Assignment 2

SOSC 13200-2

Vincent Zhang, Collaborated with: Tessie Dong

Due 1/24 @ 11:59pm

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 100), tidy = TRUE)

# packages
library(ggplot2)
library(stringr)
# library(tidyr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.6     v purrr   0.3.4
## v tidyr   1.1.4     v dplyr   1.0.7
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
# load data
public <- read.table("Datasets/card-krueger_1994/njmin/public.dat", na.strings=c(".","NA"))
codebook <- readLines("Datasets/card-krueger_1994/njmin/codebook.txt")
```

1 Mean & variance

Examine the summary table below. The table is based on Card & Krueger's (1994) minimum wage and employment data. The table gives the proportion of restaurants in each region that are Kentucky Fried Chicken restaurants (KFC).

Region	% KFC
South NJ	12.91
Central NJ	11.62
North NJ	15.59
NE Philadelphia, PA	12.25
Easton, etc., PA	13.75

1.1 (5pts) In the chunk below, calculate the mean proportion. Do not use any R functions. Use only arithmetic operators (+, -, /, *, ^).

```
# calculate the sample mean without using any R functions
South_NJ <- 12.91
Central_NJ <- 11.62
North_NJ <- 15.59
NE_Philadelphia <- 12.25
Easton_etc <- 13.75

sample_mean <- (South_NJ + Central_NJ + North_NJ + NE_Philadelphia + Easton_etc) / 5

sample_mean
```

```
## [1] 13.224
```

1.2 (5pts) Calculate the sample variance. As above, do not use any R functions. Use only arithmetic operators. You may also use any object that you created when calculating the sample mean above.

```
# calculate the sample standard deviation without using any R functions

n <- 5
regions <- c(South_NJ, Central_NJ, North_NJ, NE_Philadelphia, Easton_etc)

sum <- 0
for (region in regions) {
  sum <- sum + (sample_mean - region)^2
}

sample_variance = sum / (n - 1)

sample_variance
```

```
## [1] 2.37368
```

1.3 (5pts) Calculate the sample standard deviation. As above, do not use any R functions. Use only arithmetic operators. You may also use any objects that you created when calculating the sample mean and standard variance above.

```
# calculate the sample standard deviation without using any R functions.

sample_STD = sample_variance^0.5 # raising to the half power is equivalent to sqrt()

sample_STD
```

```
## [1] 1.540675
```

2 Setup & cleaning

2.1 (No points) In the remainder of the Assignment, you will work with Card & Krueger's (1994) minimum wage and employment data. In the `setup` chunk, load the authors' dataset. You should also load the `ggplot2` package. Later on, you may wish to return to the `setup` chunk to load additional packages. But that is not strictly necessary.

2.2 (No points) In the chunk below, repeat your cleaning procedure from Assignment 1. Specifically: (a) Rename the variables in the dataset using the names given in the codebook; (b) create the authors' full-time-equivalent employment variable (fte), adding it to the dataset. You should create the fte variable for waves 1 and 2. Be sure to fix any errors in your cleaning procedure from Assignment 1.

```
# rename the variables using the names given in the codebook
variables <- c()


for (line in codebook) {
  first_word <- word(line, 1)

  if (length(first_word) == 0) {
    next
  }

  if (str_detect(first_word, "[[:upper:]]") && !str_detect(first_word, "[[:lower:]]")) {
    # if the first word is UPPER case
    # print("Inserted into variables")
    variables <- append(variables, first_word)
  }
}

colnames(public) <- variables
rownames(public) <-  make.names(public$SHEET, unique=TRUE)
# create the full-time-equivalent-employment variable for waves 1 and 2

public$FTE1 <- (public$EMPFT + 0.5*(public$EMPPT) + public$NMGRS)
public$FTE2 <- (public$EMPFT2 + 0.5*(public$EMPPT2) + public$NMGRS2)
```

3 Summarizing variables numerically

3.1 (25pts) Create a table that summarizes the following variables:

- chain
- fte
- wage_st
- bonus
- fte2
- wage_st2
- bonus2

The challenge here is that your table should summarize these variables for NJ and PA, separately. For each variable, your table should give:

- mean
- standard deviation
- minimum value
- maximum value

The table should be professional in appearance. You may build it by hand (see my "pipe" table above for a simple example). Or, you can search for and use a function from an external pack

