Welcome to coreR

NCEAS Learning Hub

for

Delta Stewardship Council

June 2024

Week's Schedule

Session Time	RM 2-310 Monday		RM 2-309 Tuesday		RM 2-310 Wednesday		RM 2-310 Thursday		Friday
8:30-10:00	Introduction Set up	Camila	Cleaning & Wrangling Data	Angel	Publishing to the Web Intro to Data Viz	Angel	Shiny cont'	Camila	
10:00-10:30	BREAK		BREAK		BREAK		BREAK		
10:30-12:00	Literate Analysis with Quarto	Camila	Practice Session I		Working with Spatial Data	Angel	Wrap-up: Reproducibility & Provenance Survey + Q&A	Camila	
12:00-1:00	LUNCH		LUNCH		LUNCH		ADJOURN		
1:00-2:30	Introduction to Git & GitHub	Angel	Collaborationg with Git & GitHub	Angel	Practice Session II		*Technical* *Non-Technical*		
2:30-3:00	BREAK		BREAK		BREAK		*Practice*		
3:00-4:30	Tidy Data	Camila	Data Management	Camila	Intro to Shiny	Camila			

About this course

An immersion course in **R programming for environmental data science.**

You will gain experience on how to leverage the use of data science tools to increase your capacity to **collaborate** with your team, create **reproducible workflows**, and learn **best practices for open science**.

About this course

C

O

R

E

About this course

Collaborative

O

R

E

About this course

Collaborative

Open

R

E

About this course

Collaborative

Open

Reproducible

E

About this course

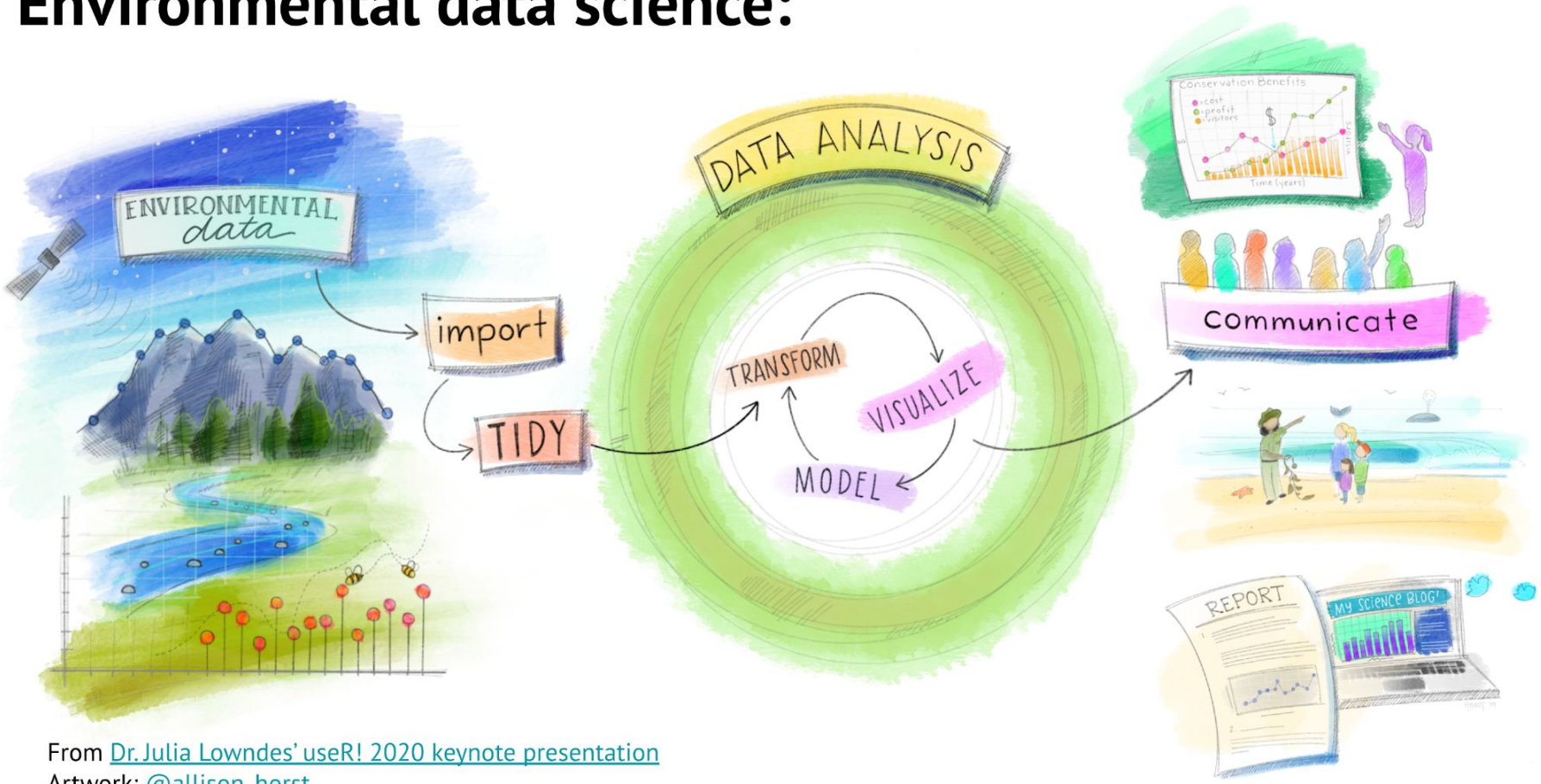
C ollaborative

O pen

R eproducible

E nvironment

Environmental data science:



From Dr. Julia Lowndes' useR! 2020 keynote presentation
Artwork: @allison_horst

Let’s talk about reproducibility…

And building robust workflows.

What is reproducibility?

U.S National Science Foundation (NSF) subcommittee on replicability in science: “reproducibility refers to the **ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator**”

Goodman et al 2016

Types of reproducibility

- "**Computational reproducibility:** When detailed information is provided about code, software, hardware and implementation details."

- "**Empirical reproducibility:** when detailed information is provided about non-computational empirical scientific experiments and observations. In practice, this enabled by making data freely available as well as details of how data was collected."

- "**Statistical reproducibility:** when detailed information is provided about the choice of statistical tests, model parameters, threshold, values etc. This mostly related to pre-registration of study design to prevent p-values hacking and manipulations."

Victoria Stodden, 2014, ROpenSci Reproducibility Guide

Types of reproducibility

- "**Computational reproducibility:** When detailed information is provided about code, software, hardware and implementation details."

