Analysis of the effect of PM10 Concentration, Temperature, Geographic Location, Population, and distance to Electricity-Generation Combustion Points on the concentration of fine particulate matter (PM2.5) in North Carolina during the year 2018.

https://github.com/fr55/DataAnalytics_FinalProject

Felipe Raby Amadori

Abstract

Experimental overview. This section should be no longer than 250 words.

Contents

List of Tables

List of Figures

1 Research Question and Rationale

Nowadays air pollution is one of the most relevant health issues in the world. It refers to the contamination of the air by chemicals, biological materials, and other types of pollutants that are harmful to human health. To solve the problem of air pollution, it's necessary to understand the problem, what are the causes, and search for solutions based on the findings.

Particulate matter with a diameter of less than 2.5 micrometers is called PM2.5, and it is a extremely harmful air pollutant because it consists of particles with diameters that are less than or equal to 2.5 microns in size, which can get deeply into the lung, and ultimately impair lung function.

This study focus on trying to understand how PM2.5 concentration in North Carolina vary with temperature, PM10 concentration, zoning (piedmont, coastal, mountain), population, elevation, and distance to combustion points for electricity generation. This last variable was included because acording to the EPA combustion for electricity generation is the major point-source sector for PM2.5 in the USA (EPA, 2019).

The research question is: What are the effects of temperature, PM10 concentration, zoning (piedmont, coastal, mountain), population, elevation, distance to combustion points for electricity generation, in PM2.5 concentrations within North Carolina in the year 2018?

2 Dataset Information

For the analysis the following datasets were considered:

2.1 EPA PM2.5 Dataset

This dataset contains data from air quality monitoring of PM2.5 in North Carolina in 2018, and it was obtained using the Download Daily Data Tool in the United States Environmental Protection Agency (EPA) webpage https://www.epa.gov/outdoor-air-quality-data/download-daily-data where the options showed in Table 1 were selected:

Option	Selection
Pollutant	PM2.5
Year	2018
Geographic Area	North Carolina
Monitor Site	All Sites
Download	Download CSV (spreadsheet)

Table 1: Selections

The downloaded file was saved in the project folder path ./Data/Raw/ as EPAair_PM25_NC2018_raw.csv on 2019-03-31.

2.1.1 Data Content Information

The dataset contains daily mean PM2.5 concentration in ug/m3 in 2018. Data from 24 stations in 21 different counties of North Carolina with their location in NAD83 lat/long coordinates.

The dataset contains 19 columns, which are shown in Table 2. Column names without description are self-explanatory.

2.2 EPA PM10 Dataset

This dataset contains data from air quality monitoring of PM10 in North Carolina in 2018, and it was obtained using the Download Daily Data Tool in the United States Environmental Protection Agency (EPA) webpage https://www.epa.gov/outdoor-air-quality-data/download-daily-data where the options showed in Table 3 were selected:

The downloaded file was saved in the project folder path ./Data/Raw/ as EPAair_PM10_NC2018_raw.csv on 2019-03-31.

Column	Description
Date	mm/dd/YY
Source	AQS (Air Quality System)
Site ID	A unique number identifying the site.
POC	"Parameter Occurrence Code", distinguishes different instruments that measure the same parameter at the same site.
Daily Mean PM2.5 Concentration	
Units	Concentration Units
Site Name	
DAILY_OBS_COUNT	
PERCENT_COMPLETE	
AQS_PARAMETER_CODE	
AQS_PARAMETER_DESC	
CBSA_CODE	
CBSA_NAME	
STATE_CODE	
COUNTY CODE	A unique number identifying the County.
COUNTY	
SITE_LATITUDE	NAD83
SITE_LONGITUDE	NAD83

Table 2: Dataset content

Option	Selection
Pollutant	PM10
Year	2018
Geographic Area	North Carolina
Monitor Site	All Sites
Download	Download CSV (spreadsheet)

Table 3: Selections

2.2.1 Data Content Information

The dataset contains daily mean PM10 concentration in ug/m3 in 2018. Data from 9 stations in 8 different counties of North Carolina with their location in NAD83 lat/long coordinates.

