

# S&DS 361 Homework 0: Software Prep

Due Tue Jan 23, 2024

## Part 1: Course prep and software installation

**1. Complete the Course Survey in the Quizzes section of Canvas** After you submit the quiz, take a screenshot of the time stamp and Question 1. It should look something like this.

---

❗ Correct answers are hidden.

Submitted Jan 21 at 8:31pm

Question 1

Describe what program you are in.

☐ Undergraduate student, S&DS major

☐ Undergraduate student, S&DS concentration

☐ Undergraduate student, Other major (please describe in next question)

☐ Graduate student, S&DS MA or MS program

☒ Graduate student, S&DS Ph.D. program

You Answered

Edit the above file name and path to show your screenshot and ensure that it appears when you knit your document.

**2. Download and install the latest version of R** See <https://bmacgtpm.github.io/notes/software-installation.html> for some potentially useful tips.

The following code will show your version of R when you knit the document. It should say **R version 4.3.2** or later. Make sure it appears when you knit your document.

```
R.Version()$version.string
```

```
[1] "R version 4.3.2 (2023-10-31 ucrt)"
```

**3. Download and install the latest version of RStudio.** See <https://bmacgtpm.github.io/notes/software-installation.html> for some potentially useful tips.

This code will show your version of R when you knit the document. Make sure it appears when you knit your document. It should say 2023.12.0+369 (or later).

```
rstudioapi::versionInfo()$long_version
```

```
[1] "2023.12.0+369"
```

**4. Install/update packages** See <https://bmacgtpm.github.io/notes/software-installation.html> for the packages to install.

Do not write R code for installing packages in this R Markdown. You don't want packages to install every time you knit this document.

Check that you can load all of the libraries by running this chunk of code and showing that it executes without error. There may be some messages, and maybe warnings about versions. Those are ok. Make sure the output appears when you knit the document.

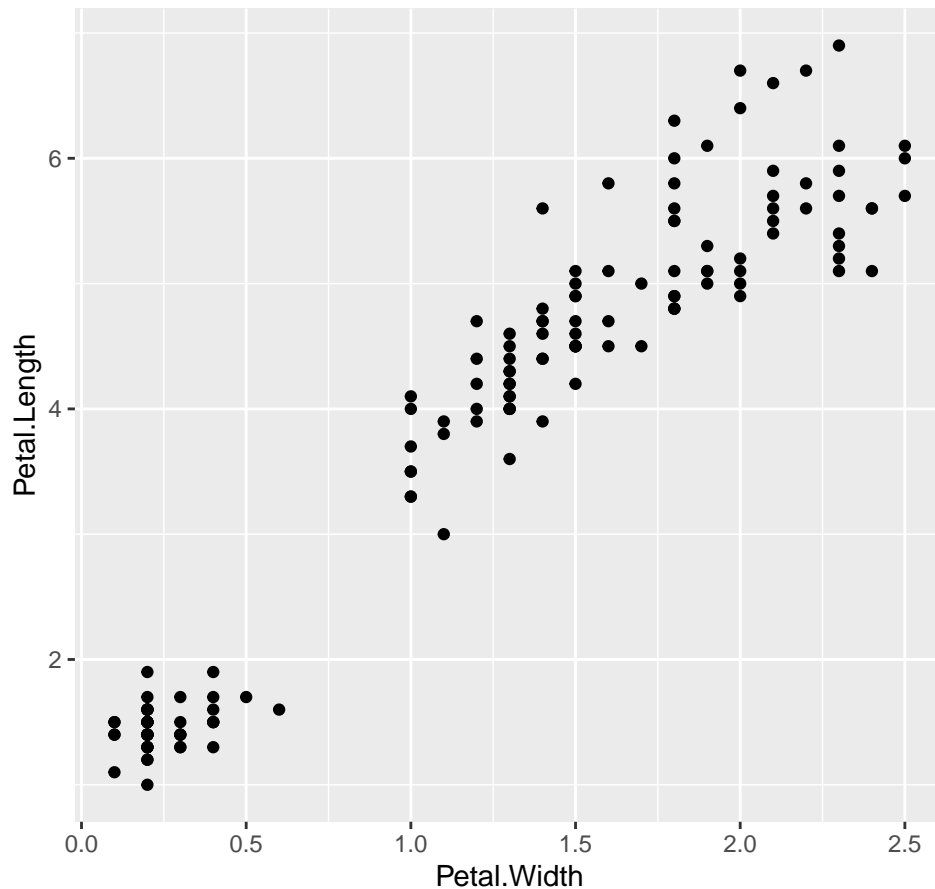
```
library(knitr)
library(plotly)
library(scales)
library(DT)
library(leaflet)
library(gganimate)
library(gifski)
library(png)
library(corrplot)
library(GGally)
library(ggmap)
library(shiny)
library(MASS)
library(lme4)
library(arm)
library(pROC)
library(MLmetrics)
library(viridis)
library(RSelenium)
library(rvest)
library(randomForest)
library(FNN)
library(caret)
library(pls)
library(devtools)
library(splines)
library(RecordLinkage)
library(rsconnect)
library(grid)
library(foreign)
library(maps) ## leave uncommented. For some reason GitHub Actions had a problem when this wasn't explicit

## load tidyverse last!
library(tidyverse)
library(pubtheme)
```