```
# create a summary table that is professional in appearance

NJ_public <- filter(public, public$STATE == 1)
NJ_public <- NJ_public %>% select(SHEET, CHAIN, FTE1, WAGE_ST, BONUS, FTE2, WAGE_ST2)
colnames(NJ_public) <- c("ID", "NJ_CHAIN", "NJ_FTE1", "NJ_WAGE_ST",
                         "NJ_BONUS", "NJ_FTE2", "NJ_WAGE_ST2")

PA_public <- filter(public, public$STATE == 0)
PA_public <- PA_public %>% select(SHEET, CHAIN, FTE1, WAGE_ST, BONUS, FTE2, WAGE_ST2)
colnames(PA_public) <- c("ID", "PA_CHAIN", "PA_FTE1", "PA_WAGE_ST",
                         "PA_BONUS", "PA_FTE2", "PA_WAGE_ST2")

combined_public <- full_join(x = NJ_public, y = PA_public, by = "ID")
# combined_public
# tbl_NJ <- stargazer(NJ_public, type="text", title="Summary for NJ")
# tbl_PA <- stargazer(PA_public, type="text", title="Summary for PA")

# tbl_combined <- merge(tbl_NJ, tbl_PA, all = TRUE)
tbl_combined <- stargazer(select(combined_public, -"ID"), type="latex",
                          title="Summary for NJ and PA",
                          omit.summary.stat = c("N", "p25", "p75"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Mon, Jan 24, 2022 - 22:50:57

3.2 (20pts) Think about the variables in your table: (a) What type of variable is each one? (b) For which variables does it make sense to calculate the mean and standard deviation? (c) Why?

a.

1. CHAIN is the type of fast food restaurant, and is thus a **categorical nominal** variable.
2. FTE is the calculated / estimated number of employees, and is thus a **numerical discrete** variable.
3. WAGE_ST is starting wage of the employees and is thus a **numerical continuous** variable, since ideally, the wage can range infinitely between two numbers (even though in practice, wages are usually whole numbers).
4. BONUS depicts whether or not starting workers get a bonus, and is thus a **categorical nominal** variable.

b. For only numerical variables (FTE, WAGE) does it make sense to calculate the mean. In a way, calculating the mean of BONUS coincidentally results in a meaningful result (the proportion of words that get a bonus), but this does not generalize for most categorical variables

Table 2: Summary for NJ and PA

Statistic	Mean	St. Dev.	Min	Max
NJ_CHAIN	2.109	1.093	1.000	4.000
NJ_FTE1	20.439	9.106	5.000	85.000
NJ_WAGE_ST	4.612	0.346	4.250	5.750
NJ_BONUS	0.236	0.425	0.000	1.000
NJ_FTE2	21.027	9.293	0.000	60.500
NJ_WAGE_ST2	5.081	0.105	5.000	5.750
PA_CHAIN	2.152	1.188	1.000	4.000
PA_FTE1	23.331	11.856	7.500	70.500
PA_WAGE_ST	4.630	0.352	4.250	5.500
PA_BONUS	0.291	0.457	0.000	1.000
PA_FTE2	21.166	8.277	0.000	43.500
PA_WAGE_ST2	4.617	0.357	4.250	6.250

c. Categorical variables only use numbers as a way to *depict* different categories of stuff, and thus averaging out the values of those variables is not meaningful in anyway. Consider the example of the CHAINs, where 1 is BK, 2 is KFC, 3 is ROY, 4 is Wendys. Suppose the average is 2.5. That has no meaningful significant because it could mean there were a lot of KFC and ROY, but it could also mean there were a lot of BK and Wendys in equal proportion. What about the BONUS example? In a way, we can argue that the mean of BONUS tells you what percentage of words get bonuses. For instance, of the mean of bonus is .7, we can say that 70% of people get bonuses. However, this is simply due to fact that there are only two categories, and thus allows it to be easy to utilize mean in a more meaningful way; Add a third condition, then we will realize that calculating the mean is nonsense!

