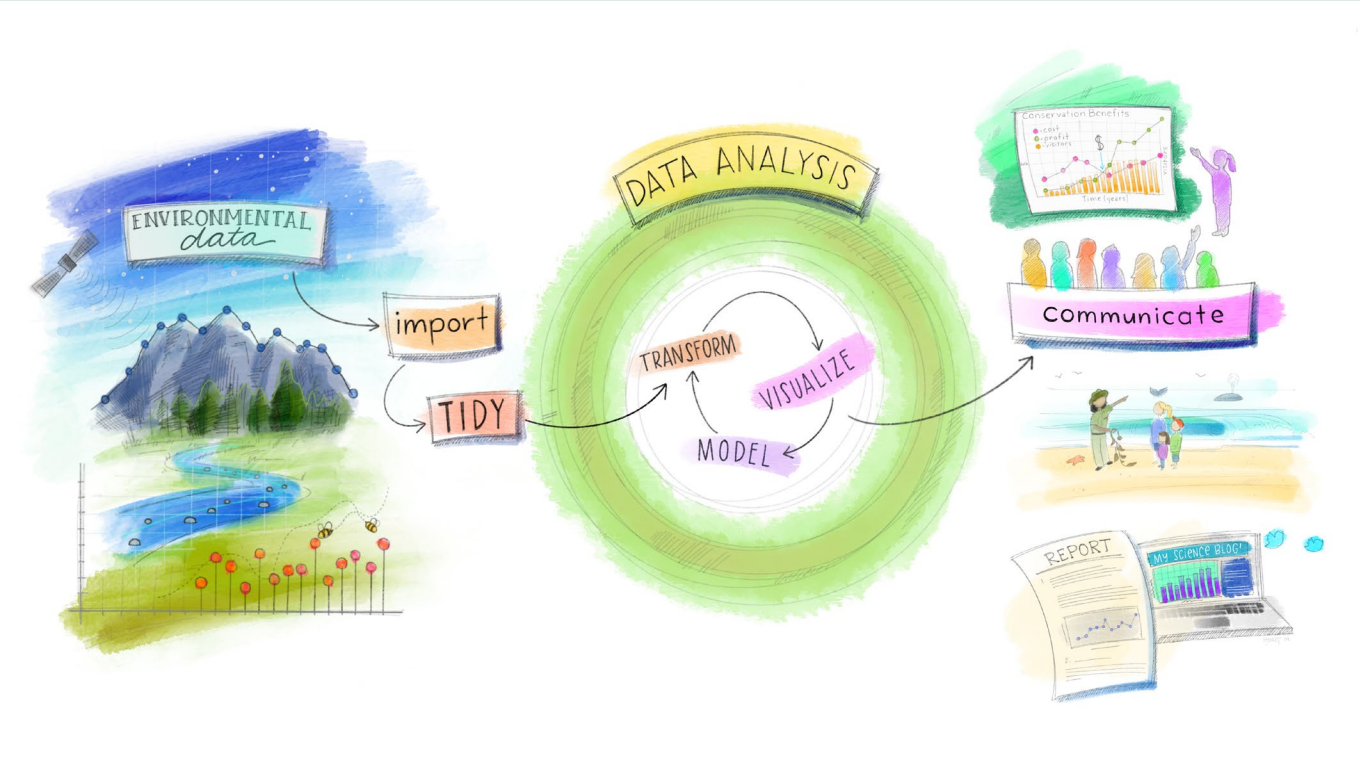
- Importance of Reproducible Science
- Project Organization
- Rprojects and GitHub
- Connecting to GitHub

