S2208 MATH8050 Data Analysis: Homework 1 Due on 09/07/22

Adithya Ravi, C09059838

2022-09-06

Solutions

Question1

1a

```
getwd()
```

```
## [1] "C:/Users/ravia/OneDrive/Documents"
```

```
rain.df <- read.table("rnf6080.dat")
head(rain.df)
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 1 60  4  1  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
## 2 60  4  2  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
## 3 60  4  3  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
## 4 60  4  4  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
## 5 60  4  5  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
## 6 60  4  6  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
##   V22 V23 V24 V25 V26 V27
## 1   0   0   0   0   0   0
## 2   0   0   0   0   0   0
## 3   0   0   0   0   0   0
## 4   0   0   0   0   0   0
## 5   0   0   0   0   0   0
## 6   0   0   0   0   0   0
```

1b

```
nrow(rain.df)
```

```
## [1] 5070
```

```
ncol(rain.df)
```

```
## [1] 27
```

1c

```
colnames(rain.df)
```

```
##  [1] "V1"  "V2"  "V3"  "V4"  "V5"  "V6"  "V7"  "V8"  "V9"  "V10" "V11" "V12"
## [13] "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23" "V24"
## [25] "V25" "V26" "V27"
```

1d

```
rain.df[2,4]
```

```
## [1] 0
```

1e

```
rain.df[2,]
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 2 60  4  2  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
##   V22 V23 V24 V25 V26 V27
## 2   0   0   0   0   0   0
```

1f

```
names(rain.df) <- c("year","month","day",seq(0,23))

#head(rain.df)
#tail(rain.df)
#rain.df[20,]
```

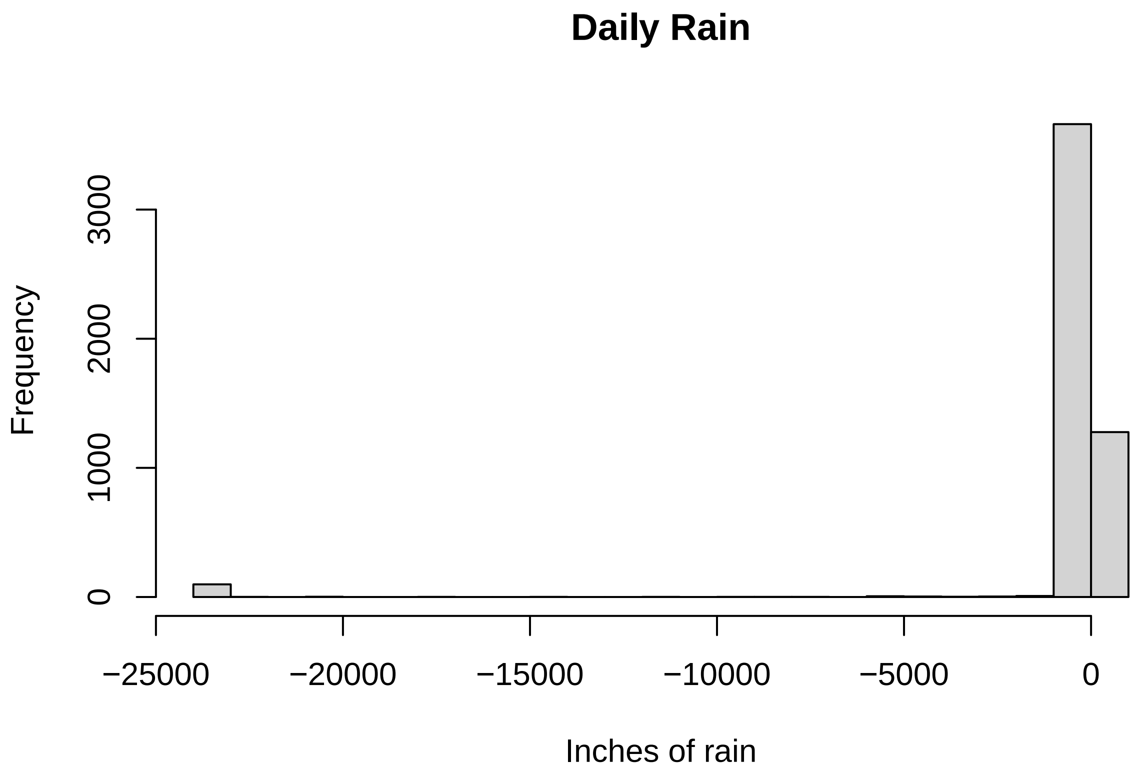
Executing this line of code on the dataframe adds the column names and from the 4th to 27th column, as it represents the hours of the day, the column names are 0 through 23.

1g

```
rain.df$daily <- apply(rain.df[,c(4:27)], 1, function(x) sum(x))
```

1h

```
hist(rain.df[,28], breaks=25, xlab ="Inches of rain", main = "Daily Rain")
```



1i

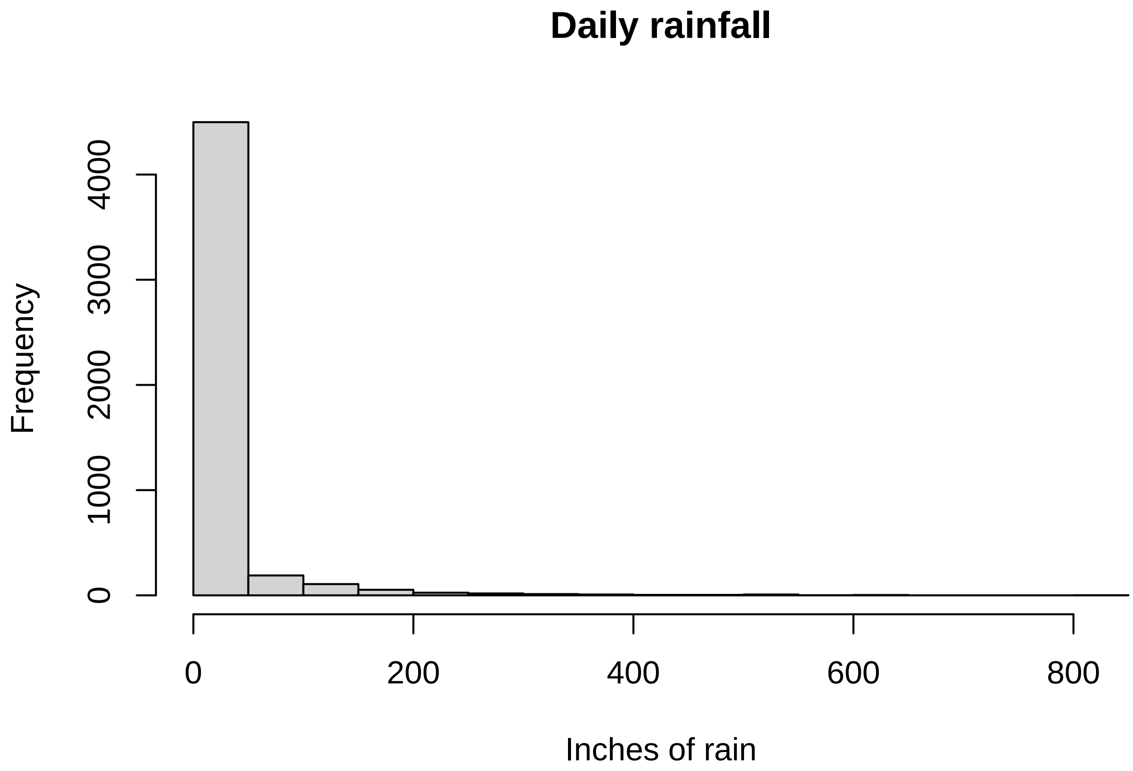
The above generated histogram is erroneous because it generate the histogram in the negative quadrant and the rainfall measure cant be negative.