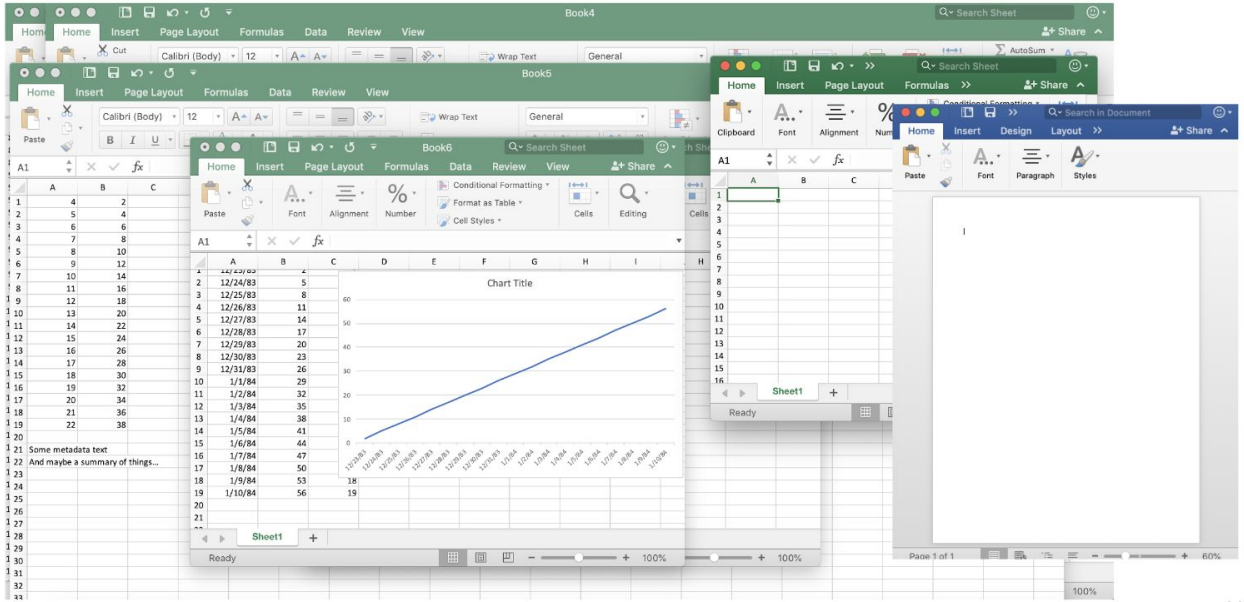
- "**Empirical reproducibility:** when detailed information is provided about non-computational empirical scientific experiments and observations. In practice, this enabled by making data freely available as well as details of how data was collected."

- "**Statistical reproducibility:** when detailed information is provided about the choice of statistical tests, model parameters, threshold, values etc. This mostly related to pre-registration of study design to prevent p-values hacking and manipulations."

Victoria Stodden, 2014, ROpenSci Reproducibility Guide

Does this look familiar?

Is this ↓ how you've been working with data? Cool!
If it has been working for you, feel good about it.



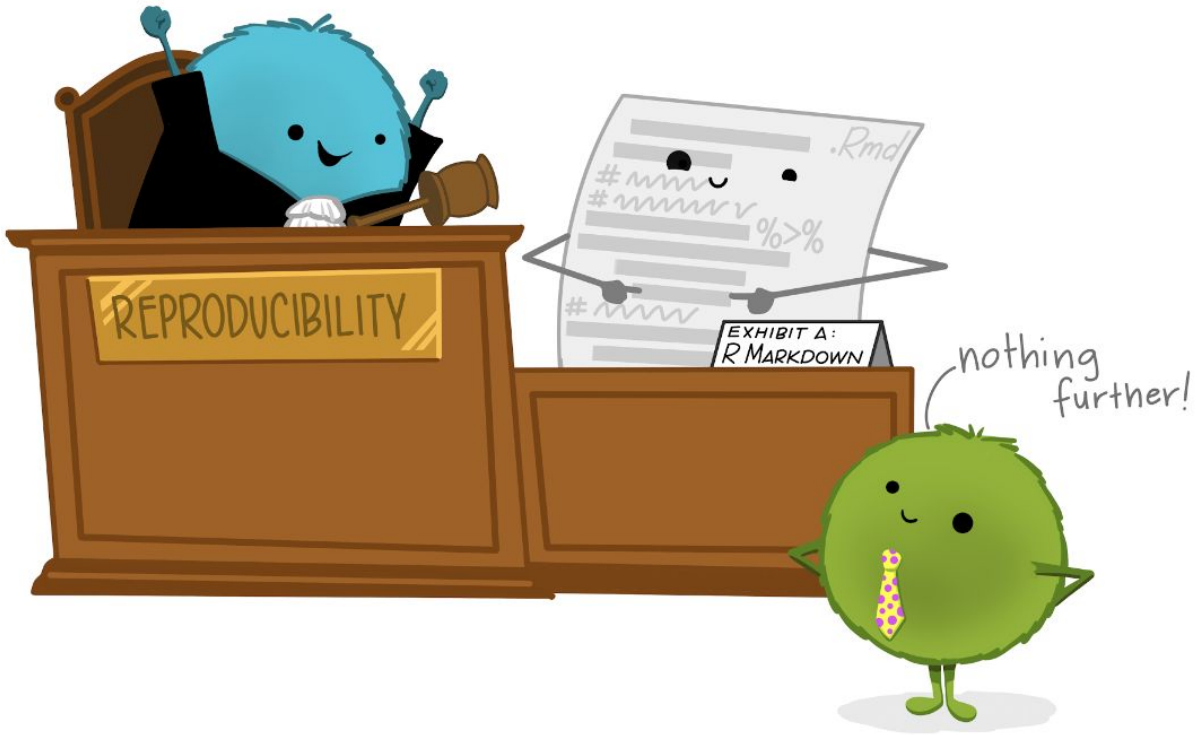
A common workflow

A common workflow

Room for improvement

- No history of what has been done to the data. From raw data to final figures/results.
- Lack of documentation on the step by step process.
- What happens if data is updated? We need to repeat every process?
- How do we collaborate with colleagues? Back and forth emails and versions of files with inevitably long file names (final_report_v1_CVP_AC_review_new_this_one.docx)
- How do we transfer analysis to final reports? Is this reproducible?

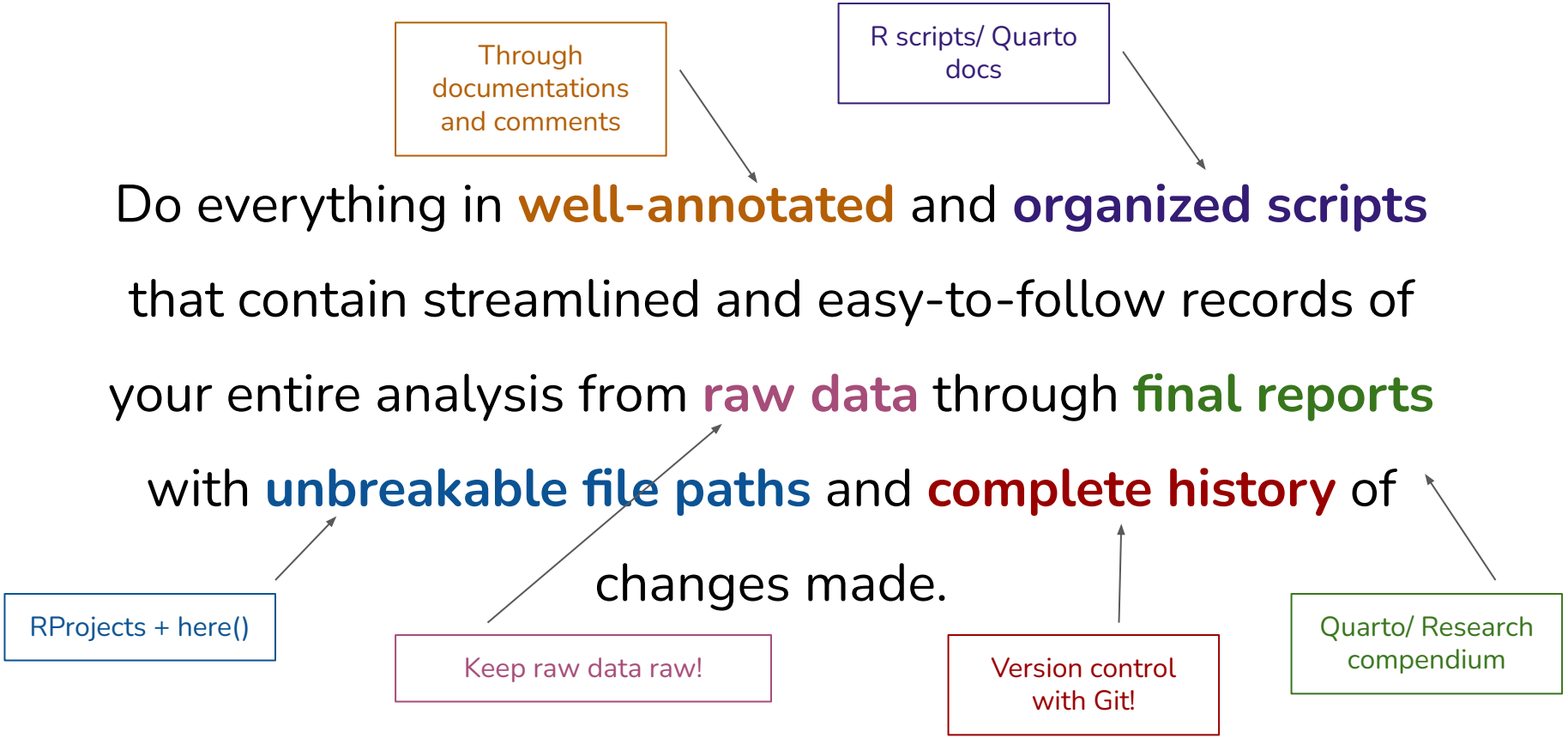
Work with your data like it's going to need an alibi



