Assignment 8: Time Series Analysis

Ayden Schirmacher

Spring 2025

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on generalized linear models.

Directions

1. Rename this file `<FirstLast>_A08_TimeSeries.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up

1. Set up your session:

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

```
remove(list=ls())

getwd()
```

```
## [1] "/home/guest/EDA_Spring 2025"
```

```
library(tidyverse)
library(lubridate)
library(dplyr)
library(ggthemes)
library(ggplot2)
library(zoo)
library(trend)
library(here)

theme_A08 <- theme_base() +
  theme(
    text = element_text(color='black', size=10, face = 'italic'),
    panel.grid.minor = element_line(color="gray87"),
    panel.grid.major = element_line(color="gray87"),
    plot.background = element_rect(color='black', fill='snow1'),
    axis.ticks = element_line(linewidth=0.25))

theme_set(theme_A08) #creating theme for outputs
```

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
file_paths <- c("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv",
                "Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv")

GaringerOzone <- file_paths %>% # Read and combine all CSV files
  map(~ read_csv(here(.))) %>%
  bind_rows()
```

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
GaringerOzone<-GaringerOzone%>%
  mutate(Date=mdy(Date))

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

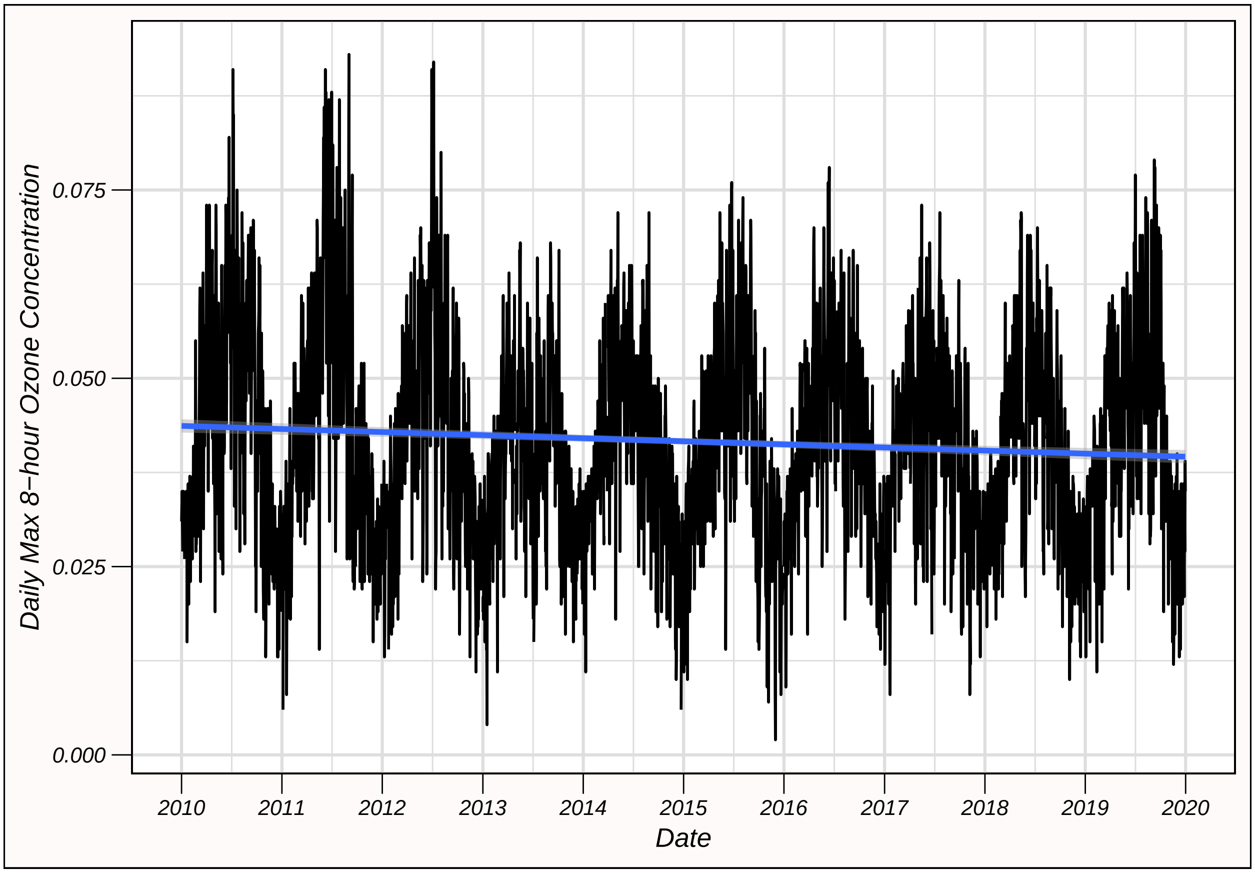
```
# 5
Days <- as.data.frame(seq(mdy("01-01-2010"),
                          mdy("12-31-2019"),
                          by = "day"))
colnames(Days) <- "Date"