1j

```
rain.df.fixed <- rain.df
is.na(rain.df.fixed) <- rain.df.fixed < 0
```

1k

```
hist(rain.df.fixed[,28], breaks = 25, xlab = "Inches of rain" , main = "Daily rainfall")
```



The histogram has been fixed by removing all the N/A values and arranging it properly.

Question2

2a

```
#vector1 <- c("5", "12", "7", "32")
#max(vector1)
#sort(vector1)
#sum(vector1)
```

The error with the max() statement is, since the numbers are put in quotation marks, they are treated as characters and hence are arranged lexicographically. When executing the sort() statement, since the values are characters and not numeric - they are sorted based on their first values i.e 12, 32, 5, 7 and not 5, 7, 12, 32. Finally the sum() statement doesn't get executed because it just accepts numeric values and returns the invalid 'type' error.

2b

```
#y <- c("5",7,12)
#y[2] + y[3]
```

When executing the adding of vector elements, it returns invalid type error because when initializing the y vector, the 1st element is a character and the other two elements are integers. Vectors doesnt accept data elements of different types and hence it converts the integers to characters and addition of characters is not possible.

2c

```
z <- data.frame(z1="5",z2=7,z3=12)
z[1,2] + z[1,3]
```

```
## [1] 19
```

The variable 'z' is assigned with a dataframe with 1 row, 3 columns with the mentioned values. The second line is an addition of the element in the first row second column and first row third column, which are integers and when added it returns the result 19.

Question3

3a

```
mysolver <- function(A,b) {
  X <- solve(A,b)
  return (X)
}
```

3b

```
n = 100
set.seed(123)
A = rWishart(1, 150, diag(n))[ , ,1]
b = rnorm(n,1)
mysolver(A,b)
```

```
##   [1]  0.0304904446  0.0513126996  0.0154465983  0.0880140982  0.0206193282
##   [6] -0.0188615518  0.0154455967 -0.0114977415  0.0154554764  0.0013277842
##  [11] -0.0144490153  0.0117708385 -0.0365008032 -0.0118364385 -0.0268731752
##  [16]  0.0088594658  0.0342928779 -0.0373493390  0.0755887348  0.0334924362
##  [21]  0.0643525626  0.0329045990  0.0538493081  0.0584607262  0.0132776492
##  [26]  0.0591025796  0.0386174543  0.0291968521  0.0155404267  0.0300184430
## [31]  0.0479344631  0.0181094963  0.0137618715  0.0020543864  0.0329946099
## [36]  0.0700977751 -0.0103798766 -0.0607553014  0.0060087534  0.0045095703
## [41]  0.0406824191 -0.0126976864  0.0413099886  0.0497211390  0.0136004108
## [46]  0.0168871768 -0.0337439148 -0.0045213855 -0.0246322197  0.0270570372
## [51] -0.0115487391  0.0078318445 -0.0065892127  0.0342191375  0.0044861429
## [56] -0.0015665577  0.0077317216 -0.0139819360  0.0114902547 -0.0002086674
## [61]  0.0195829631 -0.0103309045 -0.0089335358  0.0014125713  0.0256850181
## [66]  0.0417217201  0.0262774133  0.0098425096  0.0336859799 -0.0513870724
## [71] -0.0411449266  0.0133773333 -0.0198858733  0.0443756387 -0.0230688916
## [76]  0.0264735075  0.0315870253  0.0334322871  0.0235810646  0.0072452152
## [81]  0.0430647635  0.0599169758 -0.0185225796  0.0102451144 -0.0321602843
## [86] -0.0163589049  0.0715600447  0.0935536826  0.0038482350  0.0524514264
## [91]  0.0072533977 -0.0389588986  0.0534358539  0.0236271251  0.0689938328
## [96]  0.0206482494  0.0022481242  0.0497799362  0.0446726451  0.0007246325
```

Question4

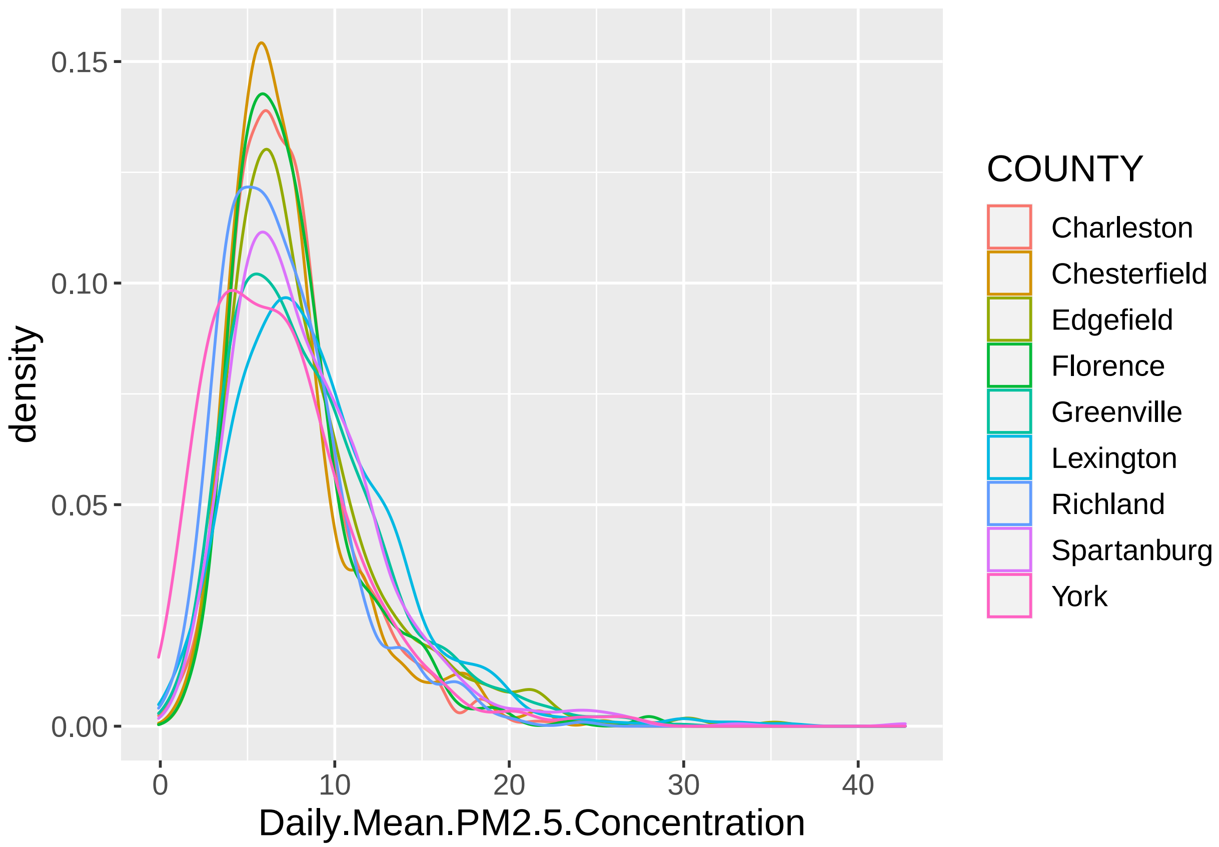
4a

```
data <- read.csv("AQSdata.csv")
```

