Assignment 4: Data Wrangling

Tasha Griffiths

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Wrangling

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Fay_A04_DataWrangling.Rmd") prior to submission.

The completed exercise is due on Monday, Feb 7 @ 7:00pm.

Set up your session

1. Check your working directory, load the `tidyverse` and `lubridate` packages, and upload all four raw data files associated with the EPA Air dataset. See the README file for the EPA air datasets for more information (especially if you have not worked with air quality data previously).

2. Explore the dimensions, column names, and structure of the datasets.

```
#1
getwd()
```

```
## [1] "C:/Users/Tasha Griffiths/Documents/Duke Year 1/Spring 22 Classes/Environmental Data Analytics/G
```

```
#setwd("C:/Users/Tasha Griffiths/Documents/Duke Year 1/Spring 22 Classes/Environmental Data Analytics/G

#install.packages(tidyverse)
library(tidyverse)
#install.packages(lubridate)
library(lubridate)
#install.packages(dplyr)
library(dplyr)

#load all 4 raw data files
EPAair_O3_NC2018 <- read.csv("./Data/Raw/EPAair_O3_NC2018_raw.csv",
                             stringsAsFactors = TRUE)
EPAair_O3_NC2019 <- read.csv("./Data/Raw/EPAair_O3_NC2019_raw.csv",
```

```
                              stringsAsFactors = TRUE)
EPAair_PM25_NC2018 <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv",
                               stringsAsFactors = TRUE)
EPAair_PM25_NC2019 <- read.csv("./Data/Raw/EPAair_PM25_NC2019_raw.csv",
                               stringsAsFactors = TRUE)

#2
#basic exploration, repeat for all datasets
colnames(EPAair_03_NC2018)
```

```
##  [1] "Date"
##  [2] "Source"
##  [3] "Site.ID"
##  [4] "POC"
##  [5] "Daily.Max.8.hour.Ozone.Concentration"
##  [6] "UNITS"
##  [7] "DAILY_AQI_VALUE"
##  [8] "Site.Name"
##  [9] "DAILY_OBS_COUNT"
## [10] "PERCENT_COMPLETE"
## [11] "AQS_PARAMETER_CODE"
## [12] "AQS_PARAMETER_DESC"
## [13] "CBSA_CODE"
## [14] "CBSA_NAME"
## [15] "STATE_CODE"
## [16] "STATE"
## [17] "COUNTY_CODE"
## [18] "COUNTY"
## [19] "SITE_LATITUDE"
## [20] "SITE_LONGITUDE"
```

```
head(EPAair_03_NC2018)
```

```
##         Date Source   Site.ID POC Daily.Max.8.hour.Ozone.Concentration UNITS
## 1 03/01/2018    AQS 370030005   1                                 0.043   ppm
## 2 03/02/2018    AQS 370030005   1                                 0.046   ppm
## 3 03/03/2018    AQS 370030005   1                                 0.047   ppm
## 4 03/04/2018    AQS 370030005   1                                 0.049   ppm
## 5 03/05/2018    AQS 370030005   1                                 0.047   ppm
## 6 03/06/2018    AQS 370030005   1                                 0.030   ppm
##   DAILY_AQI_VALUE            Site.Name DAILY_OBS_COUNT PERCENT_COMPLETE
## 1              40 Taylorsville Liledoun              17              100
## 2              43 Taylorsville Liledoun              17              100
## 3              44 Taylorsville Liledoun              17              100
## 4              45 Taylorsville Liledoun              17              100
## 5              44 Taylorsville Liledoun              17              100
## 6              28 Taylorsville Liledoun              17              100
##   AQS_PARAMETER_CODE AQS_PARAMETER_DESC CBSA_CODE                    CBSA_NAME
## 1              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
## 2              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
## 3              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
## 4              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
## 5              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
```

```
## 6              44201              Ozone     25860 Hickory-Lenoir-Morganton, NC
##   STATE_CODE          STATE COUNTY_CODE    COUNTY SITE_LATITUDE SITE_LONGITUDE
## 1         37 North Carolina           3 Alexander       35.9138        -81.191
## 2         37 North Carolina           3 Alexander       35.9138        -81.191
## 3         37 North Carolina           3 Alexander       35.9138        -81.191
## 4         37 North Carolina           3 Alexander       35.9138        -81.191
## 5         37 North Carolina           3 Alexander       35.9138        -81.191
## 6         37 North Carolina           3 Alexander       35.9138        -81.191
```

```
summary(EPAair_03_NC2018)
```

```