All the slides and additional resources available

https://github.com/kpgund/github_workshop

Importance of Reproducible Science

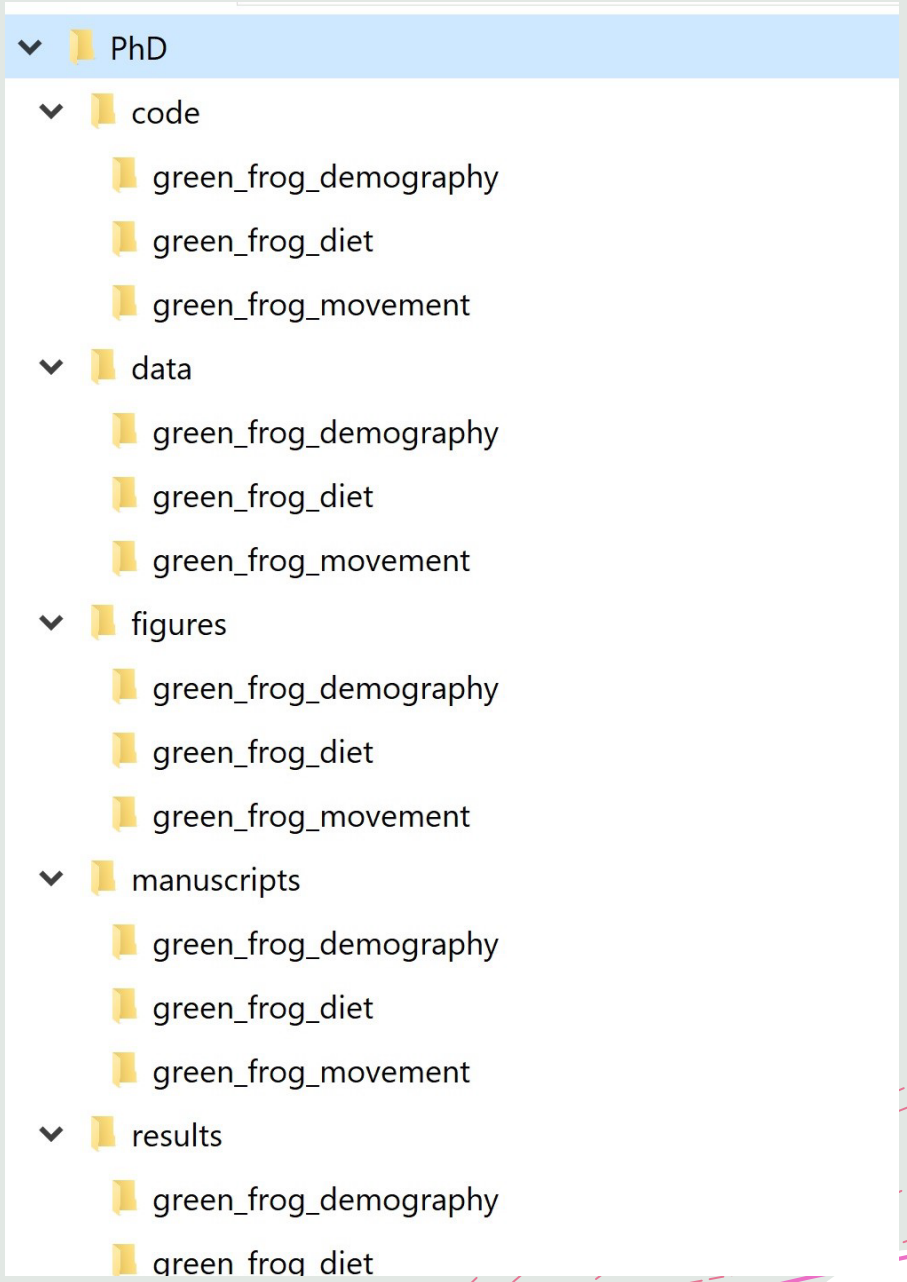
- ***"Reproducibility is a pillar of the scientific method" – Dr. Picardi***
- Being proficient in the use of programming tools and effectively apply them to store, process, manage, analyze, and visualize data have become **must-have skills** to take part in the scientific discourse.



