Assignment 4: Data Wrangling

Jinglin Zhang

Spring 2023

======= date: "Spring 2023" output: pdf_document geometry: margin=2.54cm editor_options: chunk_output_type: console —

a6d638b49c38a27960fa1bb235956abd244afafa ## OVERVIEW

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

Directions

«««< HEAD 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 @ 8:00am.

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] "C:/Users/wwwla/Documents"
```

```
library(tidyverse)
library(lubridate)
air.pm.19 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_PM25_NC2019_raw.csv",stringsAsFactors = TRUE)
air.pm.18 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_PM25_NC2018_raw.csv",stringsAsFactors = TRUE)
air.o3.19 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_O3_NC2019_raw.csv",stringsAsFactors = TRUE)
air.o3.18 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_O3_NC2018_raw.csv",stringsAsFactors = TRUE)
#2
dim(air.pm.19)
```

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

```
dim(air.pm.18)
```

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

```
dim(air.o3.19)
```

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

```
dim(air.o3.18)
```

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

```
colnames(air.pm.19)
```

```
##  [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(air.pm.18)
```

```
##  [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(air.o3.19)
```

```
##  [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(air.o3.18)
```

```
##  [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"
```

```
str(air.pm.19)
```

```
## '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 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 1 ..
##  $ SITE_LATITUDE                 : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE                : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

```
str(air.pm.18)
```

```
## '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 ...
##  $ 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(air.o3.19)
```

```
## '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 ...
##  $ 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 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 ...
##  $ 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(air.o3.18)
```

```
## '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 ...
##  $ 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  0.043 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 ...
##  $ 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 ...
##  $ 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
##  $ 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 ...
```

Wrangle individual datasets to create processed files.

3. Change date to date

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. Assign a useful **name to each code chunk** and include ample **comments** with your code.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.
7. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai.

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
library(tidyverse)
library(lubridate)
library(here)
#1b
getwd()
```

```
## [1] "C:/Users/wwwla/Documents"
```

```
#1c
air.pm.19 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_PM25_NC2019_raw.csv",stringsAsFactors = TRUE)
air.pm.18 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_PM25_NC2018_raw.csv",stringsAsFactors = TRUE)
air.o3.19 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_O3_NC2019_raw.csv",stringsAsFactors = TRUE)
air.o3.18 <- read.csv("~/EDA-Spring2023/Data/Raw/EPAair_O3_NC2018_raw.csv",stringsAsFactors = TRUE)
#2
glimpse(air.pm.19)
```

```
## 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~
```

```
glimpse(air.pm.18)
```

```
## 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~
```

```
glimpse(air.o3.19)
```

```
## 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~
```

```
glimpse(air.o3.18)
```

```
## 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~
```

Wrangle individual datasets to create processed files.

3. Change date columns to be date objects »»»> a6d638b49c38a27960fa1bb235956abd244afafa
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".

«««< HEAD

```
#3
air.pm.19$Date <- mdy(air.pm.19$Date)
air.pm.18$Date <- mdy(air.pm.18$Date)
air.o3.19$Date <- mdy(air.o3.19$Date)
air.o3.18$Date <- mdy(air.o3.18$Date)
```

```
#4
pm.19.select <- air.pm.19 %>% select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_
pm.18.select <- air.pm.18 %>% select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_
o3.19.select <- air.o3.19 %>% select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_
o3.18.select <- air.o3.18 %>% select(Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_
#5
pm.19.select$AQS_PARAMETER_DESC <- "pm2.5"
pm.18.select$AQS_PARAMETER_DESC <- "pm2.5"
#6
write.csv(pm.19.select, row.names = FALSE, file = "~/EDA-Spring2023/Data/Processed/EPAair_PM25_NC2019_pr
write.csv(pm.18.select, row.names = FALSE, file = "~/EDA-Spring2023/Data/Processed/EPAair_PM25_NC2018_pr
write.csv(o3.19.select, row.names = FALSE, file = "~/EDA-Spring2023/Data/Processed/EPAair_O3_NC2019_proc
write.csv(o3.18.select, row.names = FALSE, file = "~/EDA-Spring2023/Data/Processed/EPAair_O3_NC2018_proc
```

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_NC1819_Processed.csv"

```
#7
air.NC <- rbind(pm.19.select, pm.18.select, o3.19.select, o3.18.select)
#8
#=======
#TIP: At the end, ensure that your four dataframes each have different number of records, and that numb
library(dbplyr)
```

```
##
## 'dbplyr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     ident, sql
```

```
air.NC.processed <- air.NC %>%
  filter(Site.Name == "Linville Falls" | Site.Name == "Durham Armory" | Site.Name == "Leggett" | Site.Na
  group_by(Date, COUNTY, AQS_PARAMETER_DESC, Site.Name) %>%
  summarise( Mean.AQI = mean(DAILY_AQI_VALUE),
             Mean.Latitude = mean(SITE_LATITUDE),
             Mean.Longitdue = mean(SITE_LONGITUDE)) %>%
  mutate(month = month(Date), year = year(Date))
```

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

```
dim(air.NC.processed)
```

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

