Assignment 7: Time Series Analysis

Jess Garcia

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. Check working directory, install & load packages, &  set ggplot theme
getwd()
```

```
## [1] "C:/Users/93jes/Documents/ENV872/Environmental_Data_Analytics_2021"
```

```
library(tidyverse)
library(lubridate)
#install.packages("zoo")
library(zoo)
```

```
## Warning: package 'zoo' was built under R version 4.0.4
```

```
#install.packages("trend")
library(trend)
```

```
## Warning: package 'trend' was built under R version 4.0.4
```

```
#install.packages("Kendall")
library(Kendall)
```

```
## Warning: package 'Kendall' was built under R version 4.0.4
```

```
JG_default_theme <- theme_bw(base_size = 12)+
  theme(axis.text = element_text(color = "black"),
        plot.title = element_text(color = "black", size = 16, face = "bold",
        hjust = 0.5))


#2. Import & combine data sets
Garinger2010 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv", stringsAsFacto

Garinger2011 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv", stringsAsFacto

Garinger2012 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv", stringsAsFacto

Garinger2013 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv", stringsAsFacto

Garinger2014 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv", stringsAsFacto

Garinger2015 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv", stringsAsFacto

Garinger2016 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv", stringsAsFacto

Garinger2017 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv", stringsAsFacto

Garinger2018 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv", stringsAsFacto

Garinger2019 <- read.csv("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv", stringsAsFacto


GaringerOzone <- rbind(Garinger2010, Garinger2011, Garinger2012, Garinger2013, Garinger2014, Garinger201
```

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. Set date column as a date class
GaringerOzone$Date <- as.Date(GaringerOzone$Date, format = "%m/%d/%Y")


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

```r
# 5.Generate a daily dataset

Days.df <- as.data.frame(seq(as.Date("2010-01-01"), as.Date("2019-12-31"), "day"))


#rename column
colnames(Days.df) <- "Date"


# 6. Left join to combine data frames
GaringerOzone2 <- left_join(Days.df, GaringerOzone_wrangled)
```

```
## 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?

