# Course2 R Programming - Assignment 1

Haoyi Wei

2022-12-21

## Contents

## Instruction

For this first programming assignment you will write three functions that are meant to interact with dataset that accompanies this assignment. The dataset is contained in a zip file specdata.zip that you can download from the Coursera web site.

## Reviwe Criteria

Although this is a programming assignment, you will be assessed using a separate quiz.

## Part 1

Write a function named 'pollutantmean' that calculates the mean of a pollutant (sulfate or nitrate) across a specified list of monitors. The function 'pollutantmean' takes three arguments: 'directory', 'pollutant', and 'id'. Given a vector monitor ID numbers, 'pollutantmean' reads that monitors' particulate matter data from the directory specified in the 'directory' argument and returns the mean of the pollutant across all of the monitors, ignoring any missing values coded as NA. A prototype of the function is as follows

```
pollutantmean <- function(directory, pollutant, id = 1:332) {
        ## 'directory' is a character vector of length 1 indicating
        ## the location of the CSV files

        ## 'pollutant' is a character vector of length 1 indicating
        ## the name of the pollutant for which we will calculate the
        ## mean; either "sulfate" or "nitrate".

        ## 'id' is an integer vector indicating the monitor ID numbers
        ## to be used

        ## Return the mean of the pollutant across all monitors list
        ## in the 'id' vector (ignoring NA values)
        ## NOTE: Do not round the result!
}
```

Figure 1: Example

Answer:

```
pollutantmean <- function(directory,pollutant, id=1:332) {

  full <-NA

  for (x in id){
    path <- paste("./","data/",directory,"/",sprintf("%.3d",x),".csv",sep="")
    rcsv <- read.csv(path)
    full<-rbind(full,rcsv)
  }

  x<-mean(full[,pollutant],na.rm=T)
  x
}
```

# Part 2

Write a function that reads a directory full of files and reports the number of completely observed cases in each data file. The function should return a data frame where the first column is the name of the file and the second column is the number of complete cases. A prototype of this function follows

```
complete <- function(directory, id = 1:332) {
        ## 'directory' is a character vector of length 1 indicating
        ## the location of the CSV files

        ## 'id' is an integer vector indicating the monitor ID numbers
        ## to be used

        ## Return a data frame of the form:
        ## id nobs
        ## 1  117
        ## 2  1041
        ## ...
        ## where 'id' is the monitor ID number and 'nobs' is the
        ## number of complete cases
}
```

Figure 2: Example

Answer:

```
library(dplyr)
```

```
##
## 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
```

```
complete <- function(directory, id=1:332){

  full <-NA

  for (x in id){
    path <- paste("./","data/",directory,"/",sprintf("%.3d",x),".csv",sep="")
    rcsv <- read.csv(path)
    full<-rbind(full,rcsv)
  }

  results <-full[complete.cases(full),] %>% # remove rows with missing values in any column of data fra
    mutate(a=1) %>%
    group_by(ID) %>%
    mutate(nobs=sum(a)) %>%
```

```
    filter(row_number()==1) %>%
    ungroup() %>%
    select(ID,nobs) %>%
    relocate(ID,nobs)
  results
}
```

# Part 3

Write a function that takes a directory of data files and a threshold for complete cases and calculates the correlation between sulfate and nitrate for monitor locations where the number of completely observed cases (on all variables) is greater than the threshold. The function should return a vector of correlations for the monitors that meet the threshold requirement. If no monitors meet the threshold requirement, then the function should return a numeric vector of length 0. A prototype of this function follows

```
corr <- function(directory, threshold = 0) {
        ## 'directory' is a character vector of length 1 indicating
        ## the location of the CSV files

        ## 'threshold' is a numeric vector of length 1 indicating the
        ## number of completely observed observations (on all
        ## variables) required to compute the correlation between
        ## nitrate and sulfate; the default is 0

        ## Return a numeric vector of correlations
        ## NOTE: Do not round the result!
}
```

Figure 3: Example

Answer:

```
source("./DS2-HW1-complete.R")
corr <- function(directory, threshold=0){

  for (x in 1:332){
    path <- paste("./","data/",directory,"/",sprintf("%.3d",x),".csv",sep="")
    rcsv <- read.csv(path)
    full<-rbind(full,rcsv)
  }

  nobs <- complete(directory)

  full_nobs <- left_join(full,nobs, by="ID")

  results <-    full_nobs %>%
    group_by(ID) %>%
    mutate(correlation=cor(sulfate,nitrate,use="na.or.complete")) %>%
    filter(row_number()==1) %>%
    ungroup() %>%
    select(ID,nobs,correlation) %>%
    filter(nobs>threshold)
```

```
  vec <-results$correlation
}
```