Assignment 4: Data Wrangling

Jaleesia Amos

Spring 2023

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A04_DataWrangling.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, Feb 20th @ 5:00pm.

Set up your session

1a. Load the `tidyverse`, `lubridate`, and `here` packages into your session.

1b. Check your working directory.

1c. Read in 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. Apply the `glimpse()` function to reveal the dimensions, column names, and structure of each dataset.

```
# 1a

#----Load packages into session & download if not already installed----#
pacman::p_load(tidyverse, lubridate, here)

# 1b
#----Check working directory----#
getwd()
```

```
## [1] "/Users/jaleesiad.amos/Documents/EDA-Spring2023"
```

```
# 1c

#----Load in four EPA Air datasets & read columns as factors----#

# _____EPAair NC 2018 data______#
EPAair_NC2018 <- read.csv("./Data/Raw/EPAair_O3_NC2018_raw.csv", stringsAsFactors = TRUE)

# _____EPAair NC 2019 data______#
EPAair_NC2019 <- read.csv("./Data/Raw/EPAair_O3_NC2019_raw.csv", stringsAsFactors = TRUE)

# _____EPAair NC 2018 Pollutant data______#
EPAair_PM_NC2018 <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv", stringsAsFactors = TRUE)

# _____EPAair NC 2019 Pollutant data______#
EPAair_PM_NC2019 <- read.csv("./Data/Raw/EPAair_PM25_NC2019_raw.csv", stringsAsFactors = TRUE)

# 2 _____Overview of EPAair NC 2018 data______#
glimpse(EPAair_NC2018)
```

```
## Rows: 9,737
## Columns: 20
## $ Date                                 <fct> 03/01/2018, 03/02/2018, 03/03/201~
## $ Source                               <fct> AQS, AQS, AQS, AQS, AQS, AQS, AQS~
## $ Site.ID                              <int> 370030005, 370030005, 370030005, ~
## $ POC                                  <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ Daily.Max.8.hour.Ozone.Concentration <dbl> 0.043, 0.046, 0.047, 0.049, 0.047~
## $ UNITS                                <fct> ppm, ppm, ppm, ppm, ppm, ppm, ppm~
## $ DAILY_AQI_VALUE                      <int> 40, 43, 44, 45, 44, 28, 33, 41, 4~
## $ Site.Name                            <fct> Taylorsville Liledoun, Taylorsvil~
## $ DAILY_OBS_COUNT                      <int> 17, 17, 17, 17, 17, 17, 17, 17, 1~
## $ PERCENT_COMPLETE                     <dbl> 100, 100, 100, 100, 100, 100, 100~
## $ AQS_PARAMETER_CODE                   <int> 44201, 44201, 44201, 44201, 44201~
## $ AQS_PARAMETER_DESC                   <fct> Ozone, Ozone, Ozone, Ozone, Ozone~
## $ CBSA_CODE                            <int> 25860, 25860, 25860, 25860, 25860~
## $ CBSA_NAME                            <fct> "Hickory-Lenoir-Morganton, NC", "~
## $ STATE_CODE                           <int> 37, 37, 37, 37, 37, 37, 37, 37, 3~
## $ STATE                                <fct> North Carolina, North Carolina, N~
## $ COUNTY_CODE                          <int> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, ~
## $ COUNTY                               <fct> Alexander, Alexander, Alexander, ~
## $ SITE_LATITUDE                        <dbl> 35.9138, 35.9138, 35.9138, 35.913~
## $ SITE_LONGITUDE                       <dbl> -81.191, -81.191, -81.191, -81.19~
```

```
# _____Overview of EPAair NC 2019 data______#
glimpse(EPAair_NC2019)
```

```
## Rows: 10,592
## Columns: 20
## $ Date                                 <fct> 01/01/2019, 01/02/2019, 01/03/201~
## $ Source                               <fct> AirNow, AirNow, AirNow, AirNow, A~
## $ Site.ID                              <int> 370030005, 370030005, 370030005, ~
## $ POC                                  <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ Daily.Max.8.hour.Ozone.Concentration <dbl> 0.029, 0.018, 0.016, 0.022, 0.037~
## $ UNITS                                <fct> ppm, ppm, ppm, ppm, ppm, ppm, ppm~
```

```