4b

```
library(ggplot2)

ggplot(data, aes(x = Daily.Mean.PM2.5.Concentration, colour = COUNTY)) + geom_density() + theme(text = e
```

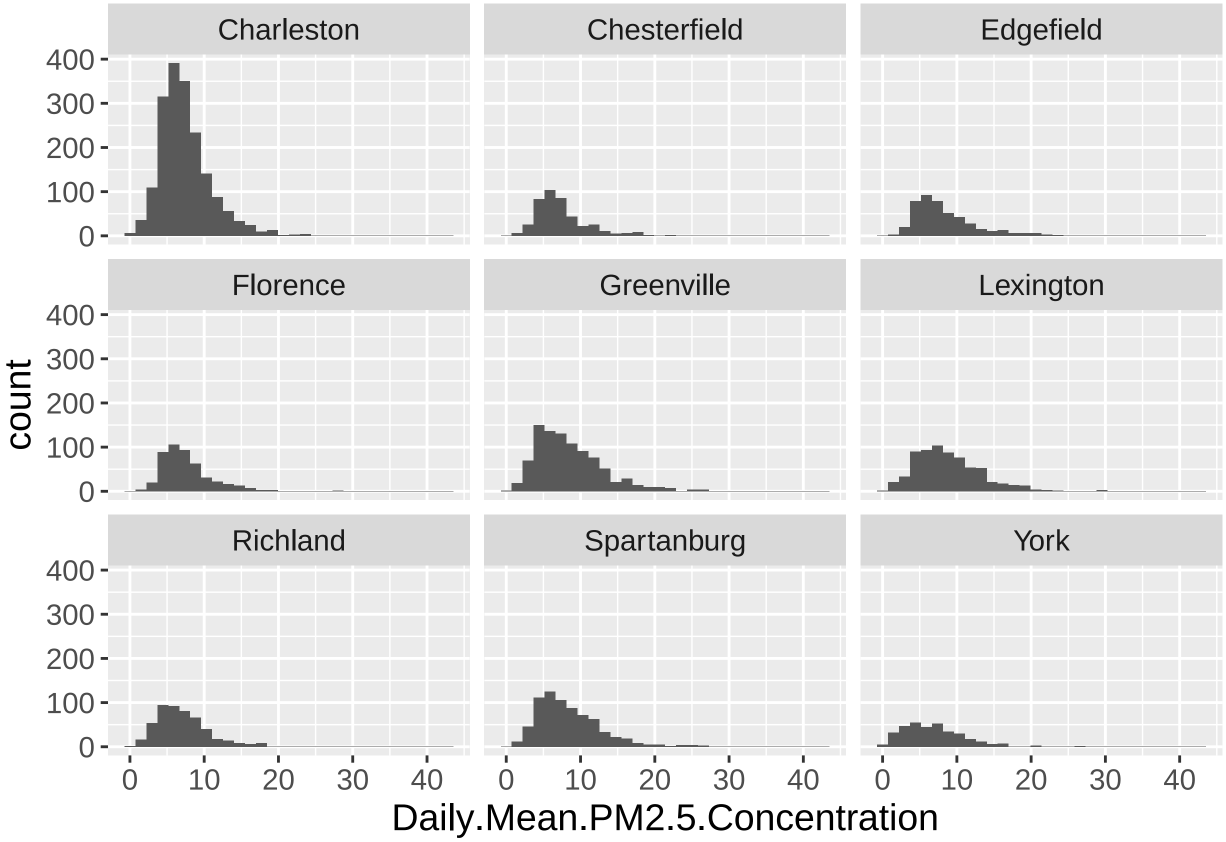


From the graph it can be inferred that at the range 0-10, especially at Chesterfield and Charleston, the graph has peaked which is evident that the PM2.5 density is great and the pollution is worse compared to any other zones.