The dataset contains 19 columns, which are shown in Table 4. Column names without description are self-explanatory.

2.3 NOAA Average Temperature Dataset

This dataset contains data from temperature monitoring in North Carolina in 2018, and it was obtained using the Data Search Tool in the National Center for Environmental Information

Column	Description
Date	mm/dd/YY
Source	AQS (Air Quality System)
Site ID	A unique number identifying the site.
POC	“Parameter Occurrence Code”, distinguishes different instruments that measure the same parameter at the same site.
Daily Mean PM10 Concentration	
Units	Concentration Units
Site Name	
DAILY_OBS_COUNT	
PERCENT_COMPLETE	
AQS_PARAMETER_CODE	
AQS_PARAMETER_DESC	
CBSA_CODE	
CBSA_NAME	
STATE_CODE	A unique number identifying the County.
COUNTY CODE	A unique number identifying the County.
COUNTY	
SITE_LATITUDE	NAD83
SITE_LONGITUDE	NAD83

Table 4: Dataset content

of the National Oceanic and Atmospheric Administration (NOAA). Webpage https://www.ncdc.noaa.gov/cdo-web. Options showed in Table 5 were selected: XXXXXXArreglar

Option	Selection
Pollutant	PM10
Year	2018
Geographic Area	North Carolina
Monitor Site	All Sites
Download	Download CSV (spreadsheet)

Table 5: Selections

The downloaded file was saved in the project folder path ./Data/Raw/ as NOAA_TAVG_NC2018_raw.csv on 2019-03-28.

2.3.1 Data Content Information

The dataset contains daily mean air temperature in Farenheit in 2018. Data from 39 stations in North Carolina with their location in NAD83 lat/long coordinates. No county information.

The dataset contains 7 columns, which are shown in Table 6. Column names without description are self-explanatory.

Column	Description
STATION	A unique code identifying the site.
NAME	Station Name
Site ID	A unique number identifying the site.
LATITUDE	NAD83
LONGITUDE	NAD83
DATE	dd/mm/YY
TAVG	Daily Average Temperature in °F

Table 6: Dataset content

2.4 US Census Bureau Us counties shapefile

This dataset contains geographic and geometric information of all the counties of the US. The data is in NAD83 lat/long coordinates. De file was provided by John Fay in the Environmental Data Analytics (ENV 872L) course at Duke University, Spring 2019.

The files containing the information were saved in the project folder path ./Data/Spatial/ as cb_2017_us_county_20m on 2019-03-28.

2.4.1 Data Content Information

The dataset contains geographic and geometric information of all the counties of the US in NAD83 lat/long coordinates.

The dataset contains 10 columns, which are shown in Table 7. Column names without description are self-explanatory.

Column	Description
STATEFP	A unique number identifying the State.
COUNTYFP	County Federal Information Processing Standards (FIPS) Code
COUNTYNS	Provides the American National Standards Institute (ANSI) code for the county or equivalent entity, as used by GNIS.
AFFGEOID	AFF Summary Level Code
GEOID	NAD83
NAME	County Name
LSAD	Legal/statistical area description
ALAND	County Land Area in square meters
AWATER	County Water Area in square meters
Geometry	Geometry and geographic information

Table 7: Dataset content

2.5 EPA major point-source sectors for PM2.5 Dataset

This dataset contains facility-level locations for the top 8 major point-source sectors for PM2.5 in the US, and it was obtained from the United States Environmental Protection Agency (EPA) webpage https://www3.epa.gov/air/emissions/where.htm. The Top PM2.5 emitting sectors link was selected.

The downloaded file was saved in the project folder path ./Data/Raw/ as EPA_ElecGenComb_US_raw.kml on 2019-03-31.

2.5.1 Data Content Information

The dataset is a kml file that contains point location for the top 8 major point-source sectors for PM2.5 in the US. The data is in WGS84 lat/long coordinates.

