# Assignment 7: Time Series Analysis

Sara Sayed

## OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on time series analysis.

### 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_A07_TimeSeries.Rmd") prior to submission.

The completed exercise is due on Tuesday, March 16 at 11:59 pm.

### Set up

1. Set up your session:

- Check your working directory
- Load the tidyverse, lubridate, zoo, and trend packages
- Set your ggplot theme

2. Import the ten datasets from the Ozone_TimeSeries folder in the Raw data folder. These contain ozone concentrations at Garinger High School in North Carolina from 2010-2019 (the EPA air database only allows downloads for one year at a time). Import these either individually or in bulk and then combine them into a single dataframe named `GaringerOzone` of 3589 observation and 20 variables.

```
#1

library(tidyverse)
library(dplyr)
library(lubridate)
library(zoo)
library(trend)

mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)

#2

Ozone2010 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv")
Ozone2011 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv")
Ozone2012 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv")
```

```
Ozone2013 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv")
Ozone2014 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv")
Ozone2015 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv")
Ozone2016 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv")
Ozone2017 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv")
Ozone2018 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv")
Ozone2019 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv")

Ozone_Concentrations <- rbind(Ozone2010, Ozone2011, Ozone2012, Ozone2013, Ozone2014,
                              Ozone2015,Ozone2016, Ozone2017, Ozone2018, Ozone2019)
```

## Wrangle

3. Set your date column as a date class.

4. Wrangle your dataset so that it only contains the columns Date, Daily.Max.8.hour.Ozone.Concentration, and DAILY_AQI_VALUE.

5. Notice there are a few days in each year that are missing ozone concentrations. We want to generate a daily dataset, so we will need to fill in any missing days with NA. Create a new data frame that contains a sequence of dates from 2010-01-01 to 2019-12-31 (hint: `as.data.frame(seq())`). Call this new data frame Days. Rename the column name in Days to "Date".

6. Use a `left_join` to combine the data frames. Specify the correct order of data frames within this function so that the final dimensions are 3652 rows and 3 columns. Call your combined data frame GaringerOzone.

```
# 3

Ozone_Concentrations$Date <- as.Date(Ozone_Concentrations$Date,format = "%m/%d/%Y")

# 4

Ozone_Concentrations_AQI <-Ozone_Concentrations %>%
  select(Date, Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)

summary(Ozone_Concentrations_AQI)
```

```
##       Date            Daily.Max.8.hour.Ozone.Concentration DAILY_AQI_VALUE
##  Min.   :2010-01-01   Min.   :0.00200                      Min.   :  2.00
##  1st Qu.:2012-07-03   1st Qu.:0.03200                      1st Qu.: 30.00
##  Median :2015-01-04   Median :0.04100                      Median : 38.00
##  Mean   :2015-01-01   Mean   :0.04163                      Mean   : 41.57
##  3rd Qu.:2017-07-02   3rd Qu.:0.05100                      3rd Qu.: 47.00
##  Max.   :2019-12-31   Max.   :0.09300                      Max.   :169.00
```

```
# 5

Days <-as.data.frame(seq(as.Date('2010-01-01'), by = 'day', length.out = 3652))
colnames(Days) <-c("Date")

# 6

GaringerOzone <- left_join(Days, Ozone_Concentrations_AQI)
```

```
## Joining, by = "Date"
```