4c

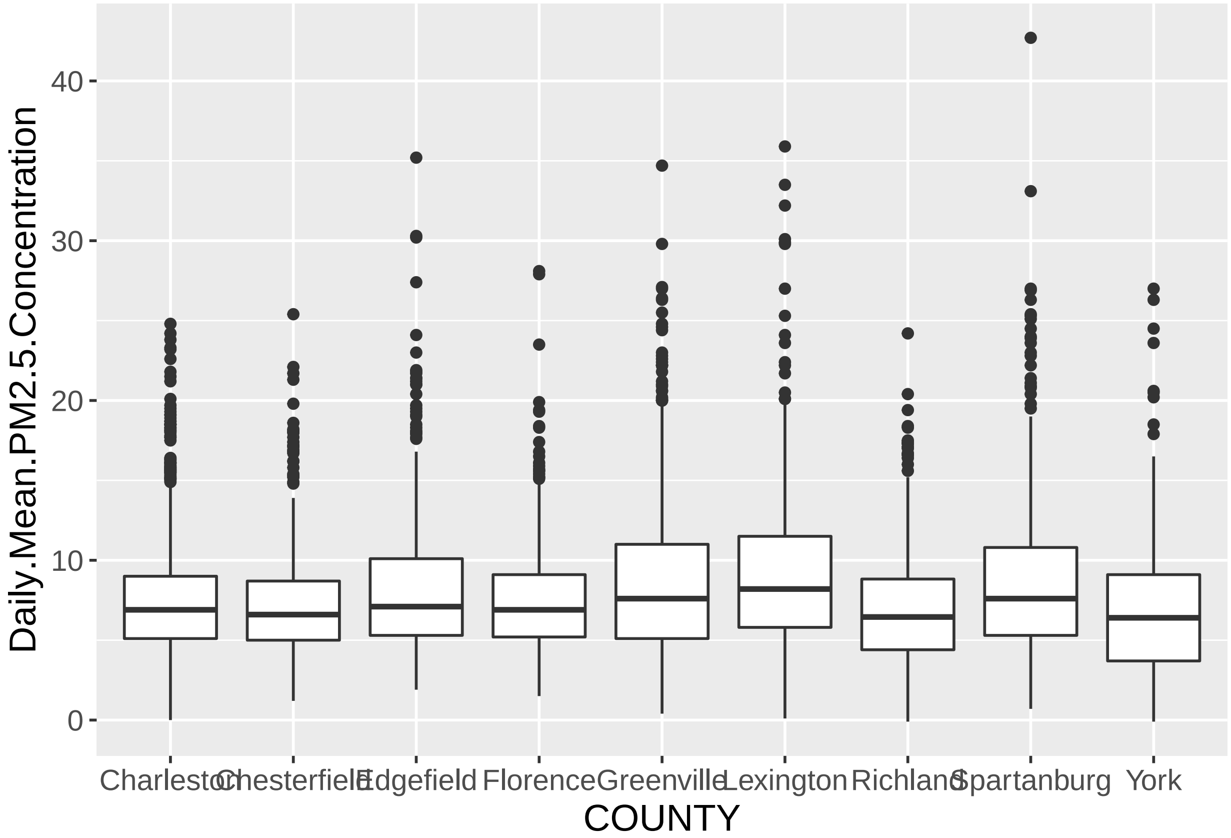
```
ggplot(data, aes(x = Daily.Mean.PM2.5.Concentration)) + geom_histogram() + facet_wrap('COUNTY') + theme
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



4d

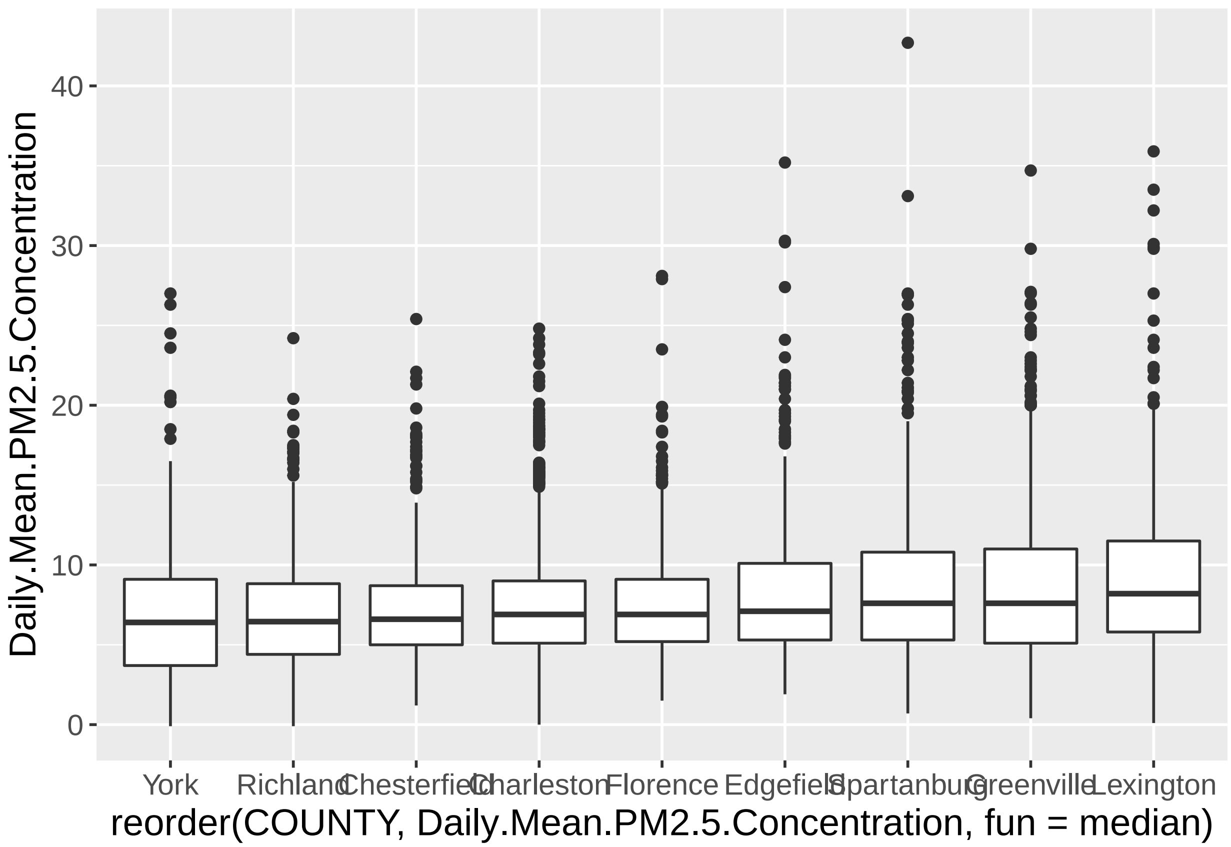
```
ggplot(data, aes(x = COUNTY, y = Daily.Mean.PM2.5.Concentration)) + geom_boxplot() + theme(text = elemen
```



Out of all the County's, Greenville and Lexington has the most interquartile range. The overall range of PM2.5 is greater for Spartanburg if the outlier is included

4e

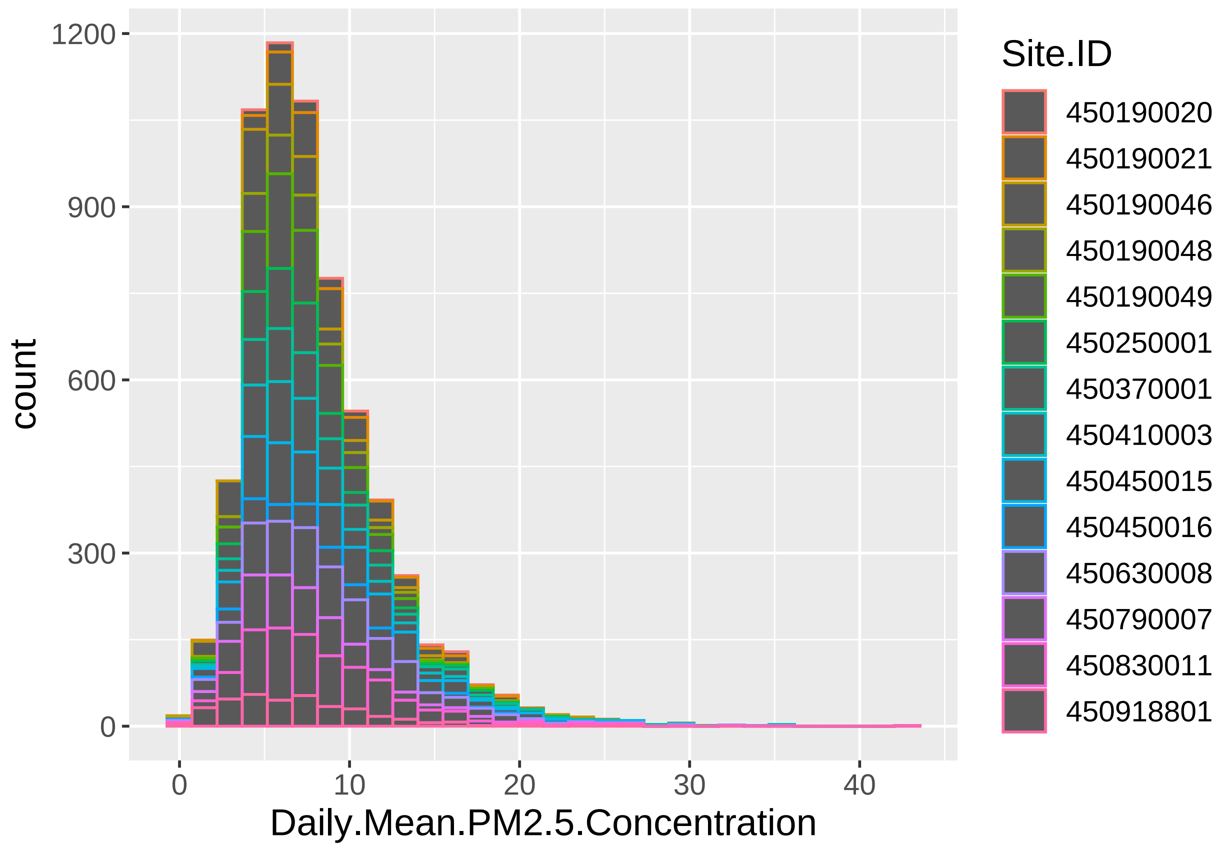
```
ggplot(data, aes(x =reorder(COUNTY, Daily.Mean.PM2.5.Concentration, fun = median), y = Daily.Mean.PM2.5
```



4f

```
data[,"Site.ID"] <- as.factor(as.numeric(data[, "Site.ID"]))
ggplot(data, aes(x= Daily.Mean.PM2.5.Concentration, color = Site.ID)) + geom_histogram() + theme(text =
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```

