Assignment 7: Time Series Analysis

Li Jia Go

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
#checking working directory
getwd()
```

```
## [1] "/home/guest/R/EDA-Fall2022"
```

```
#load required packages
library(tidyverse)
library(lubridate)
library(zoo)
library(trend)

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

#2
```

```
#import files
GaringerFiles = list.files(path = "./Data/Raw/Ozone_TimeSeries/", pattern="*.csv", full.names=TRUE)
GaringerFiles
```

```
##  [1] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2010_raw.csv"
##  [2] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2011_raw.csv"
##  [3] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2012_raw.csv"
##  [4] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2013_raw.csv"
##  [5] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2014_raw.csv"
##  [6] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2015_raw.csv"
##  [7] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2016_raw.csv"
##  [8] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2017_raw.csv"
##  [9] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2018_raw.csv"
## [10] "./Data/Raw/Ozone_TimeSeries//EPAair_O3_GaringerNC2019_raw.csv"
```

```
GaringerOzone <- GaringerFiles %>%
  plyr::ldply(read.csv)
```

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".

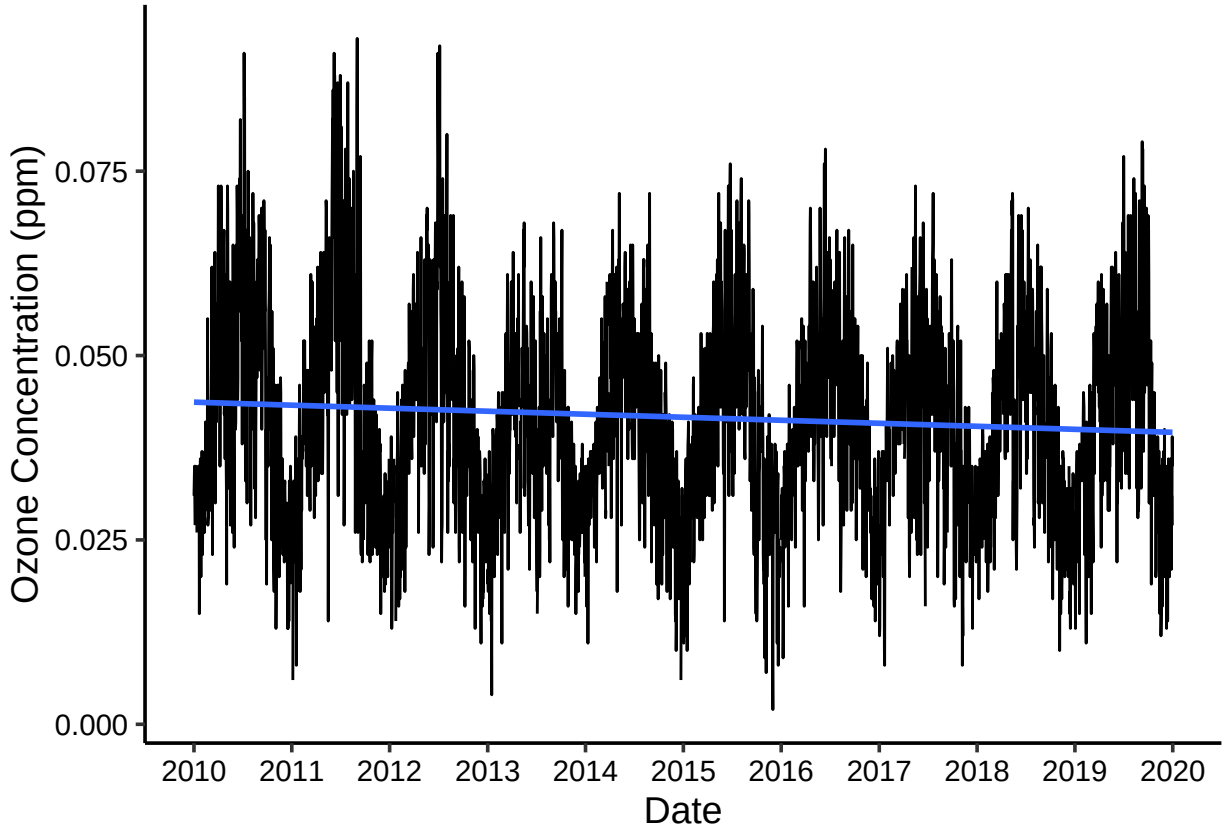
6. Use a `left_join` to combine the data frames. Specify the correct order of data frames within this function so that the final dimensions are 3652 rows and 3 columns. Call your combined data frame GaringerOzone.

