

Data Cleaning: Redwood Lab

January 29, 2025

Today's plan

- 1 **Review: Git/GitHub + Reproducible Environments**
- 2 **Hands-on practice: Redwood data cleaning**

Review: Git/GitHub + Reproducible Environments

Setting up git, GitHub, and a reproducible environment

Details at: <https://tiffanymtang.github.io/dsip-s25/>

- + We set up two repositories:
 - + `dsip-s25`: for me to distribute course materials (lecture slides, **code**, etc) to you
 - + You should only *pull* to retrieve information
 - + `dsip`: **your** repository to do all your work in
- + We introduced **renv** and **conda** to create reproducible environments

Overview of **renv** and **conda**

Details at: <https://tiffanymtang.github.io/dsip-s25/>

renv

conda

1. Create environment:

Create .Rproj and open it
`renv::init()`

`conda create --name env_name`

2. Activate:

`renv::activate("pkg_name")`
not necessary if you're working in an .Rproj
(automatically activated if in .Rproj)

`conda activate env_name`

3. Add packages:

`renv::install("pkg_name")`

`conda install pkg_name`

4. Create/update lock file:

`renv::snapshot()`

`conda env export --from-history >
environment.yml
conda lock`
to run `conda lock`, need to have installed
conda-lock package beforehand

Why bother with reproducible environments?

- + Exact reproduction of the packages, software versions, etc.
- + Different projects might use/require different package versions
 - + E.g., older projects might use older package versions
- + Ease of portability to different computers and operating systems:

renv

1. Clone GitHub repository
2. Open R project
3. Install renv:
`install.packages("renv")`
4. Restore environment:
`renv::restore()`

conda

1. Clone GitHub repository
2. Navigate to directory with lock file
3. Install conda-lock (and conda if not already available)
`conda install conda-lock`
4. Restore environment:
`conda-lock install --name "new_env_name"`

A couple extra bells and whistles from last time...

- + How do you choose a particular conda environment in VS Code?
 - + Open command palette (Ctrl+Shift+P or Cmd+Shift+P)
 - + Search for "Python: Select Interpreter"
- + What is quarto and how do you use quarto in VS Code?
 - + Details: <https://tiffanymtang.github.io/dsip-s25/#using-quarto>
 - + Note: to do this, usually need to install jupyterlab and ipykernel in your conda environment:
`conda install ipykernel`
`conda install jupyterlab`

Pushing changes to GitHub

We've completed our basic setup – A great time to pause and take a snapshot of our project.

But before doing so, if you check your git status,

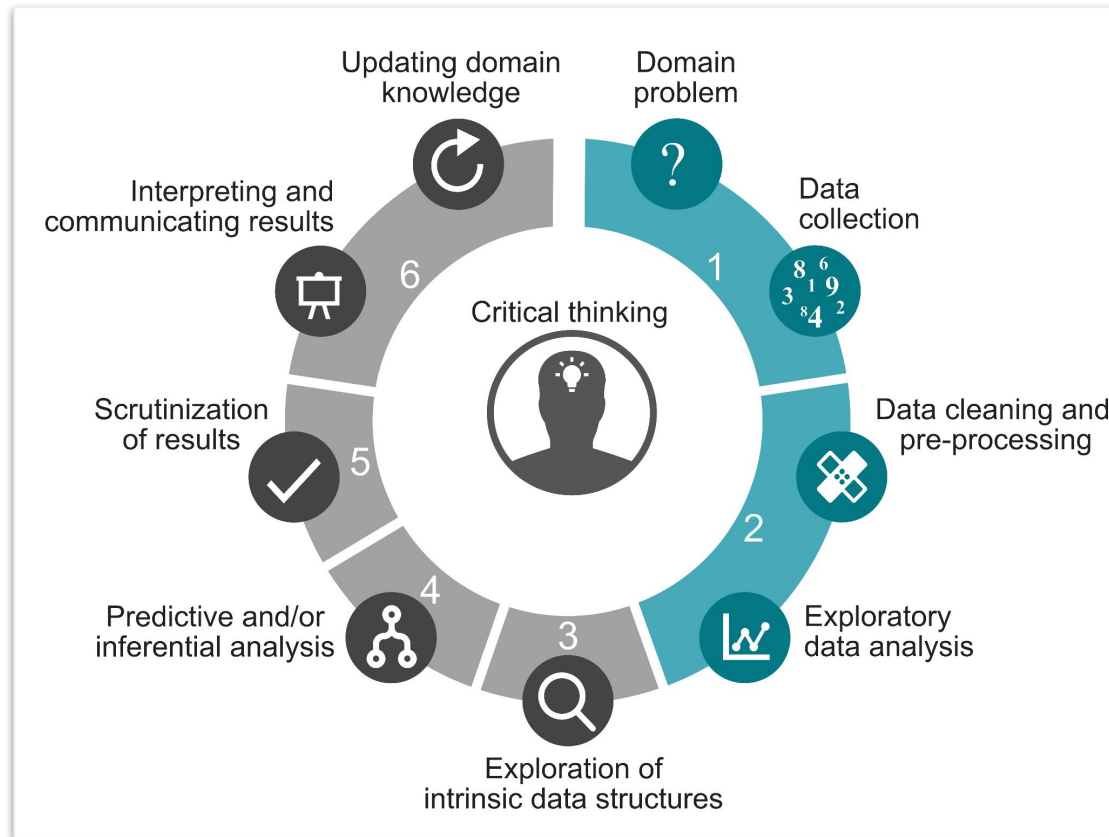
- + You might notice a lot of "junk" files that aren't worth tracking (e.g., they are useless to collaborators and/or they change too frequently)
- + Add these files to your .gitignore, e.g.,

```
*.DS_Store  
*/data/*  
*__pycache__*  
*.ipynb_checkpoints*
```