**5. Check gganimate** See <https://bmacgtpm.github.io/notes/software-installation.html>. The code from that page is below, except a custom title has been added. Replace my name with yours, uncomment the animation code, run all of this code.

```
# We'll start with a static plot
g = ggplot(iris,
           aes(x = Petal.Width,
               y = Petal.Length)) +
  geom_point() +
  ggtitle("Edoardo Marchionni's animation")
g
```

Edoardo Marchionni's animation

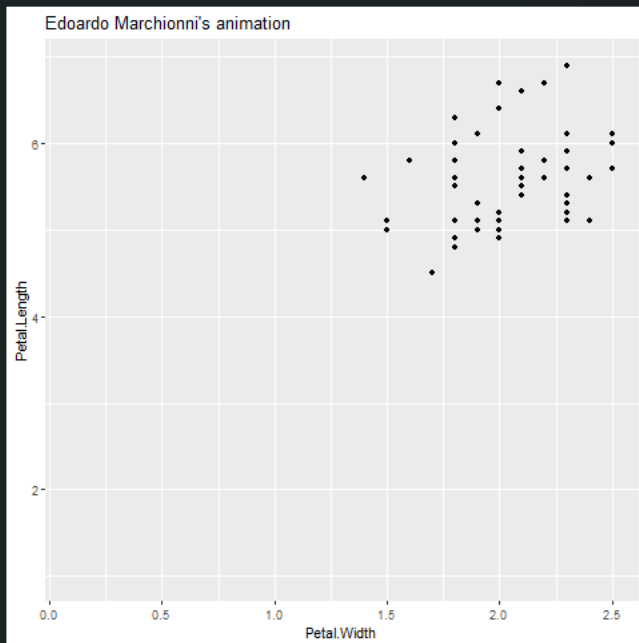


```
# a = g +
#   transition_states(Species,
#                     transition_length = 2,
#                     state_length = 1)

# a ## check that the animation works

## save the animation
# anim_save(a,
#           filename = 'img/test animation.gif')
```

There should be a static plot and an animated plot above. If the `anim_save` worked properly there should be a new `test.gif` in the `img` folder that has your name. Take a screen shot of your animated gif when the points are near the upper right and show the screenshot here:



If all of that works, `gganimate` is good to go! If that doesn't work, see the tips at <https://bmacgtpm.github.io/notes/software-installation.html>.

Once you have created the animation, comment out the code that creates the animation (as I have done above). This document won't knit to PDF with the animation code in it. You can only knit to HTML.

**6. Bookmarks** See <https://bmacgtpm.github.io/notes/software-installation.html>.

## Part 2: Github

**7. Create a GitHub account at <https://github.com/> if you don't have one. Submit your GitHub username in Quizzes -> Course Survey on Canvas.**

**8. Download GitHub Desktop at <https://desktop.github.com/>.** Take a screenshot showing Github Desktop (or different software, or the command line) and show it here.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tutti i diritti riservati.