```
#9
air.NC.processed <- pivot_wider(air.NC.processed, names_from = AQS_PARAMETER_DESC, values_from = Mean.A
air.NC.processed
```

```
## # A tibble: 8,976 x 9
## # Groups:   Date, COUNTY [8,246]
##    Date       COUNTY  Site.Name   Mean.Latitude Mean.Longitdue month  year pm2.5
##    <date>     <fct>   <fct>               <dbl>          <dbl> <dbl> <dbl> <dbl>
##  1 2018-01-01 Durham  Durham Arm~          36.0          -78.9     1  2018    31
##  2 2018-01-01 Edgeco~ Leggett              36.0          -77.6     1  2018    14
##  3 2018-01-01 Forsyth Clemmons M~          36.0          -80.3     1  2018    24
##  4 2018-01-01 Forsyth Hattie Ave~          36.1          -80.2     1  2018    22
##  5 2018-01-01 Johnst~ West Johns~          35.6          -78.5     1  2018    24
##  6 2018-01-01 Meckle~ Garinger H~          35.2          -80.8     1  2018    20
##  7 2018-01-01 New Ha~ Castle Hay~          34.4          -77.8     1  2018    13
##  8 2018-01-01 Pitt    Pitt Agri.~          35.6          -77.4     1  2018    15
##  9 2018-01-01 Swain   Bryson City          35.4          -83.4     1  2018    35
## 10 2018-01-01 Wake    Millbrook ~          35.9          -78.6     1  2018    28
## # ... with 8,966 more rows, and 1 more variable: Ozone <dbl>
```

```
#10
dim(air.NC.processed)
```

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

```
#11
write.csv(air.NC.processed, row.names = FALSE, file = "~/EDA-Spring2023/Data/Processed/EPAair_O3_PM25_N
```

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

#8

#9

#10

#11
```

Generate summary tables

«««< HEAD 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). ======= 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. »»»> a6d638b49c38a27960fa1bb235956abd244afafa

13. Call up the dimensions of the summary dataset.

```
#12a
air.NC.summary <- air.NC.processed %>%
  group_by(Site.Name, month, year) %>%
  summarise(mean.sum.pm2.5 = mean(pm2.5),
            mean.sum.o3 = mean(Ozone),
            mean.sum.latitude = mean(Mean.Latitude),
            mean.sum.longitude = mean(Mean.Longitdue)) %>%
  drop_na(month, year)
```

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

```
air.NC.summary
```

```
## # A tibble: 308 x 7
## # Groups:   Site.Name, month [156]
##    Site.Name   month  year mean.sum.pm2.5 mean.sum.o3 mean.sum.latitude
##    <fct>       <dbl> <dbl>          <dbl>       <dbl>              <dbl>
##  1 Bryson City     1  2018           38.9        NA                 35.4
##  2 Bryson City     1  2019           29.8        NA                 35.4
##  3 Bryson City     2  2018           27.2        NA                 35.4
##  4 Bryson City     2  2019           33.0        NA                 35.4
##  5 Bryson City     3  2018           34.7        41.6               35.4
##  6 Bryson City     3  2019           NA          42.5               35.4
##  7 Bryson City     4  2018           28.2        44.5               35.4
##  8 Bryson City     4  2019           26.7        45.4               35.4
##  9 Bryson City     5  2018           NA          NA                 35.4
## 10 Bryson City     5  2019           NA          39.6               35.4
## # ... with 298 more rows, and 1 more variable: mean.sum.longitude <dbl>
```

```
#12b
air.NC.summary2 <- air.NC.processed %>%
  group_by(Site.Name, month, year) %>%
  summarise(mean.sum.pm2.5 = mean(pm2.5),
            mean.sum.o3 = mean(Ozone),
            mean.sum.latitude = mean(Mean.Latitude),
            mean.sum.longitude = mean(Mean.Longitdue)) %>%
  drop_na(mean.sum.o3)
```

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

```
air.NC.summary2
```

```
## # A tibble: 182 x 7
## # Groups:   Site.Name, month [109]
##    Site.Name   month  year mean.sum.pm2.5 mean.sum.o3 mean.sum.latitude
##    <fct>       <dbl> <dbl>          <dbl>       <dbl>              <dbl>
##  1 Bryson City     3  2018           34.7        41.6               35.4
##  2 Bryson City     3  2019           NA          42.5               35.4
##  3 Bryson City     4  2018           28.2        44.5               35.4
##  4 Bryson City     4  2019           26.7        45.4               35.4
##  5 Bryson City     5  2019           NA          39.6               35.4
##  6 Bryson City     6  2018           NA          37.8               35.4
##  7 Bryson City     6  2019           NA          34.0               35.4
##  8 Bryson City     7  2018           NA          34.6               35.4
##  9 Bryson City     7  2019           33.6        30.4               35.4
## 10 Bryson City     8  2018           NA          30.8               35.4
## # ... with 172 more rows, and 1 more variable: mean.sum.longitude <dbl>
```

```
#13
dim(air.NC.summary)
```

```
## [1] 308   7
```

```
dim(air.NC.summary2)
```

```
## [1] 182   7
```

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

    Answer: na.omit aims to remove all rows that comtain incomplete data, drop_na can drop rows where any specified colum comtains missing value.