Allison Horst

Do everything in **well-annotated and organized scripts** that contain streamlined and easy-to-follow records of your entire analysis from **raw data** through **final reports** with **unbreakable file paths** and **complete history** of changes made.

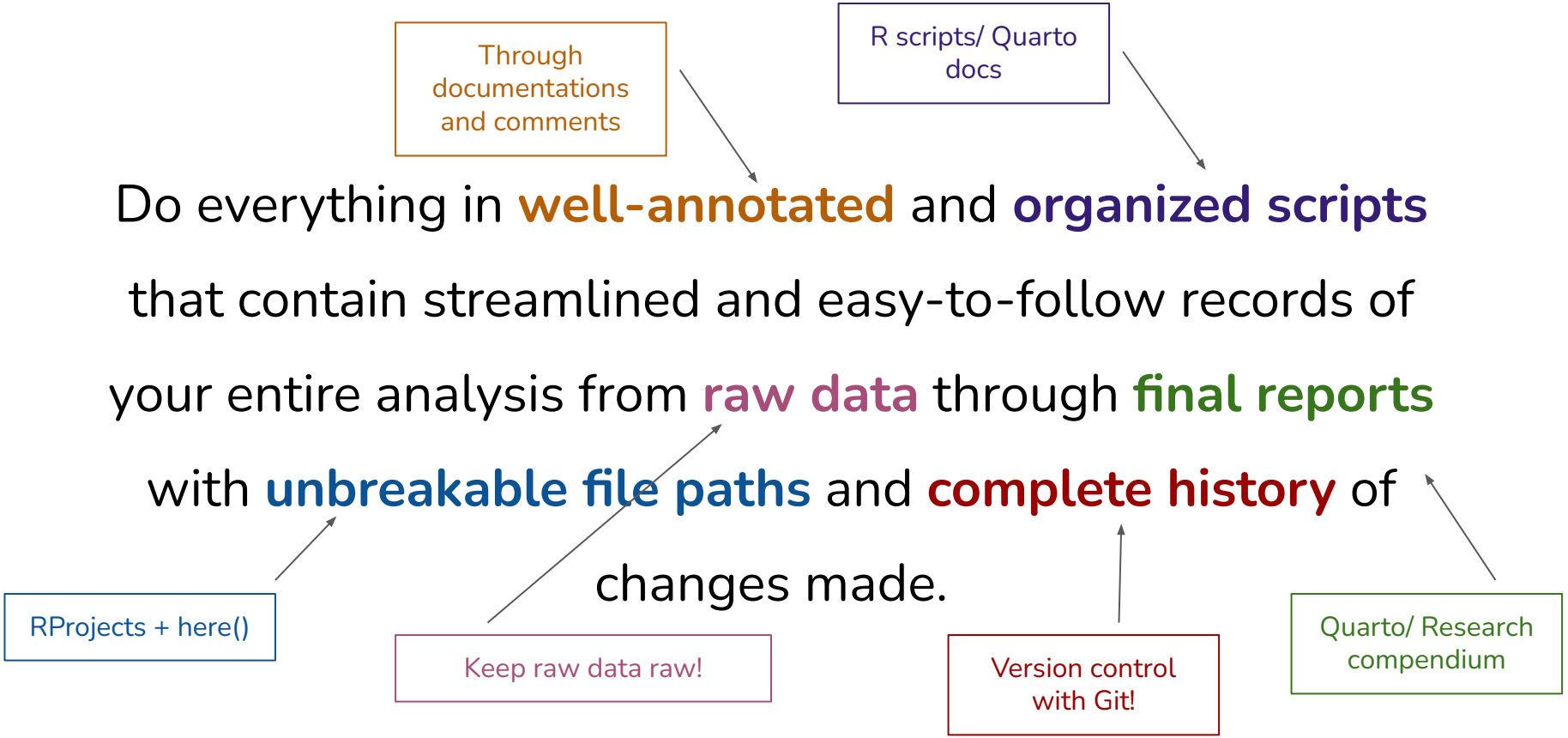
Adapted from Allison Horst, EDS 221, Scientific Programming Essentials



Adapted from Allison Horst, EDS 221, Scientific Programming Essentials

"...you're actually always collaborating with future-you; and past-you doesn't respond to emails."

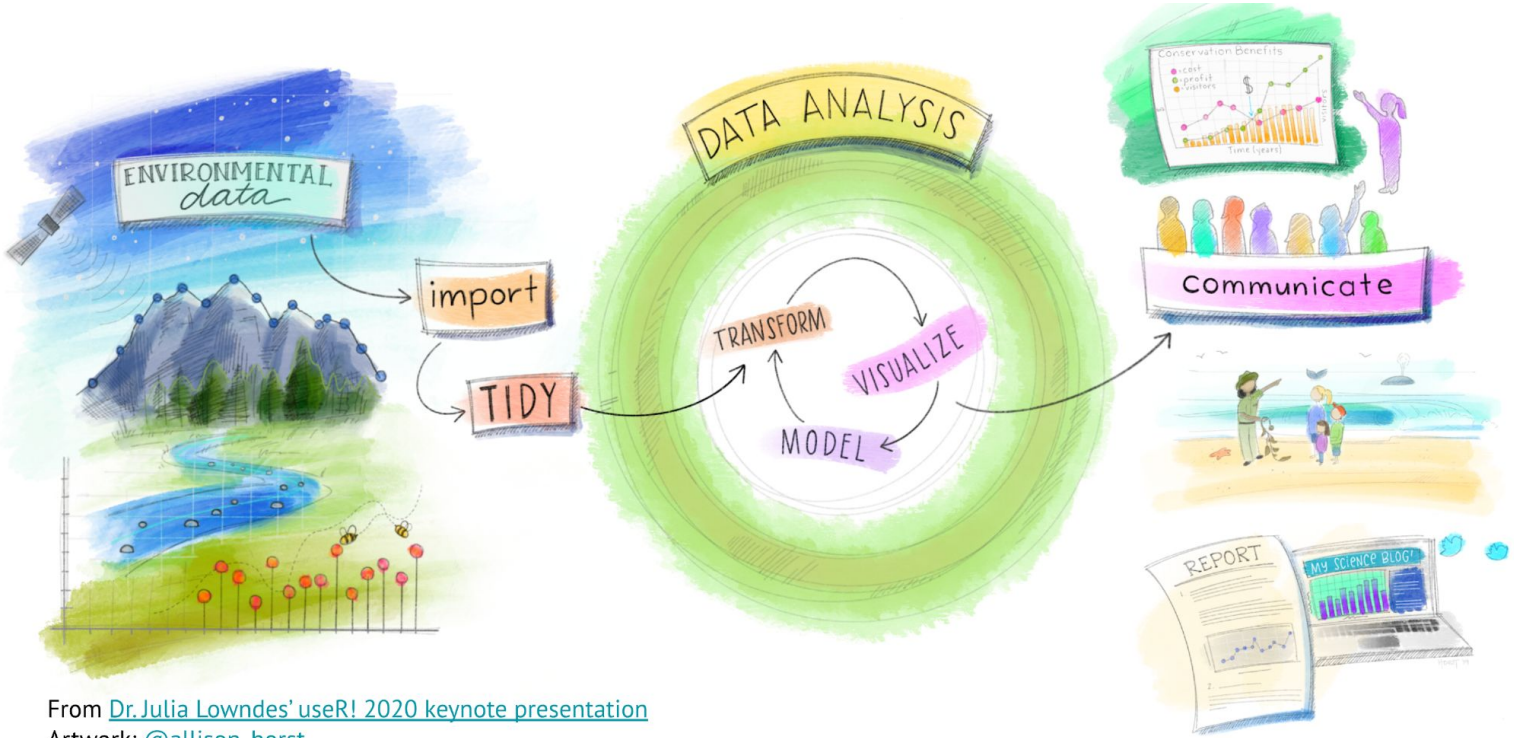
- Hadley Wickham (from fivebooks.com interview)



