Project #2-Conrardy

Anthony Conrardy

2024-03-03

Project #2 Details

IS 607 – Project 2

The goal of this assignment is to give you practice in preparing different datasets for downstream analysis work.

Your task is to:

(1) Choose any three of the "wide" datasets identified in the Week 6 Discussion items. (You may use your own dataset; please don't use my Sample Post dataset, since that was used in your Week 6 assignment!) **For each of the three chosen datasets:**

- Create a .CSV file (or optionally, a MySQL database!) that includes all of the information included in the dataset. You're encouraged to use a "wide" structure similar to how the information appears in the discussion item, so that you can practice tidying and transformations as described below.
- Read the information from your .CSV file into R, and use tidyr and dplyr as needed to tidy and transform your data. [Most of your grade will be based on this step!]
- Perform the analysis requested in the discussion item.
- Your code should be in an R Markdown file, posted to rpubs.com, and should include narrative descriptions of your data cleanup work, analysis, and conclusions.

(2) Please include in your homework submission, **for each of the three chosen datasets**:

- The URL to the .Rmd file in your GitHub repository, and
- The URL for your rpubs.com web page.

Dataset #1 Analysis

Importing the Dataset

We will pull school weather data from the website https://www.collegetransitions.com/dataverse/weather-data-by-college/. As directed, the table available on that website was converted to a CSV file and placed on the GitHub Repo. However, we could have read the data directly into a data frame by using the read_html function and scraped the data directly from the site.

```
# Pull the data from the CSV file located on the GitHub Repo
school_weather <- read.csv("https://raw.githubusercontent.com/Aconrard/DATA607/main/Project_2/school_wea
head(school_weather)
```

```
##   UnitID          Institution TAG        City State Avg.Jan.Temp Avg.April.Temp
## 1 188429   Adelphi University     Garden City    NY      39°/26°        59°/42°
## 2 138600 Agnes Scott College          Decatur    GA      52°/33°        73°/49°
## 3 168546       Albion College          Albion    MI      30°/16°        58°/37°
## 4 188641    Alfred University          Alfred    NY      32°/13°        56°/31°
## 5 210669    Allegheny College       Meadville    PA      32°/17°        58°/35°
## 6 131159  American University      Washington    DC      43°/27°        67°/45°
##   Avg.July.Temp Avg.Oct.Temp Days.w.Precipitation Sunny.Days
## 1       83°/67°      64°/48°                  127        173
## 2       88°/70°      72°/52°                  109        218
## 3       82°/60°      60°/40°                  127        169
## 4       80°/55°      59°/36°                  123        162
## 5       81°/58°      61°/40°                  160        160
## 6       89°/69°      69°/48°                  114        203
```

Review of the Data Structure

The suggested analysis involves identifying the school with the largest amount of sunny days, and to find out which institution has the same average weather for all four months. The first thing we must do is look over the data and see how it is structured and what may be possible to achieve this goal.

A quick review identifies a "Tag" column that can possibly be removed. There are four (4) months during the year for which temperatures are available, indicating the month following the solstice or equinox discriminating the seasons of winter, spring, summer and fall. There are two (2) temperatures given for each season, indicating the average high and average low temperature separated by "/". These will have to be parsed separately for analysis. There are an additional two(2) columns identifying the average days with precipitation and sun. The column of "sunny.days" allows us to immediately determine the school(s) with the greatest number of sunny and rainy days.

Tidy and Transform

We are going to tidy and transform the data set for analysts. We are going to pivot longer the seasonal temperatures, remove irrelevant text from the column names, remove the "°" symbol from temperature entries, and split the temperatures into High and Low for the different sites. This will tidy a data set from 470 observations and 11 variables to 1880 observations and 7 variables.

```
# Clean up the column headers to make pivot_longer easier and make variables clearer.
school_weather1 <- school_weather |> rename_with(~str_remove(.,'Avg.'), starts_with('Avg.'))
```

```r
# Clean up additional column names to be used for sunny day calculations.
school_weather2 <- school_weather1 |> mutate(Precipitation=sub("\\Days.w.", "", Days.w.Precipitation))

# Remove unnecessary columns since we used mutate to create new columns.
school_weather3 <- subset(school_weather2, select = -c(Days.w.Precipitation, Sunny.Days))

# Pivot longer the temperatures, remove the "°" symbol from the entries, and the divide high and low te
school_weather_Winter <- school_weather3 |>
  pivot_longer(cols = Jan.Temp,
 names_to = c("Month"),
 values_to = "High_Low") |> mutate(High_Low = gsub("°", "", High_Low)) |>
  mutate(Month = sub("\\.Temp", "", Month),) |>
  separate(High_Low, into = c("High", "Low"), sep="/") |>
  select(UnitID, Institution, City, State, Month, High, Low)

school_weather_Spring <- school_weather3 |>
  pivot_longer(cols = April.Temp,
 names_to = c("Month"),
 values_to = "High_Low") |> mutate(High_Low = gsub("°", "", High_Low)) |>
  mutate(Month = sub("\\.Temp", "", Month),) |>
  separate(High_Low, into = c("High", "Low"), sep="/") |>
  select(UnitID, Institution, City, State, Month, High, Low)

school_weather_Summer <- school_weather3 |>
  pivot_longer(cols = July.Temp,
 names_to = c("Month"),
 values_to = "High_Low") |> mutate(High_Low = gsub("°", "", High_Low)) |>
  mutate(Month = sub("\\.Temp", "", Month),) |>
  separate(High_Low, into = c("High", "Low"), sep="/") |>
  select(UnitID, Institution, City, State, Month, High, Low)

school_weather_Fall <- school_weather3 |>
  pivot_longer(cols = Oct.Temp,
 names_to = c("Month"),
 values_to = "High_Low") |> mutate(High_Low = gsub("°", "", High_Low)) |>
  mutate(Month = sub("\\.Temp", "", Month),) |>
  separate(High_Low, into = c("High", "Low"), sep="/") |>
  select(UnitID, Institution, City, State, Month, High, Low)

# Perform a row bind in preparation for analysis.
school_weather_year <- rbind(school_weather_Winter, school_weather_Spring, school_weather_Summer, schoo
head(school_weather_year)
```

```
## # A tibble: 6 x 7
##   UnitID Institution         City        State Month High  Low
##    <int> <chr>               <chr>       <chr> <chr> <chr> <chr>
## 1 188429 Adelphi University  Garden City NY    Jan   39    26
## 2 138600 Agnes Scott College Decatur     GA    Jan   52    33
## 3 168546 Albion College      Albion      MI    Jan   30    16
## 4 188641 Alfred University   Alfred      NY    Jan   32    13
## 5 210669 Allegheny College   Meadville   PA    Jan   32    17
## 6 131159 American University Washington  DC    Jan   43    27
```