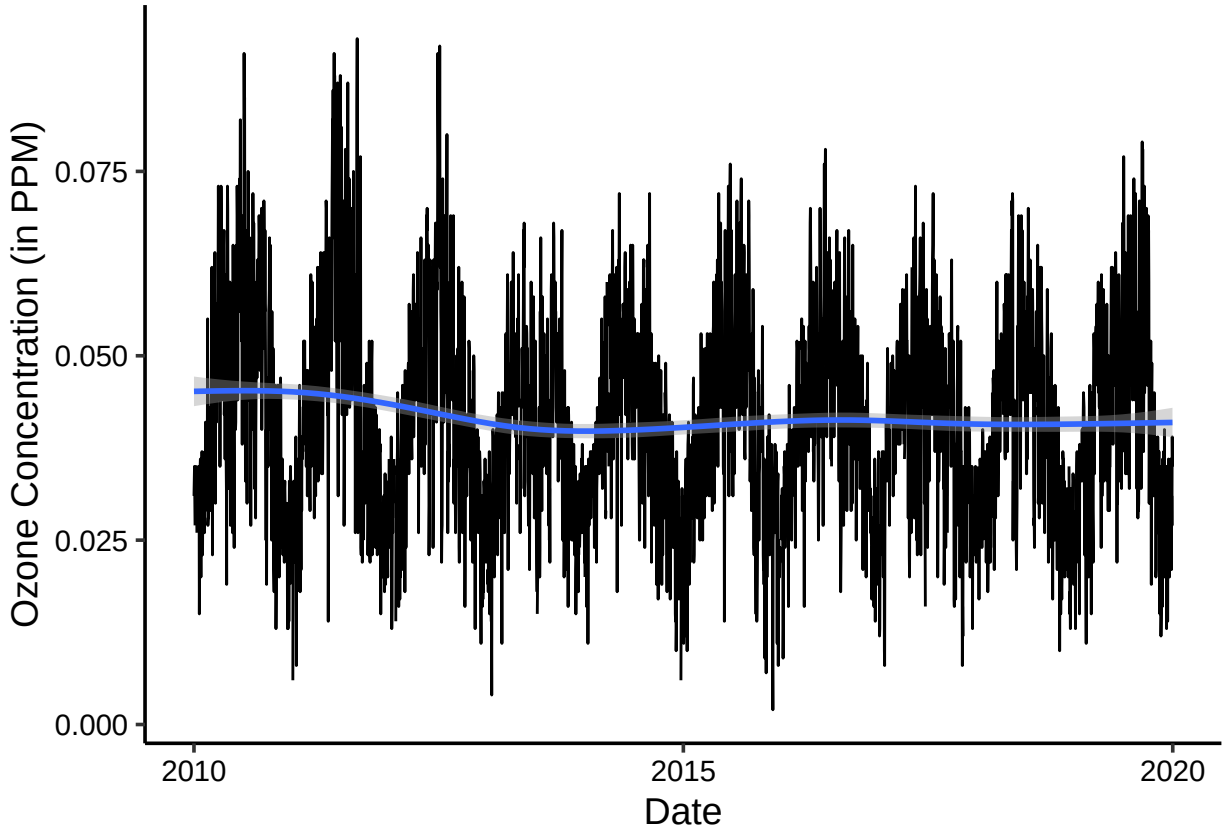
## Visualize

7. Create a line plot depicting ozone concentrations over time. In this case, we will plot actual concentrations in ppm, not AQI values. Format your axes accordingly. Add a smoothed line showing any linear trend of your data. Does your plot suggest a trend in ozone concentration over time?

```
#7

GaringerOzone.Time <- ggplot(GaringerOzone,
                             aes(x = Date, y = Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line() + geom_smooth() +
  labs(x = "Date", y = "Ozone Concentration (in PPM)")
print(GaringerOzone.Time)
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 63 rows containing non-finite values (stat_smooth).
```




Answer:It shows a slight downward trend, but generally the trend remains the same.

## Time Series Analysis

Study question: Have ozone concentrations changed over the 2010s at this station?

8. Use a linear interpolation to fill in missing daily data for ozone concentration. Why didn't we use a piecewise constant or spline interpolation?

```
#8

GaringerOzone$Daily.Max.8.hour.Ozone.Concentration <-
  na.approx(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration)
```

Answer: Since the data uses dates, it is assumed for each point there will be a point following the prior one. By using linear interpolation, we are essentially connecting the dates and filling in for the dates that have missing datapoints.

9. Create a new data frame called `GaringerOzone.monthly` that contains aggregated data: mean ozone concentrations for each month. In your pipe, you will need to first add columns for year and month to form the groupings. In a separate line of code, create a new Date column with each month-year combination being set as the first day of the month (this is for graphing purposes only)

