Psych 205 - Lab 2 - Prof. Key

The goal of this lab is to get practice describing data and working in Quarto.

Download this file as a .qmd from the course page.

In Lecture.

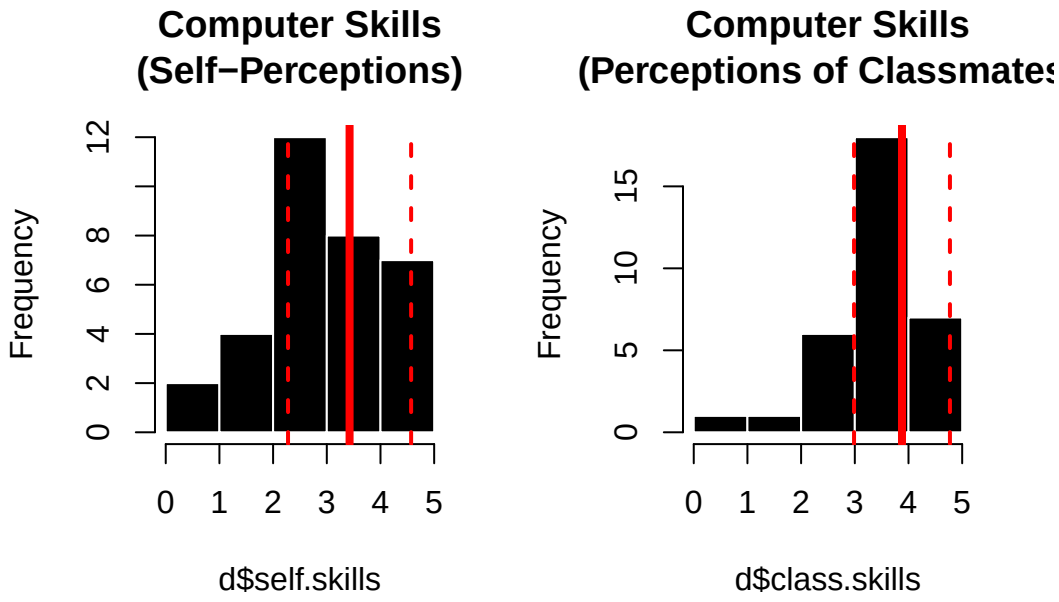
1. One advantage of Quarto (and R Markdown) is that you can run code in a document. You do this using a "code block". In the space below, insert an R code block, type out a math equation that used to give you difficulty as a kid into the code block below, and run the code to see the result. Below the code block, add text for humans to determine whether R got the math problem correct or not. Then render the document as a .pdf and .html file. Did this work?
2. Load the "grad onboarding" survey into the code block below, and answer the following questions. *Note : to successfully render code in the document, you must a) explicitly load the dataset into your Quarto document and b) make sure you have no errors in your code. :)*
    - Graph the variables `d$self.skills` and `d$class.skills` side by side using the par() function. Change the formatting of the graph to make it look ready for presentation. Add vertical lines to each graph to illustrate the mean (solid line) and standard deviation (dashed lines).
    - Below each graph, report the mean and standard deviation of both variables, and interpret what these statistics tell you about the individuals in our class. (Who cares? What do these statistics tell us?)

```
d <- read.csv("~/Downloads/grad_onboard_SP25.csv", stringsAsFactors = T, na.strings = "")
par(mfrow = c(1,2))

hist(d$self.skills, breaks = c(0:5),
     col = 'black', bor = 'white', main = "Computer Skills\n(Self-Perceptions)")
abline(v = mean(d$self.skills), lwd = 4, col = 'red')
abline(v = mean(d$self.skills) + sd(d$self.skills),
       lwd = 2, lty = "dashed", col = 'red')
abline(v = mean(d$self.skills) - sd(d$self.skills),
       lwd = 2, lty = "dashed", col = 'red')

hist(d$class.skills, breaks = c(0:5),
     col = 'black', bor = 'white', main = "Computer Skills\n(Perceptions of Classmates)")
abline(v = mean(d$class.skills), lwd = 4, col = 'red')
abline(v = mean(d$class.skills) + sd(d$class.skills),
```

```
       lwd = 2, lty = "dashed", col = 'red')
abline(v = mean(d$class.skills) - sd(d$class.skills),
       lwd = 2, lty = "dashed", col = 'red')
```