Analysis

We can now start the analysis portion of this project. It was suggested that we identify the school with the largest amount of sunny days, and to find out which institution has the same average weather for all four months. The first part seems fairly simple by arranging one of the initially transformed data sets to identify the institution with the largest number of sunny days. We will also identify the institution with the greatest number of days with precipitation.

```
# In this step we will drop unnecessary columns and arrange the "Sunny" column to identify the school w
most_sunny <- school_weather3 |> select(-c(TAG, Jan.Temp, April.Temp, July.Temp, Oct.Temp)) |> arrange(d
head(most_sunny,5)
```

```
##   UnitID                                            Institution
## 1 104151                          Arizona State University-Tempe
## 2 182281                          University of Nevada-Las Vegas
## 3 188030                 New Mexico State University-Main Campus
## 4 123961                       University of Southern California
## 5 110422 California Polytechnic State University-San Luis Obispo
##              City State Precipitation Sunny
## 1           Tempe    AZ            36   300
## 2       Las Vegas    NV            26   294
## 3      Las Cruces    NM            43   293
## 4     Los Angeles    CA            32   292
## 5 San Luis Obispo    CA            51   287
```

```
# In this step we will do the same for the most rainy days
most_rainy <- school_weather3 |> select(-c(TAG, Jan.Temp, April.Temp, July.Temp, Oct.Temp)) |> arrange(d
head(most_rainy,5)
```

```
##   UnitID                           Institution         City State
## 1 126818 Colorado State University-Fort Collins Fort Collins    CO
## 2 240727                 University of Wyoming      Laramie    WY
## 3 231624           College of William and Mary Williamsburg    VA
## 4 213385                     Lafayette College       Easton    PA
## 5 186399             Rutgers University-Newark       Newark    NJ
##   Precipitation Sunny
## 1           237   237
## 2           231   233
## 3           212   116
## 4           206   206
## 5           206   206
```

We can then start the analysis of the second part by trying to identify the institutions with the same average weather for all four seasons. That means we need to consider both high and low temperatures for the different sites. We can try by filtering the data to see what happens. However, it is extremely unlikely that we will identify a single case where the average temperatures will be the same for each season, especially in North America above the equator. Therefore, we will change the parameters a bit and try to identify any institution where the seasonal average high and low temperatures are the same for at least 2 or more seasons. This will allow us to identify any institutions with 2 or more seasonal highs and matching lows.

```
# In this step we will create a new variable as a seasonal average of both the high and low temperature
school_season <- school_weather_year |> mutate(season_avg = round((as.integer(High) + as.integer(Low))/2
```

```r
# In this step we will group the data by Institution and High Temperature, and then filter the result t
duplicates <- school_season |> group_by(Institution, High) |> filter(n()>1) |> group_by(Institution, Lo
duplicates
```

```
## # A tibble: 42 x 8
## # Groups:   Institution, Low [21]
##    UnitID Institution                   City  State Month High  Low   season_avg
##     <int> <chr>                         <chr> <chr> <chr> <chr> <chr>      <dbl>
##  1 189705 Canisius College              Buff~ NY    July  81    61            71
##  2 189705 Canisius College              Buff~ NY    Oct   81    61            71
##  3 126818 Colorado State University-Fo~ Fort~ CO    April 87    57            72
##  4 126818 Colorado State University-Fo~ Fort~ CO    July  87    57            72
##  5 190150 Columbia University in the C~ New ~ NY    April 84    69            76
##  6 190150 Columbia University in the C~ New ~ NY    July  84    69            76
##  7 163046 Loyola University Maryland    Balt~ MD    July  88    69            78
##  8 163046 Loyola University Maryland    Balt~ MD    Oct   88    69            78
##  9 239105 Marquette University          Milw~ WI    July  83    63            73
## 10 239105 Marquette University          Milw~ WI    Oct   83    63            73
## # i 32 more rows
```

```r
# In this step we will see if the seasonal average match.  We will group by institution and seasonal av
duplicates1 <- school_season |> group_by(Institution, season_avg) |> filter(n()>1) |> arrange(Instituti
duplicates1
```

```
## # A tibble: 62 x 8
## # Groups:   Institution, season_avg [31]
##    UnitID Institution                   City  State Month High  Low   season_avg
##     <int> <chr>                         <chr> <chr> <chr> <chr> <chr>      <dbl>
##  1 189705 Canisius College              Buff~ NY    July  81    61            71
##  2 189705 Canisius College              Buff~ NY    Oct   81    61            71
##  3 126818 Colorado State University-Fo~ Fort~ CO    April 87    57            72
##  4 126818 Colorado State University-Fo~ Fort~ CO    July  87    57            72
##  5 190150 Columbia University in the C~ New ~ NY    April 84    69            76
##  6 190150 Columbia University in the C~ New ~ NY    July  84    69            76
##  7 198695 High Point University         High~ NC    April 72    47            60
##  8 198695 High Point University         High~ NC    Oct   72    49            60
##  9 159391 Louisiana State University    Bato~ LA    April 79    56            68
## 10 159391 Louisiana State University    Bato~ LA    Oct   80    57            68
## # i 52 more rows
```