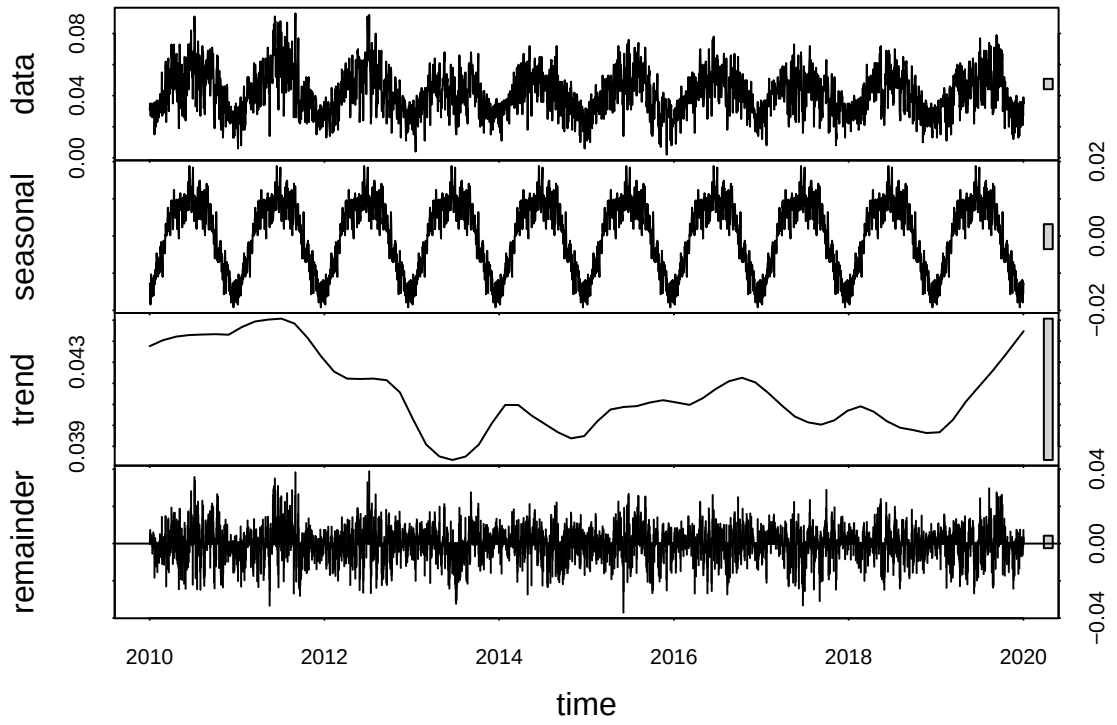
```
#9

GaringerOzone.monthly <-
  GaringerOzone %>%
   mutate(Month = as.yearmon(Date, "%m/%Y"))%>%
   group_by(Month) %>%
   summarise(Mean.Ozone=mean(Daily.Max.8.hour.Ozone.Concentration))

GaringerOzone.monthly$Date <-as.Date(GaringerOzone.monthly$Month, format= "%b %Y")
```

10. Generate two time series objects. Name the first `GaringerOzone.daily.ts` and base it on the dataframe of daily observations. Name the second `GaringerOzone.monthly.ts` and base it on the monthly average ozone values. Be sure that each specifies the correct start and end dates and the frequency of the time series.

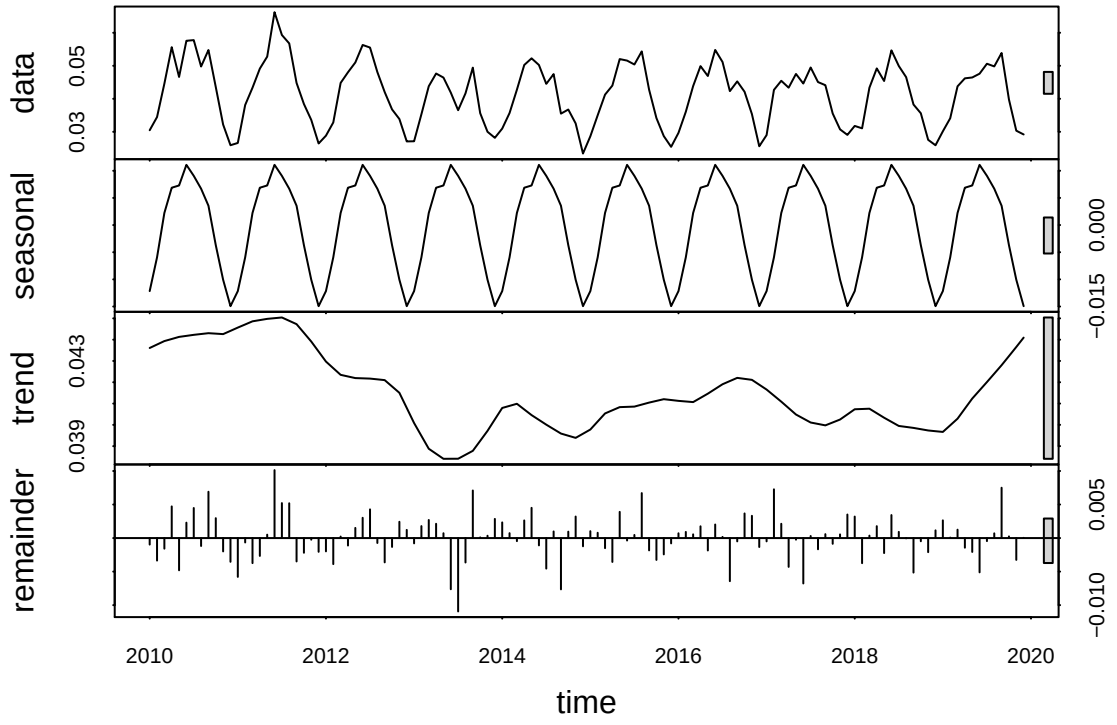
```
#10

GaringerOzone.daily.ts <- ts(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration,
                             start = c(2010,1), frequency = 365)
GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$Mean.Ozone,
                               start = c(2010,1), frequency = 12)
```

