Assignment 4: Data Wrangling

Megan McClaugherty

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A03_DataExploration.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

The completed exercise is due on Friday, Oct7th @ 5:00pm.

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, being sure to set string columns to be read in a factors. 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] "/home/guest/EDA-Fall2022/Assignments"
```

```
library(tidyverse)
library(lubridate)
EPAair03_2018 <- read.csv("/home/guest/EDA-Fall2022/Data/Raw/EPAair_03_NC2018_raw.csv",
    stringsAsFactors = TRUE)
EPAair03_2019 <- read.csv("/home/guest/EDA-Fall2022/Data/Raw/EPAair_03_NC2019_raw.csv",
    stringsAsFactors = TRUE)
EPAairPM25_2018 <- read.csv("/home/guest/EDA-Fall2022/Data/Raw/EPAair_PM25_NC2018_raw.csv",
    stringsAsFactors = TRUE)
EPAairPM25_2019 <- read.csv("/home/guest/EDA-Fall2022/Data/Raw/EPAair_PM25_NC2019_raw.csv",
    stringsAsFactors = TRUE)

# 2 Dimensions of all four datasets
dim(EPAair03_2018)
```

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

```
dim(EPAair03_2019)
```

```
## [1] 10592    20
```

```
dim(EPAairPM25_2018)
```

```
## [1] 8983   20
```

```
dim(EPAairPM25_2019)
```

```
## [1] 8581   20
```

```
# Column names of all 4 datasets
colnames(EPAairO3_2018)
```

```
##  [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"
```

```
colnames(EPAairO3_2019)
```

```
##  [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"
```

```
colnames(EPAairPM25_2018)
```

```
##  [1] "Date"                           "Source"
##  [3] "Site.ID"                        "POC"
##  [5] "Daily.Mean.PM2.5.Concentration" "UNITS"
##  [7] "DAILY_AQI_VALUE"                "Site.Name"
##  [9] "DAILY_OBS_COUNT"                "PERCENT_COMPLETE"
## [11] "AQS_PARAMETER_CODE"             "AQS_PARAMETER_DESC"
## [13] "CBSA_CODE"                      "CBSA_NAME"
## [15] "STATE_CODE"                     "STATE"
## [17] "COUNTY_CODE"                    "COUNTY"
## [19] "SITE_LATITUDE"                  "SITE_LONGITUDE"
```

```
colnames(EPAairPM25_2019)
```

```
##  [1] "Date"                           "Source"
##  [3] "Site.ID"                        "POC"
##  [5] "Daily.Mean.PM2.5.Concentration" "UNITS"
##  [7] "DAILY_AQI_VALUE"                "Site.Name"
##  [9] "DAILY_OBS_COUNT"                "PERCENT_COMPLETE"
## [11] "AQS_PARAMETER_CODE"             "AQS_PARAMETER_DESC"
## [13] "CBSA_CODE"                      "CBSA_NAME"
## [15] "STATE_CODE"                     "STATE"
## [17] "COUNTY_CODE"                    "COUNTY"
## [19] "SITE_LATITUDE"                  "SITE_LONGITUDE"
```

```
# structure of each dataset
str(EPAairO3_2018)
```

```
## '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 1
##  $ 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 ...
```

```
str(EPAairO3_2019)
```

```
## 'data.frame':    10592 obs. of  20 variables:
##  $ Date                                : Factor w/ 365 levels "01/01/2019","01/02/2019",..: 1 2 3 4 5
##  $ Source                              : Factor w/ 2 levels "AirNow","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 ...
##  $ Daily.Max.8.hour.Ozone.Concentration: num  0.029 0.018 0.016 0.022 0.037 0.037 0.029 0.038 0.038 0
##  $ UNITS                             : Factor w/ 1 level "ppm": 1 1 1 1 1 1 1 1 1 ...
##  $ DAILY_AQI_VALUE                   : int  27 17 15 20 34 34 27 35 35 28 ...
##  $ Site.Name                         : Factor w/ 38 levels "","Beaufort",..: 33 33 33 33 33 33 33 3
##  $ DAILY_OBS_COUNT                   : int  24 24 24 24 24 24 24 24 24 24 ...
##  $ 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 ...
##  $ CBSA_CODE                         : int  25860 25860 25860 25860 25860 25860 25860 25860 25860 1
##  $ CBSA_NAME                         : Factor w/ 15 levels "","Asheville, NC",..: 8 8 8 8 8 8 8 8
##  $ STATE_CODE                        : int  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 ...
##  $ COUNTY_CODE                       : int  3 3 3 3 3 3 3 3 3 ...
##  $ COUNTY                            : Factor w/ 30 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 ...
```