Findings and Conclusions

From the initial output identifying the sunniest days, we find that Arizona State University-Tempe is the sunniest with 300 days of sun. Looking at the days involving precipitation, we see that Colorado State University-Fort Collins enjoys 237 days of rain. However, we also see that they also enjoy 237 days of sun, which would account for 474 days of weather a year. This could be simply an error in the data source, or we may not be aware of how the source considered the precipitation (i.e. rain occurs 24 hours a day where the sun is only up part of the day). In any case, the suggestion was to find the school with the greatest amount of sun and that was Arizona State University-Tempe.

A review of the data frame results clearly indicate there are no institutions where all the seasonal temperatures are the same. The most we can observe are two matching seasons, either April and July or July and October. No season match the January temperatures, which would indicate the winter season. This data frame returned 21 institutions. A review of the data frame where we matched the seasonal average temperatures, returned an additional 10 institutions which brings up the total of 31 institutions that have matching seasonal temperatures. However, regardless of choosing to match the high/low temperatures or the daily average temperature, the same pattern of pairing of either April/July or July/October occurs.

Dataset #2 Analysis

Importing the Dataset

For the purposes of this analysis, we will pull the CSV data from the GitHub repo. The CSV was downloaded directly from the NYC OpenData site at https://data.cityofnewyork.us/Environment/Air-Quality/c3uy-2p5r/about_data and placed on the GitHub repo with supporting documentation at GitHub.https://raw.githubusercontent.com/Aconrard/DATA607/main/Project_2/Air_Quality_20240226.csv

```
air_quality_data <- read.csv("https://raw.githubusercontent.com/Aconrard/DATA607/main/Project_2/Air_Qual
head(air_quality_data,5)
```

```
##   Unique.ID Indicator.ID                   Name Measure Measure.Info
## 1    172653          375 Nitrogen dioxide (NO2)    Mean          ppb
## 2    172585          375 Nitrogen dioxide (NO2)    Mean          ppb
## 3    336637          375 Nitrogen dioxide (NO2)    Mean          ppb
## 4    336622          375 Nitrogen dioxide (NO2)    Mean          ppb
## 5    172582          375 Nitrogen dioxide (NO2)    Mean          ppb
##   Geo.Type.Name Geo.Join.ID                     Geo.Place.Name
## 1         UHF34         203 Bedford Stuyvesant - Crown Heights
## 2         UHF34         203 Bedford Stuyvesant - Crown Heights
## 3         UHF34         204                      East New York
## 4         UHF34         103                 Fordham - Bronx Pk
## 5         UHF34         104                Pelham - Throgs Neck
##           Time.Period Start_Date Data.Value Message
## 1 Annual Average 2011  12/1/2010      25.30      NA
## 2 Annual Average 2009  12/1/2008      26.93      NA
## 3 Annual Average 2015   1/1/2015      19.09      NA
## 4 Annual Average 2015   1/1/2015      19.76      NA
## 5 Annual Average 2009  12/1/2008      22.83      NA
```

Review of the Data Structure

There are four (4) outdoor air pollutants of note in this data set: Fine Particulate Matter (PM2.5), Nitrogen Dioxide (NO2), Sulfur Dioxide (SO2) and Ozone (O3). However, it also contains data on the health burden of the different air pollutants in the same variable category. The column names should be renamed to be more accurate to the data contained therein, and the time period variable needs to be transformed into something that can be graphically displayed. While there is a lot that can be obtained with this dataset, we will focus on the extraction of one air pollutant for analysis, fine particulate matter 2.5 (PM2.5).

Tidy and Transform

```
# First we are going to filter the data to identify the PM2.5 entries and them select the columns we wa
pm_25 <- air_quality_data |> filter(Name == "Fine particles (PM 2.5)") |> select(-c(Indicator.ID, Geo.Ty
```

```r
# Now we are going to parse out the different possible entries for the time periods of Winter, Summer, 
# Starting with the annual average
pm_25_annual <- pm_25 |> filter(grepl("Annual", Time.Period)) |> separate(Time.Period, into = c("Interva

# Then with the summer.
pm_25_summer <- pm_25 |> filter(grepl("Summer", Time.Period))|> separate(Time.Period, into = c("Interval

# Finally with the winter.
pm_25_winter <- pm_25 |> filter(grepl("Winter", Time.Period))|> separate(Time.Period, into = c("Interval

# Now we will bind the rows into a complete set.
pm_25_complete <- bind_rows(pm_25_annual, pm_25_summer, pm_25_winter)
head(pm_25_complete)
```

```
##   Unique.ID                    Name                             Location
## 1    212069 Fine particles (PM 2.5)                        East New York
## 2    214164 Fine particles (PM 2.5)      St. George and Stapleton (CD1)
## 3    742182 Fine particles (PM 2.5)            Hunts Point - Mott Haven
## 4    214123 Fine particles (PM 2.5)      Highbridge and Concourse (CD4)
## 5    214110 Fine particles (PM 2.5) Lower East Side and Chinatown (CD3)
## 6    170402 Fine particles (PM 2.5) Lower East Side and Chinatown (CD3)
##   Interval Year Start_Date mean_mcg_m_3
## 1   Annual 2014  12/1/2013         9.04
## 2   Annual 2014  12/1/2013         8.61
## 3   Annual 2021   1/1/2021         6.98
## 4   Annual 2014  12/1/2013         9.99
## 5   Annual 2014  12/1/2013        10.31
## 6   Annual 2013  12/1/2012         9.90
```