Artwork by Allison Horst

Project Organization

- Everything meaningful in one folder structure
- Folder Structure
  - Project-based
  - Activity-based
- File Naming



Golden rules

1. **Raw data should never be changed**. Save it into a “data” folder and treat it as immutable. You can even set it as read-only to make sure there is no room for accidents.
2. The processed, clean version of your data will go into a dedicated “processed_data” folder.
3. Anything that can be generated from code goes into its own folder. This includes basically everything but the raw data and the code itself. You can have an “output” folder, or separate folders for output files and figures (e.g., “output” and “figures”)
4. If there are text documents, put them in their own folder (e.g., “docs”)
5. Code also has its own folder. If you write a lot of functions, it can be helpful to have a “funs” folder to store those and a “src” (for ‘source’) folder to save processing/analysis code.
6. If processing/analysis scripts are meant to be used in a certain order, you can number them (more on this in a minute). Sometimes the pipeline is not linear but branched, so numbering may not always make sense. Function scripts should not be numbered.
7. Modularize your code: instead of having a giant script to run your entire analysis from data cleaning to final figures, break up your workflow into several short, single-purpose scripts with well-defined inputs and outputs.

File Naming

- Good file names are computer-readable, human-readable, and work well with default ordering.

Ex: What we don't want:

data.csv

data_cleaned_March-22-2012.csv

analysis code.R

Green Frogs Manuscript_Final_edits.docx

final.docx

Ex: What we do want:

20221009_GitHub-workshop_powerpoint.ppt

01_DataOrganization.R

02_DataCleaning.R

03_Analysis.R

Simona Picardi

I am an ecologist with expertise in the analysis of animal movement, behavior, and space-use.

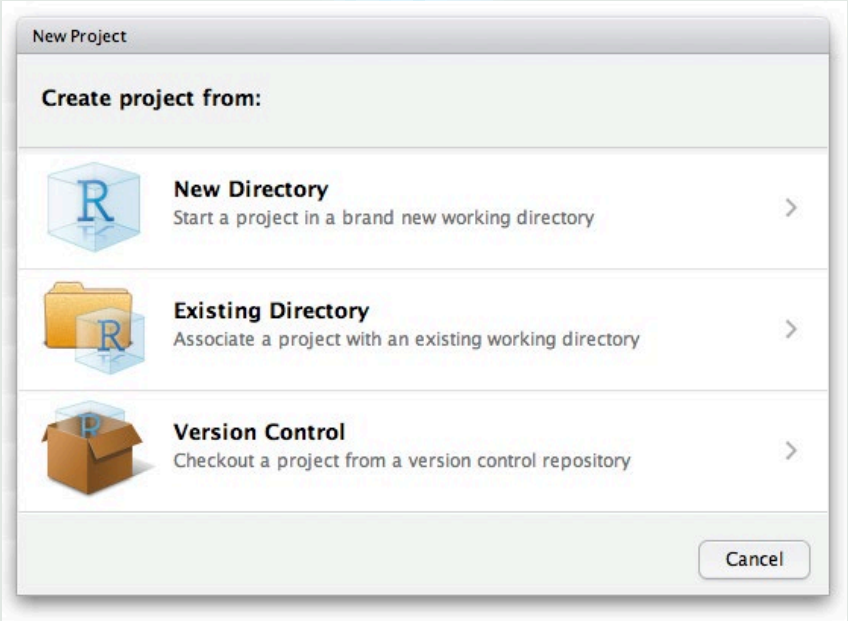
My research aims to quantify wildlife responses to environmental change and anthropogenic pressures with the goal of informing their management and conservation.

I am a passionate teacher and educator with several years of successful experience teaching data science and programming to ecologists at all levels.