All data sets, variable, and files are named according to the following naming convention: *databasename_datatype_details_stage.format*, where:

- databasename refers to the database from where the data originated
- datatype is a description of data
- details are additional descriptive details, particularly important for processed data
- stage refers to the stage in data management pipelines (e.g., raw, cleaned, temp or processed)

2.6 Analized data structure

With these datasets an exploratory data analysis was done and for the study. The datasets were wrangled and a file called PM2.5_Full_Elev_utm.shp was created, which has the data structure shown in Table 8.

Variable	Units	N.Elements	Range	Source.File
Date	YY-mm-dd	343	From 2018-01-01 to 2018-12-09	EPAair_PM25_NC2018_raw.csv
Site_ID	-	24	-	EPAair_PM25_NC2018_raw.csv
COUNTY	-	21	-	EPAair_PM25_NC2018_raw.csv
Population	People	21	From 5,507 to 1,034,290	https://en.wikipedia.org/
Zone	-	3	Coastal, Piedmont, and Mountains	NC County Maps
PM2_5	ug/m3	6499	From -2.5 to 34.2	EPAair_PM25_NC2018_raw.csv
PM10	ug/m3	926	From 0 to 35	EPAair_PM10_NC2018_raw.csv
TAVG	Farenheit	4011	From 11 to 87	NOAA_TAVG_NC2018_raw.csv
Emiss_Dist	meters	24	From 813.5 to 81800.9	Self made
Elevation	meters	24	From 0.04 to 1418.8	Package elevatr

Table 8: Summary of data structure

3 Exploratory Data Analysis and Wrangling

Upload PM2.5 and PM10 2018 data raw data files associated with EPA Air dataset.

```
EPA_AQPM25_NC2018_raw <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv")
EPA_AQPM10_NC2018_raw <- read.csv("./Data/Raw/EPAair_PM10_NC2018_raw.csv")

#Formatting Dates
EPA_AQPM25_NC2018_raw$Date <- as.Date(EPA_AQPM25_NC2018_raw$Date, format = "%m/%d/%y")
EPA_AQPM10_NC2018_raw$Date <- as.Date(EPA_AQPM10_NC2018_raw$Date, format = "%m/%d/%Y")

#Selecting Columns
EPA_AQ_PM25_NC2018_Temp <- select(EPA_AQPM25_NC2018_raw, Date, Site.ID,
                                  Daily.Mean.PM2.5.Concentration, AQS_PARAMETER_DESC,
                                  COUNTY:SITE_LONGITUDE)

#Changing column name
colnames(EPA_AQ_PM25_NC2018_Temp)[colnames(EPA_AQ_PM25_NC2018_Temp)
                                  =="Daily.Mean.PM2.5.Concentration"]<-"Daily.Mean.Concent

#Selecting Columns
EPA_AQ_PM10_NC2018_Temp <- select(EPA_AQPM10_NC2018_raw, Date, Site.ID,
                                  Daily.Mean.PM10.Concentration, AQS_PARAMETER_DESC,
                                  COUNTY:SITE_LONGITUDE)

#Changing column name
colnames(EPA_AQ_PM10_NC2018_Temp)[colnames(EPA_AQ_PM10_NC2018_Temp)
                                  =="Daily.Mean.PM10.Concentration"]<-"Daily.Mean.Concentr

#Create AQS_PARAMETER_DESC Column with Contaminant description.
EPA_AQ_PM25_NC2018_Temp$AQS_PARAMETER_DESC <- "PM2.5"
EPA_AQ_PM10_NC2018_Temp$AQS_PARAMETER_DESC <- "PM10"

#Eliminates duplicate dates
EPA_AQ_PM25_NC2018_Cleaned <- EPA_AQ_PM25_NC2018_Temp [!duplicated(EPA_AQ_PM25_NC2018_Te

EPA_AQ_PM10_NC2018_Cleaned <- EPA_AQ_PM10_NC2018_Temp [!duplicated(EPA_AQ_PM10_NC2018_Te

# Combine the data.
EPA_AQ_PM2.5PM10_NC2018_Cleaned <- rbind(EPA_AQ_PM25_NC2018_Cleaned, EPA_AQ_PM10_NC2018_

#Save the data in the processed folder
write.csv(EPA_AQ_PM2.5PM10_NC2018_Cleaned,
          "./Data/Processed/EPA_AQ_PM2.5PM10_NC2018_Cleaned.csv")
```

```
#Spread PM2.5 and PM10
EPA_AQ_PM2.5PM10_NC2018_Spread <-
  EPA_AQ_PM2.5PM10_NC2018_Cleaned %>%
  spread(AQS_PARAMETER_DESC, Daily.Mean.Concentration)

#Remove rows without PM2.5 data
EPA_AQ_PM2.5PM10_NC2018_Spread <- EPA_AQ_PM2.5PM10_NC2018_Spread[!is.na(EPA_AQ_PM2.5PM10

#Convert the dataset to a spatially enabled "sf" data frame
PM2.5_PM10_sf <- st_as_sf(EPA_AQ_PM2.5PM10_NC2018_Spread,coords = c('SITE_LONGITUDE','SI

#Convert all to UTM Zone 17 (crs = 26917)
PM2.5_PM10_sf_utm <- st_transform(PM2.5_PM10_sf, c=26917)
```

