Assignment 4: Data Wrangling

Danlei Zou

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

# install.packages('formatR')

library(formatR)
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 80), tidy = TRUE)

# checking working directory and loading lubridate and tidyverse
getwd()
```

```
## [1] "/Users/danleizou/EDA-Fall2022"
```

```
library(lubridate)
library(tidyverse)

# loading the four EPA Air raw datasets with stringAsFactors
O3NC2018 <- read.csv("./Data/Raw/EPAair_O3_NC2018_raw.csv", stringsAsFactors = TRUE)
```

```
O3NC2019 <- read.csv("./Data/Raw/EPAair_O3_NC2019_raw.csv", stringsAsFactors = TRUE)
PM25NC2018 <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv", stringsAsFactors = TRUE)
PM25NC2019 <- read.csv("./Data/Raw/EPAair_PM25_NC2019_raw.csv", stringsAsFactors = TRUE)

# 2

# exploring dimensions of datasets
dim(O3NC2018)
```

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

```
dim(O3NC2019)
```

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

```
dim(PM25NC2018)
```

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

```
dim(PM25NC2019)
```

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

```
# exploring column names of datasets
colnames(O3NC2018)
```

```
##  [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(O3NC2019)
```

```
##  [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(PM25NC2018)
```

```
##  [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(PM25NC2019)
```

```
##  [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"
```

```
# exploring structure of datasets
str(O3NC2018)
```

```
## '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 37003
##  $ 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 ...
```

```
str(O3NC2019)
```

```
## '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 37003
##  $ POC                                 : int  1 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 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 1 ...
##  $ CBSA_CODE                           : int  25860 25860 25860 25860 25860 25860 25860 25860 25860 2
##  $ CBSA_NAME                           : Factor w/ 15 levels "","Asheville, NC",..: 8 8 8 8 8 8 8 8 8
##  $ 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/ 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(PM25NC2018)
```

```
## '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 1 ...
##  $ 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  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 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 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(PM25NC2019)
```

```
## '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 ...
##  $ 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 ...
##  $ 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 date to date format in datasets
```

```
O3NC2018$Date <- mdy(O3NC2018$Date)
O3NC2019$Date <- mdy(O3NC2019$Date)
PM25NC2018$Date <- mdy(PM25NC2018$Date)
PM25NC2019$Date <- mdy(PM25NC2019$Date)

# checking to make sure date is now in date format
class(O3NC2018$Date)
```

```
## [1] "Date"
```

```
# 4

# selecting columns for each dataset

O3NC2018.subset <- select(O3NC2018, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
    COUNTY, SITE_LATITUDE, SITE_LONGITUDE)

O3NC2019.subset <- select(O3NC2019, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
    COUNTY, SITE_LATITUDE, SITE_LONGITUDE)

PM25NC2018.subset <- select(PM25NC2018, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
    COUNTY, SITE_LATITUDE, SITE_LONGITUDE)

PM25NC2019.subset <- select(PM25NC2019, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC,
    COUNTY, SITE_LATITUDE, SITE_LONGITUDE)

# checking dimensions of new subsets to make sure there are 7 columns in each
dim(O3NC2018.subset)
```

```
## [1] 9737    7
```

```
dim(O3NC2019.subset)
```

```
## [1] 10592     7
```

```
dim(PM25NC2018.subset)
```

```
## [1] 8983    7
```

```
dim(PM25NC2019.subset)
```

```
## [1] 8581    7
```

```
# 5

# filling cells in AQS_PARAMETER_DESC to 'PM2.5'
PM25NC2018.subset$AQS_PARAMETER_DESC = "PM2.5"

PM25NC2019.subset$AQS_PARAMETER_DESC = "PM2.5"
```

```
# checking length of columns in PM2.5 datasets after filling
view(PM25NC2018.subset$AQS_PARAMETER_DESC)
length(PM25NC2018.subset$AQS_PARAMETER_DESC)
```

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

```
view(PM25NC2019.subset$AQS_PARAMETER_DESC)
length((PM25NC2019.subset$AQS_PARAMETER_DESC))
```

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