Adapted from Allison Horst, EDS 221, Scientific Programming Essentials

Reproducibility starting point: Set up a robust structure

Reproducibility starting point: Set up a robust structure



From Dr. Julia Lowndes' useR! 2020 keynote presentation
Artwork: @allison_horst

Reproducibility starting point: Set up a robust structure

- The fundamental idea behind a reproducible analysis is a **clean, repeatable script-based workflow**.
- This will allow you to **re-run your analysis as many times** as needed before (and after) the completion of your project.
- The smoother and more **automated the workflow,** the easier, faster and more robust the process of repeating it will be.

Key Points

1. Use a scripted (programming) language
2. Use one folder per project
3. Organize the content of your project with sub-folders
4. Set up robust file paths

Talk to your neighbor

- How do you generally organize your files for a project?
- What do you like about your system?
- Do you see any limitations to your system?

1. Use a scripted (programming) language

How do you tell your code where to find files?

Artwork by Allison Horst

File System Structure

How do you tell your code where to find files?

```
some_data <- read.csv("/home/vargas-poulsen/Documentes/Workshops/RLadies-SB/reproducible-workflows/some_data.csv")
```

File System Structure

How do you tell your code where to find files?

```
some_data <- read.csv("/home/vargas-poulsen/Documentes/Workshops/RLadies-SB/reproducible-workflows/some_data.csv")
```

If I **share my script with this file path to my colleagues**, would they be able to open the file?

Probably not.

A better way...

Projects

A better way...

- Provides a **self contained working directory (folder)** that does not depend on the absolute location of your computer.

Projects

A better way...

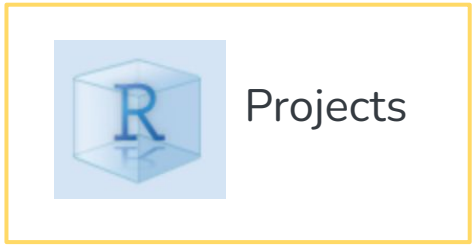
Projects

- Provides a **self contained working directory (folder)** that does not depend on the absolute location of your computer.
- **Bundles all your work within a working directory**, pointing to relative locations within the project.

A better way...

- Provides a **self contained working directory (folder)** that does not depend on the absolute location of your computer.
- **Bundles all your work within a working directory**, pointing to relative locations within the project.
- Within this centralize location **we can organize all the files** involved in our project (inputs data, scripts, outputs, etc.)

A better way...



When you create an *R Project*, it **creates and Rproj file and a folder in your computer that will be the *working directory*** when your are working in your Rproj.

✅ 2. Use one folder per project

A note on file paths

- An absolute path always **starts with the root of your file system** and locates files from there.

```
/home/vargas-poulsen/Documents/Workshops/RLadies-SB/reproducible-workflows/data/data.csv
```

A note on file paths

- An absolute path always **starts with the root of your file system** and locates files from there.

```
/home/vargas-poulsen/Documents/Workshops/RLadies-SB/reproducible-workflows/data/data.csv
```

- Relative paths start from some **location in your file system that is below the root**. That is the starting point to locate a file on your system

A note on file paths

- An absolute path always **starts with the root of your file system** and locates files from there.

`/home/vargas-poulsen/Documents/Workshops/RLadies-SB/reproducible-workflows/data/data.csv`

- Relative paths start from some **location in your file system that is below the root**. That is the starting point to locate a file on your system

If my R project is named `reproducible-workflows`, then the relative path to `data.csv`, starting from the project directory will be `data/data.csv`.

A note on file paths

- An absolute path always **starts with the root of your file system** and locates files from there.

`/home/vargas-poulsen/Documents/Workshops/RLadies-SB/reproducible-workflows/data/data.csv`

- Relative paths start from some **location in your file system that is below the root**. That is the starting point to locate a file on your system

If my R project is named `reproducible-workflows`, then the relative path to `data.csv`, starting from the project directory will be `data/data.csv`.

****R projects set the file path relative to the project's directory (folder)****

Keys Points

1. ☑ Use a scripted (programming) language
2. ☑ Use one folder per project
3. Organize the content of your project with sub-folders
4. Set up robust file paths

Organizing files inside your project

- Ensure that the structure of the folders and location of files in your project are **consistent.**

Organizing files inside your project

- Ensure that the structure of the folders and location of files in your project are **consistent.**

- The location of files should be as **informative** as possible on what a file contains.

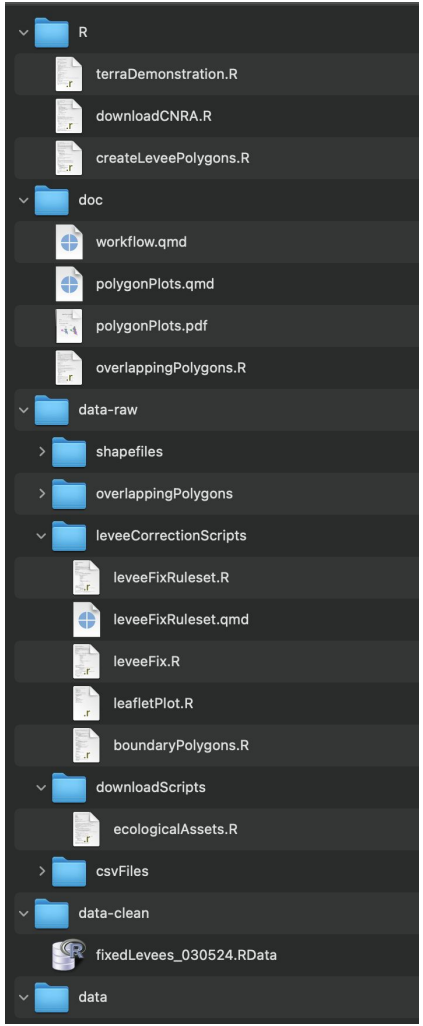
Organizing files inside your project

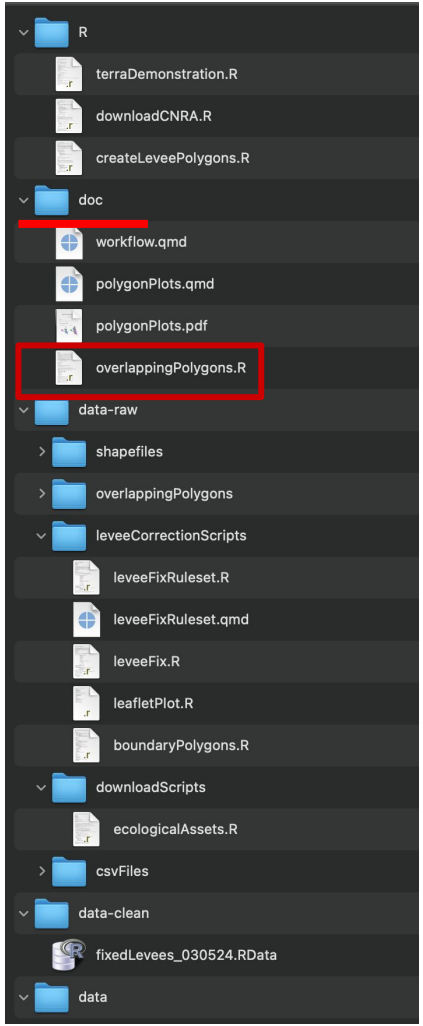
- Ensure that the structure of the folders and location of files in your project are **consistent.**
- The location of files should be as **informative** as possible on what a file contains.
- The idea is to organize your research into a compendium that **has all of the digital parts needed to replicate your analysis**, like code, figures, the manuscript, and data access.