See in **??** the locations of the PM2.5 monitoring stations.

In the next chunk I am getting the list of North Carolina Counties and Population from an URL.

```
#North Carolina Counties
url <- "https://en.wikipedia.org/wiki/List_of_counties_in_North_Carolina"
webpage <- read_html(url)

County_Name <- webpage %>% html_nodes("th:nth-child(1)") %>% html_text()
County_Population <- webpage %>% html_nodes("tr :nth-child(7)") %>% html_text()

#Remove unwanted info and characters
County_Info <- data_frame(County = County_Name[9:108])
County_Info$County <- str_replace(County_Info$County, " County", "")
County_Info$County <- str_replace(County_Info$County, "\n", "")

County_Info$Population <- NULL

Population <- data_frame(Population=County_Population[2:101])

County_Info <- cbind(County_Info, Population)

County_Info$Population <- str_replace(County_Info$Population,",","")
County_Info$Population <- str_replace(County_Info$Population,",","")

County_Info$Population <- as.numeric(County_Info$Population)
```

In the next chunk I am assigning the corresponding zone to each county. Info from: Rudersdorf, Amy. 2010. "NC County Maps." Government & Heritage Library, State Library of North Carolina.

```r
#North Carolina Zones
County_Info$Zone<-ifelse(County_Info$County == 'Ashe'|County_Info$County =='Alleghany'|C
                          ifelse(County_Info$County == 'Surry'|County_Info$County =='Stok
                                 ifelse(County_Info$County == 'Scotland'|County_Info$Coun
```

```r
PM2.5_PM10_Info_sf_utm <- PM2.5_PM10_sf_utm %>%
left_join(County_Info, by = c("COUNTY"="County"))
```

In the next chunk I am reading the USA county shapefile, sub-setting for NC.

```r
counties_sf<- st_read('./Data/Spatial/cb_2017_us_county_20m.shp') %>%
  filter(STATEFP == 37) #Filter for just NC Counties
```

```
## Reading layer `cb_2017_us_county_20m' from data source `C:\Users\Felipe\OneDrive - Du
## Simple feature collection with 3220 features and 9 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -179.1743 ymin: 17.91377 xmax: 179.7739 ymax: 71.35256
## epsg (SRID):    4269
## proj4string:    +proj=longlat +datum=NAD83 +no_defs
```

```r
#CRS
st_crs(counties_sf) #crs=4269 = NAD83.
```

```
## Coordinate Reference System:
##   EPSG: 4269
##   proj4string: "+proj=longlat +datum=NAD83 +no_defs"
```

Converting the counties_sf to UTM Zone 17

```r
#Convert all to UTM Zone 17 (crs = 26917)
counties_sf_utm <- st_transform(counties_sf, c=26917)

#Join the two datasets using "NAME" from the left dataset and "County" from the right
#counties_sf_utm_join <- counties_sf_utm %>%
#  left_join(y = County_Info,by = c("NAME" =  "County"))

#counties_sf_utm_simple <- select(counties_sf_utm_join, NAME, Zone, Population, geomet
```