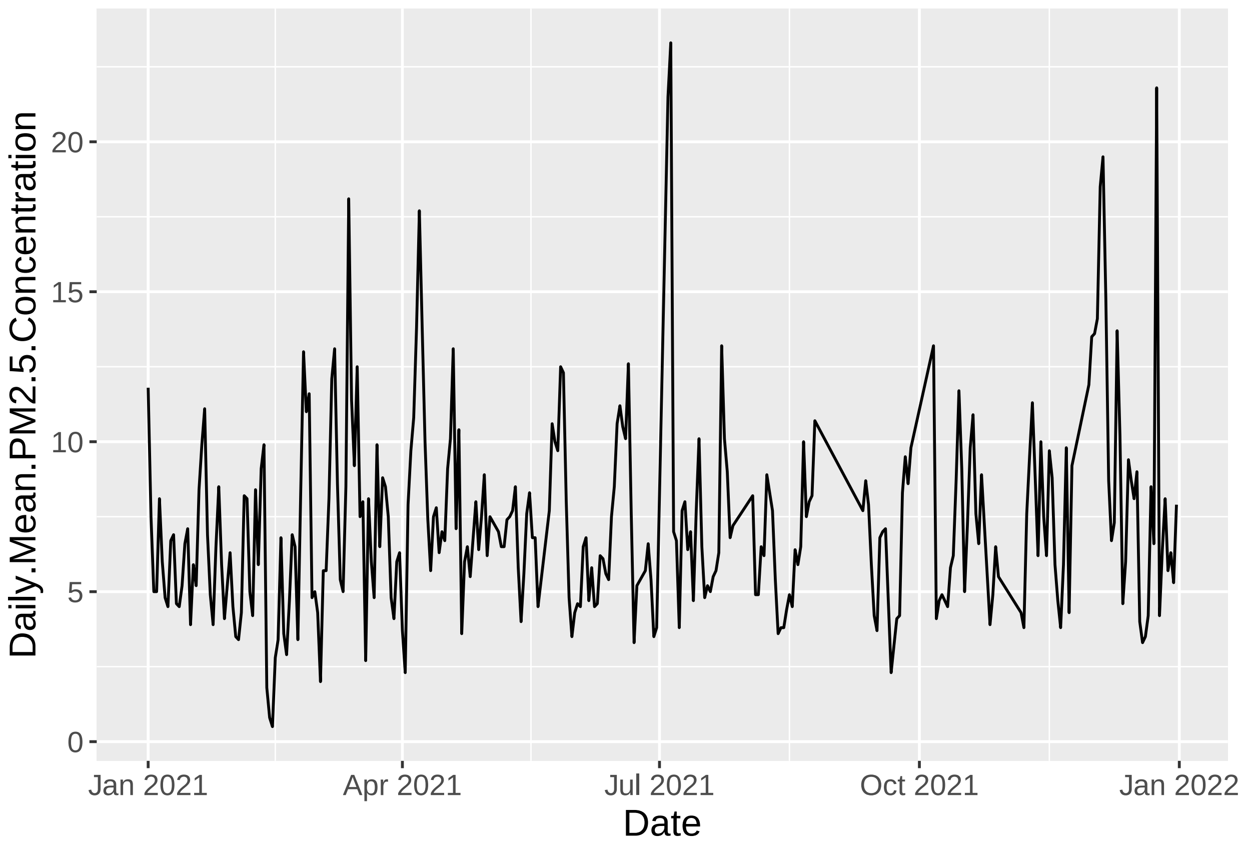


4g

```
library(tidyverse)
```

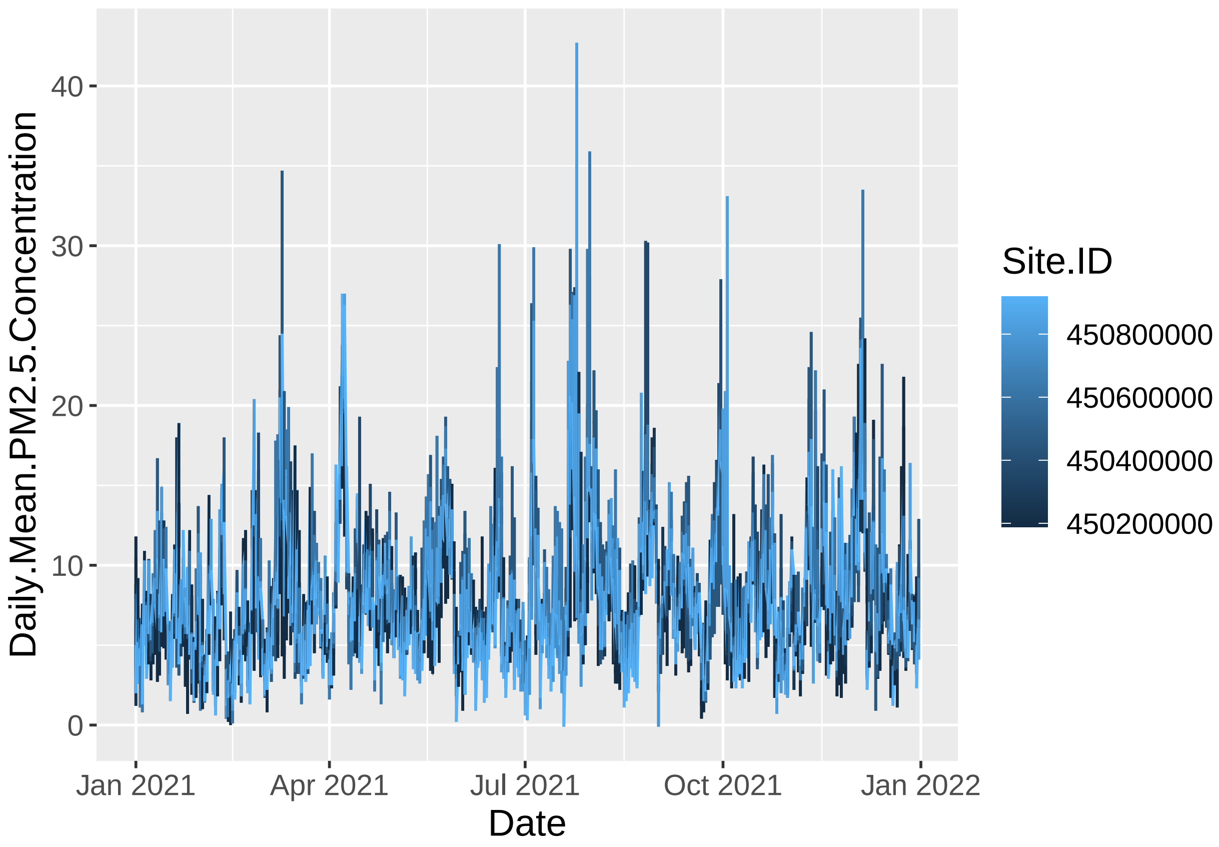
```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.7      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## v purrr   0.3.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
new_data_3=read.csv("AQSdata.csv")
new_data_3$Date<-as.Date(new_data_3$Date,format="%m/%d/%Y")
ggplot(filter(new_data_3, Site.ID == "450190048"), aes(Date,Daily.Mean.PM2.5.Concentration)) + geom_lin
```



