Session 1 - R Basics

Ya-Feng Wen

2021-05-26

Contents

Objectives

- Project Management
  - Git and version control
  - Command Line
- RStudio and R markdown
- Loops/Functions
- ggplot2
- dplyr
- lubridate

Git and version control

- Install Git
  - Mac: use terminal, `git --version`. If Git is not found, you will be asked to install it.
  - Windows: search *Git for Windows*. Download the most recent build.
    * Select the *nano* as default editor. If familiar with *vi* or *vim*, you can use these.
    * Select *Git and optional Unix tools from the Windows Command Prompt* so you can use Git from within RStudio
    * Once Git/Git Bash is installed, in Tools -> Global Options -> Terminal -> select *Git Bash*.

- Connect RStudio with GitHub

```
# Bash, run in Terminal
# Not run, modify accordingly
git config --global user.name "Your Name"
git config --global user.mail "your@email.com"
```

- Tools -> Global Options -> Git/SVN, enter a path for the Git executable. (Default on the Windows: *C:/Program File/Git/bin/git.exe*)
- Create *SSH RSA Key* by clicking the *Create RSA Key* button. This enable you to avoid entering password each time trying to access GitHub repository.
- Github setup
  - Create a repository on GitHub or GitHub Enterprise
- Initialize a Git directory
  - Initialize a Git directory and connect it to the upsteam repository
  - Set up a R project

```
# Bash, run in Terminal
# Not run, modify accordingly
pwd
mkdir directory
git init https://github.com/yafengwen/project.git
cd project
```

- Overview of Git

Main actions:

1. **clone** an existing GitHub Upstream Repository, including the entire Git structure: *Working Directory*, *Staging Area*, and *Local Repository*.
2. **pull** changes from the GitHub repo
3. **stage** (add) files
4. **commit** changes to the local repo
5. **push** changes to the GitHub repo
6. **branch** and **merge** to facilitate collaboration. See more details here

Figure 1: Git Overview

Some usefual functions:

```
# Bash, run in Terminal

# Compare files in the Working Directory with GitHub repo
git status
```

```
# Add a file to the Staging Area
git add new-file.txt
git status

# Commit the changes to the local repo
git commit -m "add a message"
git status

# keep track of all the changes
git log new-file.txt

# Push the changes to the upstream repo
git push

# Pull changes from the upstream repo to working direcotry
git pull
```

- Use Git and GitHub in RStudio
- Other software to facilitate the version control
  - GitHub Desktop
  - GitKraken

Command Line

- `pwd`: show full path of the working directory
- `ls`: list directory content
  - Arugment: -a (all), -l (long), -t (chronological order), -r (reverse order), -lart (combine all the aguments)
- `mkdir`, `rmdir`: make and remove a directory
- `cd`: change directory
  - `cd ~`, `cd ..`, `cd ../..`
- `mv`: move files or rename files
- `cp`: copy files
- `rm`: remove files
  - Argument: -r (recursive), -f (force), -rf (force to remove files recrusively)
- `less`, `more`: view files

Practice

1. Create the following folders using terminal: data, rds, figs
2. Create relevant `.R` and `.Rmd` files: download-data.R, wrangle-data.R, analysis.R, report.Rmd

RStudio and R markdown

Loops/Functions

Practice

ggplot2

Practice

dplyr

- `filter()`: select rows
- `select()`: select columns
- `arrange()`: reorder rows
- `mutate()`: create new variables based on existing variables
- `summarize()`: summary values within a columns
  - `group_by()`: use with the main 5 functions

Syntax

1. The first argument: a data frame
2. The subsequent arugments: what to do with the data frame, using the variable names without quotes
3. The result: a new data frame

Practice

lubridate

- `today()`
- `now()`
- `ymd_hms()`
- `make_date()`

Practice

Resources

1. R for Data Science by Hadley Wickham and Garrett Grolemund, online at https://r4ds.had.co.nz/
2. Introduction to Data Science by Rafael A. Irizarry, online at https://rafalab.github.io/dsbook/
3. R cheetsheets https://rstudio.com/resources/cheatsheets/
   - Data Transformation Cheatsheet
   - Dates and Times Cheatsheet
   - R Markdown Cheatsheet
   - Data Visualization Cheatsheet
4. R Markdown: The Definitive Guide by Yihui Xie, J. J. Allaire, Garrett Grolemund, online at https://bookdown.org/yihui/rmarkdown/
5. R Packages by Hadley Wickham and Jenny Bryan, online at https://r-pkgs.org/

Session Info

```
sessionInfo()
```

```
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
```

```
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## loaded via a namespace (and not attached):
##  [1] compiler_4.1.0    magrittr_2.0.1    tools_4.1.0       htmltools_0.5.1.1
##  [5] yaml_2.2.1        stringi_1.6.1     rmarkdown_2.8     knitr_1.33
##  [9] stringr_1.4.0     xfun_0.23         digest_0.6.27     rlang_0.4.11
## [13] evaluate_0.14
```