```
str(EPAairPM25_2018)
```

```
## 'data.frame':    8983 obs. of  20 variables:
##  $ Date                          : Factor w/ 365 levels "01/01/2018","01/02/2018",..: 2 5 8 11 14 17
##  $ Source                        : Factor w/ 1 level "AQS": 1 1 1 1 1 1 1 1 1 ...
##  $ Site.ID                       : int  370110002 370110002 370110002 370110002 370110002 370110002 3
##  $ POC                           : int  1 1 1 1 1 1 1 1 1 ...
##  $ Daily.Mean.PM2.5.Concentration: num  2.9 3.7 5.3 0.8 2.5 4.5 1.8 2.5 4.2 1.7 ...
##  $ UNITS                         : Factor w/ 1 level "ug/m3 LC": 1 1 1 1 1 1 1 1 1 ...
##  $ DAILY_AQI_VALUE               : int  12 15 22 3 10 19 8 10 18 7 ...
##  $ Site.Name                     : Factor w/ 25 levels "","Blackstone",..: 15 15 15 15 15 15 15 15 15
##  $ DAILY_OBS_COUNT               : int  1 1 1 1 1 1 1 1 1 ...
##  $ PERCENT_COMPLETE              : num  100 100 100 100 100 100 100 100 100 100 ...
##  $ AQS_PARAMETER_CODE            : int  88502 88502 88502 88502 88502 88502 88502 88502 88502 88502
##  $ AQS_PARAMETER_DESC            : Factor w/ 2 levels "Acceptable PM2.5 AQI & Speciation Mass",..: 1
##  $ CBSA_CODE                     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ CBSA_NAME                     : Factor w/ 14 levels "","Asheville, NC",..: 1 1 1 1 1 1 1 1 1 ..
##  $ 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 ...
##  $ COUNTY_CODE                   : int  11 11 11 11 11 11 11 11 11 11 ...
##  $ COUNTY                        : Factor w/ 21 levels "Avery","Buncombe",..: 1 1 1 1 1 1 1 1 1 ..
##  $ SITE_LATITUDE                 : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE                : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

```
str(EPAairPM25_2019)
```

```
## 'data.frame':    8581 obs. of  20 variables:
##  $ Date                          : Factor w/ 365 levels "01/01/2019","01/02/2019",..: 3 6 9 12 15 18
##  $ Source                        : Factor w/ 2 levels "AirNow","AQS": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Site.ID                       : int  370110002 370110002 370110002 370110002 370110002 370110002 3
##  $ POC                           : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Daily.Mean.PM2.5.Concentration: num  1.6 1 1.3 6.3 2.6 1.2 1.5 1.5 3.7 1.6 ...
##  $ UNITS                         : Factor w/ 1 level "ug/m3 LC": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DAILY_AQI_VALUE               : int  7 4 5 26 11 5 6 6 15 7 ...
##  $ Site.Name                     : Factor w/ 25 levels "","Board Of Ed. Bldg.",..: 14 14 14 14 14 14
##  $ DAILY_OBS_COUNT               : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ PERCENT_COMPLETE              : num  100 100 100 100 100 100 100 100 100 100 ...
##  $ AQS_PARAMETER_CODE            : int  88502 88502 88502 88502 88502 88502 88502 88502 88502 88502
##  $ AQS_PARAMETER_DESC            : Factor w/ 2 levels "Acceptable PM2.5 AQI & Speciation Mass",..: 1
##  $ CBSA_CODE                     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ CBSA_NAME                     : Factor w/ 14 levels "","Asheville, NC",..: 1 1 1 1 1 1 1 1 1 ..
##  $ 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  11 11 11 11 11 11 11 11 11 11 ...
##  $ COUNTY                        : Factor w/ 21 levels "Avery","Buncombe",..: 1 1 1 1 1 1 1 1 1 ..
##  $ SITE_LATITUDE                 : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE                : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

Wrangle individual datasets to create processed files.

3. Change date to date
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 Changing dates
EPAair03_2018$Date <- as.Date(EPAair03_2018$Date,
    format = "%m/%d/%Y")
EPAair03_2019$Date <- as.Date(EPAair03_2019$Date,
    format = "%m/%d/%Y")
EPAairPM25_2018$Date <- as.Date(EPAairPM25_2018$Date,
    format = "%m/%d/%Y")
EPAairPM25_2019$Date <- as.Date(EPAairPM25_2019$Date,
    format = "%m/%d/%Y")

# 4 Selecting specific columns
EPAair03_2018Processed <- EPAair03_2018 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
        COUNTY, SITE_LATITUDE, SITE_LONGITUDE)
EPAair03_2019Processed <- EPAair03_2019 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
        COUNTY, SITE_LATITUDE, SITE_LONGITUDE)
EPAairPM25_2018Processed <- EPAairPM25_2018 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
        COUNTY, SITE_LATITUDE, SITE_LONGITUDE)
EPAairPM25_2019Processed <- EPAairPM25_2019 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
        COUNTY, SITE_LATITUDE, SITE_LONGITUDE)

# 5 Changing the AQS Parameter to a
# consistent value
EPAairPM25_2018Processed <- EPAairPM25_2018Processed %>%
    mutate(AQS_PARAMETER_DESC = "PM2.5")
EPAairPM25_2019Processed <- EPAairPM25_2019Processed %>%
    mutate(AQS_PARAMETER_DESC = "PM2.5")


# 6 Saving the 4 processed datasets
```

```
write.csv(EPAairO3_2018Processed, row.names = FALSE,
    file = "/home/guest/EDA-Fall2022/Data/Processed/EPAair_O3_NC2018_processed.csv")
write.csv(EPAairO3_2019Processed, row.names = FALSE,
    file = "/home/guest/EDA-Fall2022/Data/Processed/EPAair_O3_NC2019_processed.csv")
write.csv(EPAairPM25_2018Processed, row.names = FALSE,
    file = "/home/guest/EDA-Fall2022/Data/Processed/EPAairPM25_NC2018processed.csv")
write.csv(EPAairPM25_2019Processed, row.names = FALSE,
    file = "/home/guest/EDA-Fall2022/Data/Processed/EPAairPM25_NC2019processed.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:

- Include all 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 function `intersect` can figure out common factor levels)
- 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_NC1718_Processed.csv"