##          Date      Source        Site.ID              POC
##  04/01/2018:  40   AQS:9737   Min.   :370030005   Min.   :1
##  04/12/2018:  40              1st Qu.:370650099   1st Qu.:1
##  04/13/2018:  40              Median :371010002   Median :1
##  04/14/2018:  40              Mean   :370969118   Mean   :1
##  04/15/2018:  40              3rd Qu.:371290002   3rd Qu.:1
##  04/18/2018:  40              Max.   :371990004   Max.   :1
##  (Other)   :9497
##  Daily.Max.8.hour.Ozone.Concentration UNITS      DAILY_AQI_VALUE
##  Min.   :0.00200                      ppm:9737   Min.   :  2.00
##  1st Qu.:0.03400                                 1st Qu.: 31.00
##  Median :0.04200                                 Median : 39.00
##  Mean   :0.04194                                 Mean   : 40.22
##  3rd Qu.:0.04900                                 3rd Qu.: 45.00
##  Max.   :0.07700                                 Max.   :122.00
##
##                 Site.Name    DAILY_OBS_COUNT PERCENT_COMPLETE
##  Coweeta              : 355   Min.   :12.00   Min.   : 71.00
##  Garinger High School: 354   1st Qu.:17.00   1st Qu.:100.00
##  Millbrook School     : 352   Median :17.00   Median :100.00
##  Candor               : 335   Mean   :16.94   Mean   : 99.65
##  Rockwell             : 335   3rd Qu.:17.00   3rd Qu.:100.00
##  Cranberry            : 323   Max.   :17.00   Max.   :100.00
##  (Other)              :7683
##  AQS_PARAMETER_CODE AQS_PARAMETER_DESC   CBSA_CODE
##  Min.   :44201      Ozone:9737         Min.   :11700
##  1st Qu.:44201                         1st Qu.:16740
##  Median :44201                         Median :24660
##  Mean   :44201                         Mean   :27247
##  3rd Qu.:44201                         3rd Qu.:39580
##  Max.   :44201                         Max.   :49180
##                                        NA's   :2609
##                               CBSA_NAME      STATE_CODE            STATE
##                                    :2609   Min.   :37   North Carolina:9737
##  Charlotte-Concord-Gastonia, NC-SC:1338   1st Qu.:37
##  Asheville, NC                    : 927   Median :37
##  Winston-Salem, NC                : 725   Mean   :37
##  Raleigh, NC                      : 585   3rd Qu.:37
##  Hickory-Lenoir-Morganton, NC     : 477   Max.   :37
##  (Other)                          :3076
##   COUNTY_CODE             COUNTY     SITE_LATITUDE   SITE_LONGITUDE
##  Min.   :  3.00   Forsyth    : 725   Min.   :34.36   Min.   :-83.80
##  1st Qu.: 65.00   Haywood    : 683   1st Qu.:35.26   1st Qu.:-82.05
```

```
##  Median :101.00   Mecklenburg: 592   Median :35.55   Median :-80.34
##  Mean   : 96.78   Avery      : 558   Mean   :35.62   Mean   :-80.42
##  3rd Qu.:129.00   Swain      : 483   3rd Qu.:36.03   3rd Qu.:-78.90
##  Max.   :199.00   Cumberland : 444   Max.   :36.31   Max.   :-76.62
##                   (Other)    :6252
```

```
str(EPAair_03_NC2018)
```

```
## 'data.frame':    9737 obs. of  20 variables:
##  $ Date                                : Factor w/ 364 levels "01/01/2018","01/02/2018",..: 60 61 62
##  $ Source                              : Factor w/ 1 level "AQS": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Site.ID                             : int  370030005 370030005 370030005 370030005 370030005 3700
##  $ POC                                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Daily.Max.8.hour.Ozone.Concentration: num  0.043 0.046 0.047 0.049 0.047 0.03 0.036 0.044 0.049 0
##  $ UNITS                               : Factor w/ 1 level "ppm": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DAILY_AQI_VALUE                     : int  40 43 44 45 44 28 33 41 45 40 ...
##  $ Site.Name                           : Factor w/ 40 levels "","Beaufort",..: 35 35 35 35 35 35 35 3
##  $ DAILY_OBS_COUNT                     : int  17 17 17 17 17 17 17 17 17 17 ...
##  $ PERCENT_COMPLETE                    : num  100 100 100 100 100 100 100 100 100 100 ...
##  $ AQS_PARAMETER_CODE                  : int  44201 44201 44201 44201 44201 44201 44201 44201 44201 4
##  $ AQS_PARAMETER_DESC                  : Factor w/ 1 level "Ozone": 1 1 1 1 1 1 1 1 1 1 ...
##  $ CBSA_CODE                           : int  25860 25860 25860 25860 25860 25860 25860 25860 25860 2
##  $ CBSA_NAME                           : Factor w/ 17 levels "","Asheville, NC",..: 9 9 9 9 9 9 9 9 9
##  $ STATE_CODE                          : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ STATE                               : Factor w/ 1 level "North Carolina": 1 1 1 1 1 1 1 1 1 1 ...
##  $ COUNTY_CODE                         : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ COUNTY                              : Factor w/ 32 levels "Alexander","Avery",..: 1 1 1 1 1 1 1 1
##  $ SITE_LATITUDE                       : num  35.9 35.9 35.9 35.9 35.9 ...
##  $ SITE_LONGITUDE                      : num  -81.2 -81.2 -81.2 -81.2 -81.2 ...
```

```
dim(EPAair_03_NC2018)
```

```
## [1] 9737   20
```

Wrangle individual datasets to create processed files.

