««< HEAD ## Wrangle individual datasets to create processed files. ======= — title: "Assignment 4: Data Wrangling" author: "Elise Harrigan" output: pdf_document geometry: margin=2.54cm —

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 Tuesday, Feb 16 @ 11:59pm.

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
library(tidyverse)
library(lubridate)

EPA_PM25_19<-read.csv("../Data/Raw/EPAair_PM25_NC2019_raw.csv", stringsAsFactors=FALSE)

EPA_PM25_18<-read.csv("../Data/Raw/EPAair_PM25_NC2018_raw.csv", stringsAsFactors=FALSE)

EPA_03_2018<-read.csv("../Data/Raw/EPAair_O3_NC2018_raw.csv", stringsAsFactors=FALSE)

EPA_03_2019<-read.csv("../Data/Raw/EPAair_O3_NC2019_raw.csv", stringsAsFactors=FALSE)

#2
#explore EPA_PM25_2018
colnames(EPA_PM25_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"
```

```
dim(EPA_PM25_18)
```

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

```r
str(EPA_PM25_18)
```

```
## 'data.frame':    8983 obs. of  20 variables:
##  $ Date                          : chr  "01/02/2018" "01/05/2018" "01/08/2018" "01/11/2018" ...
##  $ Source                        : chr  "AQS" "AQS" "AQS" "AQS" ...
##  $ 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                         : chr  "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 ...
##  $ Site.Name                     : chr  "Linville Falls" "Linville Falls" "Linville Falls" "Linville
##  $ 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            : chr  "Acceptable PM2.5 AQI & Speciation Mass" "Acceptable PM2.5 AQ
##  $ CBSA_CODE                     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ CBSA_NAME                     : chr  "" "" "" "" ...
##  $ STATE_CODE                    : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ STATE                         : chr  "North Carolina" "North Carolina" "North Carolina" "North Ca
##  $ COUNTY_CODE                   : int  11 11 11 11 11 11 11 11 11 11 ...
##  $ COUNTY                        : chr  "Avery" "Avery" "Avery" "Avery" ...
##  $ SITE_LATITUDE                 : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE                : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

```r
#explore EPA_PM25_2019
colnames(EPA_PM25_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"
```

```r
dim(EPA_PM25_19)
```

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

```r
str(EPA_PM25_19)
```

```
## 'data.frame':    8581 obs. of  20 variables:
##  $ Date                          : chr  "01/03/2019" "01/06/2019" "01/09/2019" "01/12/2019" ...
##  $ Source                        : chr  "AQS" "AQS" "AQS" "AQS" ...
##  $ 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                         : chr  "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 ...
##  $ Site.Name                     : chr  "Linville Falls" "Linville Falls" "Linville Falls" "Linville
##  $ 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           : chr  "Acceptable PM2.5 AQI & Speciation Mass" "Acceptable PM2.5 A
##  $ CBSA_CODE                    : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ CBSA_NAME                    : chr  "" "" "" "" ...
##  $ STATE_CODE                   : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ STATE                        : chr  "North Carolina" "North Carolina" "North Carolina" "North Ca
##  $ COUNTY_CODE                  : int  11 11 11 11 11 11 11 11 11 11 ...
##  $ COUNTY                       : chr  "Avery" "Avery" "Avery" "Avery" ...
##  $ SITE_LATITUDE                : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE               : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

```r
#explore EPA_03_2018
colnames(EPA_03_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"
```

