Assignment One

DanielH

July 5, 2017

Part One

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

```
# R version
print(R.version.string)
```

```
## [1] "R version 3.4.0 (2017-04-21)"
```

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

  pollutant <- enquo(pollutant)
  list.files(directory, full.names = TRUE) %>%
    map_df(read_csv,
           cols(Date = col_date(),
                sulfate = col_double(),
                nitrate = col_double(),
                ID = col_integer()),
           col_names = T) %>%
    select(ID, !!pollutant) %>%
    filter(ID %in% id) %>%
    select(-ID) %>%
    unlist() %>%
    mean(na.rm = TRUE)

}


# Test function
pollutantmean("specdata", "sulfate", 1:10)
```

```
## [1] 4.064128
```

```
pollutantmean("specdata", "nitrate", 70:72)
```

```
## [1] 1.706047
```

```
pollutantmean("specdata", "nitrate", 23)
```

```
## [1] 1.280833
```

Part Two

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

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

  list.files(directory, full.names = TRUE) %>%
    map_df(read_csv,
           cols(Date = col_date(),
                sulfate = col_double(),
                nitrate = col_double(),
                ID = col_integer()),
           col_names = T) %>%
    filter(!is.na(sulfate) & !is.na(nitrate)) %>%
    mutate(ID = factor(ID)) %>%
    select(-Date) %>%
    group_by(ID) %>%
    count() %>%
    rename(nobs = n) %>%
    filter(ID %in% id)

}


# Test function
complete("specdata", c(2, 4, 8, 10, 12))
```

```
## # A tibble: 5 x 2
## # Groups:   ID [5]
##       ID  nobs
##   <fctr> <int>
## 1      2  1041
## 2      4   474
## 3      8   192
## 4     10   148
## 5     12    96
```

```
complete("specdata", 30:25)
```

```
## # A tibble: 6 x 2
## # Groups:   ID [6]
##       ID  nobs
##   <fctr> <int>
## 1     25   463
## 2     26   586
## 3     27   338
## 4     28   475
## 5     29   711
## 6     30   932
```

Part three

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.

```
# Define function
corr <- function(directory, threshold = 0) {

  list.files(directory, full.names = TRUE) %>%
    map_df(read_csv,
           cols(Date = col_date(),
                sulfate = col_double(),
                nitrate = col_double(),
                ID = col_integer()),
           col_names = T) %>%
    filter(!is.na(sulfate) & !is.na(nitrate)) %>%
    select(-Date) %>%
    group_by(ID) %>%
    mutate(crl = cor(sulfate, nitrate)) %>%
    summarise(nbo = n(), crl = mean(crl)) %>%
    filter(nbo >= threshold) %>%
    select(crl, -nbo) %>% unlist()
}


# Test function
cr <- corr("specdata", 150)
head(cr)
```

```
##        crl1        crl2        crl3        crl4        crl5        crl6
## -0.01895754 -0.14051254 -0.04389737 -0.06815956 -0.12350667 -0.07588814
```

```
summary(cr)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.21057 -0.05147  0.09333  0.12401  0.26836  0.76313
```

```
cr <- corr("specdata", 400)
head(cr)
```

```
##        crl1        crl2        crl3        crl4        crl5        crl6
## -0.01895754 -0.04389737 -0.06815956 -0.07588814  0.76312884 -0.15782860
```

```
cr <- corr("specdata", 5000)
summary(cr)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##
```

```
cr <- corr("specdata")
summary(cr)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -1.00000 -0.05282  0.10718  0.13684  0.27831  1.00000
```