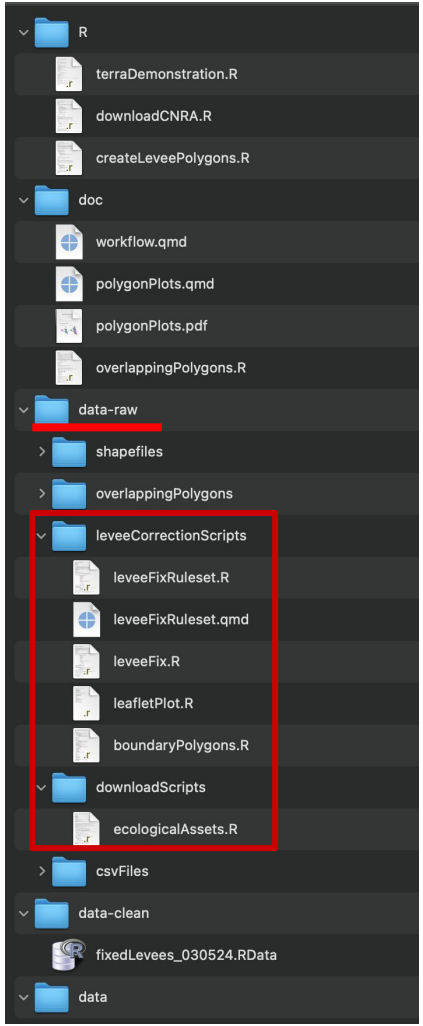
Organizing files inside your project

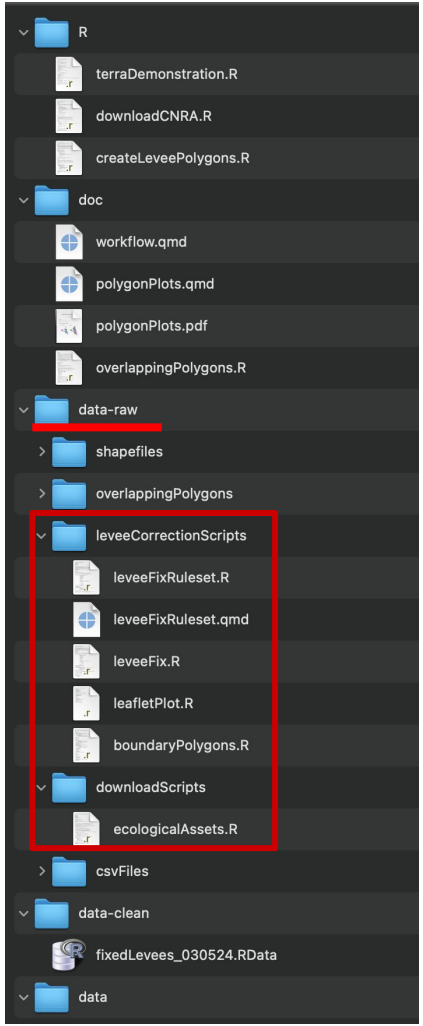
- Ensure that the structure of the folders and location of files in your project are **consistent.**
- The location of files should be as **informative** as possible on what a file contains.
- The idea is to organize your research into a compendium that **has all of the digital parts needed to replicate your analysis**, like code, figures, the manuscript, and data access.

Let’s take a look at one example

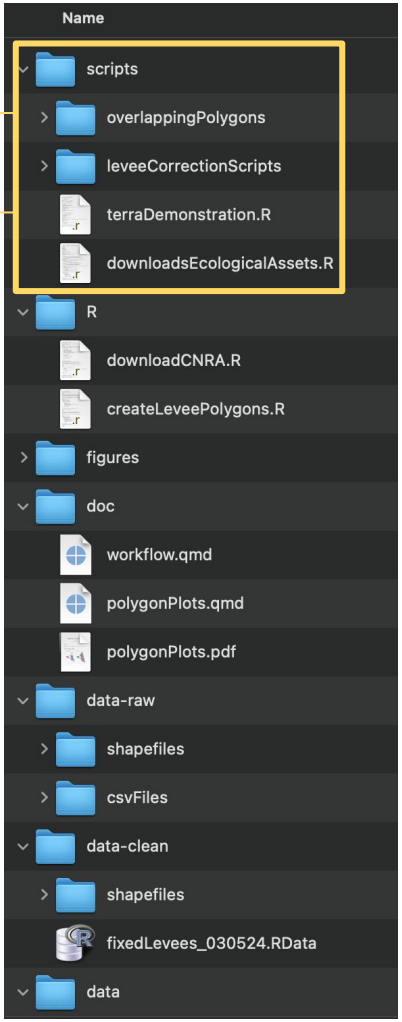

R
terraDemonstration.R
downloadCNRA.R
createLeveePolygons.R
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
overlappingPolygons.R
data-raw
shapefiles
overlappingPolygons
leveeCorrectionScripts
leveeFixRuleset.R
leveeFixRuleset.qmd
leveeFix.R
leafletPlot.R
boundaryPolygons.R
downloadScripts
ecologicalAssets.R
csvFiles
data-clean
fixedLevees_030524.RData
data

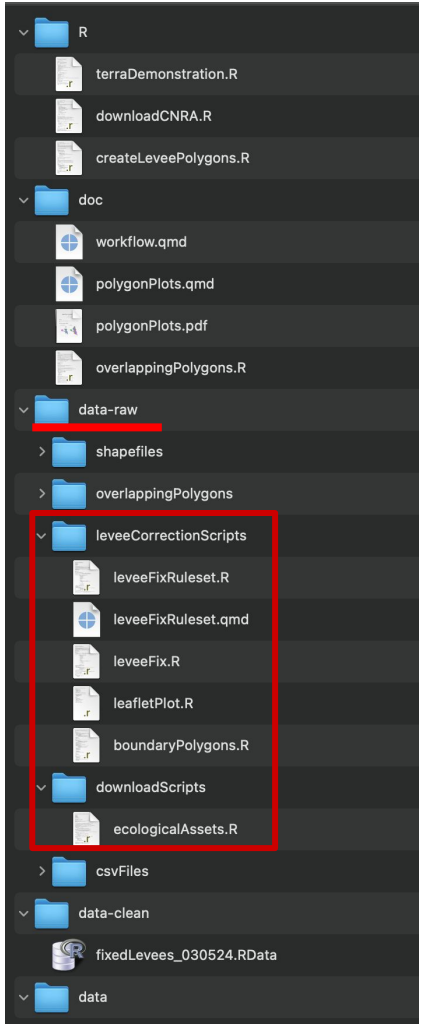

R
terraDemonstration.R
downloadCNRA.R
createLeveePolygons.R
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
overlappingPolygons.R
data-raw
shapefiles
overlappingPolygons
leveeCorrectionScripts
leveeFixRuleset.R
leveeFixRuleset.qmd
leveeFix.R
leafletPlot.R
boundaryPolygons.R
downloadScripts
ecologicalAssets.R
csvFiles
data-clean
fixedLevees_030524.RData
data