```
# 3
#setting date column
GaringerOzone$Date <- as.Date(GaringerOzone$Date, format ="%m/%d/%Y")

# 4
GaringerOzone.processed <- GaringerOzone %>%
  select(Date, Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)

# 5
Days <- as.data.frame(seq.Date(from = as.Date("2010-01-01"), to = as.Date("2019-12-31"), by="day"))
colnames(Days) <- c("Date")

# 6
GaringerOzone2 <- left_join(Days, GaringerOzone.processed)
```

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
Ozone_Time <- ggplot(GaringerOzone2, aes(x=Date, y=Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line() +
  geom_smooth(method= "lm", se=FALSE) +
  labs (y= "Ozone Concentration (ppm)") +
  scale_x_date(date_breaks = "1 year", date_labels="%Y")
print(Ozone_Time)
```



Answer: The plot suggests that no trend is visible, given that the linear line is relatively flat. This is likely due to seasonality.

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
#fill in missing data for ozone concentration
GaringerOzone2.filled <- GaringerOzone2 %>%
  mutate(Daily.Max.8.hour.Ozone.Concentration=zoo::na.approx((Daily.Max.8.hour.Ozone.Concentration)))
```

Answer: Given that we were looking at environmental data (i.e. ozone concentrations), linear interpolation made the most sense as we would expect the missing data to fall between known data points on either side of the unknown point. The piecewise constant/nearest neighbour approach would be best used for categorical data (e.g. land use classification), and the spline is used to minimise overall curvature of an interpolated line which was not necessary in this case.

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)

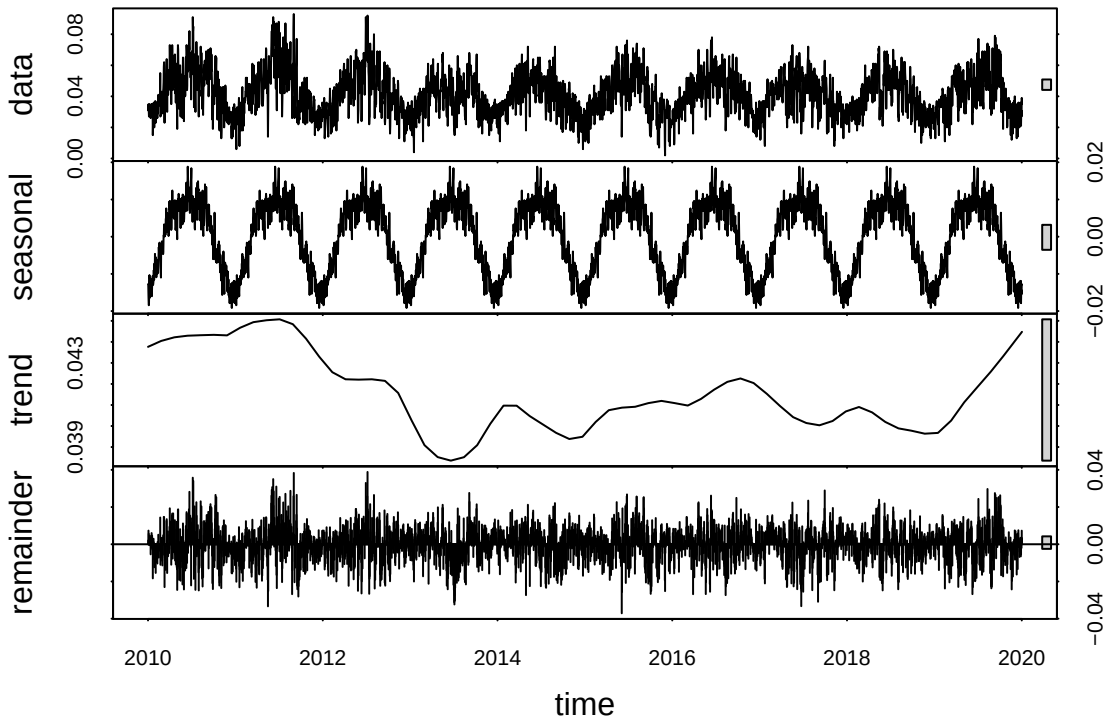
```
#9

#creating a separate month and year column, using the split-apply-combine
#technique to get mean O3 values
#creating a new Date column with month and year set as first day of the month
GaringerOzone.monthly <- GaringerOzone2.filled %>%
  mutate(Month=month(Date)) %>%
  mutate(Year=year(Date)) %>%
  group_by(Month, Year) %>%
  summarize (meanO3conc=mean(Daily.Max.8.hour.Ozone.Concentration)) %>%
  mutate(Day="01") %>%
  mutate(Date = as.Date(paste(Year, Month, Day, sep = "-"), "%Y-%m-%d"))