4 Summarizing variables visually

4.1 (15pts) Examine Figure 2 from Card & Krueger (1994) on p. 777. What is the key takeway from the Figure? Lay out your answer in a few sentences (up to a short paragraph) below these lines.

The key takeaway that is immediately obvious in the figure is that 1: New Jersey had a minimum wage requirement after February 1992 that resulted in the majority (90%) of the wages of employees to be 5.05 dollars, and 2: meanwhile, Pennsylvania (intended to be some kind of control for comparison) had relatively the same proportions of wages ranged between 4.25 dollars to 5.55 dollars.

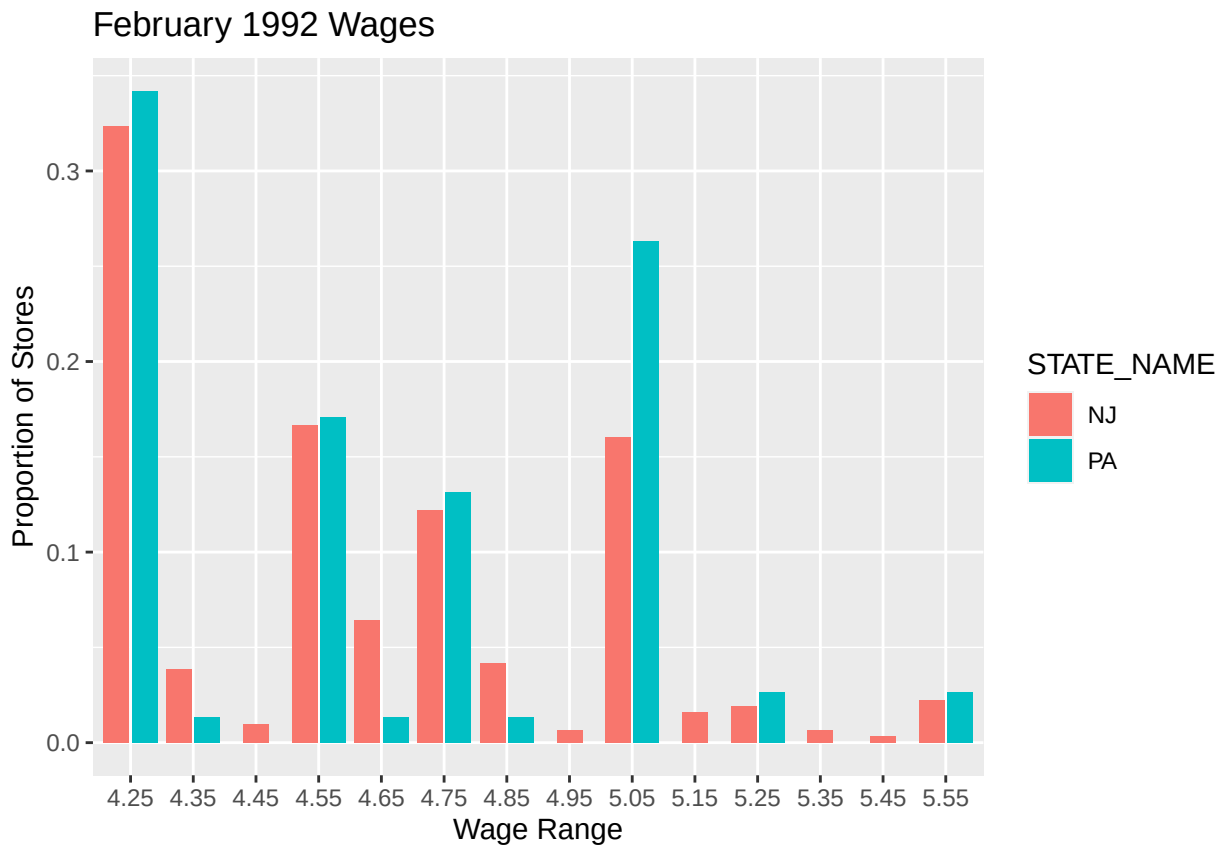
4.2 (25pts) Use `ggplot2` to replicate Figure 1 from Card & Krueger (1994). Your replication should be professional in appearance. (HINT: If you have not already completed the practice exercise for this week, I highly recommend that you do so.)

```
breaks <- c(4.20, 4.30, 4.40, 4.50, 4.60, 4.70, 4.80, 4.90, 5.00,
            5.10, 5.20, 5.30, 5.40, 5.50, 5.60)
# breaks <- c(4.25, 4.35, 4.45, 4.55, 4.65, 4.75, 4.85, 4.95, 5.05,
#             5.15, 5.25, 5.35, 5.45, 5.55, 5.65)
labels <- c(4.25, 4.35, 4.45, 4.55, 4.65, 4.75, 4.85, 4.95, 5.05,
            5.15, 5.25, 5.35, 5.45, 5.55)

public <- public %>%
  mutate(STATE_NAME = if_else(STATE == 1, "NJ", "PA"))
```