Analysis

As we start the analysis portion of this dataset, we need to narrow down what to look at since this dataset has a large amount of information that could potentially be investigated, For our purposes, we will attempt to identify the number of locations that record the data for PM2.5, and then choose several locations to perform a limited analysis of the winter, summer and annual average of PM2.5 measurements for the years of 2009-2021. The areas of Bedford Stuyvesant (Brooklyn), Willowbrook (Staten Island), Rego Park and Forest Hills (Queens), Fordham (Bronx), and Midtown (Manhattan) as proxies for the geographic centers for each of the five (5) boroughs. We provide a plot of the winter, summer and average annual PM2.5 measurements in µg/m^3 for each of these areas, as well as individual plots for further investigation.

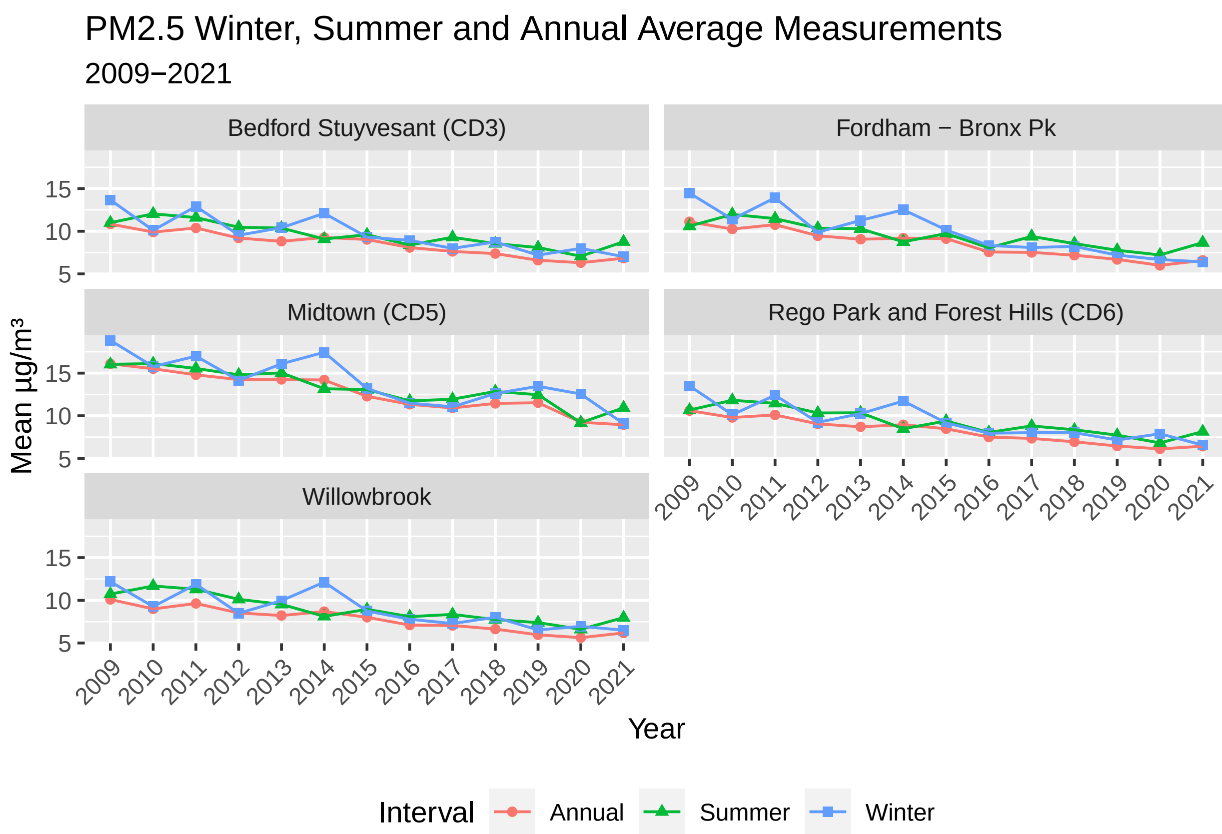
```r
# First we are going to find out how many groups based on location are in the dataset.
pm_25_complete |> group_by(Location)
```

```
## # A tibble: 5,499 x 7
## # Groups:   Location [114]
##    Unique.ID Name                 Location Interval Year  Start_Date mean_mcg_m_3
##        <int> <chr>                <chr>    <chr>    <chr> <chr>             <dbl>
##  1    212069 Fine particles (PM~ East Ne~ Annual   2014  12/1/2013          9.04
##  2    214164 Fine particles (PM~ St. Geo~ Annual   2014  12/1/2013          8.61
##  3    742182 Fine particles (PM~ Hunts P~ Annual   2021  1/1/2021           6.98
##  4    214123 Fine particles (PM~ Highbri~ Annual   2014  12/1/2013          9.99
```