R
terraDemonstration.R
downloadCNRA.R
createLeveePolygons.R
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
overlappingPolygons.R
data-raw
shapefiles
overlappingPolygons
leveeCorrectionScripts
leveeFixRuleset.R
leveeFixRuleset.qmd
leveeFix.R
leafletPlot.R
boundaryPolygons.R
downloadScripts
ecologicalAssets.R
csvFiles
data-clean
fixedLevees_030524.RData
data



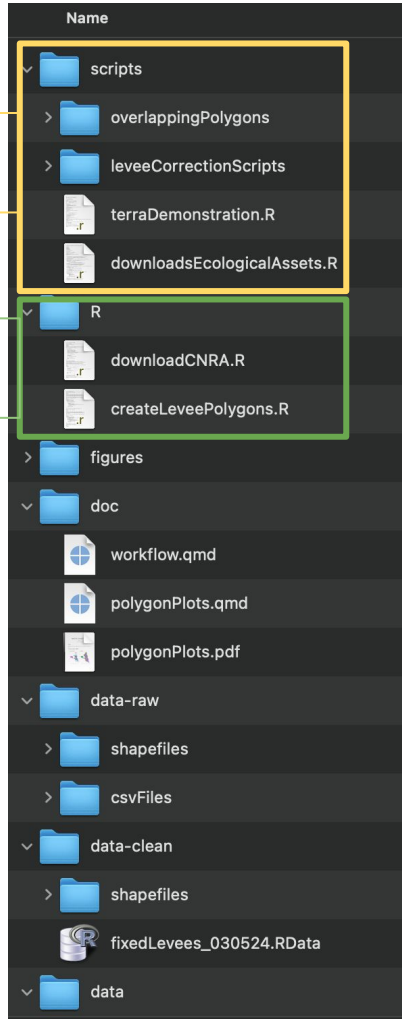
Analysis related scripts

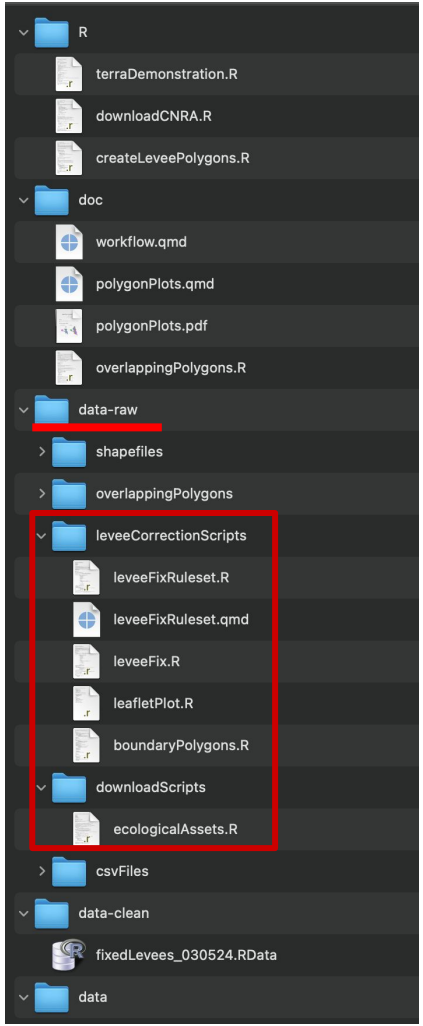




Analysis related scripts

Functions scripts

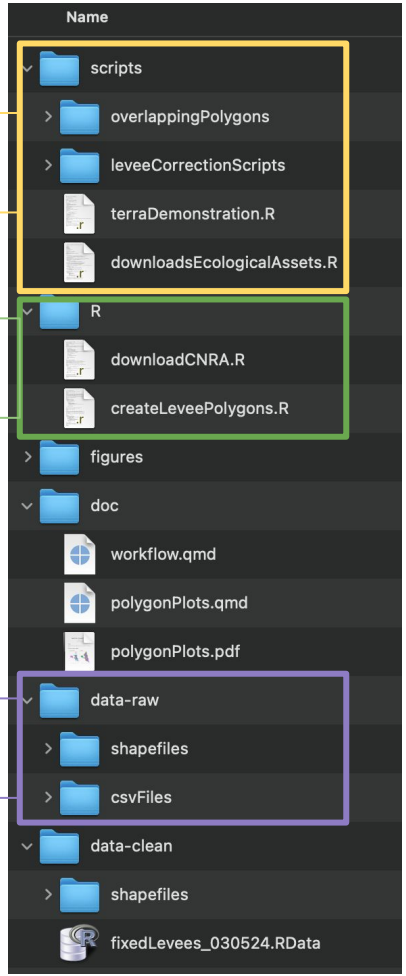


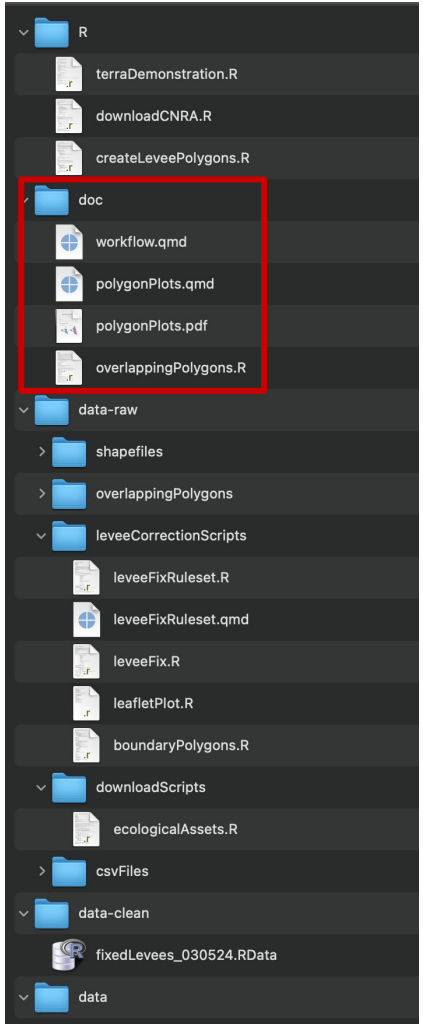


Analysis related scripts

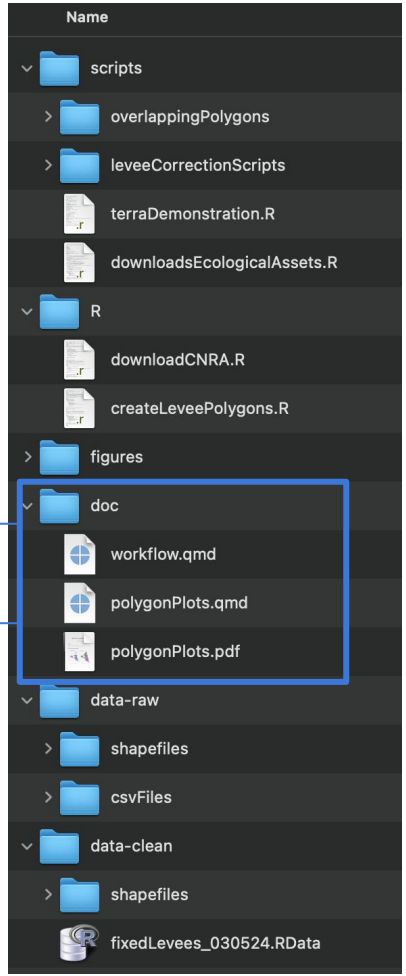
Functions scripts

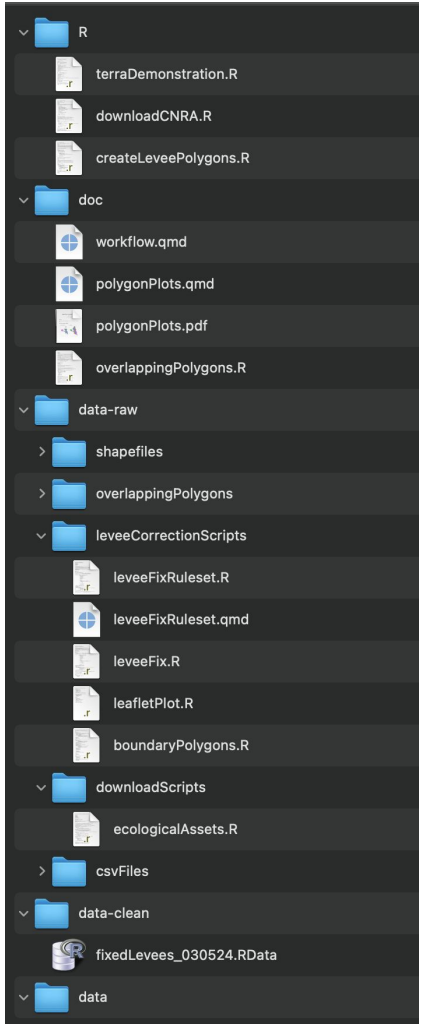
Raw Data