3. Change date to a date object
4. Select the following columns: Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_LATITUDE, SITE_LONGITUDE
5. For the PM2.5 datasets, fill all cells in AQS_PARAMETER_DESC with "PM2.5" (all cells in this column should be identical).
6. Save all four processed datasets in the Processed folder. Use the same file names as the raw files but replace "raw" with "processed".

```
#3 check data class
class(EPAair_03_NC2018$Date)
```

```
## [1] "factor"
```

```
class(EPAair_03_NC2019$Date)
```

```
## [1] "factor"
```

```
class(EPAair_PM25_NC2018$Date)
```

```
## [1] "factor"
```

```
class(EPAair_PM25_NC2019$Date)
```

```
## [1] "factor"
```

```
#Format all data file date columns as date
EPAair_03_NC2018$Date <- as.Date(EPAair_03_NC2018$Date, format = "%m/%d/%Y")
EPAair_03_NC2019$Date <- as.Date(EPAair_03_NC2019$Date, format = "%m/%d/%Y")
EPAair_PM25_NC2018$Date <- as.Date(EPAair_PM25_NC2018$Date, format = "%m/%d/%Y")
EPAair_PM25_NC2019$Date <- as.Date(EPAair_PM25_NC2019$Date, format = "%m/%d/%Y")


#4 select a subset of columns
EPAair_03_NC2018 <- select(EPAair_03_NC2018, Date, DAILY_AQI_VALUE, Site.Name,
                           AQS_PARAMETER_DESC, COUNTY, SITE_LATITUDE,
                           SITE_LONGITUDE)

EPAair_03_NC2019 <- select(EPAair_03_NC2019, Date, DAILY_AQI_VALUE, Site.Name,
                           AQS_PARAMETER_DESC, COUNTY, SITE_LATITUDE,
                           SITE_LONGITUDE)

EPAair_PM25_NC2018 <- select(EPAair_PM25_NC2018, Date, DAILY_AQI_VALUE,
                             Site.Name, AQS_PARAMETER_DESC, COUNTY,
                             SITE_LATITUDE, SITE_LONGITUDE)

EPAair_PM25_NC2019 <- select(EPAair_PM25_NC2019, Date, DAILY_AQI_VALUE,
                             Site.Name, AQS_PARAMETER_DESC, COUNTY,
                             SITE_LATITUDE, SITE_LONGITUDE)

#5 fill cells in MP25 data sets
EPAair_PM25_NC2018$AQS_PARAMETER_DESC <- 'PM2.5'

EPAair_PM25_NC2019$AQS_PARAMETER_DESC <- 'PM2.5'


#6 saved as data files as processed
write.csv(EPAair_03_NC2018, row.names = FALSE,
          file = "./Data/Processed/EPAair_03_NC2018_processed.csv")

write.csv(EPAair_03_NC2019, row.names = FALSE,
          file = "./Data/Processed/EPAair_03_NC2019_processed.csv")

write.csv(EPAair_PM25_NC2018, row.names = FALSE,
          file = "./Data/Processed/EPAair_PM25_NC2018_processed.csv")
```

```
write.csv(EPAair_PM25_NC2019, row.names = FALSE,
          file = "./Data/Processed/EPAair_PM25_NC2019_processed.csv")
```

Combine datasets

7. Combine the four datasets with `rbind`. Make sure your column names are identical prior to running this code.
8. Wrangle your new dataset with a pipe function (%>%) so that it fills the following conditions:

- Filter records to include just the sites that the four data frames have in common: "Linville Falls", "Durham Armory", "Leggett", "Hattie Avenue", "Clemmons Middle", "Mendenhall School", "Frying Pan Mountain", "West Johnston Co.", "Garinger High School", "Castle Hayne", "Pitt Agri. Center", "Bryson City", "Millbrook School". (The `intersect` function can figure out common factor levels if we didn't give you this list...)
- Some sites have multiple measurements per day. Use the split-apply-combine strategy to generate daily means: group by date, site, aqs parameter, and county. Take the mean of the AQI value, latitude, and longitude.
- Add columns for "Month" and "Year" by parsing your "Date" column (hint: `lubridate` package)
- Hint: the dimensions of this dataset should be 14,752 x 9.