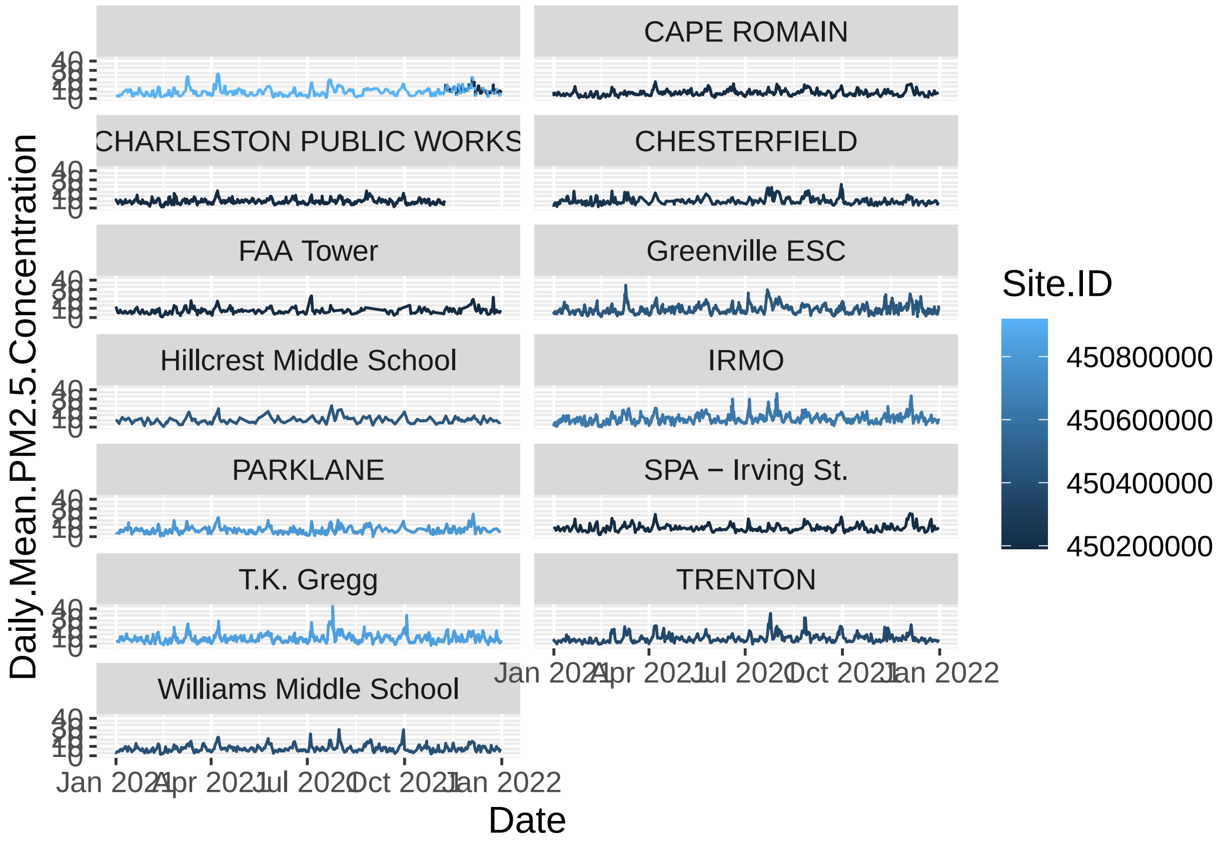
4h

```
new_data_3=read.csv("AQSdata.csv")
new_data_3$Date<-as.Date(new_data_3$Date,format="%m/%d/%Y")
ggplot(new_data_3, aes(Date,Daily.Mean.PM2.5.Concentration,color=Site.ID))+geom_line() + theme(text = el
```



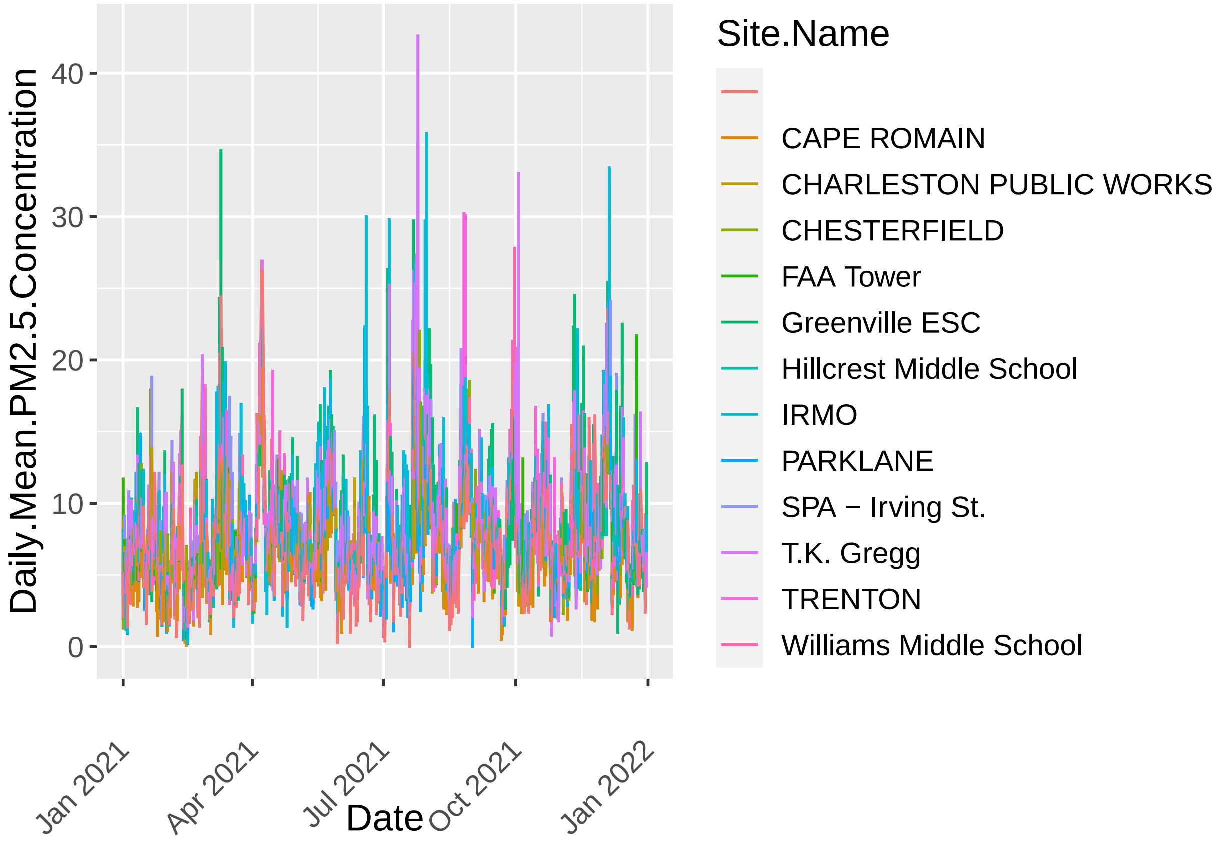
4i

```
ggplot(new_data_3, aes(x= Date, y=Daily.Mean.PM2.5.Concentration, col = Site.ID, group = 1)) + geom_lin
```