Read the 2018 Air Temperature data.

```r
#Read the 2018 Air Temperature data
NOAA_DTAVG_NC2018_raw <- read.csv("./Data/Raw/NOAA_TAVG_NC2018_raw.csv")

#Remove stations without Temperature information
NOAA_DTAVG_NC2018_Complete <- na.omit(NOAA_DTAVG_NC2018_raw)

#Convert the dataset to a spatially enabled "sf" data frame
```

```r
NOAA_DTAVG_NC2018_sf <- st_as_sf(NOAA_DTAVG_NC2018_Complete,coords = c('LONGITUDE','LATI

#Convert all to UTM Zone 17 (crs = 26917)
NOAA_DTAVG_NC2018_sf_utm <- st_transform(NOAA_DTAVG_NC2018_sf, c=26917)

#Formatting dates
NOAA_DTAVG_NC2018_sf_utm$DATE <- as.Date(NOAA_DTAVG_NC2018_sf_utm$DATE, format = "%d/%m/
```

The 2018 Air Temperature data does not have County information, so the location is used to locate the county of each station.

```r
#Adding the county and zone information to the Temperature dataframe

#Index of the matching feature
county_index <- st_nearest_feature(NOAA_DTAVG_NC2018_sf_utm, counties_sf_utm)

#Eliminates geo info
aux1 <- st_set_geometry(counties_sf_utm[county_index,"NAME"], value=NULL)

#adds the columns
NOAA_DTAVG_NC2018_sf_utm$COUNTY <- aux1$NAME

#Reordering
NOAA_DTAVG_NC2018_sf_utm <- NOAA_DTAVG_NC2018_sf_utm[,c(1,2,3,4,5,7,6)]
```

For each PM2.5 Station we assign the Temperature of the nearest Temperature Station.

```r
#Use again PM2.5_Stations with only the PM2.5 station info
#Data frame with only the PM2.5 station info
PM2.5_Stations <- PM2.5_PM10_Info_sf_utm %>%
  select(Site.ID, geometry) %>%
  subset(!duplicated(Site.ID))

#Distances bewteen the PM2.5 stations and the Temperature Stations
Nearest <- st_nearest_feature(PM2.5_Stations, NOAA_DTAVG_NC2018_sf_utm)

a <- length(unique(PM2.5_Stations$Site.ID))

NOAA_DTAVG_NC2018_sf_utm$NAME <- as.character(NOAA_DTAVG_NC2018_sf_utm$NAME)

#Assingning the nearest Temperature Station to each PM2.5 station.
for (i in 1:a){
  PM2.5_Stations$Temp_Est[i] <- NOAA_DTAVG_NC2018_sf_utm$NAME[Nearest[i]]
}

#Drop the geo data
```

```r
aux2 <- st_set_geometry(PM2.5_Stations, value=NULL)

#Left_join the data
PM2.5_PM10_Temp_sf_utm <- PM2.5_PM10_Info_sf_utm %>%
left_join(aux2)

#Assingning the Temperature of the nearest Temperature Station to each PM2.5 station.

#Drops the geo data
aux3 <- st_set_geometry(NOAA_DTAVG_NC2018_sf_utm, value=NULL)

#Left_join the data
PM2.5_PM10_Temp_sf_utm <- PM2.5_PM10_Temp_sf_utm %>%
left_join(aux3, by = c("Temp_Est"="NAME", "Date"="DATE", "COUNTY")) %>%
  select(Date,Site.ID,COUNTY,Population,Zone,PM2.5,PM10,TAVG,geometry)
```