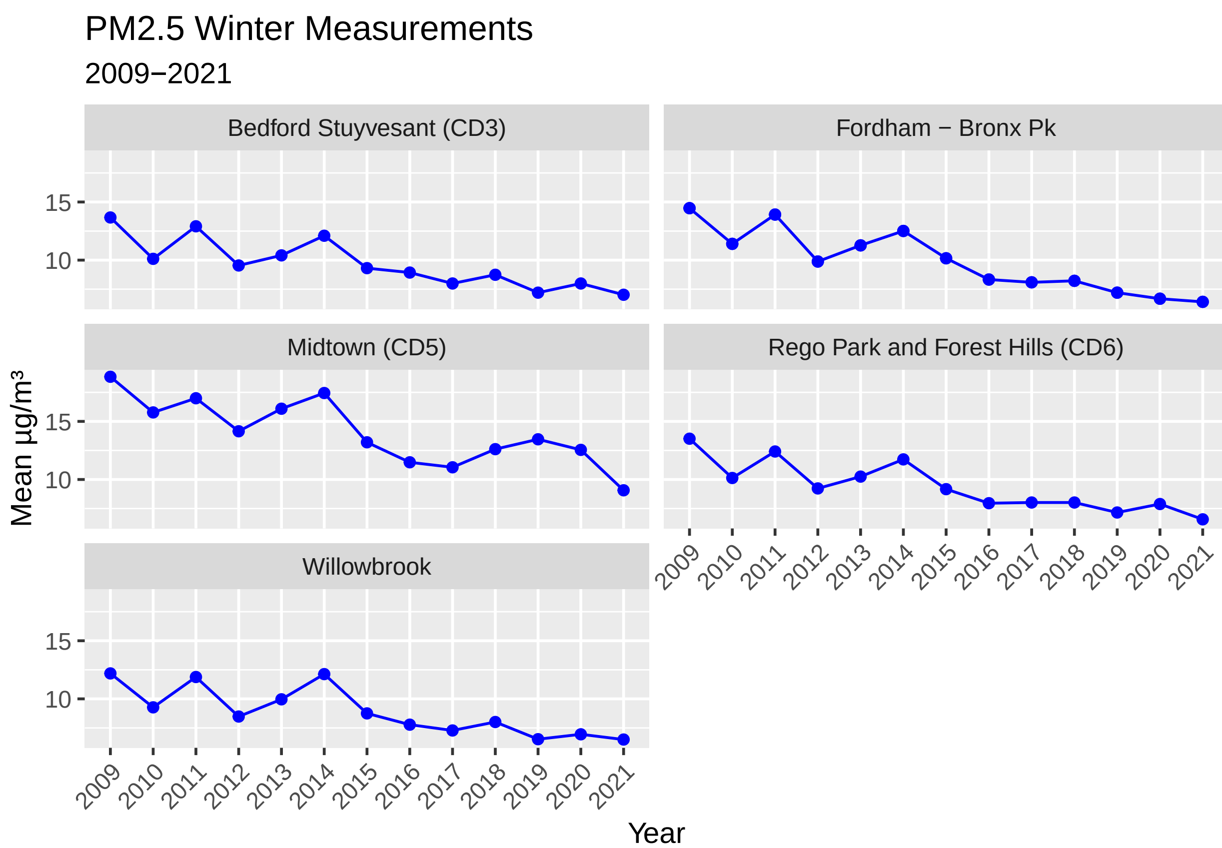
```
##  5    214110 Fine particles (PM~ Lower E~ Annual   2014  12/1/2013         10.3
##  6    170402 Fine particles (PM~ Lower E~ Annual   2013  12/1/2012          9.9
##  7    212933 Fine particles (PM~ East Fl~ Annual   2014  12/1/2013          8.97
##  8    547390 Fine particles (PM~ Greenpo~ Annual   2017  1/1/2017           8.98
##  9    742185 Fine particles (PM~ Greenpo~ Annual   2021  1/1/2021           7.67
## 10    547621 Fine particles (PM~ Upper E~ Annual   2017  1/1/2017           8.87
## # i 5,489 more rows
```

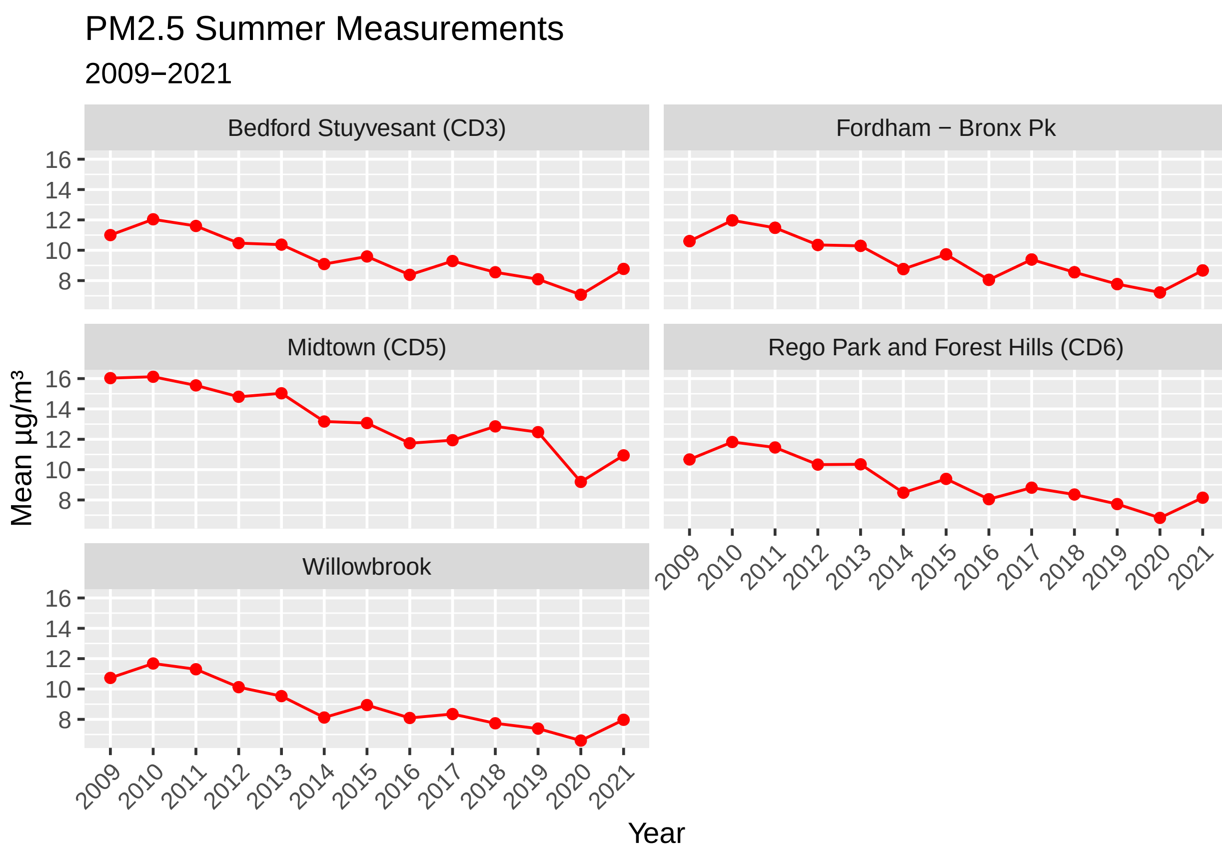
```
# Now that we have identified there are 114 different locations throughout the city of New York, we now

groups_pm <- pm_25_complete |> group_by(Location)

# Once we have identified the individual neighborhoods near the centers of the five boroughs, we are go

borough_centers <- c("Bedford Stuyvesant (CD3)", "Willowbrook", "Rego Park and Forest Hills (CD6)", "Fo

pm_boro_centers <- pm_25_complete |> filter(Location %in% borough_centers) |> arrange(Location)
head(pm_boro_centers,5)
```