```
# 7 Combing the four datasets using rbind
EPAairO3_PM25_18_19 <- rbind(EPAairO3_2018Processed,
    EPAairO3_2019Processed, EPAairPM25_2018Processed,
    EPAairPM25_2019Processed)


# 8 filtering for the sites all datasets had
# in common, combining multiple measurements

EPAairO3_PM25_18_19 <- filter(EPAairO3_PM25_18_19,
    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) %>%
    summarise(meanDailyAQI = mean(DAILY_AQI_VALUE),
        meanLatitude = mean(SITE_LATITUDE), meanLongitude = mean(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 the `.groups` argument.
```

```
# verifying the dimensions
dim(EPAairO3_PM25_18_19)
```

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

```
# 9 spreading dataset to put Ozone and PM2.5
# into their own columns
EPAairO3_PM25_18_19 <- pivot_wider(EPAairO3_PM25_18_19,
    names_from = AQS_PARAMETER_DESC, values_from = meanDailyAQI)


# 10 checking dimensions of new dataset
dim(EPAairO3_PM25_18_19)
```

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

```
# 11 saving wrangled dataset
EPAair_O3_PM25_NC1819_Processed <- EPAairO3_PM25_18_19
write.csv(EPAair_O3_PM25_NC1819_Processed, row.names = FALSE,
    file = "/home/guest/EDA-Fall2022/Data/Processed/EPAair_O3_PM25_NC1819_Processed.csv")
EPAair_O3_PM25_NC1819_Processed
```

```
## # A tibble: 8,976 x 9
## # Groups:   Date, Site.Name [8,976]
##    Date       Site.Name           COUNTY meanL~1 meanL~2 Month  Year PM2.5 Ozone
##    <date>     <fct>               <fct>    <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 2018-01-01 Bryson City         Swain     35.4   -83.4     1  2018    35    NA
##  2 2018-01-01 Castle Hayne        New H~    34.4   -77.8     1  2018    13    NA
##  3 2018-01-01 Clemmons Middle     Forsy~    36.0   -80.3     1  2018    24    NA
##  4 2018-01-01 Durham Armory       Durham    36.0   -78.9     1  2018    31    NA
##  5 2018-01-01 Garinger High Scho~ Meckl~    35.2   -80.8     1  2018    20    32
##  6 2018-01-01 Hattie Avenue       Forsy~    36.1   -80.2     1  2018    22    NA
##  7 2018-01-01 Leggett             Edgec~    36.0   -77.6     1  2018    14    NA
##  8 2018-01-01 Millbrook School    Wake      35.9   -78.6     1  2018    28    34
##  9 2018-01-01 Pitt Agri. Center   Pitt      35.6   -77.4     1  2018    15    NA
## 10 2018-01-01 West Johnston Co.   Johns~    35.6   -78.5     1  2018    24    NA
## # ... with 8,966 more rows, and abbreviated variable names 1: meanLatitude,
## #   2: meanLongitude
```

Generate summary tables

12. Use the split-apply-combine strategy to generate a summary data frame. Data should be grouped by site, month, and year. Generate the mean AQI values for ozone and PM2.5 for each group. Then, add a pipe to remove instances where a month and year are not available (use the function `drop_na` in your pipe).

13. Call up the dimensions of the summary dataset.

```
# 12a Creating the Summaries dataframe of
# mean ozone and PM2.5 AQI values grouped by
# site, month, and year.
EPAair_O3_PM25_1819_Summaries <- EPAair_O3_PM25_NC1819_Processed %>%
    group_by(Site.Name, Month, Year) %>%
    summarise(MeanOzone = mean(Ozone), MeanPM2.5 = mean(PM2.5))
```

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

```
# 12b Removing NAs from MeanOzone and
# MeanPM2.5 columns
EPAair_O3_PM25_1819_Summaries <- EPAair_O3_PM25_1819_Summaries %>%
    drop_na(MeanOzone) %>%
    drop_na(MeanPM2.5)

# 13 checking dimensions of finalized
# dataset
dim(EPAair_O3_PM25_1819_Summaries)
```

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

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

    Answer: The na.omit function removes NAs from the entire dataframe, while the drop_na function allows you to remove NAs from the columns you specify.