#rearranging rows of monthly dataset in ascending order by date
GaringerOzone.monthly2 <- GaringerOzone.monthly %>%
  dplyr::arrange(Date)
```

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.

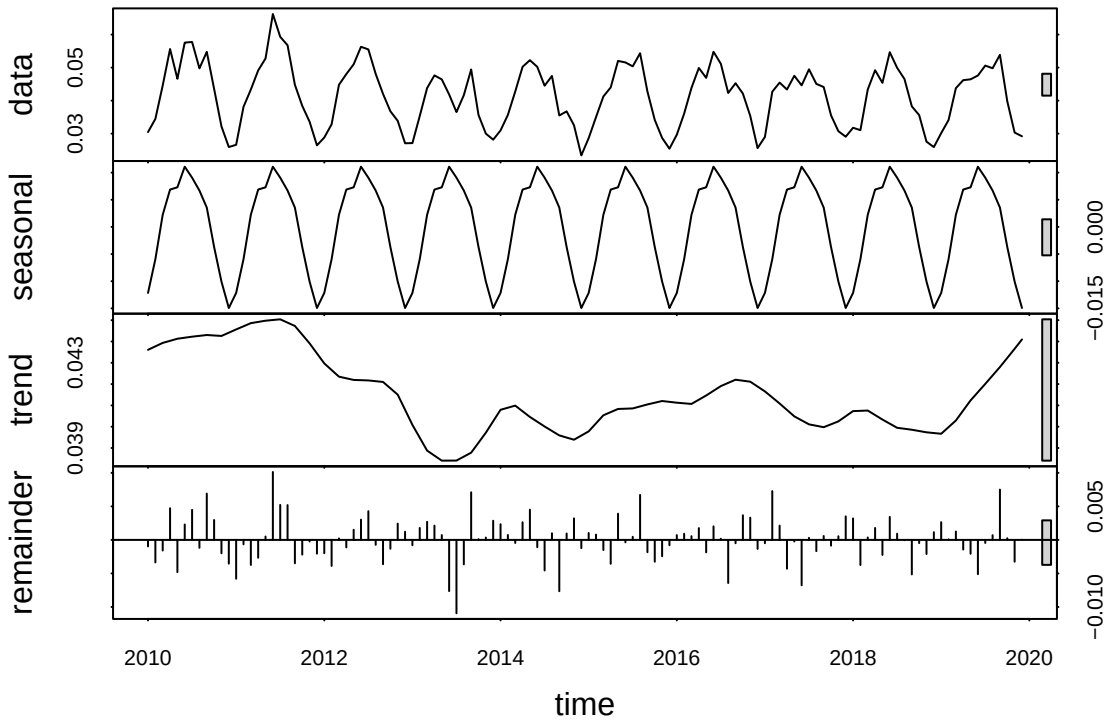
```
# 10

GaringerOzone.daily.ts <- ts(GaringerOzone2.filled$Daily.Max.8.hour.Ozone.Concentration,
    start = c(2010, 1), frequency = 365)

GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly2$meanO3conc,
    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.ts_decomposed <- stl(GaringerOzone.daily.ts, s.window="periodic")
plot(GaringerOzone.daily.ts_decomposed)
```




```
GaringerOzone.monthly.ts_decomposed <- stl(GaringerOzone.monthly.ts, s.window="periodic")
plot(GaringerOzone.monthly.ts_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?