```
# 6

# saving processed datasets to Data/Processed folder
write.csv(O3NC2018.subset, row.names = FALSE, file = "./Data/Processed/EPAair_O3_NC2018_processed.csv")
write.csv(O3NC2019.subset, row.names = FALSE, file = "./Data/Processed/EPAair_O3_NC2019_processed.csv")
write.csv(PM25NC2018.subset, row.names = FALSE, file = "./Data/Processed/EPAair_PM25_NC2018_processed.cs
write.csv(PM25NC2019.subset, row.names = FALSE, file = "./Data/Processed/EPAair_PM25_NC2019_processed.cs
```

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

# combining processed EPA air datasets
EPAairCombined <- rbind(O3NC2018.subset, O3NC2019.subset, PM25NC2018.subset, PM25NC2019.subset)

# checking dimensions of combined dataset
dim(EPAairCombined)
```

```
## [1] 37893     7
```

```
# 8

# wrangling dataset with pipe function to satisfy the conditions given
EPAairCombined2 <- EPAairCombined %>%
    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) %>%
    summarise(meanAQIValue = mean(DAILY_AQI_VALUE), meanLatitude = mean(SITE_LATITUDE),
        meanLongitude = mean(SITE_LONGITUDE)) %>%
    mutate(month = month(Date), year = year(Date))
```

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

```
# checking summary and dimensions of new dataset after pipe function
summary(EPAairCombined2)
```

```
##       Date                         Site.Name     AQS_PARAMETER_DESC
##  Min.   :2018-01-01   Millbrook School    :1435   Ozone:6830
##  1st Qu.:2018-07-01   Garinger High School:1429   PM2.5:7922
##  Median :2019-01-08   West Johnston Co.   :1222
##  Mean   :2018-12-30   Clemmons Middle     :1211
##  3rd Qu.:2019-06-28   Durham Armory       :1211
##  Max.   :2019-12-31   Hattie Avenue       :1207
##                       (Other)             :7037
##          COUNTY      meanAQIValue     meanLatitude   meanLongitude
##  Forsyth    :2418   Min.   :  0.00   Min.   :34.36   Min.   :-83.44
##  Wake       :1435   1st Qu.: 25.00   1st Qu.:35.43   1st Qu.:-80.79
##  Mecklenburg:1429   Median : 35.00   Median :35.86   Median :-79.80
##  Johnston   :1222   Mean   : 35.19   Mean   :35.68   Mean   :-79.67
##  Durham     :1211   3rd Qu.: 44.00   3rd Qu.:36.03   3rd Qu.:-78.46
##  Edgecombe  :1184   Max.   :129.00   Max.   :36.11   Max.   :-77.36
##  (Other)    :5853
##      month             year
##  Min.   : 1.000   Min.   :2018
##  1st Qu.: 4.000   1st Qu.:2018
##  Median : 6.000   Median :2019
##  Mean   : 6.402   Mean   :2019
##  3rd Qu.: 9.000   3rd Qu.:2019
##  Max.   :12.000   Max.   :2019
##
```

```
dim(EPAairCombined2)
```

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

```
# 9

# spreading ozone and PM2.5 AQI values into separate columns
```

```
EPAairCombined.spread <- pivot_wider(EPAairCombined2, names_from = AQS_PARAMETER_DESC,
    values_from = meanAQIValue)

# 10

# calling up dimensions of spread dataset
dim(EPAairCombined.spread)
```

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

```
# 11

# saving processed dataset to Data/Processed folder
write.csv(EPAairCombined.spread, row.names = FALSE, file = "./Data/Processed/EPAair_O3_PM25_NC1718_Proc
```

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 and 12b

# using split-apply-combine to generate summary dataset with meanAQI values of
# ozone and PM25
EPAairCombined.spread.summary <- EPAairCombined.spread %>%
    group_by(Site.Name, month, year) %>%
    summarise(meanAQIozone = mean(Ozone), meanAQIPM25 = mean(PM2.5)) %>%
    # pipe function to remove instances without month and year
drop_na(meanAQIozone, meanAQIPM25)
```

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

```
# 13

# checking dimensions of summary dataset
dim(EPAairCombined.spread.summary)
```

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

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

    Answer: The drop_na function will get rid of the rows with NAs, whereas the na.omit function will delete instances of NAs from the dataframe.