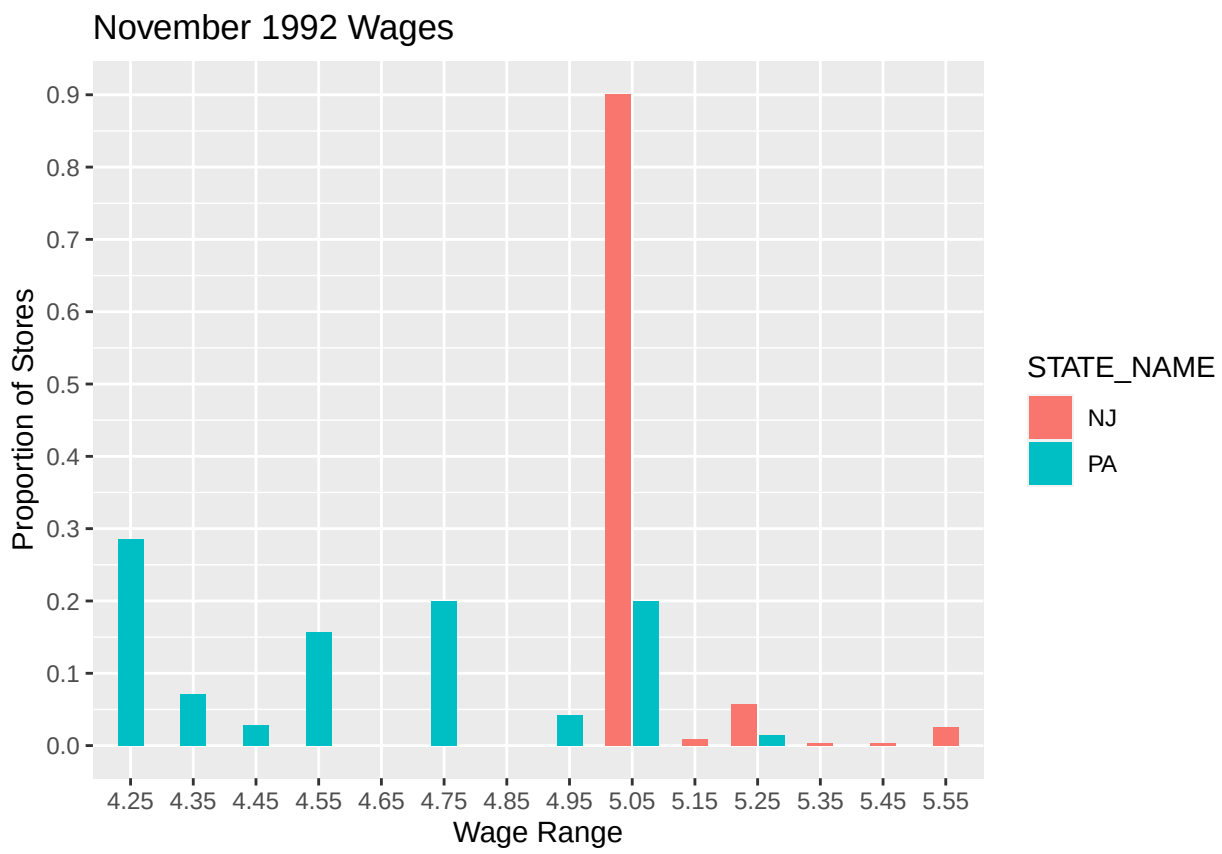
```
public$WAGE_ST_binned <- cut(public$WAGE_ST, breaks, labels, right=FALSE, na.rm=TRUE)
public$WAGE_ST2_binned <- cut(public$WAGE_ST2, breaks, labels, right=FALSE, na.rm=TRUE)
# table(public$WAGE_ST_binned)
# table(public$WAGE_ST2_binned)

# One thing I had trouble with is trying to get the bar graphs to be the same
# with even when there is only one in the interval. I tried using geom_col(),
# which supports positiondodge2() that gives additional customization of
# preserve="single" that what do what we would want, but when I use that,
# I get an "Error in FUN(X[[i]], ...) : object 'prop' not found"
# that I can't resolve so any feedback on that would be appreciated!
barplot1 <- ggplot(data = public %>% drop_na(WAGE_ST_binned),
                   aes(x = WAGE_ST_binned, y = ..prop..,
                       group = STATE_NAME, fill=STATE_NAME)) +
                   scale_y_continuous(breaks = seq(0, 1, by = .10)) +
                   scale_x_discrete(drop=FALSE) +
                   ggtitle("February 1992 Wages") +
                   xlab("Wage Range") + ylab("Proportion of Stores") +
                   geom_bar(position = position_dodge2(width = 0.9, preserve = "single"))
barplot1
```




```
barplot2 <- ggplot(data = public %>% drop_na(WAGE_ST2_binned),
                   aes(x = WAGE_ST2_binned, y = ..prop..,
                       group = STATE_NAME, fill=STATE_NAME)) +
                   scale_y_continuous(breaks = seq(0, 1, by = .10)) +
```

```
                  scale_x_discrete(drop=FALSE) +
                  ggtitle("November 1992 Wages") +
                  xlab("Wage Range") + ylab("Proportion of Stores") +