```r
dim(EPA_03_2018)
```

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

```r
str(EPA_03_2018)
```

```
## 'data.frame':    9737 obs. of  20 variables:
##  $ Date                                : chr  "03/01/2018" "03/02/2018" "03/03/2018" "03/04/2018" ..
##  $ Source                              : chr  "AQS" "AQS" "AQS" "AQS" ...
##  $ 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                               : chr  "ppm" "ppm" "ppm" "ppm" ...
##  $ DAILY_AQI_VALUE                     : int  40 43 44 45 44 28 33 41 45 40 ...
##  $ Site.Name                           : chr  "Taylorsville Liledoun" "Taylorsville Liledoun" "Taylo
##  $ 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 
##  $ AQS_PARAMETER_DESC                  : chr  "Ozone" "Ozone" "Ozone" "Ozone" ...
##  $ CBSA_CODE                           : int  25860 25860 25860 25860 25860 25860 25860 25860 25860 
##  $ CBSA_NAME                           : chr  "Hickory-Lenoir-Morganton, NC" "Hickory-Lenoir-Morgant
##  $ STATE_CODE                          : int  37 37 37 37 37 37 37 37 37 37 ...
```

```
##  $ STATE                               : chr  "North Carolina" "North Carolina" "North Carolina" "No
##  $ COUNTY_CODE                         : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ COUNTY                              : chr  "Alexander" "Alexander" "Alexander" "Alexander" ...
##  $ 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 ...
```

```r
#explore EPA_03_2019
colnames(EPA_03_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"
```

```r
dim(EPA_03_2019)
```

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

```r
str(EPA_03_2019)
```

```
## 'data.frame':    10592 obs. of  20 variables:
##  $ Date                                : chr  "01/01/2019" "01/02/2019" "01/03/2019" "01/04/2019" ..
##  $ Source                              : chr  "AirNow" "AirNow" "AirNow" "AirNow" ...
##  $ 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  0.029 0.018 0.016 0.022 0.037 0.037 0.029 0.038 0.038 0
##  $ UNITS                               : chr  "ppm" "ppm" "ppm" "ppm" ...
##  $ DAILY_AQI_VALUE                     : int  27 17 15 20 34 34 27 35 35 28 ...
##  $ Site.Name                           : chr  "Taylorsville Liledoun" "Taylorsville Liledoun" "Taylo
##  $ 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                  : chr  "Ozone" "Ozone" "Ozone" "Ozone" ...
##  $ CBSA_CODE                           : int  25860 25860 25860 25860 25860 25860 25860 25860 25860 2
##  $ CBSA_NAME                           : chr  "Hickory-Lenoir-Morganton, NC" "Hickory-Lenoir-Morganto
##  $ STATE_CODE                          : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ STATE                               : chr  "North Carolina" "North Carolina" "North Carolina" "No
##  $ COUNTY_CODE                         : int  3 3 3 3 3 3 3 3 3 3 ...
##  $ COUNTY                              : chr  "Alexander" "Alexander" "Alexander" "Alexander" ...
##  $ 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
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".

```
#vignette("dplyr")
```

```
#3. Change date to date

EPA_03_2018$Date <- as.Date(EPA_03_2018$Date, format = "%m/%d/%Y")
EPA_03_2019$Date <- as.Date(EPA_03_2019$Date, format = "%m/%d/%Y")
EPA_PM25_18$Date <- as.Date(EPA_PM25_18$Date, format = "%m/%d/%Y")
EPA_PM25_19$Date <- as.Date(EPA_PM25_19$Date, format = "%m/%d/%Y")

#4. Select the following columns: Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, COUNTY, SITE_LA

EPA_03_2018.selected <- dplyr::select(EPA_03_2018, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC

EPA_03_2019.selected <- dplyr::select(EPA_03_2018, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC

EPA_PM25_18.selected <- dplyr::select(EPA_PM25_18, Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC

EPA_PM25_19.selected<- dplyr::select(EPA_PM25_19,Date, DAILY_AQI_VALUE, Site.Name, AQS_PARAMETER_DESC, C

#5. For the PM2.5 datasets, fill all cells in AQS_PARAMETER_DESC with "PM2.5"
EPA_PM25_18_fill<-EPA_PM25_18.selected$AQS_PARAMETER_DESC <-  "PM2.5"

EPA_PM25_19_fill<-EPA_PM25_19.selected$AQS_PARAMETER_DESC <-  "PM2.5"

#6. Save processed datasets

write.csv(EPA_03_2018.selected, row.names = FALSE, file = "../Data/Processed/EPAair_O3_NC2018_Processed

write.csv(EPA_03_2019.selected, row.names = FALSE, file = "../Data/Processed/EPAair_O3_NC2019_Processed

write.csv(EPA_PM25_18.selected, row.names = FALSE, file = "../Data/Processed/EPAair_PM25_NC2018_Process

write.csv(EPA_PM25_19.selected, row.names = FALSE, file = "../Data/Processed/EPAair_PM25_NC2019_Process
```