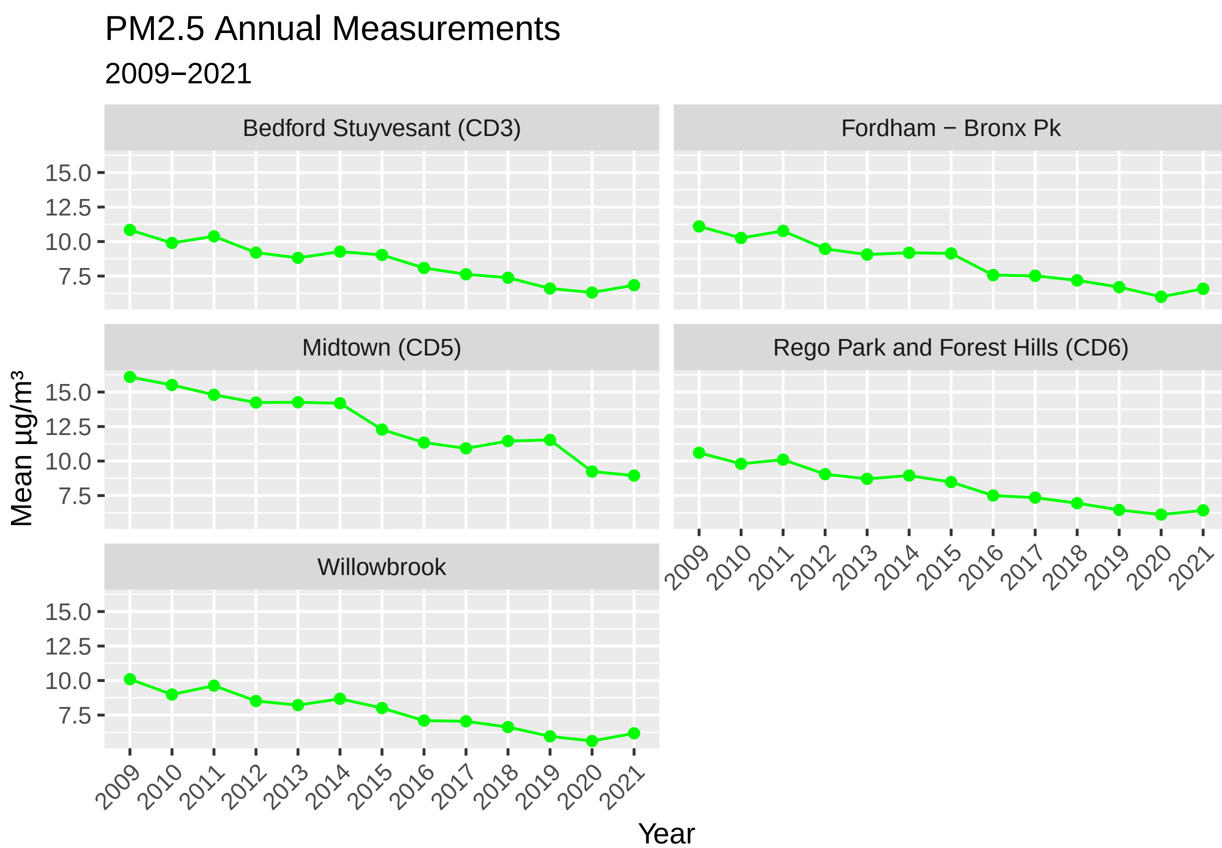
```
##   Unique.ID                    Name                 Location Interval Year
## 1    411046 Fine particles (PM 2.5) Bedford Stuyvesant (CD3)   Annual 2016
## 2    170426 Fine particles (PM 2.5) Bedford Stuyvesant (CD3)   Annual 2013
## 3    170308 Fine particles (PM 2.5) Bedford Stuyvesant (CD3)   Annual 2011
## 4    170190 Fine particles (PM 2.5) Bedford Stuyvesant (CD3)   Annual 2009
## 5    214134 Fine particles (PM 2.5) Bedford Stuyvesant (CD3)   Annual 2014
##   Start_Date mean_mcg_m_3
## 1 12/31/2015         8.09
## 2  12/1/2012         8.82
## 3  12/1/2010        10.38
## 4  12/1/2008        10.84
## 5  12/1/2013         9.27
```