R
terraDemonstration.R
downloadCNRA.R
createLeveePolygons.R
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
overlappingPolygons.R
data-raw
shapefiles
overlappingPolygons
leveeCorrectionScripts
leveeFixRuleset.R
leveeFixRuleset.qmd
leveeFix.R
leafletPlot.R
boundaryPolygons.R
downloadScripts
ecologicalAssets.R
csvFiles
data-clean
fixedLevees_030524.RData
data

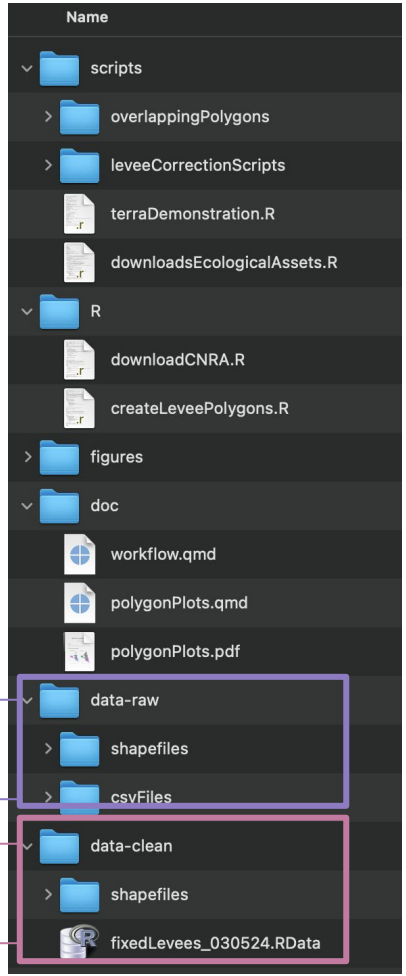
Documents

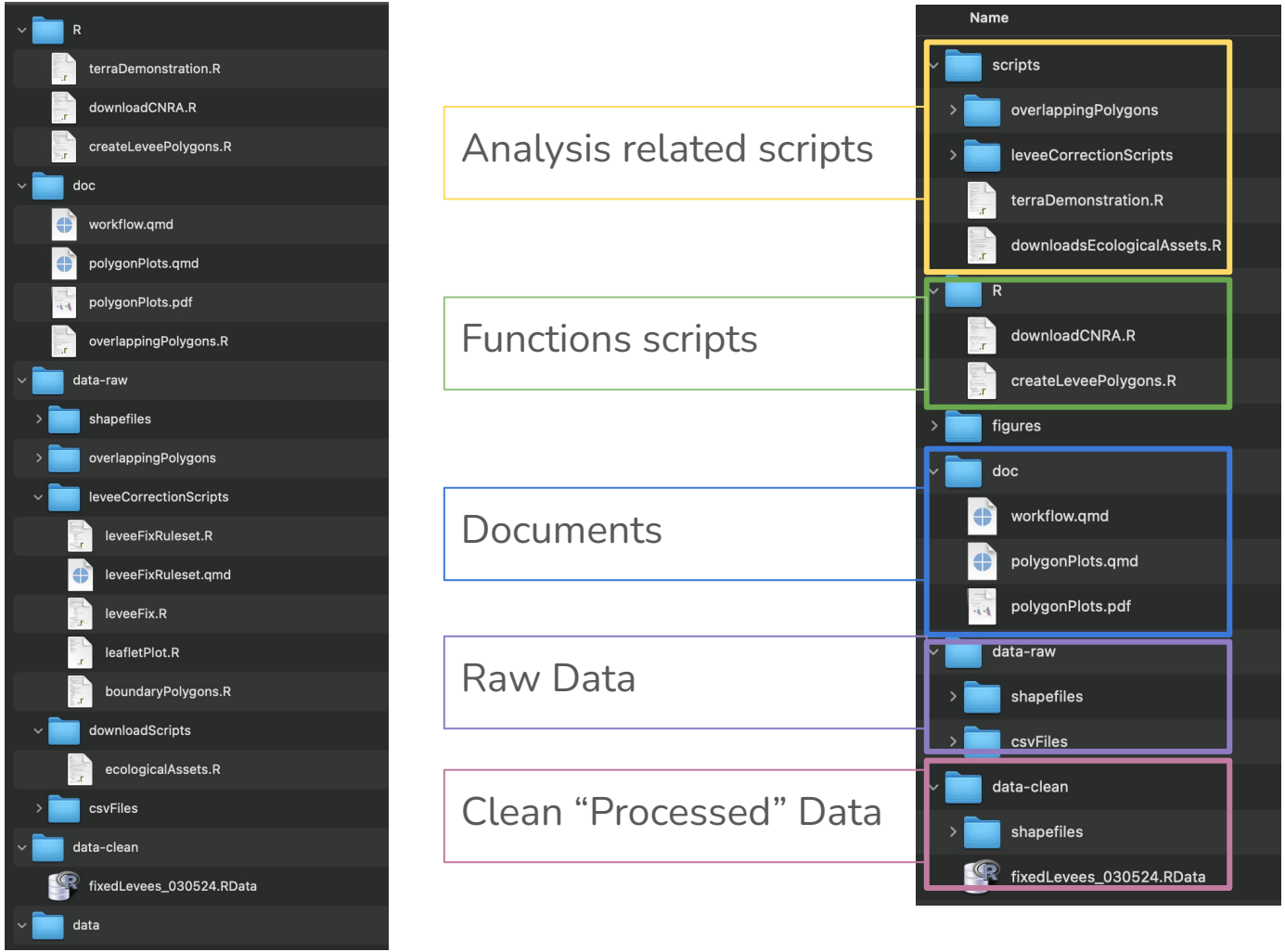

Name
scripts
overlappingPolygons
leveeCorrectionScripts
terraDemonstration.R
downloadsEcologicalAssets.R
R
downloadCNRA.R
createLeveePolygons.R
figures
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
data-raw
shapefiles
csvFiles
data-clean
shapefiles
fixedLevees_030524.RData