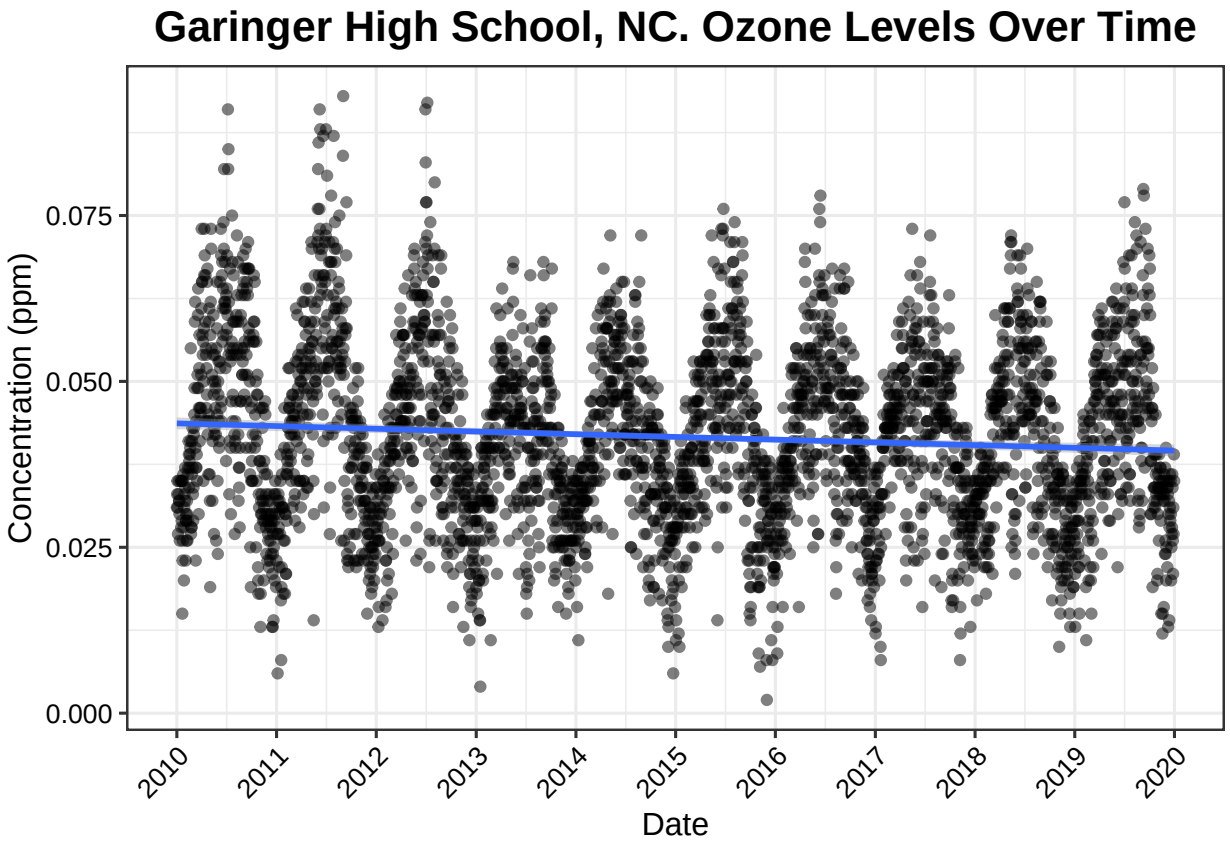
```r
#7. Create line plot of ozone concentrations in ppm over time
GaringerOzone2 <- GaringerOzone2 %>%
  rename(Concentration = Daily.Max.8.hour.Ozone.Concentration)


ggplot(GaringerOzone2, aes(x= Date, y = Concentration))+
  geom_point(alpha = 0.5)+
  geom_smooth(method = lm)+
  JG_default_theme +
  ggtitle("Garinger High School, NC. Ozone Levels Over Time")+
  labs(x = "Date", y="Concentration (ppm)")+
  scale_x_date(date_breaks ="1 year", date_labels = "%Y")+
  theme(axis.text.x=element_text(angle = 45, hjust = 1))
```

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

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

```
## Warning: Removed 63 rows containing missing values (geom_point).
```



Answer: My plot roughly suggests a minor downward trend in ozone concentrations over time when looking at the smoothed line for linear trend.

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. Linear interpolation for missing daily data
GaringerOzone2_filled <- GaringerOzone2 %>%
  mutate(Concentration_filled = zoo:: na.approx(Concentration))
```

Answer: We used linear interpolation to fill in the missing data. We did not use piecewise constant interpolation because it assumes that any missing data is equal to that of the next nearest measurement and we don't want to assume they are the same. It's better to assume the missing data falls somewhere between the included data with linear interpolation. Ozone concentration is going to rise and fall which means it makes more sense to assume the measurements missing fall between what is recorded. We also didn't use spline interpolation because spline assumes a quadratic function and there isn't anything about our data that necessarily makes it seem like quadratic would make sense.

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. Create aggregated data frame
GaringerOzone.monthly <- GaringerOzone2_filled %>%
  mutate(year = year(Date)) %>%
  mutate(month = month(Date))%>%
  mutate(Date = my(paste0(month,"-",year)))

GaringerOzone.monthly2 <- GaringerOzone.monthly %>%
  group_by(Date)%>%
  summarize(mean_Ozone.monthly = mean(Concentration_filled))
```

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.