## $ DAILY_AQI_VALUE                      <int> 27, 17, 15, 20, 34, 34, 27, 35, 3~
## $ Site.Name                            <fct> Taylorsville Liledoun, Taylorsvil~
## $ DAILY_OBS_COUNT                      <int> 24, 24, 24, 24, 24, 24, 24, 24, 2~
## $ PERCENT_COMPLETE                     <dbl> 100, 100, 100, 100, 100, 100, 100~
## $ AQS_PARAMETER_CODE                   <int> 44201, 44201, 44201, 44201, 44201~
## $ AQS_PARAMETER_DESC                   <fct> Ozone, Ozone, Ozone, Ozone, Ozone~
## $ CBSA_CODE                            <int> 25860, 25860, 25860, 25860, 25860~
## $ CBSA_NAME                            <fct> "Hickory-Lenoir-Morganton, NC", "~
## $ STATE_CODE                           <int> 37, 37, 37, 37, 37, 37, 37, 37, 3~
## $ STATE                                <fct> North Carolina, North Carolina, N~
## $ COUNTY_CODE                          <int> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, ~
## $ COUNTY                               <fct> Alexander, Alexander, Alexander, ~
## $ SITE_LATITUDE                        <dbl> 35.9138, 35.9138, 35.9138, 35.913~
## $ SITE_LONGITUDE                       <dbl> -81.191, -81.191, -81.191, -81.19~
```

```
# _____Overview of EPAair NC 2018 Pollutant data_____#
glimpse(EPAair_PM_NC2018)
```

```
## Rows: 8,983
## Columns: 20
## $ Date                           <fct> 01/02/2018, 01/05/2018, 01/08/2018, 01/~
## $ Source                         <fct> AQS, AQS, AQS, AQS, AQS, AQS, AQS, AQS,~
## $ Site.ID                        <int> 370110002, 370110002, 370110002, 370110~
## $ POC                            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ Daily.Mean.PM2.5.Concentration <dbl> 2.9, 3.7, 5.3, 0.8, 2.5, 4.5, 1.8, 2.5,~
## $ UNITS                          <fct> ug/m3 LC, ug/m3 LC, ug/m3 LC, ug/m3 LC,~
## $ DAILY_AQI_VALUE                <int> 12, 15, 22, 3, 10, 19, 8, 10, 18, 7, 24~
## $ Site.Name                      <fct> Linville Falls, Linville Falls, Linvill~
## $ DAILY_OBS_COUNT                <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ PERCENT_COMPLETE               <dbl> 100, 100, 100, 100, 100, 100, 100, 100,~
## $ AQS_PARAMETER_CODE             <int> 88502, 88502, 88502, 88502, 88502, 8850~
## $ AQS_PARAMETER_DESC             <fct> Acceptable PM2.5 AQI & Speciation Mass,~
## $ CBSA_CODE                      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ CBSA_NAME                      <fct> "", "", "", "", "", "", "", "", "", "",~
## $ STATE_CODE                     <int> 37, 37, 37, 37, 37, 37, 37, 37, 37, 37,~
## $ STATE                          <fct> North Carolina, North Carolina, North C~
## $ COUNTY_CODE                    <int> 11, 11, 11, 11, 11, 11, 11, 11, 11, 11,~
## $ COUNTY                         <fct> Avery, Avery, Avery, Avery, Avery, Aver~
## $ SITE_LATITUDE                  <dbl> 35.97235, 35.97235, 35.97235, 35.97235,~
## $ SITE_LONGITUDE                 <dbl> -81.93307, -81.93307, -81.93307, -81.93~
```

```
# _____Overview of EPAair NC 2019 Pollutant data_____#
glimpse(EPAair_PM_NC2019)
```

```
## Rows: 8,581
## Columns: 20
## $ Date                           <fct> 01/03/2019, 01/06/2019, 01/09/2019, 01/~
## $ Source                         <fct> AQS, AQS, AQS, AQS, AQS, AQS, AQS, AQS,~
## $ Site.ID                        <int> 370110002, 370110002, 370110002, 370110~
## $ POC                            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ Daily.Mean.PM2.5.Concentration <dbl> 1.6, 1.0, 1.3, 6.3, 2.6, 1.2, 1.5, 1.5,~
## $ UNITS                          <fct> ug/m3 LC, ug/m3 LC, ug/m3 LC, ug/m3 LC,~
## $ DAILY_AQI_VALUE                <int> 7, 4, 5, 26, 11, 5, 6, 6, 15, 7, 14, 20~
```

```
## $ Site.Name                    <fct> Linville Falls, Linville Falls, Linvill~
## $ DAILY_OBS_COUNT              <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ PERCENT_COMPLETE             <dbl> 100, 100, 100, 100, 100, 100, 100, 100,~
## $ AQS_PARAMETER_CODE           <int> 88502, 88502, 88502, 88502, 88502, 8850~
## $ AQS_PARAMETER_DESC           <fct> Acceptable PM2.5 AQI & Speciation Mass,~
## $ CBSA_CODE                    <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ CBSA_NAME                    <fct> "", "", "", "", "", "", "", "", "", "",~
## $ STATE_CODE                   <int> 37, 37, 37, 37, 37, 37, 37, 37, 37, 37,~
## $ STATE                        <fct> North Carolina, North Carolina, North C~
## $ COUNTY_CODE                  <int> 11, 11, 11, 11, 11, 11, 11, 11, 11, 11,~
## $ COUNTY                       <fct> Avery, Avery, Avery, Avery, Avery, Aver~
## $ SITE_LATITUDE                <dbl> 35.97235, 35.97235, 35.97235, 35.97235,~
## $ SITE_LONGITUDE               <dbl> -81.93307, -81.93307, -81.93307, -81.93~
```

Wrangle individual datasets to create processed files.

