Reference notes

# Basic Git workflow

https://swcarpentry.github.io/git-novice/reference

--- open git bash on desired folder (i.e. Version 2 - Manuscript)

> git status

> git add .

> git commit –m “<commit message goes here>”

> git push origin master

> nano file.txt } open in a text editor

> cd . } change directory to parent folder?

> ls -a } list all contents of the folder, -a show all including hidden folders

Lab notebook

# Nov 14

## Notes about project structuring

* Pick a naming convention and stick with it. Do not mix them. Perhaps consider writing your own style guide.
* Use functions whenever you can, and keep them short.
* Use **packrat** to manage dependencies to keep things consistent and reproducible.
* Versioning systems like git keep you from holding onto old versions out of fear of needing them in the future. Use versioning software.

https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000424

In practice, therefore, the scripts that I write tend to fall into these four categories:

1. **Driver script.** This is a top-level script; hence, each directory contains only one or two scripts of this type.
2. **Single-use script.** This is a simple script designed for a single use. For example, the script might convert an arbitrarily formatted file associated with this project into a format used by some of your existing scripts. This type of script resides in the same directory as the driver script that calls it.
3. **Project-specific script.** This type of script provides a generic functionality used by multiple experiments within the given project. I typically store such scripts in a directory immediately below the project root directory (e.g., the msms/bin/parse-sqt.py file in [Figure 1](https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000424#pcbi-1000424-g001)).
4. **Multi-project script.** Some functionality is generic enough to be useful across many projects. I maintain a set of these generic scripts, which perform functions such as extracting specified sequences from a FASTA file, generating an ROC curve, splitting a file for *n*-fold cross-validation, etc.

* I’m interested to try writing my manuscript as an r markdown file. One of the biggest barriers seems to be sharing with collaborators, but apparently rmd files can be knit as a word document: https://rmarkdown.rstudio.com/articles\_docx.html

## Setting up Git

Links from the Software carpentry collaborate document

**Resources**

<https://swcarpentry.github.io/git-novice/reference>

<https://www.atlassian.com/git>

**Git cheat sheets**

<https://www.atlassian.com/git/tutorials/atlassian-git-cheatsheet>

<https://github.github.com/training-kit/downloads/github-git-cheat-sheet.pdf>

Following the first link

* Set user name, email, line endings
* Come back to setting the default text editor if it’s not what I like when I try it

-I can click on a folder and say Git Bash here to start in that directory

-Could use GUI or Bash. I think for now I will use Bash since that’s what I learned before and what I know I have good instructions for. Maybe when I’m ready I can write down a list of the commands that I would use regularly when committing etc

-Created a .gitignore file. Set it up to ignore everything that is in the “data” folder

-what adding all of the files that I had in the repository (at this point, just folders, r project files, and the git ignore file), I got the following warning message from git bash. I am going to ignore it based on what I read on the internet.

warning: LF will be replaced by CRLF in .Rhistory.

The file will have its original line endings in your working directory

-set up and connected folder to a github repository

# Nov 15, 2019

## Setting up Git

-added a README file

-did another commit/push of edits

-created the reference notes at the beginning of this document

# Nov 19, 2019

## Preparing PRISM data – getting started

-added the new version of the raw prism data to data folder: unedited excel file and CSV file with the top two rows that were full of Jennie and Isabelle’s comments removed.

-Laurent gave me his script for cleaning the PRISM data

-created a library script for attaching packages, user defined functions, defining objects that will be used regularly such as coordinate reference systems

-started working on a prism data cleaning script, based on scripts Laurent and I have written already

# Nov 20, 2019

## Preparing PRISM data – just looking around

-started a list of things that I don’t understand about the PRISM data, or errors, @ top of this document

-looking through columns I might want to find inconsistencies/surprises (recorded above)

# Nov 21, 2019

## Preparing PRISM data – getting on the same page as Laurent and Natalie

-Added temporary files starting with ~ and .Rproj.user to my git ignore file

-tried to add git configuration setting to make comparing version of .docx files within git more useful. Not 100% sure if it worked or if it matters. Instructions found here:

<https://stackoverflow.com/questions/22439517/view-docx-file-on-github-and-use-git-diff-on-docx-file-format>

<https://github.com/vigente/gerardus/wiki/Integrate-git-diffs-with-word-docx-files>

-spent a while talking with Laurent to figure out why prism GIS files of zones and plots don’t align well, and possible solutions

-spent an hour on the phone with Natalie and Laurent to talk about what we are finding with the PRISM data. Tried running one of Natalie’s scripts for her afterwards

-moved all of my PRISM notes from top of this document into a shared google doc with Natalie and Laurent.

-read the document describing the GIS analyses that Tyler did

# Nov 22, 2019

-worked on filtering prism dataset, particularly time and date columns. Some of them are annoying to deal with and I’m going to put it off until I’m sure that I actually need those columns.

# Nov 25, 2019

-making a plan of what I need to do next. Finished making this document so that I could talk to Paul about it

-this seems like a good reference for SDM process: <https://damariszurell.github.io/SDM-Intro/>

-worked a bit on filtering prism dataset

-received another new version of the PRISM dataset. I had to re-check some of the columns that I had checked last week. Some of the have changed slightly

# Nov 26, 2019

## Bi-Weekly Meeting with Paul

-came up with the idea of using Path or SEM modelling to look at the interrelationships between the predictor variables

-made me realize (again) that my grid doesn’t fully need to line up with the plots, I can just make my own

-better idea to start modelling, then fix my layers once I know what I really need

-talked about time lags ideas but we both were confused about what you actually measure and how it would all work

-talked about computational issues I might run into – options if I need more power are 5th floor, Carleton physics, ECCC supercomputers, ACEnet

-which one has the shortest queue?

## Other tasks

-Made notes about SEM and Path Analysis in literature review document

# Nov 27, 2019

## Trying real hard to finish cleaning up the PRISM data

-changing variables classes

-fixing mistakes (species in wrong group etc)

-trying to edit mutate\_cond function so that it would work with NA values. The test script I made suggested to me that it didn’t work the way I thought it did. Did not figure this out.

# Nov 28, 2019

## Finally finished cleaning up PRISM data

-finally finished going through all of the columns: looking for errors, filtering out plots that don’t meet my criteria for inclusion, cleaning formatting

-recorded all of these errors on shared doc / to send to Isabel

# Nov 29, 2019

## Comparing and extracting data from PRISM GIS

-checked the specs of the INGO environment so that I can compare to other options:

-24GB RAM, 6 processors

-GIS files and excel file have different plot names. GIS file names are likely in some of the historic plot name columns

-maybe the proportion of plot surveyed is more reliable than the areas?

-SIR 38D isn’t in any of the GIS files. Would be useful to write a script to match gis files with excel file plots to see where they match up

-first create my list of all plots in the excel file so that I don’t have to look at multiple rows?

# Dec 2, 2019

## Itemizing all the plots that we need to make decisions about how/whether to include them

-Made lists of plots that were surveyed multiple times in the same year, plots that were surveyed in multiple years, and some of the plots that were field selected

# Dec 3, 2019

## Continue list of plots that need decisions made about them

-Finished lists of plots that were field selected and field modified

-Read a bunch of the different comment columns to try to understand what’s happening

# Dec 4, 2019

## Continue list of plots that need decisions made about them

-started trying to write down my opinions and questions about decisions

-looking at specific plot examples to understand them

-wrote a script to count the number of surveyors in each plot

\*\*\*This video about tidyverse tricks looks worthwhile: https://www.youtube.com/watch?v=NDHSBUN\_rVU