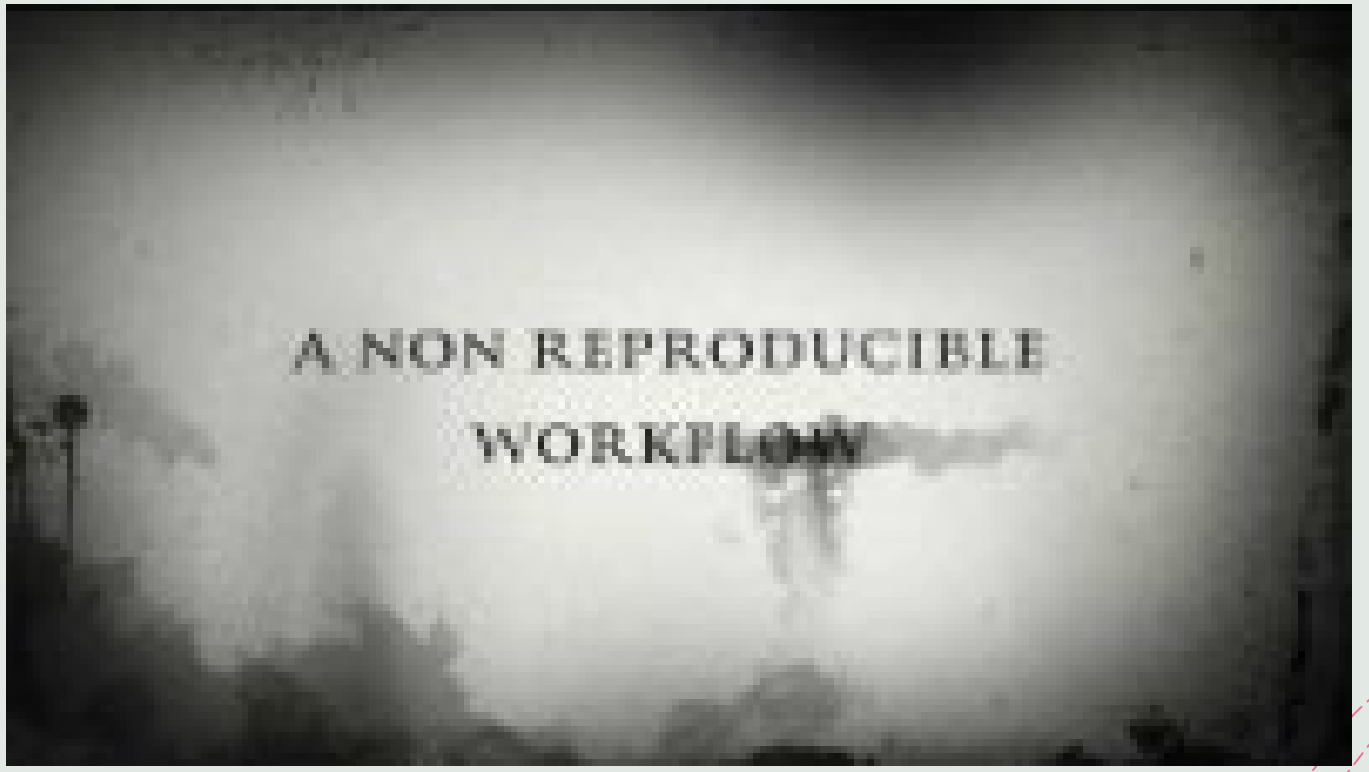
+ For more info:
+ https://ecorepsci.github.io/reproducible-science/
+ https://www.picardiecology.com/teaching

Rprojects (.Rproj)

- Working within the project directory
- Self-contained
  - Inputs, outputs, and code are all in one place
  - Relative paths