3. Change date columns to be date objects.
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

#-------Convert Date column into date objects: month-day-year format for each dataset------#

# _____Date Conversion: EPAair NC 2018 data_____#
EPAair_NC2018$Date <- mdy(EPAair_NC2018$Date)

# _____Date Conversion: EPAair NC 2019 data_____#
EPAair_NC2019$Date <- mdy(EPAair_NC2019$Date)

# _____Date Conversion: EPAair NC 2018 Pollutant data_____#
EPAair_PM_NC2018$Date <- mdy(EPAair_PM_NC2018$Date)

# _____Date Conversion: EPAair NC 2019 Pollutant data_____#
EPAair_PM_NC2019$Date <- mdy(EPAair_PM_NC2019$Date)

# 4

#-------Selecting columns: Date, DAILY_AQI_VALUE, Site.Name,AQS_PARAMETER_DESC, COUNTY,
# SITE_LATITUDE, SITE_LONGITUDE for each dataset------#

# _______using pipes to select columns: EPAair NC 2018 data_____#
EPAair_NC2018_processed <- EPAair_NC2018 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, c(COUNTY:SITE_LONGITUDE))

# _______using pipes to select columns: EPAair NC 2019 data_____#
EPAair_NC2019_processed <- EPAair_NC2019 %>%
```

```
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, c(COUNTY:SITE_LONGITUDE))

# _______using pipes to select columns: EPAair NC 2018 Pollutant data______#
EPAair_PM_NC2018_processed <- EPAair_PM_NC2018 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, c(COUNTY:SITE_LONGITUDE))

# _______using pipes to select columns: EPAair NC 2019 Pollutant data______#
EPAair_PM_NC2019_processed <- EPAair_PM_NC2019 %>%
    select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, c(COUNTY:SITE_LONGITUDE))
# 5

#---------Fill all cells in 'PM' datasets, AQS_PARAMETER_DESC with 'PM2.5'------------#

# _______using pipes to replace values: EPAair NC 2018 Pollutant data______#
EPAair_PM_NC2018_processed <- EPAair_PM_NC2018_processed %>%
    mutate(AQS_PARAMETER_DESC = "PM2.5")

# _______using pipes to select columns: EPAair NC 2019 Pollutant data______#
EPAair_PM_NC2019_processed <- EPAair_PM_NC2019_processed %>%
    mutate(AQS_PARAMETER_DESC = "PM2.5")
# 6
#----------Save processed datasets---------#

# _____Save processed EPAair NC 2018 data______#
write.csv(EPAair_NC2018_processed, row.names = FALSE, file = "./Data/Processed/EPAair_O3_NC2018_processe

# _____Save processed EPAair NC 2019 data______#
write.csv(EPAair_NC2019_processed, row.names = FALSE, file = "./Data/Processed/EPAair_O3_NC2019_processe

# _____Save processed EPAair NC 2018 Pollutant data______#
write.csv(EPAair_PM_NC2018_processed, row.names = FALSE, file = "./Data/Processed/EPAair_PM25_NC2018_pr

# _____Save processed EPAair NC 2019 Pollutant data______#
write.csv(EPAair_PM_NC2019_processed, row.names = FALSE, file = "./Data/Processed/EPAair_PM25_NC2019_pr
```

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 - but it will include sites with missing site information...)
- Some sites have multiple measurements per day. Use the split-apply-combine strategy to generate daily means: group by date, site name, 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_NC1819_Processed.csv"

```
#7

#-------Combined processed datasets-------#
EPAair_combined <- rbind(EPAair_NC2018_processed, EPAair_NC2019_processed,
                         EPAair_PM_NC2018_processed, EPAair_PM_NC2019_processed)

#8
#______Include all sites that the four data frames have in common______#
EPAair_combined_common <- EPAair_combined %>%
  filter(Site.Name == "Linville Falls" | Site.Name == "Durham Armory" |
           Site.Name == "Leggett" | Site.Name =="Hattie Avenue" | Site.Name == "Clemmons Middle" |
           Site.Name == "Mendenhall School" | Site.Name == "Frying Pan Mountain" |
           Site.Name =="West Johnston Co." | Site.Name == "Garinger High School" |
           Site.Name == "Castle Hayne" | Site.Name == "Pitt Agri. Center" |
           Site.Name == "Bryson City" | Site.Name == "Millbrook School")