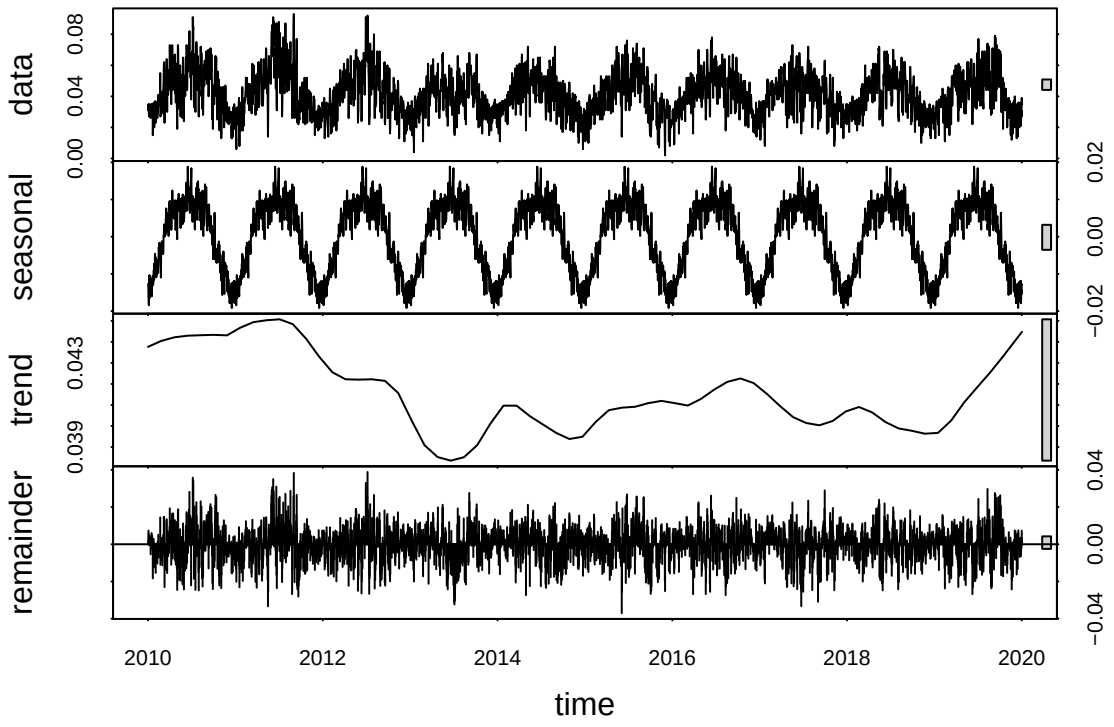
```
#10. Create new time series objects
GaringerOzone.daily.ts <- ts(GaringerOzone2_filled$Concentration_filled , start = c(2010, 1, 1), freque


GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly2$mean_Ozone.monthly , start = c(2010, 1), frequenc
```