9. Spread your datasets such that AQI values for ozone and PM2.5 are in separate columns. Each location on a specific date should now occupy only one row.
10. Call up the dimensions of your new tidy dataset.
11. Save your processed dataset with the following file name: "EPAair_O3_PM25_NC2122_Processed.csv"

```
#7 combine all dataframes
EPAair_O3_2018and2019 <- full_join(EPAair_O3_NC2018, EPAair_O3_NC2019)
```

```
## Joining, by = c("Date", "DAILY_AQI_VALUE", "Site.Name", "AQS_PARAMETER_DESC", "COUNTY", "SITE_LATITU
```

```
EPAair_PM25_2018and2019 <- full_join(EPAair_PM25_NC2018, EPAair_PM25_NC2019)
```

```
## Joining, by = c("Date", "DAILY_AQI_VALUE", "Site.Name", "AQS_PARAMETER_DESC", "COUNTY", "SITE_LATITU
```

```
EPAair_O3_PM25_NC2122 <- full_join(EPAair_O3_2018and2019, EPAair_PM25_2018and2019)
```

```
## Joining, by = c("Date", "DAILY_AQI_VALUE", "Site.Name", "AQS_PARAMETER_DESC", "COUNTY", "SITE_LATITU
```

```
#8 Filter dataset with pipe
EPAair_O3_PM25_NC2122_filtered <- EPAair_O3_PM25_NC2122 %>%
  filter(Site.Name %in% c("Linville Falls", "Durham Armory", "Leggett",
        "Hattie Avenue", "Clemmons Middle", "Mendenhall School",
        "Frying Pan Mountain", "West Johnston Co.", "Garinger High School",
        "Castle Hayne", "Pitt Agri. Center", "Bryson City",
        "Millbrook School")) %>%
  group_by(Date, Site.Name, AQS_PARAMETER_DESC, COUNTY) %>%
  summarize(AQI_mean = mean(EPAair_O3_PM25_NC2122$DAILY_AQI_VALUE),
            latitude_mean = mean(EPAair_O3_PM25_NC2122$SITE_LATITUDE),
            longitude_mean = mean(EPAair_O3_PM25_NC2122$SITE_LONGITUDE)) %>%
  mutate(Month = month(Date)) %>%
  mutate(Year = year(Date))
```

```
## `summarise()` has grouped output by 'Date', 'Site.Name', 'AQS_PARAMETER_DESC'. You can override using
```

```
dim(EPAair_O3_PM25_NC2122_filtered)
```

```
## [1] 14752     9
```

```
#9 spread ozone and PM2.5
EPAair_O3_PM25_NC2122_spread <- EPAair_O3_PM25_NC2122_filtered %>%
  pivot_wider(names_from = AQS_PARAMETER_DESC, values_from = AQI_mean)

#10
dim(EPAair_O3_PM25_NC2122_spread)
```

```
## [1] 8976    9
```

```
#11
write.csv(EPAair_O3_PM25_NC2122_spread, row.names = FALSE,
          file ="./Data/Processed/EPAair_O3_PM25_NC2122_Processed.csv")
```

Generate summary tables

12a. Use the split-apply-combine strategy to generate a summary data frame from your results from Step 9 above. Data should be grouped by site, month, and year. Generate the mean AQI values for ozone and PM2.5 for each group.

12b. BONUS: Add a piped statement to 12a that removes rows where both mean ozone and mean PM2.5 have missing values.

13. Call up the dimensions of the summary dataset.

```
#12(a,b)
EPAair_O3_PM25_NC2122_summary <- EPAair_O3_PM25_NC2122_spread %>%
  group_by(Site.Name, Month, Year) %>%
  summarise(AQI_mean_ozone = mean(Ozone),
            AQI_mean_PM2.5 = mean(PM2.5)) %>%
  filter(!is.na(AQI_mean_ozone & AQI_mean_PM2.5))
```

```
## `summarise()` has grouped output by 'Site.Name', 'Month'. You can override using the `.groups` argume
```

```
#filter(drop_na(AQI_mean_ozone & AQI_mean_PM2.5)) drop_na only works for one
#column at a time not able to use two columns at once.

#13
dim(EPAair_O3_PM25_NC2122_summary)
```

```
## [1] 101   5
```

14. Why did we use the function `drop_na` rather than `na.omit`?

    Answer: drop_na will remove NA's within the column that you specify, but does not use a logic to check between two columns. To remove rows with NA's as true in both columns, we need to use a filter at the is.na function. The na.omit works by removing rows that have any NA's within them, so if an NA exists in ozone or PM2.5 it will be dropped. However, it doesn't check for NA's within both columns.