                  geom_bar(position = position_dodge2(width = 0.9, preserve = "single"))
barplot2
```




```
# bin the wage data into intervals of 5 cents, so that they can be combined
# PA_public$PA_WAGE_ST_binned <- cut(PA_public$PA_WAGE_ST,
#                                    breaks, labels, right=FALSE, na.rm=TRUE)
# PA_public$PA_WAGE_ST2_binned <- cut(PA_public$PA_WAGE_ST2,
#                                     breaks, labels, right=FALSE, na.rm=TRUE)
# NJ_public$NJ_WAGE_ST_binned <- cut(NJ_public$NJ_WAGE_ST,
#                                    breaks, labels, right=FALSE, na.rm=TRUE)
# NJ_public$NJ_WAGE_ST2_binned <- cut(NJ_public$NJ_WAGE_ST2,
#                                     breaks, labels, right=FALSE, na.rm=TRUE)
#
# combined the PA and NJ rows into wave 1 and wave 2 proportion tables
# wave1_prop <- prop.table(table(PA_public$PA_WAGE_ST_binned))
# wave1_prop <- as.data.frame.matrix(rbind(prop.table(table(PA_public$PA_WAGE_ST_binned)),
#                    prop.table(table(NJ_public$NJ_WAGE_ST_binned))))
# wave1_prop.long <- gather(wave1_prop, variable, value)
#
# wave2_prop <- as.data.frame.matrix(rbind(prop.table(table(PA_public$PA_WAGE_ST2_binned)),
#                     prop.table(table(NJ_public$NJ_WAGE_ST2_binned))))
#
```

```
# basic_barplot <- ggplot(data = wave1_prop,
#                         aes(x = labels, group = )) +
#                         geom_bar()
# basic_barplot
#
# basic_barplot <- ggplot(data = public,
#                       aes(x = WAGE_ST_binned, group=public$STATE, fill=public$STATE)) +
#                     geom_bar(stat = "identity", position = "dodge")
# basic_barplot
```