11. Decompose the daily and the monthly time series objects and plot the components using the `plot()` function.

```
#11

GaringerOzone.daily_Decomposed <- stl(GaringerOzone.daily.ts, s.window = "periodic")

GaringerOzone.monthly_Decomposed <- stl(GaringerOzone.monthly.ts, s.window = "periodic")

plot(GaringerOzone.daily_Decomposed)
```




```
plot(GaringerOzone.monthly_Decomposed)
```




12. Run a monotonic trend analysis for the monthly Ozone series. In this case the seasonal Mann-Kendall is most appropriate; why is this?

```
#12

Garinger_Ozone_Trend <- trend::smk.test(GaringerOzone.monthly.ts)
Garinger_Ozone_Trend
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data:  GaringerOzone.monthly.ts
## z = -1.963, p-value = 0.04965
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##    S varS
##  -77 1499
```

```
print(Garinger_Ozone_Trend)
```

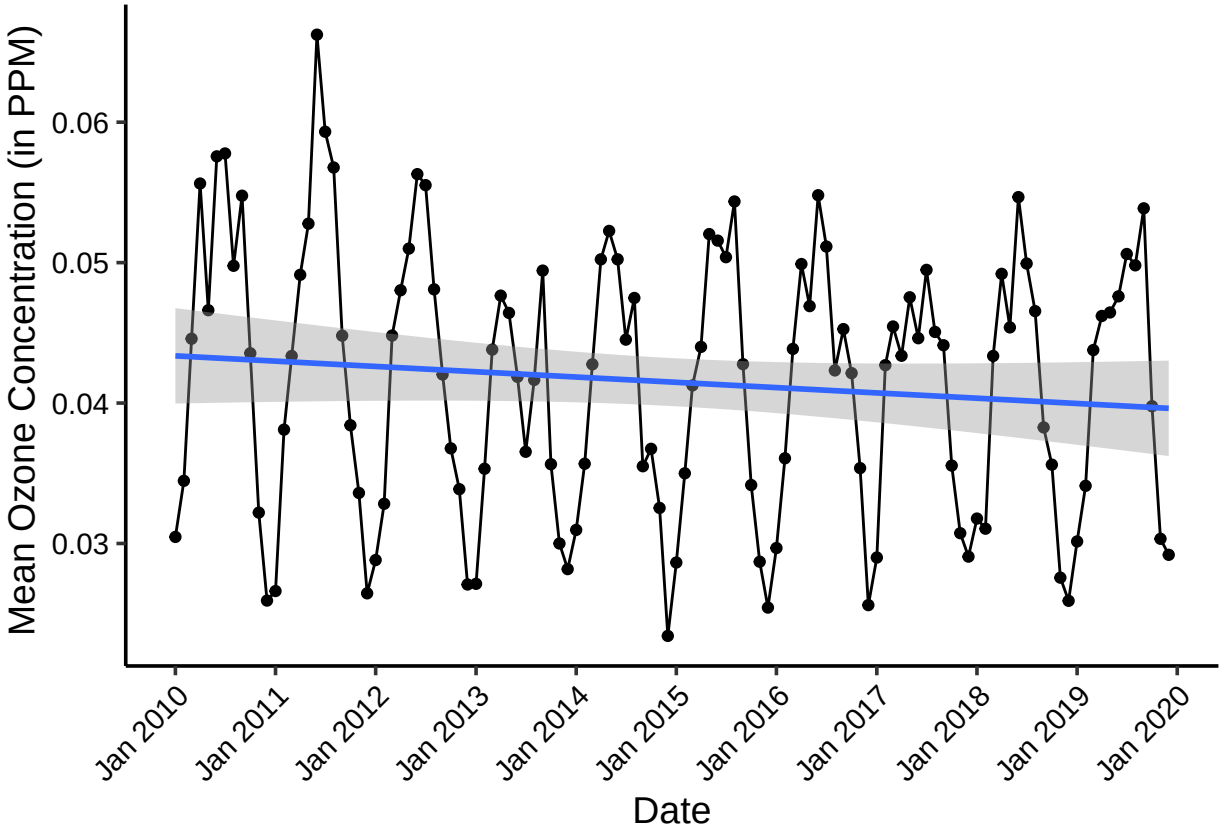
```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data:  GaringerOzone.monthly.ts
## z = -1.963, p-value = 0.04965
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##    S varS
##  -77 1499
```