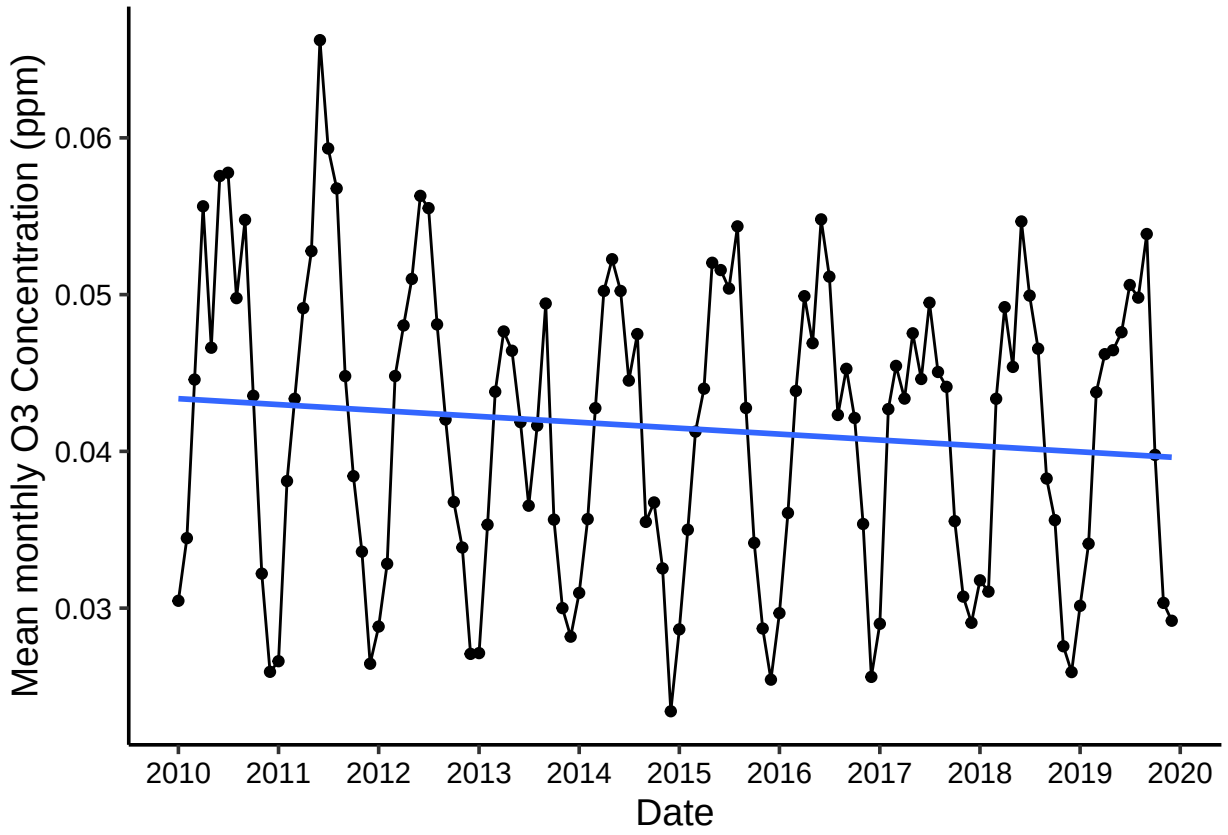
```
#12

GaringerOzone.monthly.trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)
GaringerOzone.monthly.trend
```

```
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: From the decomposed plot of the time series, we clearly see that monthly Ozone has a seasonal cycle, as such the seasonal Mann-Kendall is the most appropriate test given that the other tests cannot account for seasonal 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
monthlyO3plot <-
  ggplot(GaringerOzone.monthly, aes(x=Date, y=meanO3conc)) +
  geom_point()+
  geom_line() +
  ylab("Mean monthly O3 Concentration (ppm)") +
  geom_smooth( method = lm, se=FALSE ) +
  scale_x_date(date_breaks = "1 year", date_labels="%Y")

print(monthlyO3plot)
```



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: The graph shows a seasonal pattern in mean monthly O3 concentration which repeats yearly. The p-value of the Seasonal Mann-Kendall test was 0.0467 $<$ 0.05, which means that there is a trend in the time series. The tau value was negative, which indicates that the trend wass decreasing over the 2010s at this station. (Output of statistical test: tau=-0.143, 2-sided pvalue=0.046724)

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 converting time series object into
# data frame
GaringerOzone.monthly.components <- as.data.frame(GaringerOzone.monthly.ts_decomposed$time.series[,
    1:3])

# subtracting seasonality from original
# time series object
GaringerDeseasoned <- GaringerOzone.monthly.ts -
    GaringerOzone.monthly.components$seasonal

# 16 running the MannKendall test on
# the time series without seasonality
```

```
GaringerDeseasoned.trend <- Kendall::MannKendall(GaringerDeseasoned)
GaringerDeseasoned.trend
```

```
## tau = -0.165, 2-sided pvalue =0.0075402
```

Answer: After removing seasonality from the time series, the p-value was 0.0075 <0.05. This means that we reject the null hypothesis and conclude that there is a significant trend in the time series that is not due to seasonality. However, in comparison to the p-value obtained from the Seasonal Mann-Kendall test, we found that while the p-value was still <0.05, it was much higher than what we obtained from the Mann-Kendall test, at 0.046. This similarly suggests that there is a trend in the time series, though it is "blurred out" by seasonality.