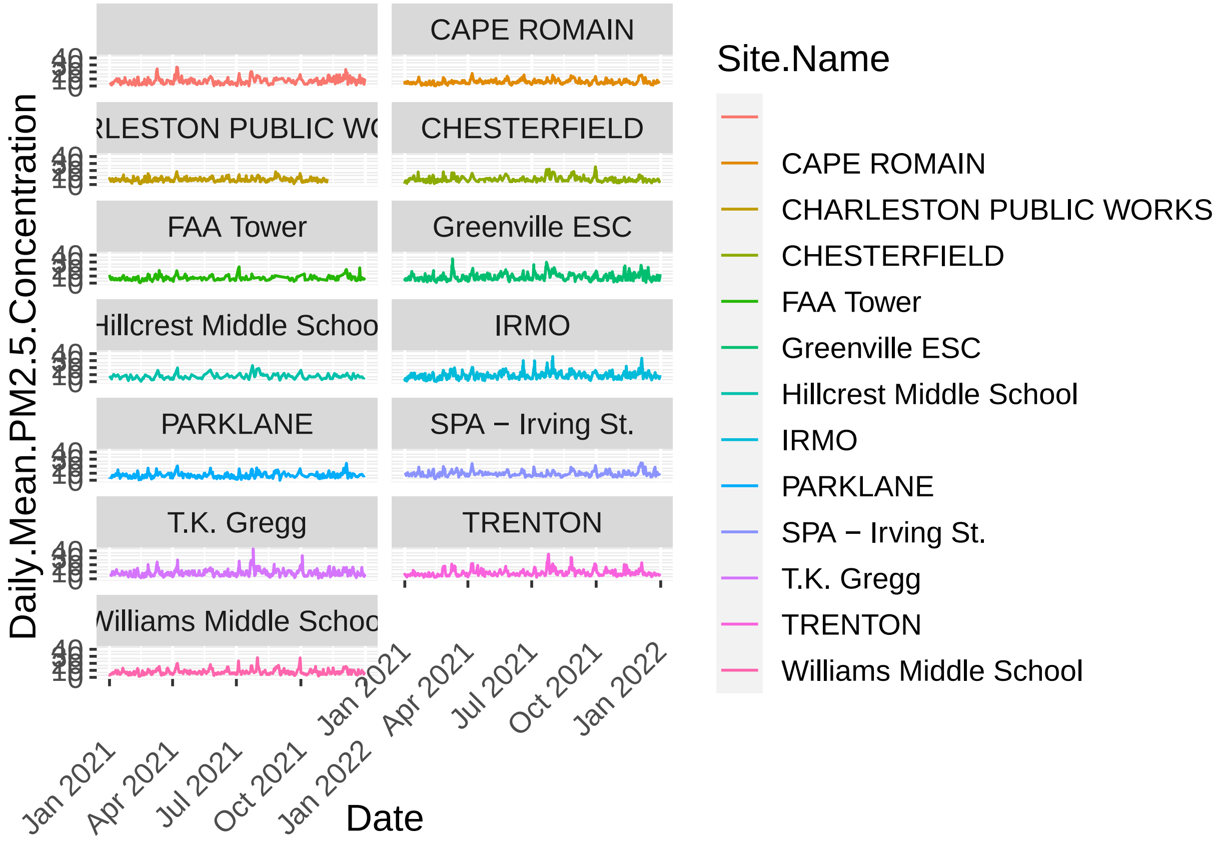


4j

```
ggplot(new_data_3, aes(x=Date, y=Daily.Mean.PM2.5.Concentration, col=Site.Name, group = 1)) + geom_line
```



```
ggplot(new_data_3, aes(x=Date, y=Daily.Mean.PM2.5.Concentration, col=Site.Name, group = 1)) + geom_line
```



Question5

5a

```
library(dplyr)

new_data1 <- filter(data, COUNTY == 'Greenville')
head(new_data1)
```

```
##         Date Source   Site.ID POC Daily.Mean.PM2.5.Concentration    UNITS
## 1 01/01/2021    AQS 450450015   1                            6.3 ug/m3 LC
## 2 01/02/2021    AQS 450450015   1                            6.6 ug/m3 LC
## 3 01/03/2021    AQS 450450015   1                            5.4 ug/m3 LC
## 4 01/05/2021    AQS 450450015   1                            7.4 ug/m3 LC
## 5 01/06/2021    AQS 450450015   1                            7.7 ug/m3 LC
## 6 01/07/2021    AQS 450450015   1                            9.5 ug/m3 LC
##   DAILY_AQI_VALUE      Site.Name DAILY_OBS_COUNT PERCENT_COMPLETE
## 1              26 Greenville ESC               1              100
## 2              28 Greenville ESC               1              100
## 3              23 Greenville ESC               1              100
## 4              31 Greenville ESC               1              100
## 5              32 Greenville ESC               1              100
## 6              40 Greenville ESC               1              100
##   AQS_PARAMETER_CODE       AQS_PARAMETER_DESC CBSA_CODE
## 1              88101 PM2.5 - Local Conditions     24860
## 2              88101 PM2.5 - Local Conditions     24860
## 3              88101 PM2.5 - Local Conditions     24860
## 4              88101 PM2.5 - Local Conditions     24860
## 5              88101 PM2.5 - Local Conditions     24860
## 6              88101 PM2.5 - Local Conditions     24860
##                         CBSA_NAME STATE_CODE          STATE COUNTY_CODE
## 1 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 2 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 3 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 4 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 5 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 6 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
##       COUNTY SITE_LATITUDE SITE_LONGITUDE
## 1 Greenville       34.8439      -82.41458
## 2 Greenville       34.8439      -82.41458
## 3 Greenville       34.8439      -82.41458
## 4 Greenville       34.8439      -82.41458
## 5 Greenville       34.8439      -82.41458
## 6 Greenville       34.8439      -82.41458
```

It displays a total of 937 observations when the dataset is filtered by COUNTY == “Greenville”.

5b

```
new_data2 <- filter(data, COUNTY == 'Greenville' & Date > '08-01-2021' & Date < '09-01-2021')

head(new_data2)
```

```
##         Date Source   Site.ID POC Daily.Mean.PM2.5.Concentration    UNITS
## 1 08/01/2021    AQS 450450015   1                           13.8 ug/m3 LC
## 2 08/02/2021    AQS 450450015   1                           19.0 ug/m3 LC
## 3 08/03/2021    AQS 450450015   1                           16.9 ug/m3 LC
## 4 08/04/2021    AQS 450450015   1                           15.6 ug/m3 LC
## 5 08/05/2021    AQS 450450015   1                           11.0 ug/m3 LC
## 6 08/06/2021    AQS 450450015   1                           10.3 ug/m3 LC
##   DAILY_AQI_VALUE      Site.Name DAILY_OBS_COUNT PERCENT_COMPLETE
## 1              55 Greenville ESC               1              100
## 2              66 Greenville ESC               1              100
## 3              61 Greenville ESC               1              100
## 4              58 Greenville ESC               1              100
## 5              46 Greenville ESC               1              100
## 6              43 Greenville ESC               1              100
##   AQS_PARAMETER_CODE       AQS_PARAMETER_DESC CBSA_CODE
## 1              88101 PM2.5 - Local Conditions     24860
## 2              88101 PM2.5 - Local Conditions     24860
## 3              88101 PM2.5 - Local Conditions     24860
## 4              88101 PM2.5 - Local Conditions     24860
## 5              88101 PM2.5 - Local Conditions     24860
## 6              88101 PM2.5 - Local Conditions     24860
##                         CBSA_NAME STATE_CODE          STATE COUNTY_CODE
## 1 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 2 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 3 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 4 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 5 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 6 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
##       COUNTY SITE_LATITUDE SITE_LONGITUDE
## 1 Greenville       34.8439      -82.41458
## 2 Greenville       34.8439      -82.41458
## 3 Greenville       34.8439      -82.41458
## 4 Greenville       34.8439      -82.41458
## 5 Greenville       34.8439      -82.41458
## 6 Greenville       34.8439      -82.41458
```