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

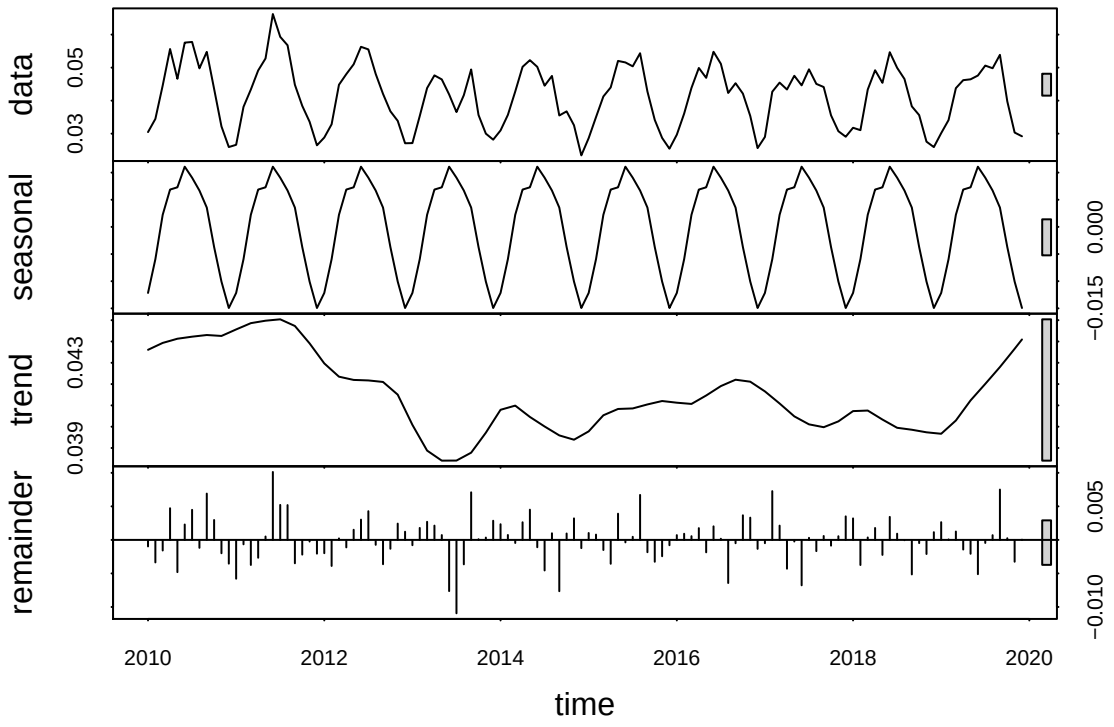
```
#11. Decompose the time series objects
GaringerOzone.daily.ts_decomposed <- stl(GaringerOzone.daily.ts, s.window = "periodic")

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. Monotonic trend analysis using seasonal Mann-Kendall
GaringerOzone.monthly.ts_trend <- Kendall:: SeasonalMannKendall (GaringerOzone.monthly.ts)

summary(GaringerOzone.monthly.ts_trend)
```

```
## Score =  -77 , Var(Score) = 1499
## denominator =  539.4972
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: The seasonal Mann-Kendall is most appropriate here for this monotonic trend analsysis because the monthly ozone series is seasonal in that ozone concentrations do vary by month (grouped by seasons). So we are seeing that over the years there is seasonality by months/seasons and taking that into consideration when asking if there is stationarity.

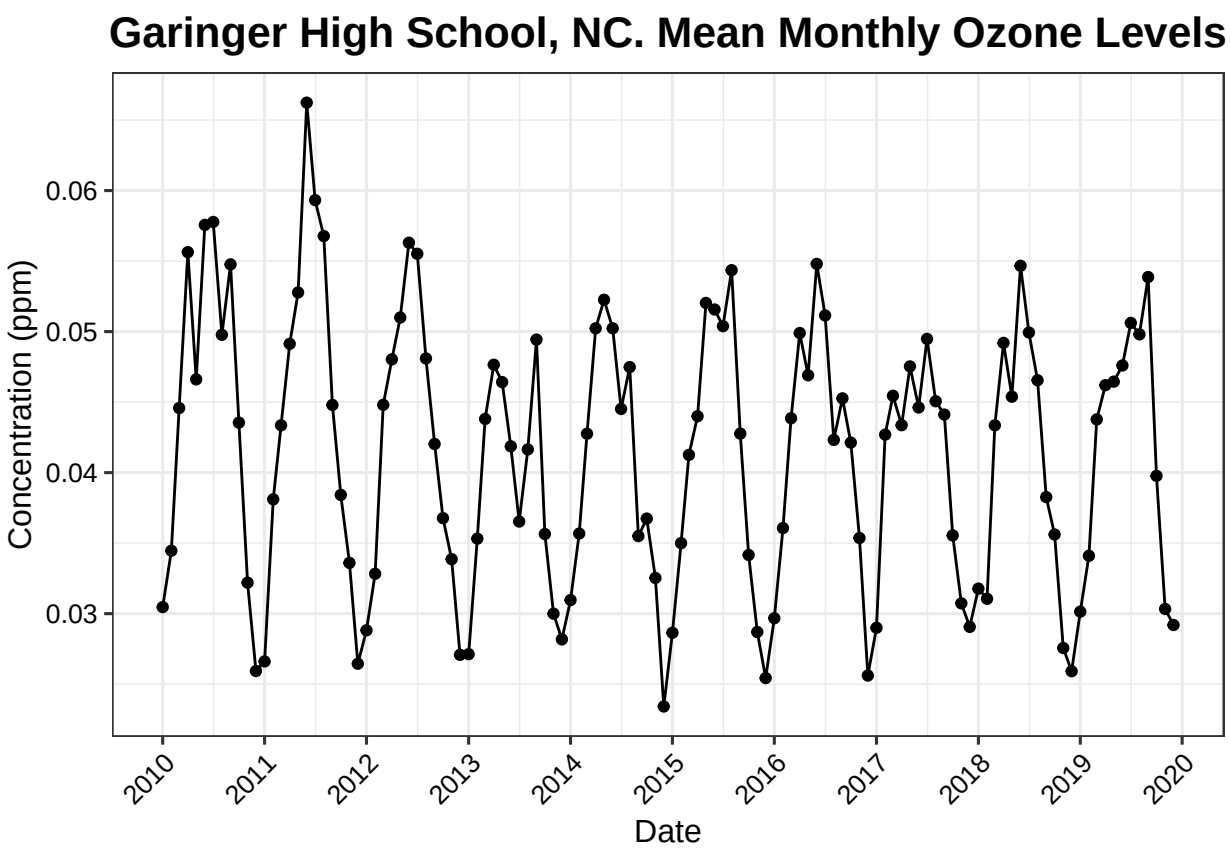
13. Create a plot depicting mean monthly ozone concentrations over time, with both a geom_point and a geom_line layer. Edit your axis labels accordingly.