PM2.5 Winter, Summer and Annual Average Measurements
2009−2021
Bedford Stuyvesant (CD3)
Fordham − Bronx Pk
Midtown (CD5)
Rego Park and Forest Hills (CD6)
Willowbrook
Mean µg/m³
5
10
15
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
Year
Interval
Annual
Summer
Winter


PM2.5 Winter Measurements
2009−2021
Bedford Stuyvesant (CD3)
Fordham − Bronx Pk
Midtown (CD5)
Rego Park and Forest Hills (CD6)
Willowbrook
Mean µg/m³
15
10
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
Year


PM2.5 Summer Measurements
2009−2021
Bedford Stuyvesant (CD3)
Fordham − Bronx Pk
Midtown (CD5)
Rego Park and Forest Hills (CD6)
Willowbrook
Mean µg/m³
16
14
12
10
8
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
Year



Findings and Conclusions

There are some points that are immediately brought forth from these plots that should be noted:

1) There is a general trend of decreasing amounts of PM2.5 noted across all five (5) boroughs of New York City for the period of 2009-2021 in all three measurements for winter, summer and annual average µg/m³.
2) While most areas have have similar levels of PM2.5, midtown Manhattan is substantially higher for all years and all measurement intervals.
3) Staten Island (Willowbrook) has substantially lower PM2.5 levels for all years and all measurement intervals.
4) There appear to be spikes in winter measurements in all five (5) boroughs in the years 2009, 2011, 2014.

Based upon this very limited analysis, we can see that air pollution as a result of PM2.5 has been trending downward for this 13 year period, which is potentially beneficial to overall population health. However, we did not investigate this in comparison to the other measured pollutants of ozone, sulfur dioxide, or nitrogen dioxide. Also, the EPA has established standards that set the level of 12µg/m³ for PM2.5, and the WHO has established guidelines recommending that it not exceed 10µg/m³. While most boroughs have been below these levels for many years, midtown Manhattan only dropped below these levels in the last 4 years. The last few years must be taken in context of the COVID pandemic and exodus of the workforce to remote working that may have temporarily influenced the measurements. Finally, the spikes in winter measurements as noted earlier need to be better explained, as they were seen citywide. While not necessarily related, those years noted substantially lower average winter temperatures than seen traditionally. A possible explanation may be the increased use of heating fuels citywide to keep the population warm. Further investigation in all areas is needed.

Dataset #3 Analysis

For the purposes of this analysis, we can import this dataset directly into R from the website using read_html. This website actually has three (3) distinctly different tables, and we can import all of them to see which one best fits our purpose to analyze dataset by comparing U.S. Vehicle Models. This could also be achieved by importing the tables into Microsoft Excel, exporting the CSV files into GitHub, and then read_csv into R. This process cuts out the intervening steps. However, since I can't guarantee the reliability of the website, I will only display the code that performs this function, and will read in a CSV file for the table data we decide to use.

Importing the Dataset

{r Site Scrape} cars <- read_html("https://www.goodcarbadcar.net/2023-us-vehicle-sales-figures-by-model/") tables <- cars |> html_table(fill = TRUE) table1 <- tables[[1]] table2 <- tables[[2]] table3 <- tables[[3]] table2

head(table1,5) head(table2,5) head(table3,5)

write.csv(table2, "brand_model_sales.csv", row.names = FALSE)

Review of the Data Structure

A review of the different tables suggests that Table 2 may be the most accessible for the data we need to perform a comparison of the US car companies. This table contains the make and model of the car, as well as Q4 and YTD data for 2022 and 2023, which allows us to compare the sales year over year. So we will use that table to create our data frame to tidy and transform for our analysis. The table was imported using the code chunk above, and then exported to GitHub fo access and importing. We notice immediately that there are some issues that need to be addressed. The manufacturers name is connected with the model type in one variable that will need to be separated. The column names are awkward to deal with and need to be simplified. There a commas located in the car sales numbers that will need to be removed. Also, the data contains foreign and domestic manufacturers and, since the suggestion was to compare US car companies, we will need to exclude identified foreign car manufactures. While there may be others, or this list inaccurate, we will assume for this analysis that it is indeed accurate.

US car companies include:

General Motors (GM) - Known for brands like Chevrolet, GMC, Cadillac, and Buick.

Ford Motor Company - Famous for its Ford brand vehicles.

Tesla - Known for electric vehicles and renewable energy products.

Stellantis North America - Formed from the merger of Fiat Chrysler Automobiles (FCA) and PSA Group, includes brands like Jeep, Ram, Chrysler, Dodge, and Fiat.

Rivian - Known for electric adventure vehicles.

Lucid Motors - Known for luxury electric vehicles.

```
table2 <- read.csv("https://raw.githubusercontent.com/Aconrard/DATA607/main/Project_2/brand_model_sales

us_car_makers <- c("GMC", "Ford", "Chevrolet", "Cadillac", "Buick", "Tesla", "Fiat", "Jeep", "Ram", "Do
```

```
filtered_data <- table2[grep(paste(us_car_makers, collapse = "|"), table2$modelName), ]
head(filtered_data,5)
```

```
##          modelName Q4.2023 Q4.2022 Year.To.Date Year.to.Date.Previous.Year
## 49   Buick Enclave  10,929   7,719       39,412                     30,532
## 50    Buick Encore     122   2,487        5,888                     13,717
## 51 Buick Encore GX  13,756   9,052       63,969                     33,349
## 52  Buick Envision   9,439   7,663       44,282                     25,870
## 53   Buick Envista   7,916       0       13,301                          0
```