5c

```
new_data <- filter(data, COUNTY == 'Greenville' & Date > '08-01-2021' & Date < '09-01-2021')

head(new_data)
```

```
##         Date Source   Site.ID POC Daily.Mean.PM2.5.Concentration    UNITS
## 1 08/01/2021    AQS 450450015   1                           13.8 ug/m3 LC
## 2 08/02/2021    AQS 450450015   1                           19.0 ug/m3 LC
## 3 08/03/2021    AQS 450450015   1                           16.9 ug/m3 LC
## 4 08/04/2021    AQS 450450015   1                           15.6 ug/m3 LC
## 5 08/05/2021    AQS 450450015   1                           11.0 ug/m3 LC
## 6 08/06/2021    AQS 450450015   1                           10.3 ug/m3 LC
##   DAILY_AQI_VALUE      Site.Name DAILY_OBS_COUNT PERCENT_COMPLETE
## 1              55 Greenville ESC               1              100
## 2              66 Greenville ESC               1              100
## 3              61 Greenville ESC               1              100
## 4              58 Greenville ESC               1              100
## 5              46 Greenville ESC               1              100
## 6              43 Greenville ESC               1              100
##   AQS_PARAMETER_CODE       AQS_PARAMETER_DESC CBSA_CODE
## 1              88101 PM2.5 - Local Conditions     24860
## 2              88101 PM2.5 - Local Conditions     24860
## 3              88101 PM2.5 - Local Conditions     24860
## 4              88101 PM2.5 - Local Conditions     24860
## 5              88101 PM2.5 - Local Conditions     24860
## 6              88101 PM2.5 - Local Conditions     24860
##                         CBSA_NAME STATE_CODE          STATE COUNTY_CODE
## 1 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 2 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 3 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 4 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 5 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
## 6 Greenville-Anderson-Mauldin, SC         45 South Carolina          45
##       COUNTY SITE_LATITUDE SITE_LONGITUDE
## 1 Greenville       34.8439      -82.41458
## 2 Greenville       34.8439      -82.41458
## 3 Greenville        34.8439       -82.41458
## 4 Greenville        34.8439       -82.41458
## 5 Greenville        34.8439       -82.41458
## 6 Greenville        34.8439       -82.41458
```

```
# head(select(new_data, Daily.Mean.PM2.5.Concentration, Date, SITE_LATITUDE, SITE_LONGITUDE))
```

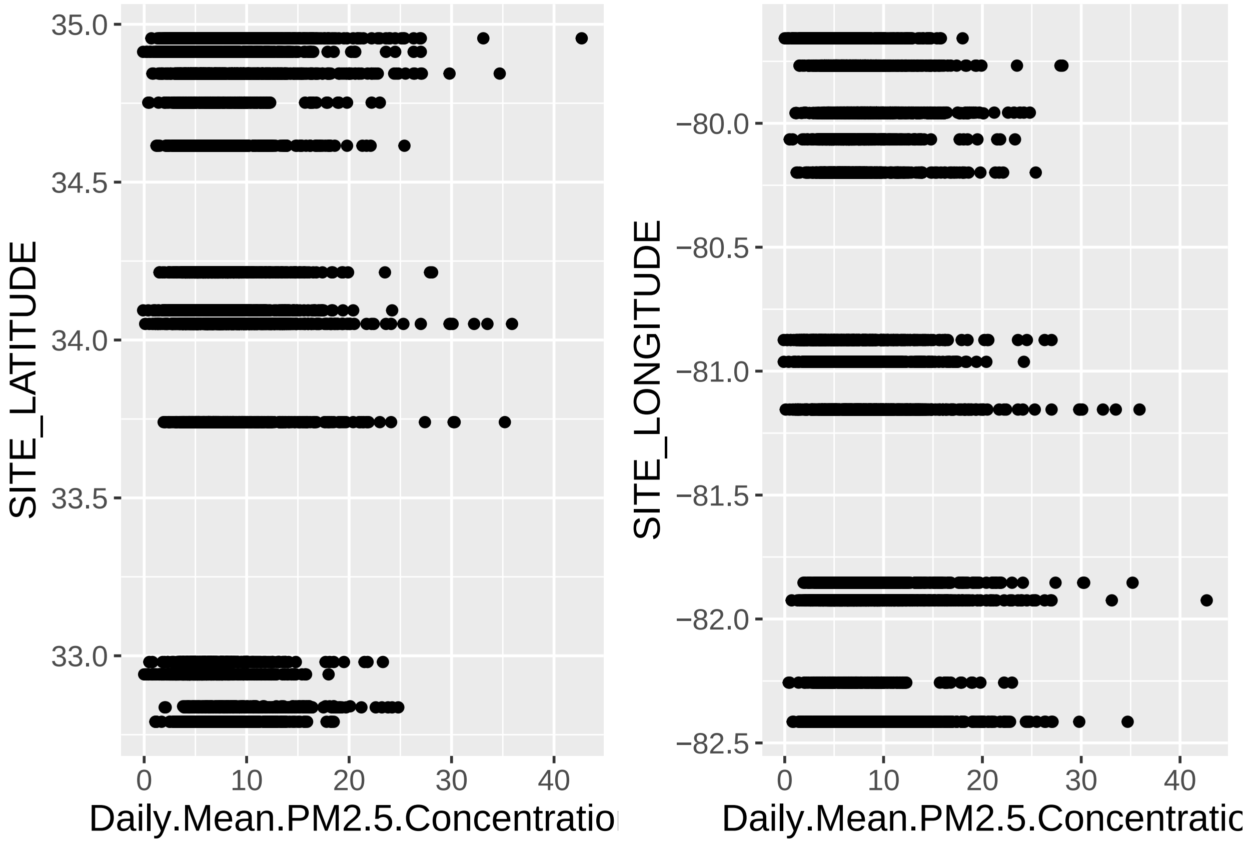
5d

```
library(ggpubr)

plot1 <- ggplot(data, aes(Daily.Mean.PM2.5.Concentration,SITE_LATITUDE)) +geom_point() + theme(text = el

plot2 <- ggplot(data, aes(Daily.Mean.PM2.5.Concentration,SITE_LONGITUDE)) +geom_point() + theme(text = e

ggarrange(plot1, plot2, ncol = 2, nrow = 1)
```



Question6

6a

If you provide a reproducible example, anybody may duplicate your problem by simply copying and pasting R code. To make your example repeatable, you must provide the following four components: the data, the code, the relevant packages, and a description of your R environment.

6b

The entire purpose of producing code reproducibility is to ensure that the code is correct. Six months from now, you'll be able to figure out what you did. You can make changes to the code or data at any time during the process and re-run any analyses. When you're ready to publish, you may perform a last double-check of your whole study, from cleaning the raw data to creating figures and tables for the publication. A project can be passed on or shared with others. People who wish to expand on your study might benefit from coding samples.

6c

Given that it is really lengthy and involves a number of subproblems, I would rate this assignment a 10/10 on a scale of 1 to 10. Additionally, it takes a lot of time, which interferes with the timeline set for doing assignments for other classes. Because the assignment deadline is at 12pm, as opposed to other courses where we are given time until 11:59pm on the day it is due, it is significantly more difficult to manage time for other classes.