Read the Electricity Generation via Combustion data.

```r
EPA_US_CombEmissions <- st_read("./Data/Raw/EPA_ElecGenComb_US_raw.kml")
```

```
## Reading layer `Electricity Generation via Combustion' from data source `C:\Users\Feli
## Simple feature collection with 2042 features and 2 fields
## geometry type:  POINT
## dimension:      XYZ
## bbox:           xmin: -176.6593 ymin: 19.63283 xmax: -67.00325 ymax: 71.29221
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
```

```r
st_crs(EPA_US_CombEmissions) #crs=4326 = WGS 84
```

```
## Coordinate Reference System:
##   EPSG: 4326
##   proj4string: "+proj=longlat +datum=WGS84 +no_defs"
```

```r
#Convert all to UTM Zone 17 (crs = 26917)
EPA_US_CombEmissions_utm <- st_transform(EPA_US_CombEmissions, c=26917)

#Clip the EPA_US_CombEmissions data set by the NC State boundary dataset

#First create a State_sf file
#Aggregate the data using group_by and summarize, just as you would a non-spatial data
state_sf_utm <- st_union(counties_sf_utm)

#Eliminate the emission points outside NC
EPA_NC_CombEmissions_utm <- st_intersection(EPA_US_CombEmissions_utm,state_sf_utm)
```

Distance between PM2.5 stations and Electricity Generation via Combustion points

```r
#Distances between PM2.5 stations and Electricity Generation via Combustion points
Distances <- st_distance(PM2.5_Stations, EPA_NC_CombEmissions_utm)

a <- length(unique(PM2.5_Stations$Site.ID))

#Determining the minimum distance of each PM2.5 station to a combustion point in meter
for (i in 1:a){
  PM2.5_Stations$Emiss_Dist[i] <- min(Distances[i,])
}

#Filling the PM2.5_PM10_Temp_sf_utm file with the distances

#Drops the geo data
aux4 <- PM2.5_Stations %>%
  st_set_geometry(value=NULL) %>%
  select(Site.ID,Emiss_Dist)

#Left_join the data
PM2.5_Full_utm <- PM2.5_PM10_Temp_sf_utm %>%
left_join(aux4, by = c("Site.ID")) %>%
  select(Date,Site.ID,COUNTY,Population,Zone,PM2.5,PM10,TAVG,Emiss_Dist,geometry)
```

Elevations for the PM2.5 Stations

```r
prj_dd <- "+proj=utm +zone=17 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs"
PM2.5_Full_Elev_utm <- get_elev_point(PM2.5_Full_utm, prj = prj_dd, src = "epqs")

st_write(PM2.5_Full_Elev_utm,
         "./Data/Processed/PM2.5_Full_Elev_utm.shp", driver = "ESRI Shapefile")
```

```r
PM2.5_Full_Elev_utm <- st_read('./Data/Processed/PM2.5_Full_Elev_utm.shp')
```

```
## Reading layer `PM2.5_Full_Elev_utm' from data source `C:\Users\Felipe\OneDrive - Duke
## Simple feature collection with 6499 features and 11 fields
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: 278314.3 ymin: 3807066 xmax: 935107.5 ymax: 3996722
## epsg (SRID):    NA
## proj4string:    +proj=utm +zone=17 +ellps=GRS80 +units=m +no_defs
```