# 6
GaringerOzone<-left_join(Days,GaringerOzone, 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%>%
  ggplot(aes(x=Date, y=`Daily Max 8-hour Ozone Concentration`))+
  geom_line()+
  geom_smooth(method="lm")+
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")
```



Answer: There is a seasonal variation in ozone concentration, with mre in the summer and less in the winter, as well as a slight negative trend in ozone concentration over the 10-year period.

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
summary(GaringerOzone$`Daily Max 8-hour Ozone Concentration`)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## 0.00200 0.03200 0.04100 0.04163 0.05100 0.09300      63
```

```
GaringerOzone_daily<-GaringerOzone%>%
  mutate(`CLEAN Daily Max 8-hour Ozone Conc`=zoo::na.approx(`Daily Max 8-hour Ozone Concentration`))
```

> Answer: A piecewise constant interpolation would be better suited for data that is linear or non-seasonal. Spline interpolation also assumes a smoother trend in the data, and might have worked on seasonal data if the data weren't over such a large period of time. However, these two types of functions fall short where they do not account for the constant repetitive changing of the ozone concentration with the seasons. Linear interpolation on the other hand assumes the data is between the previous and next observation, which provides a more accurate analysis 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

GaringerOzone.monthly<-GaringerOzone_daily%>%
  mutate(Year = year(Date),
         Month = month(Date))%>%
  group_by(Year, Month)%>%
  summarize(Mean_Ozone = mean(`Daily Max 8-hour Ozone Concentration`, na.rm = TRUE))

GaringerOzone.monthly <- GaringerOzone.monthly%>%
  mutate(new_date = as.Date(paste(Year, Month, "01", sep = "-")))
```

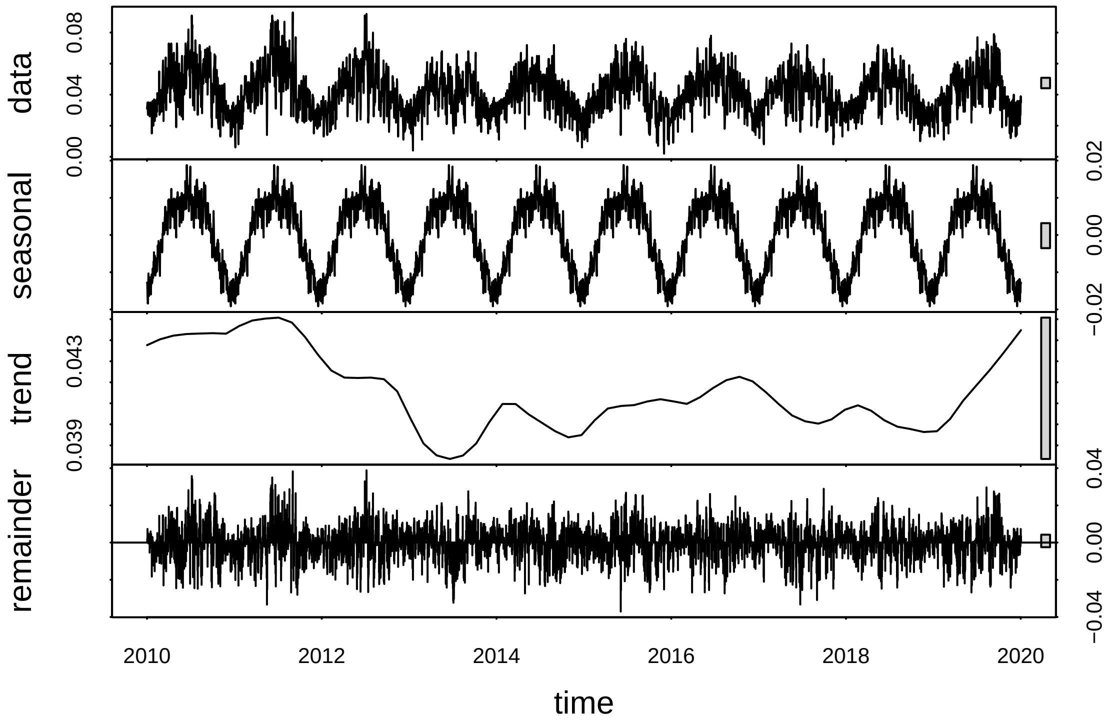
10. Generate two time series objects. Name the first `GaringerOzone.daily.ts` and base it on the dataframe of daily observations. Name the second `GaringerOzone.monthly.ts` and base it on the monthly average ozone values. Be sure that each specifies the correct start and end dates and the frequency of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