Answer: It's appropriate to use because the plot show that there are seasonal changes in the data.

13. Create a plot depicting mean monthly ozone concentrations over time, with both a geom_point and a geom_line layer. Edit your axis labels accordingly.

```
# 13

Garinger_Ozone_Plot <-
ggplot(GaringerOzone.monthly, aes(x = Date, y = Mean.Ozone)) +
  geom_point() +
  geom_line() +
  scale_x_date(date_labels = "%b %Y", date_breaks = "1 year") +
  labs(x = "Date", y = " Mean Ozone Concentration (in PPM)")+
  theme(axis.text.x = element_text(angle = 45, hjust=1)) +
  geom_smooth( method = lm)
print(Garinger_Ozone_Plot)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




14. To accompany your graph, summarize your results in context of the research question. Include output from the statistical test in parentheses at the end of your sentence. Feel free to use multiple sentences in your interpretation.

    Answer: There is not enough statistical evidence to support that Ozone concentrations have changed from the 2010. The p value generated from the Mann-Kendall test is approximately 0.05, which is the same as the alpha we are conducting the test at. Thus, the difference is neglible. While there appears to be a trend in ozone concentrations seasonally (low concentrations in the winter, high concentrations in the summer), we do not have enough statistical evidence to support this trend.

15. Subtract the seasonal component from the `GaringerOzone.monthly.ts`. Hint: Look at how we extracted the series components for the EnoDischarge on the lesson Rmd file.

16. Run the Mann Kendall test on the non-seasonal Ozone monthly series. Compare the results with the ones obtained with the Seasonal Mann Kendall on the complete series.

```
#15

GaringerOzone_Seasonal <-
  as.data.frame(GaringerOzone.monthly_Decomposed$time.series[,1:3])

GaringerOzone_Seasonal <- mutate(GaringerOzone_Seasonal,
        Observed = GaringerOzone.monthly$Mean.Ozone,
        Date = GaringerOzone.monthly$Month)

GaringerOzone_NonSeasonal <-
  ts(GaringerOzone_Seasonal$Observed - GaringerOzone_Seasonal$seasonal,
                                start = c(2010,1), frequency = 12)
```

```r
#16

GaringerOzone_NoSeasonal <- trend::mk.test(GaringerOzone_NonSeasonal)
print(GaringerOzone_NoSeasonal)
```

```
##
##  Mann-Kendall trend test
##
## data:  GaringerOzone_NonSeasonal
## z = -2.672, n = 120, p-value = 0.00754
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -1.179000e+03  1.943657e+05 -1.651376e-01
```

Answer: After preforming a Mann-Kendall test on the nonseasonal data, there is statistical evidence that Ozone concentrations have changed from 2010. The p value is 0.008, which is far below the alpha of 0.05.