```
#13. Plot mean monthly ozone concentrations over time
GaringerOzone_monthly_mean.plot <- ggplot(GaringerOzone.monthly2, (aes(x=Date, y=mean_Ozone.monthly)))+
  geom_line(method = lm)+
  geom_point()+
  JG_default_theme+
  ggtitle("Garinger High School, NC. Mean Monthly Ozone Levels")+
  labs(x = "Date", y="Concentration (ppm)")+
  scale_x_date(date_breaks ="1 year", date_labels = "%Y")+
```

```
  theme(axis.text.x=element_text(angle = 45, hjust = 1))
```

```
## Warning: Ignoring unknown parameters: method
```

```
print(GaringerOzone_monthly_mean.plot)
```



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: Concentrations have changed over the 2010s at this station at Garinger high school. There does appear to be seasonality in this data which means that the ozone concentration changes througout the year. However, the P-value is .0467, which is really close to the .05 signficance threshold. It is less than the threshold so you should probably reject the null for the seasonal MannKendall that there is stationarity, but it's super close.

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. Subtract seasonal component from monthly time series

GaringerOzone.monthly_seasonal.ts <-
GaringerOzone.monthly.ts_decomposed$time.series[,1]

GaringerOzone.monthly_nonseasonal.ts <- GaringerOzone.monthly.ts - GaringerOzone.monthly_seasonal.ts
```

```
#16. Regular Mann Kendall test on non-seasonal ozone monthly time series
MannKendall(GaringerOzone.monthly_nonseasonal.ts)
```

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

Answer: The P-value for the regular Mann Kendall test is .00754, which is less than the .05 significance threshold which means we reject the null. The regular Mann Kendall tests for stationarity and I think the null is that there is stationarity, which means there is no change to the series over time. Because we reject the null, that means that concentrations of ozone have changed over time at Garinger high school even removing seasonality from the equation. Running the seasonal Mann Kendall where the p-value was .0467 still had us reject the null of stationarity but it was closer to the significance threshold because there clearly is seasonality in this data and by taking that into account there is going to be less evidence of change over time, but the conclusion of non-stationarity is still the same.