Details: <https://tiffanymtang.github.io/dsip-s25/#pushing-your-changes-to-github>

Getting started with the Redwood Lab

Lab 1: A Macroscopic in the Redwoods [Tolle et al. (2005)]



Lab 1: A Macroscope in the Redwoods [Tolle et al. (2005)]

- + **Coastal redwood trees:**
very tall, very old
- + 44-day study in Sonoma, California
(April 27, 2004 5:10pm - June 10, 2004 2pm)

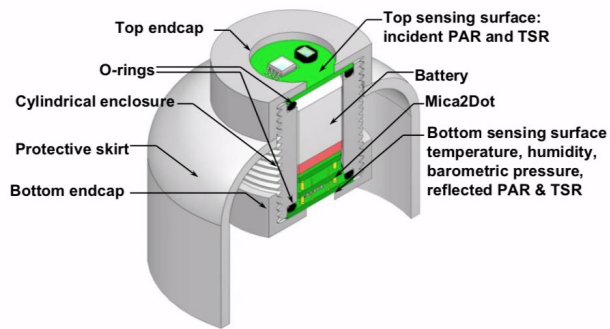


Figure 2: Sensor node and packaging

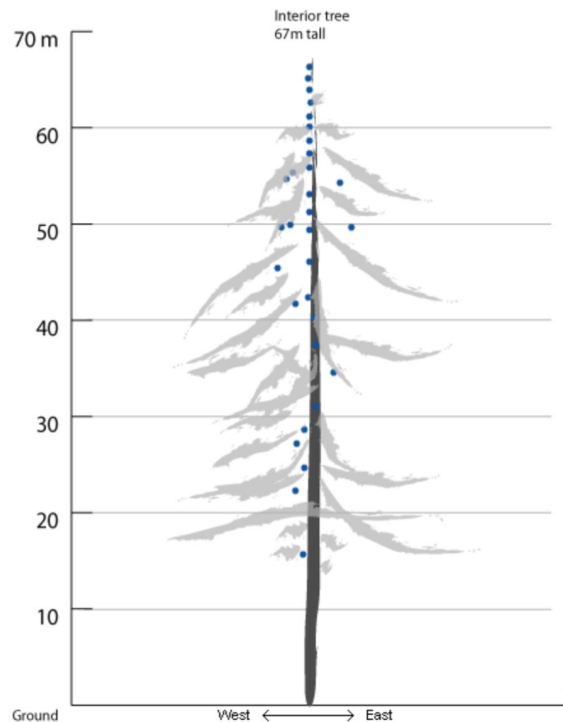
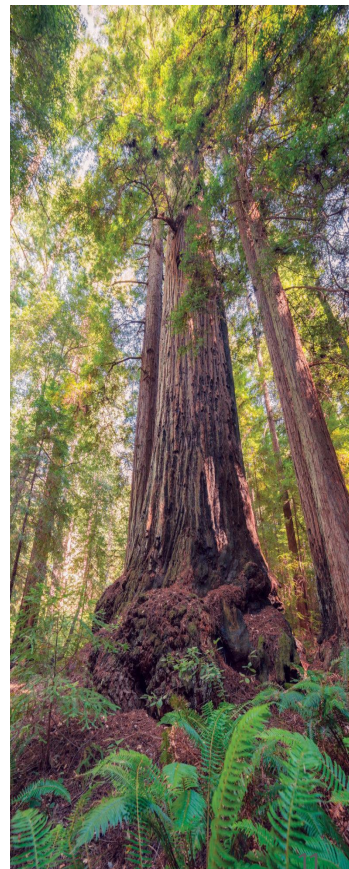


Figure 1: The placement of nodes within the tree



Getting started with the redwood lab

Available templates on the course repository ([dsip-s25](#))

- + For your **reproducible report**: you can find R Markdown (R only) and quarto templates (R, Python) in [notebooks/](#)
- + Some **loading** and **cleaning** functions have already been populated in [R/](#) and [python/](#) folders
 - Some functions have scaffolding but not yet filled in; this is your to-do

Your tasks for today

1. Load in the data

- a. Epoch/dates and redwood datasets have already been filled out for you.
- b. You need to fill out the `load_mote_location_data()` in the `load.R/load.py` file.

2. Look and "play" around with the data in order to:

- a. Try to **identify as many issues or oddities** with the data as you can.
Hint: there are many!!
- b. Also **think** about how you might address these issues and clean the data. Jot down these ideas, but no need to take the time to implement it *yet*.
 - *Time permitting:* you can start implementing your ideas, but prioritize identifying the issues over fixing them.

Your tasks for today

3. **Start** data cleaning:
 - a. Remove duplicate rows/observations
 - b. Remove missing data
 - c. Merge redwood log and network data
 - d. Merge redwood data with epoch/dates data and mote location data
 - e. *Time permitting*: other data cleaning steps that you think are appropriate

Recap + Next Time

Recap

- + **Data cleaning** is a highly iterative process.
- + My two cents:
 - Don't be afraid to ask lots of questions. Better to ask than to assume (more likely than not, incorrectly)
 - Read all documentation

Next Time

- + More hands-on practice with exploratory data analysis
[\[chapters 4 and 5 from VDS textbook\]](#)