HW: Week 4

36-350 – Statistical Computing

Week 4 – Spring 2021

Name: Cherie Hua

Andrew ID: cxhua

You must submit **your own** HW as a PDF file on Gradescope.

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Question 1

(20 points)

You are given the following matrix:

```r
set.seed(505)
mat = matrix(rnorm(900),30,30)
mat[sample(30,1),sample(30,1)] = NA
```

Compute the standard deviation for each row, using `apply()` and your own on-the-fly function, i.e., a function that is defined *within* the argument list being passed to `apply()`. **Do not use the function sd()!** Realize that since there is a missing value within the matrix, you need to define your function so as to only take into account the non-missing data in each row. If your vector of standard deviations has an `NA` in it, then your function isn't quite working yet.

```r
apply(mat, MARGIN = 1, FUN = function(x) {sqrt(sum((x - mean(x, na.rm = TRUE))^2, na.rm = TRUE) / (leng
```

```
##  [1] 1.2235111 0.9996540 0.8324186 0.7797836 0.9546933 1.1166745 1.0264495
##  [8] 0.7135952 1.0357715 0.9023740 1.2146342 0.9665977 1.1364236 0.7335094
## [15] 0.8758855 1.0529671 1.0303302 0.8857679 1.1004938 0.9636788 0.9981597
## [22] 1.1224219 1.2828417 0.9777383 0.9223948 0.8506261 0.8840344 0.6538431
## [29] 0.8304627 1.0001846
```

Below we read in the data on the political economy of strikes.

```
strikes.df = read.csv("http://www.stat.cmu.edu/~mfarag/350/strikes.csv")
```

Question 2

(20 points)

Using `split()` and `sapply()`, compute the average unemployment rate, inflation rates, and strike volume for each year represented in the `strikes.df` data frame. The output should be a matrix of dimension 3 × 35. (You need not display the matrix contents…just capture the output from `sapply()` and pass that output to `dim()`.) Provide appropriate row names (see `rownames()` to your output matrix. Display the columns for 1962, 1972, and 1982. (This can be done in one line as opposed to three.)

```
split_by_year = split(strikes.df, strikes.df$year)
avgs = sapply(split_by_year, function(df) {
  c("Unemployment Average"=mean(df$unemployment),
    "Inflation Average"=mean(df$inflation),
    "Strike Average"=mean(df$strike.volume))})
dim(avgs)
```

```
## [1]  3 35
```

Question 3

(20 points)

Utilize piping and `group_by()`, etc., to compute the average unemployment rate for each country, and display that average for only those countries with the maximum and minimum averages. To be clear: your output should only show average unemployment for Ireland and Switzerland, and nothing else. (Hint: remember `slice()`, a less-often-used `dplyr` function.) Hint: arrange your output in order of descending average unemployment, then note that `n()` applied as an argument to the right function will return the last row.

```
avg.unemp = suppressMessages(strikes.df %>% select(country, unemployment) %>% group_by(country) %>% summ
avg.unemp
```

```
## # A tibble: 2 x 2
##   country     mean_unemployment
##   <chr>                   <dbl>
## 1 Ireland                 7.77
## 2 Switzerland             0.329
```