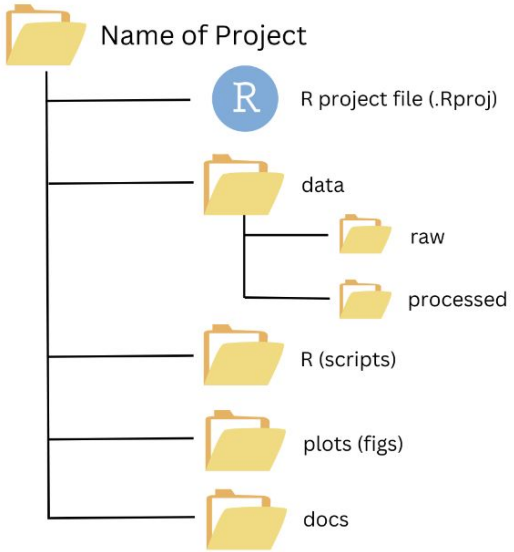
Raw Data

Clean “Processed” Data




R
terraDemonstration.R
downloadCNRA.R
createLeveePolygons.R
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
overlappingPolygons.R
data-raw
shapefiles
overlappingPolygons
leveeCorrectionScripts
leveeFixRuleset.R
leveeFixRuleset.qmd
leveeFix.R
leafletPlot.R
boundaryPolygons.R
downloadScripts
ecologicalAssets.R
csvFiles
data-clean
fixedLevees_030524.RData
data
Analysis related scripts
Functions scripts
Documents
Raw Data
Clean “Processed” Data
Name
scripts
overlappingPolygons
leveeCorrectionScripts
terraDemonstration.R
downloadsEcologicalAssets.R
R
downloadCNRA.R
createLeveePolygons.R
figures
doc
workflow.qmd
polygonPlots.qmd
polygonPlots.pdf
data-raw
shapefiles
csvFiles
data-clean
shapefiles
fixedLevees_030524.RData

Example of project organization

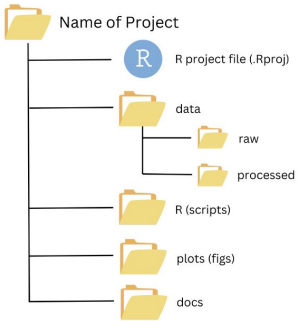


✅ 3. Organize the content of your project with sub-folders

General organization recommendations

- Keep your **Raw data Raw** - Never edit your raw data.
- Clearly **separate raw** data **from “clean”** processed data.
- **Review external inputs** to make sure they align with the established organization structure.
- Define informative **file naming conventions.**

One more thing about (reproducible) file paths



- Different operating systems use different characters to define file paths.
  - **Mac and Linux** uses slashes (e.g. `plots/diamonds.pdf`)
  - **Windows** uses backslashes (e.g. `\plots.pdf`).
  - ~ is a convenient shortcut to your home directory on mac
  - Windows doesn't really have the notion of a home directory, so it instead points to your documents directory.

R for Data Science (Grolemund & Wickham)

How to make your file paths within your project robust?

One more thing about (reproducible) file paths

Artwork by Allison Horst

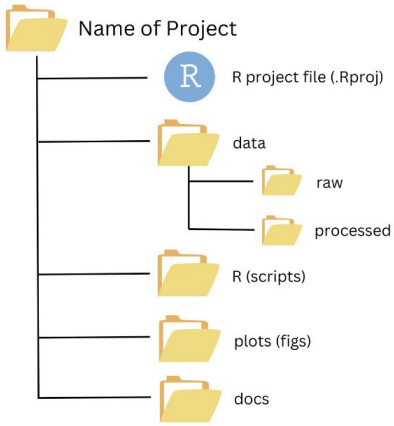
One more thing about (reproducible) file paths

"*The goal of the here package is to enable easy file referencing in project-oriented workflows. In contrast to using* `setwd()`*, which is fragile and dependent on the way you organize your files,* ***here uses the top-level directory of a project to easily build paths to files.***"

here.r Documentation

It allows us to **navigate through the files in our project** without having to worry about operating system issues.
`here()` starts from the working directory, aka your `Rproj` folder.

One more thing about (reproducible) file paths

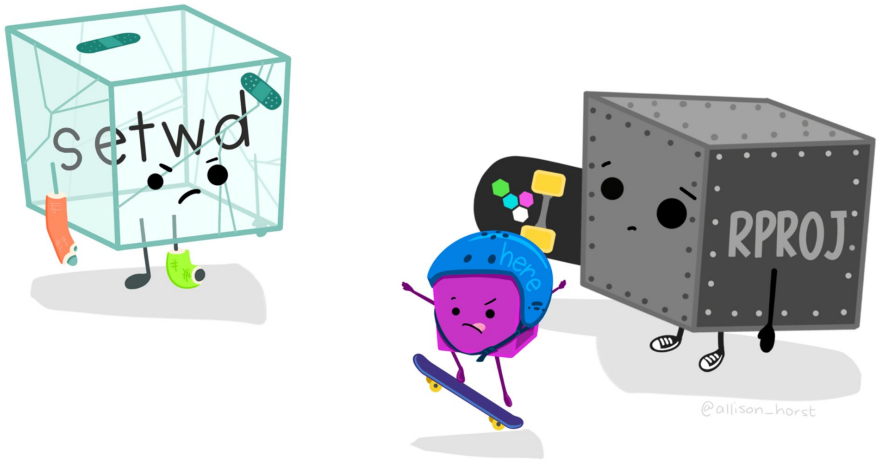


If I'm working within my R project, to read `some_data.csv`, inside the raw folder in this case, I can use the `here::here()` function.

```
some_data ← read_csv(here::here("data", "raw", "some_data.csv"))
```

R Projects + here() = robust file paths

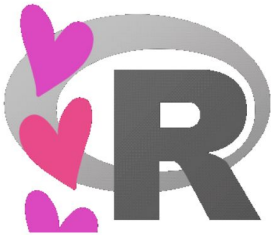
BIG first step towards reproducible workflows!



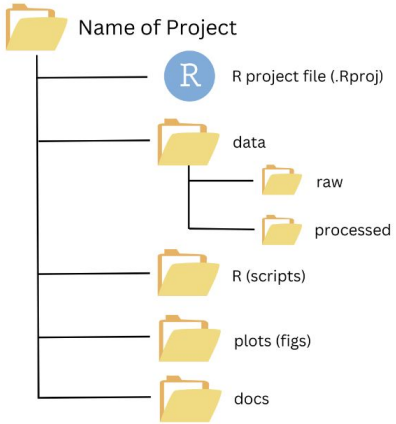
Artwork by Allison Horst

Organization Wrap up

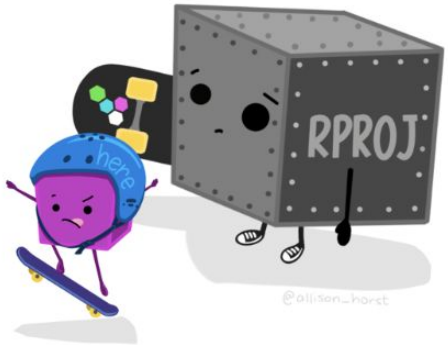
One of the first steps to achieve reproducibility is to **set up a robust structure** for our work.



Scripted analysis



One folder with organized content



Robust file paths

Artwork by Allison Horst

Organization Wrap up

Email, share, export into a different computer

`reproducible_project`

`reproducible_project`

Self contained project with all file paths relative to folders within the project, analysis can be reproduced elsewhere

Resources

- Best Practices for Writing Reproducible Code, University of Utrecht
- A Guide to Reproducible Code in Ecology and Evolution, British Ecological Society
- Reproducibility and Provenance, NCEAS Learning Hub
- Workflows, LTER Scientific Computing Workshops
- Reproducibility Lesson, LTER Synthesis Skills for Early Career Researchers
- EDS 221, Lesson 1 and Lesson 2, UCSB MEDS, By Allison Horst
- GitHub Clinic, Openscapes
- Building reproducible analytical pipelines with R, Bruno Rodrigues