The next chunk is to start looking at the data

```r
#Spatially
ggplot() +
  annotation_map_tile(zoom = 7) +
  geom_sf(data = counties_sf_utm_simple, color = 'Black', aes(fill=Zone), alpha=0.7) +
```

```r
  geom_sf(data = NOAA_DTAVG_NC2018_sf_utm, color = 'Blue') +
  geom_sf(data = EPA_NC_CombEmissions_utm, color = 'Black') +
  scale_fill_brewer(palette = "Blues", name = "Zoning")  +
  xlab(expression("Longitud")) +
  ylab(expression("Latitude")) +
  ggtitle("Data Visualization")

#Scatterplot
ggplot(NOAA_DTAVG_NC2018_sf_utm, aes(x=DATE, y=TAVG)) +
  geom_point(aes(color = ELEVATION)) +
  xlab(expression("Date")) +
  ylab(expression("Average Daily Temp °F")) +
  #labs(color = 'Station') +
  scale_color_gradient(low="cyan", high="blue4", name = "Elevation (m)")  +
  ggtitle("Average Temperature in North Carolina 2018")

ggplot(NOAA_DTAVG_NC2018_sf_utm, aes(x=DATE, y=TAVG)) +
  geom_point(aes(color = Zone)) +
  xlab(expression("Date")) +
  ylab(expression("Average Daily Temp °F")) +
  #labs(color = 'Station') +
  scale_color_brewer(palette = "Reds", name = "Zone")  +
  ggtitle("Average Temperature in North Carolina 2018")

PM2.5_Dist_Plot <- ggplot(PM2.5_PM10_sf_utm, aes(x=Date, y=PM2.5)) +
  geom_point(aes(color = Emiss_Dist)) +
  geom_smooth(color ="black") +
  xlab(expression("Date")) +
  ylab(expression("PM 2.5 Concentration (\U003BCg/m3)")) +
  scale_color_gradient(low="cyan", high="blue4", name = "Dist. to combustion point (m)")
  ggtitle("2018 Daily PM2.5 concentration, North Carolina") +
  geom_hline(yintercept=12, linetype="dashed", color = "black", size = 1)

PM2.5_Zone_Plot <- ggplot(PM2.5_PM10_sf_utm, aes(x=Date, y=PM2.5)) +
  geom_point(aes(color = Zone)) +
  geom_smooth(color ="black") +
  xlab(expression("Date")) +
  ylab(expression("PM 2.5 Concentration (\U003BCg/m3)")) +
  scale_fill_manual("red","blue","yellow")  +
  ggtitle("2018 Daily PM2.5 concentration, North Carolina") +
  geom_hline(yintercept=12, linetype="dashed", color = "black", size = 1)

PM10_Dist_Plot <- ggplot(PM2.5_PM10_sf_utm, aes(x=Date, y=PM10)) +
  geom_point(aes(color = Emiss_Dist)) +
```

```
  geom_smooth(color ="black") +
  xlab(expression("Date")) +
  ylab(expression("PM 10 Concentration (\U003BCg/m3)")) +
  scale_color_gradient(low="cyan", high="blue4", name = "Dist. to combustion point (m)")
  ggtitle("2018 Daily PM10 concentration, North Carolina")

grid.arrange(PM2.5_Dist_Plot, PM10_Dist_Plot, PM2.5_Zone_Plot, nrow = 3)

ggplot(PM2.5_PM10_Temp_sf_utm_elev, aes(x=Date, y=PM2.5)) +
  geom_point(aes(color = elevation)) +
  geom_smooth(color ="black") +
  xlab(expression("Date")) +
  ylab(expression("PM 10 Concentration (\U003BCg/m3)")) +
  scale_color_gradient(low="cyan", high="blue4", name = "Elev. (m)")  +
  ggtitle("2018 Daily PM10 concentration, North Carolina")

ggplot(PM2.5_PM10_Temp_sf_utm_elev, aes(x=Date, y=PM2.5)) +
  geom_point(aes(color = TAVG)) +
  geom_smooth(color ="black") +
  xlab(expression("Date")) +
  ylab(expression("PM 10 Concentration (\U003BCg/m3)")) +
  scale_color_gradient(low="cyan", high="blue4", name = "Temp (°F)")  +
  ggtitle("2018 Daily PM10 concentration, North Carolina")




#PM 2.5 Regulatory Standard. Based on a yearly average value set at 12 micrograms per

#Plot counties with a new variable (zone)
ggplot() +
  geom_sf(data = counties_sf_utm_simple, aes(fill=Zone))






summary(PM2.5_Stations$Emiss_Dist)
```

4 Analysis

In 2012, the United States Environmental Protection Agency (USEPA) established two complementary primary regulatory standards for PM2.5. The first is based on a yearly average value and is set at 12 micrograms per cubic meter, ug/m3,

First statistical test should look at the standard in each station.

FRA: This model structure should look familiar, with a typical linear model structure and dataframe defined. The addition here is that we have defined Week as a random variable. Essentially, we are interested not in the specific effects of each week but in the variability among weeks, so we have defined it as a random effect (essentially coming from a larger distribution of seasonal variability).

5 Summary and Conclusions