Tidy and Transform

In this section we shall now begin to tidy the data to make it more manageable and able to be transformed and analyzed. We are going to split the car brand from the model in the modelName variable, rename the variables for quarterly and yearly sales to be easier to place into code, remove commas from the sales volumes so we can deal with the variables as numeric.

```
# In this step we separate the name of the manufacturer from the character string and now call it "bran
filtered_data$brand <- sub("^([[:alpha:]]+).*", "\\1", filtered_data$modelName)

# In this step we take the remaining portion of the character string and name it "model" in a new colum
filtered_data$model <- sub("^[[:alpha:]]+\\s*", "", filtered_data$modelName)

# In this step we do some renaming conventions for the variables and the select the variables for our n
brand_model <- filtered_data |> rename(q4_2023="Q4.2023")|> rename(q4_2022="Q4.2022")|> rename(ytd_2023=

# In this step we substitute the "," with nothing, to remove them from the characters we want to use as
brand_model <- brand_model |> mutate(q4_2023=gsub(",","",q4_2023),
                      q4_2022=gsub(",","", q4_2022),
                      ytd_2023=gsub(",","", ytd_2023),
                      ytd_2022=gsub(",","", ytd_2022))
```

Analysis

The suggestion was to compare sales between U.S. vehicle models; however, there are a number of ways we can approach this. Since some automakers represent a number of different models under different brand names, we will do the comparison based upon brand name, and not the parent automaker company. Additionally, some manufacturers debut and retire models from year to year, which makes comparison of sales between models complicated. However, we can aggregate the individual model sales for a particular brand and see overall how well the brand performed year over year (YOY) from 2022 to 2023. Therefore, we are going to calculate the difference in vehicle sales for each model and brand for the fourth quarter (Q4) and year_to_date YTD, and then sum the volume losses and gains for each brand to see performance from 2022 to 2023. We will present our summary in a chart for evaluation.

```
# In this step we calculate the difference in sales for both Q4 and year-to_date (YTD)
brand_model <- brand_model |> mutate(q4_diff=as.numeric(q4_2023)-as.numeric(q4_2022),
                      ytd_diff=as.numeric(ytd_2023)-as.numeric(ytd_2022))

# In this step we are going group by bran name and then calculate the sum totals of the gains and losse
summary_brands <- brand_model |> group_by(brand) |>
  summarize(sales_lost_q4 = sum(ifelse(q4_diff < 0, q4_diff, 0)),
```

Brand	Model Sales Loss Q4 YOY	Model Sales Gain Q4 YOY	Model Sales Loss YTD YOY	M
Buick	-2365	17606	-7829	71
Cadillac	-6493	3734	-1595	14
Chevrolet	-64621	51588	-48714	23
Chrysler	-14576	5	-918	22
Dodge	-8774	5012	-14894	23
Fiat	0	59	-407	90
Ford	-45905	48811	-90872	22
GMC	-18412	9052	-23891	69
Jeep	-17118	25526	-74698	31
Lincoln	-490	2301	-9710	76
Lucid	0	101	0	11
Ram	-5159	9648	-26444	20
Rivian	-501	5502	0	33
Tesla	-140000	0	-96768	58
Total	-324414	178945	-396740	81

```
            sales_gain_q4 = sum(ifelse(q4_diff > 0, q4_diff, 0)),
            sales_lost_ytd = sum(ifelse(ytd_diff<0, ytd_diff,0)),
            sales_gain_ytd = sum(ifelse(ytd_diff>0, ytd_diff,0)),
            total_sales_q4 = sum(q4_diff),
            total_sales_ytd = sum(ytd_diff))

# In this step we are going to rename the columns for presentation purposes.

colnames(summary_brands) = c("Brand", "Model Sales Loss Q4 YOY", "Model Sales Gain Q4 YOY", "Model Sale

# In this step we are going to create a summary row of totals of all the brands
sums <- colSums(summary_brands[,2:7], na.rm = TRUE)

# Since our data frame has 7 total columns, and out sums row has only 6, we need to add on a 7th column
total_row <- c("Total", sums)

# In this step we are going to bind the total_row to the last of the summary_brands data frame.
sums_summary_brands <- rbind(summary_brands,total_row)

# In this step we create a table for analysis
sums_summary_brands |> kbl() |> kable_classic_2(full_width=F, font_size=12)
```

Findings and Conclusions

We can immediately see that almost all of the car brands had losses in Q4 model sales from 2022 to 2023, but we also see that there are offsetting increases in other model sales for the same quarter. However, 50% of the brands did not have enough offsetting gains in other model sales to demonstrate a net increase for Q4 year over year. It should be noted that Tesla accounts for 140,000 of the 145,469 overall decrease in Q4 sales YOY, or 96.2%, while the other 13 brands combined account for only a loss of 5,469 units in lost Q4 volume.

We can see the same result for the year over year results. All of the brands had some model sales that resulted in losses YTD year over year from 2022 to 2023. However, many of the brands had offsetting model sale gains that results in their YTD sales being greater for 2023 than 2022. Buick, Chevrolet, Ford, GMC, and Rivian demonstrated very positive gains in sales year over year, while Jeep and Tesla demonstrated net losses in model sales year over year.

Despite the perceived individual performance in Q4 of 2023, overall YTD sales of US brand vehicles increased by 422,618 units from 2022 to 2023. While there may be many reasons that could potentially explain the performance of Tesla, one possible consideration is the greater number of EV manufacturers entering the market, including newcomers Rivian and Lucid, as well as the already established brands.

There are many limitations to this analysis, including the source and validity of the data from the website. Also, there are also a number of different ways this data could have been analyzed, and it suggested that other interested parties conduct their own analysis and report their findings.