#_______Using split-apply-combine to generate daily means_________#
EPAair_combined_means <-
  EPAair_combined_common %>%
  group_by(Date, Site.Name, AQS_PARAMETER_DESC, COUNTY) %>%
  filter(!is.na(DAILY_AQI_VALUE) & !is.na(SITE_LATITUDE) & !is.na(SITE_LONGITUDE)) %>% #Removing NAs
  summarise(meanAQI = mean(DAILY_AQI_VALUE),
            meanlatitude = mean(SITE_LATITUDE),
            meanlongitude = mean(SITE_LONGITUDE))
```

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

```
EPAair_combined_means # output is dataframe with desired
```

```
## # A tibble: 14,752 x 7
## # Groups:   Date, Site.Name, AQS_PARAMETER_DESC [14,752]
##    Date       Site.Name           AQS_PARAMETE~1 COUNTY meanAQI meanl~2 meanl~3
##    <date>     <fct>               <fct>          <fct>    <dbl>   <dbl>   <dbl>
##  1 2018-01-01 Bryson City         PM2.5          Swain       35    35.4   -83.4
##  2 2018-01-01 Castle Hayne        PM2.5          New H~      13    34.4   -77.8
##  3 2018-01-01 Clemmons Middle     PM2.5          Forsy~      24    36.0   -80.3
##  4 2018-01-01 Durham Armory       PM2.5          Durham      31    36.0   -78.9
##  5 2018-01-01 Garinger High School Ozone         Meckl~      32    35.2   -80.8
##  6 2018-01-01 Garinger High School PM2.5         Meckl~      20    35.2   -80.8
##  7 2018-01-01 Hattie Avenue       PM2.5          Forsy~      22    36.1   -80.2
##  8 2018-01-01 Leggett             PM2.5          Edgec~      14    36.0   -77.6
##  9 2018-01-01 Millbrook School    Ozone          Wake        34    35.9   -78.6
```

```
## 10 2018-01-01 Millbrook School     PM2.5          Wake        28    35.9   -78.6
## # ... with 14,742 more rows, and abbreviated variable names
## #   1: AQS_PARAMETER_DESC, 2: meanlatitude, 3: meanlongitude
```

```
#_______Add month and year column to dataset_______#
EPAair_combined_means_expand <- mutate(EPAair_combined_means, Month = month(Date), Year = month(Date))
dim(EPAair_combined_means_expand) # output is 14752 of 9 variables
```

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

```
#9

#_______Spread your datasets such that AQI values for ozone and PM2.5 are in separate columns______#
EPAair_combined_means_expand2 <-  pivot_wider(EPAair_combined_means_expand,
            names_from = AQS_PARAMETER_DESC,
            values_from = meanAQI)

#10

#______Call up the dimensions of your new tidy dataset______#
dim(EPAair_combined_means_expand2)
```

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

```
#11

#_____Save processed EPAair combined dataset______#
write.csv(EPAair_combined_means_expand2, row.names = FALSE,
          file = "./Data/Processed/EPAair_O3_PM25_NC1819_Processed.csv")
```

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 mean **ozone** values are not available (use the function `drop_na` in your pipe). It's ok to have missing mean PM2.5 values in this result.

13. Call up the dimensions of the summary dataset.

```
# 12

# _______Using split-apply-combine to generate summary data frame_________#
EPAair_combined_means_expand_sum <- EPAair_combined_means_expand2 %>%
    group_by(Site.Name, Month, Year) %>%
    drop_na(Ozone) %>%
    summarise(meanOzone = mean(Ozone), meanPM = mean(PM2.5))
```

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

```
EPAair_combined_means_expand_sum
```

```
## # A tibble: 127 x 5
## # Groups:   Site.Name, Month [127]
##    Site.Name    Month  Year meanOzone meanPM
##    <fct>        <dbl> <dbl>     <dbl>  <dbl>
##  1 Bryson City      2     2      32.4   26.7
##  2 Bryson City      3     3      42.0   NA
##  3 Bryson City      4     4      45.0   27.4
##  4 Bryson City      5     5      37.8   NA
##  5 Bryson City      6     6      35.9   NA
##  6 Bryson City      7     7      32.5   NA
##  7 Bryson City      8     8      31.7   NA
##  8 Bryson City      9     9      30.4   NA
##  9 Bryson City     10    10      30.3   NA
## 10 Castle Hayne     2     2      35.8   12.8
## # ... with 117 more rows
```

```
# 13

dim(EPAair_combined_means_expand_sum)
```

```
## [1] 127   5
```

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

    Answer:

'drop_na' removes the rows with an 'NA' values. 'na.omit' does not include 'NA' values in calculations but does not remove 'NA's from dataframe; sometimes does not work with pipe function.