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. combine the four datasets with `rbind`. Make sure your column names are identical prior to running
colnames(EPA_03_2018.selected)
```

```
## [1] "Date"                "DAILY_AQI_VALUE"    "Site.Name"
## [4] "AQS_PARAMETER_DESC" "COUNTY"             "SITE_LATITUDE"
## [7] "SITE_LONGITUDE"
```

```
colnames(EPA_03_2019.selected)
```

```
## [1] "Date"                "DAILY_AQI_VALUE"    "Site.Name"
## [4] "AQS_PARAMETER_DESC" "COUNTY"             "SITE_LATITUDE"
## [7] "SITE_LONGITUDE"
```

```
colnames(EPA_PM25_18.selected)
```

```
## [1] "Date"                "DAILY_AQI_VALUE"    "Site.Name"
## [4] "AQS_PARAMETER_DESC" "COUNTY"             "SITE_LATITUDE"
## [7] "SITE_LONGITUDE"
```

```
colnames(EPA_PM25_19.selected)
```

```
## [1] "Date"                "DAILY_AQI_VALUE"    "Site.Name"
## [4] "AQS_PARAMETER_DESC" "COUNTY"             "SITE_LATITUDE"
## [7] "SITE_LONGITUDE"
```

```
EPA_combined <-rbind(EPA_03_2018.selected, EPA_03_2019.selected, EPA_PM25_18.selected, EPA_PM25_19.sele

#8. Wrangle your new dataset with a pipe function

EPA_combined_subset <- EPA_combined %>%
  filter(Site.Name %in% c("Linville Falls", "Durham Armory", "Leggett", "Hattie Avenue", "Clemmons Midd
  group_by(Date, Site.Name, AQS_PARAMETER_DESC, COUNTY) %>%
  summarise(meanAQI = mean(DAILY_AQI_VALUE),
            meanlat = mean(SITE_LATITUDE),
            meanlong = 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
```

```
#9. Spread your datasets such that AQI values for ozone and PM2.5 are in separate columns

EPA_combined_subset_spread <- pivot_wider(EPA_combined_subset, names_from = AQS_PARAMETER_DESC, values_

#10. Call up the dimensions of your new tidy dataset.
dim(EPA_combined_subset_spread)
```

```
## [1] 8328    9
```

```r
#11. Save your processed dataset
write.csv(EPA_combined_subset_spread, row.names = FALSE, file = "../Data/Processed/EPAair_O3_PM25_NC1718
```

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.

```r
#12a. split-apply-combine strategy to generate a summary data frame
EPA_data.summary <-
  EPA_combined_subset_spread %>%
  group_by(Site.Name, Month, Year) %>%
  summarise(meanAQI = mean(Ozone),
            meanPM25 = mean(PM2.5)
  )
```

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

```r
#12b. add a pipe to remove instances where a month and year are not available using 'drop_na'
EPA_summary_final <-
  EPA_data.summary %>%
  drop_na(Month) %>%
  drop_na(Year)

#13. call the diminsions of the final data
dim(EPA_summary_final)
```

```
## [1] 292   5
```

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

    Answer: the drop_na function just drops any data that does not have a value. na_omit removes all incomplete objects from the data. We did not use na.omit because we did not want to delete the n/a just wanted them to be dropped from the final dataset.
    »»» 3b19294e4e347d5ad7c89fd521c23c2508d65903