Installa la versione più recente di PowerShell per nuove funzionalità e miglioramenti. https://aka.ms/PSWindows

PS C:\Users\edoar\Desktop\S&DS 661 Data Analysis\361-Spring-2024> git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
      <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone             Clone a repository into a new directory
  init              Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add               Add file contents to the index
  mv                Move or rename a file, a directory, or a symlink
  restore           Restore working tree files
  rm                Remove files from the working tree and from the index
  sparse-checkout   Initialize and modify the sparse-checkout

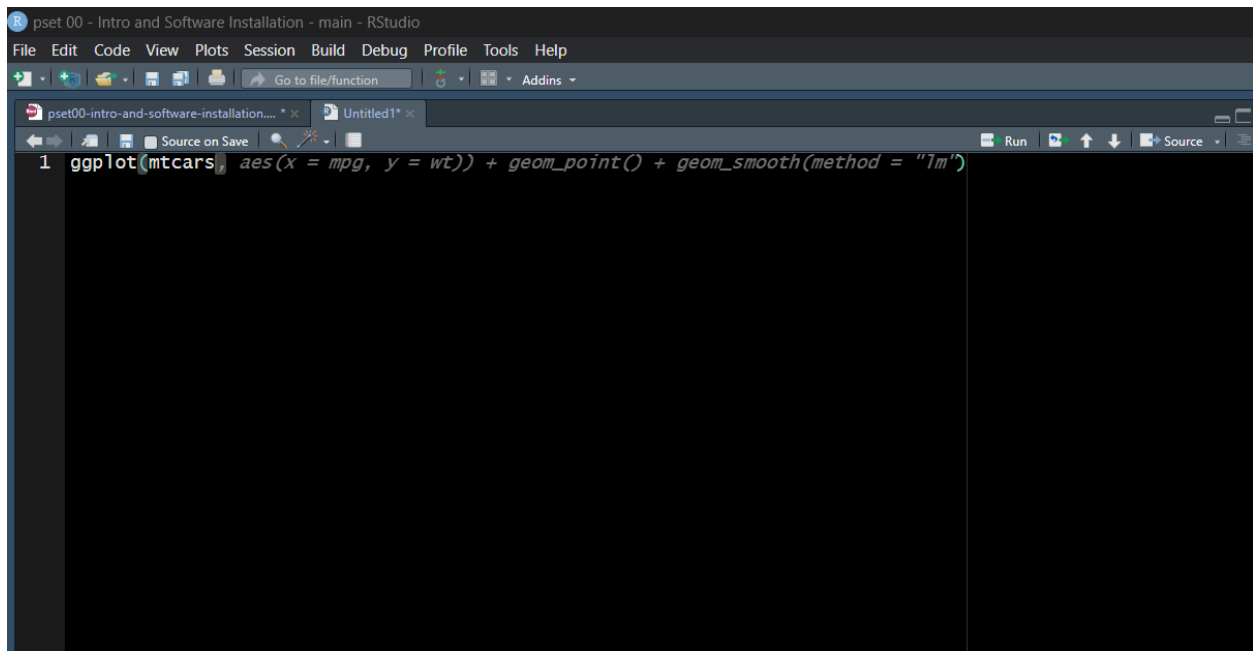
examine the history and state (see also: git help revisions)
  bisect            Use binary search to find the commit that introduced a bug
  diff              Show changes between commits, commit and working tree, etc
  grep              Print lines matching a pattern
  log               Show commit logs
```

If you have experience with Git/Github, and prefer to use different software or the command line, that's fine, but we may not be able to help if you have a problem.

**9. Clone the repo <https://github.com/bmacGTPM/361-Spring-2024> and create PR as follows.** Clone the repo, create a new branch and name the branch `Firstname Lastname` your first and last name. Make an edit to the R Markdown file `pset00-GitHub-pull-request-Firstname-Lastname.Rmd` to have your name at the top instead of mine. Commit that to your branch, push those commits to GitHub, and create a pull-request to the `main` branch on the 361-Spring-2024 repo. Make the title of the pull request your first and last name. For help getting started, see <https://docs.github.com/en/desktop/installing-and-configuring-github-desktop/overview/getting-started-with-github-desktop>.

If you find yourself getting many notification, you can go to <https://github.com/watching> to choose what notifications you get. This page has some more info on notifications/subscriptions.

**10. Set up Github Copilot in RStudio** See <https://bmacgtpm.github.io/notes/github-copilot-in-rstudio.html>.



I had troubles to set up Copilot, I wrote an email to Brian.

If you use Github Copilot elsewhere, take a screenshot of whatever software you use.