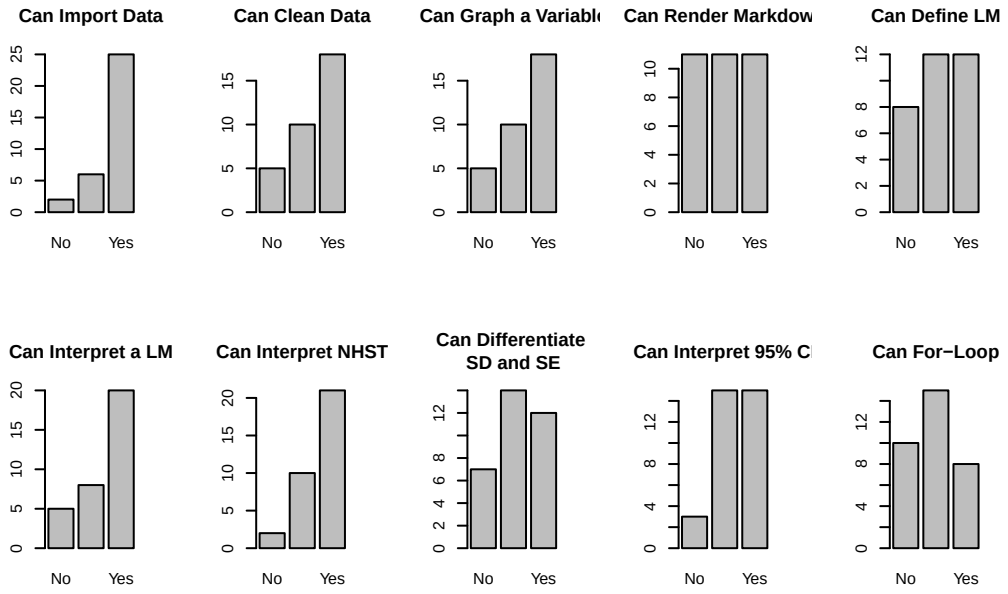
I see that students rated their classmates as having higher computer skills (mean = 3.88) than they rated themselves (mean = 3.88. There was also more variation in people's self-perceptions (sd = 1.15) than perceptions of the class 0.89. One possibility is that people were coming up with a less differentiated stereotype about the average other student's skill, but had access to more information when making their own self-perception of skill.

3. Split your graphics window into a 2x5 grid, and graph each of the 10 "can.*" variables in the dataset (e.g., "can.import", "can.clean", etc.). Make sure each graph contains the name of the variable and that the graph looks good / is intelligible. *Note : you can and should use a for-loop to do this!* Then, report the frequencies of these variables - what are some things you observe about the data? Does this make sense given what you know about the participants?

```
d[,7:16] <- lapply(d[,7:16], factor, levels = c("No", "Maybe", "Yes"))

par(mfrow = c(2,5), cex = .5)
graph.names <- c("Can Import Data", "Can Clean Data", "Can Graph a Variable", "Can Rende
                 "Can Interpret a LM", "Can Interpret NHST", "Can Differentiate \nSD and
for(i in c(7:16)){
```

```
  plot(d[,i], main = paste(graph.names[i-6]))
}
```



In Section

4. Choose another dataset from the course files (note : choose a dataset from a folder not labeled "Repeated Measures"; these data will need to be analyzed using within-person methods, which we will not learn about until later in the semester), or choose your own dataset. Look over the accompyinaning codebook / article, and identify at least one numeric and one categorical variable that seem interesting to you. Graph these variables, report the relevant descriptive statistics for each variable. Then explain what these statistics and graphs tell you about the individuals in the dataset, and what other questions you might ask about these individuals (e.g., what do the data NOT tell you?)

I'm going to work with the "Perceptions of the Wealthy" study again. [FULL KEY TBD]

```
d <- read.csv("~/Dropbox/!GRADSTATS/Datasets/Perceptions of the Wealthy/Dawtry 2015 Study 1a
```

On Your Own

5. **Data Cleaning Is So Much Fun.** Professor also collected an on-boarding survey for his undergraduate students. While he *loves* cleaning data as much as he likes painting

white fences, he's agreed to let you participate in the process. Lucky you! Complete the following steps - ideally in R (since it creates a record of every step that you've done to change data, which is very important from a transparency perspective). But if you are struggling, use excel / google sheets and save / export as a .csv file.

- Create a new .R script, and title this something that describes you are using it to clean a data file.
- Load the "raw" datafile into R and save it as an object.
- Remove the last variable in the dataset.
- Change the names of the variables in the original undergrad dataset to match the same names as in the grad_onboarding survey. This is how I change the names of a dataset. Note that the length of the new names object must match the old names, or things will get messed up. Good to be careful here and double check your work! Sure is fun cleaning this data, huh!?
  - `newnames <- c("var1", "var2")`
  - `names() <- newnames`
- Remove the "string data" that is associated with the high and low end of the likert scale numeric data (e.g., "1 (Strongly Disagree)" should just be "1"). The code below takes a datafile (`d`), and then substitute the replacement text for every instance of the pattern text. This is good for condensing long answers (e.g., "Yeah, I think so" = "Yes") or converting string data into numeric data (e.g., "1 (Strongly Disagree) –>"1"). For numeric data you will still need to use " " in the lapply code.
  - `lapply(d, gsub, pattern = "Old Text", replacement = "New Text", fixed = TRUE)`
- Use the `write.csv()` function to save your new data object as a .csv file. You will need to specify a path to store the .csv file.
- **This is the only thing to submit to show your data cleaning worked :** Load the newly cleaned data into this Lab assignment, and graph a) undergraduate student's beliefs about their own computer skills (d$self.skills), their beliefs about their classmates computer skills (d$class.skills), and their epistemological beliefs (d$epistemology). Change the formatting of each graph to make it look ready for presentation. Below the graph, report the mean, standard deviation, and range of both numeric variables. Then below these statistics, interpret what these statistics tell you about the individuals in the undergraduate class.

6. **Challenge Problem**[1]**.** Quarto (and Markdown) can also be used to make a website. Make yourself a personal website, and use one of the methods to host this website for the world to see [I use github.] Yeah!

[1] Optional, but potentially fun and useful if you have the time and interest.