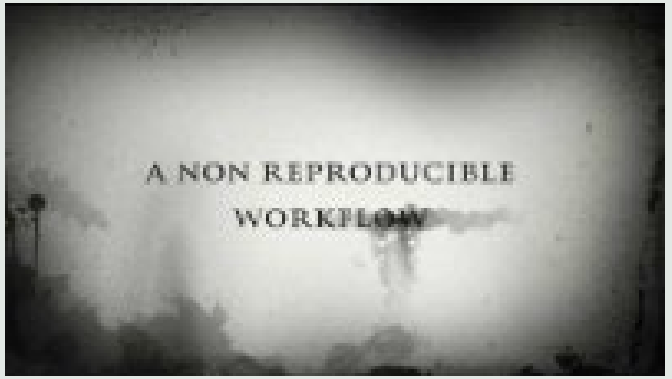


Why use GitHub

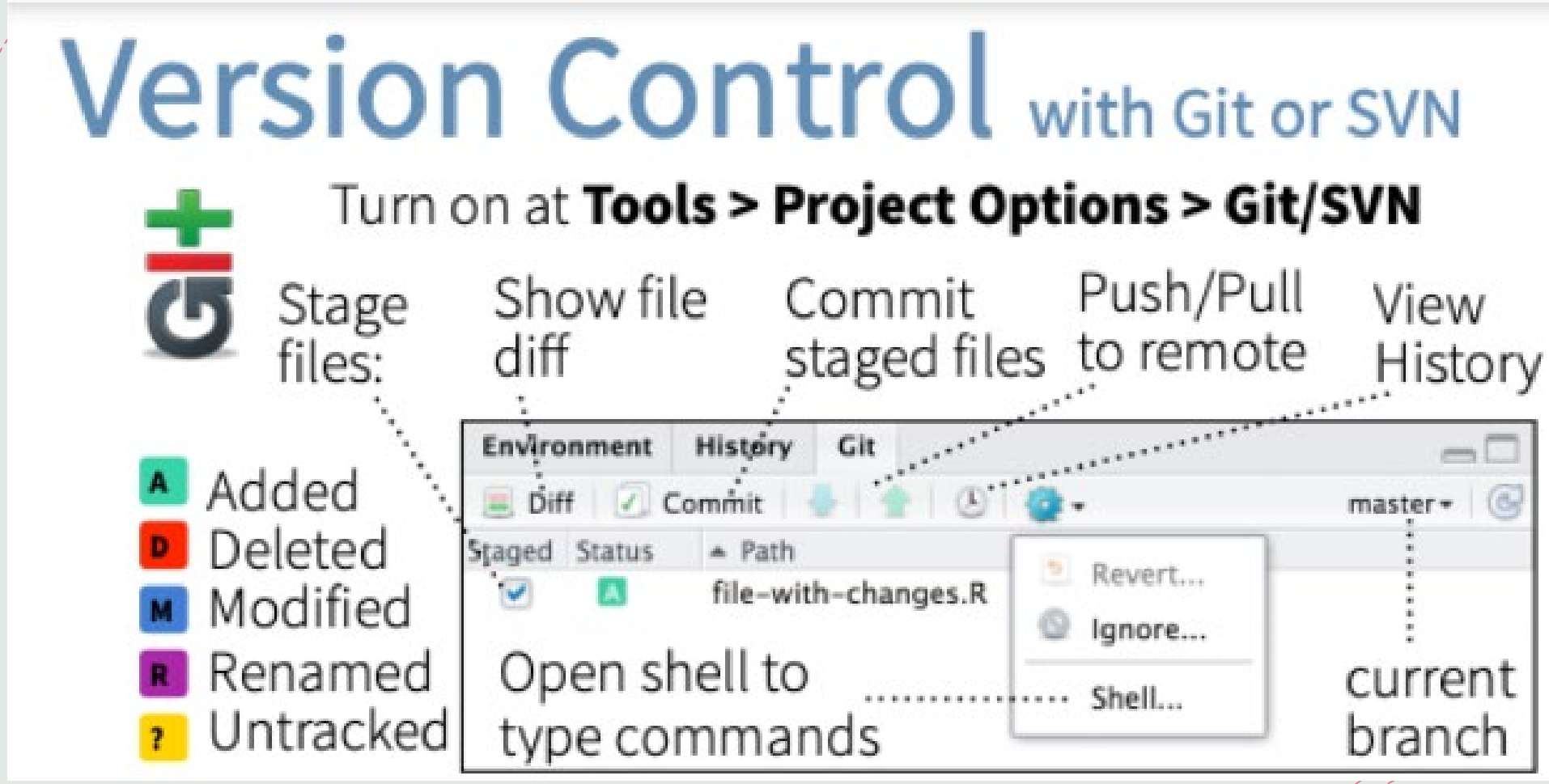


Why use GitHub

- Backup of your project
- Keep track of changes
- Network
- Share research
  Collaborate with others



Rstudio + GitHub



Rstudio + GitHub

- Track what updates are happening
- Find previous version

CREATE SSH KEY

CONNECT SSH KEY TO GITHUB ACCOUNT

CREATE A REPOSITORY IN GITHUB

CREATE AN RPROJECT AND CONNECT IT TO THE GITHUB REPOSITORY

COMMIT/ PUSH/ PULL

Let's get started!