```
#10
GaringerOzone.daily.ts<-ts(GaringerOzone_daily$`CLEAN Daily Max 8-hour Ozone Conc`,
                start=c(2010, 1),
                frequency=365) #frequency is how often the values repeat:
                              #monthly data is 12, daily is 365, etc.
```

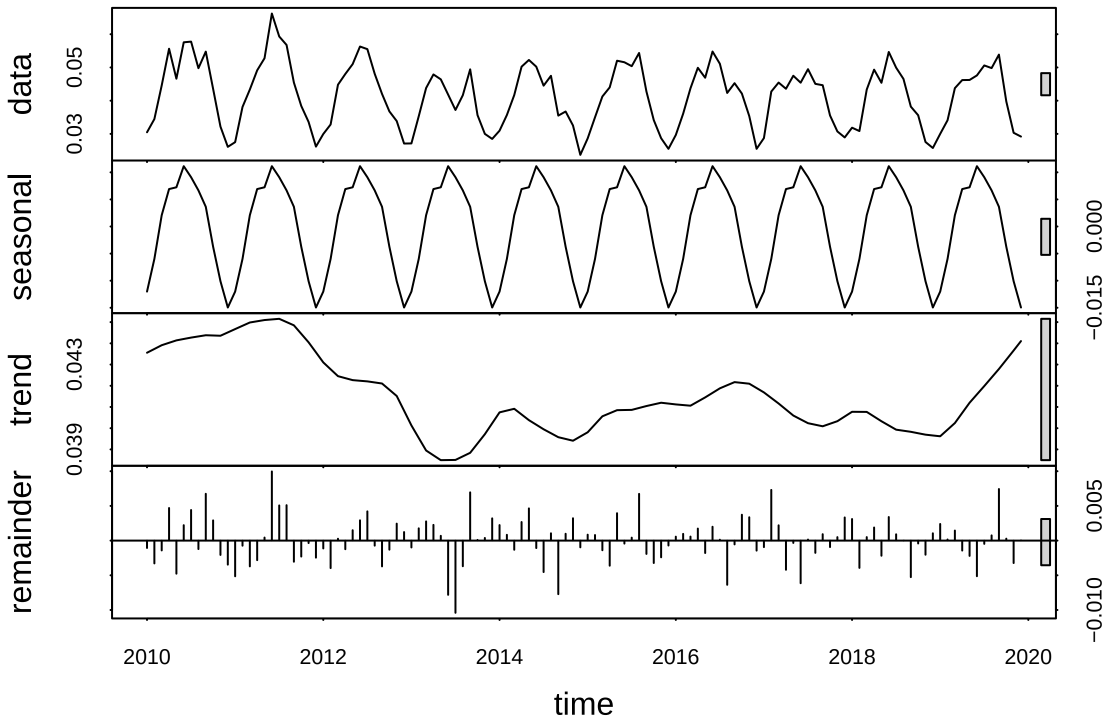
```
GaringerOzone.monthly.ts<-ts(GaringerOzone.monthly$Mean_Ozone,
            start=c(2010, 1),
            frequency=12)
```

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

```
#11
ozone_daily_decomp=stl(GaringerOzone.daily.ts, s.window = "periodic") #decomposition to find components
plot(ozone_daily_decomp)
```



```
ozone_monthly_decomp=stl(GaringerOzone.monthly.ts, s.window = "periodic") #decomposition to find compon
plot(ozone_monthly_decomp)
```



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

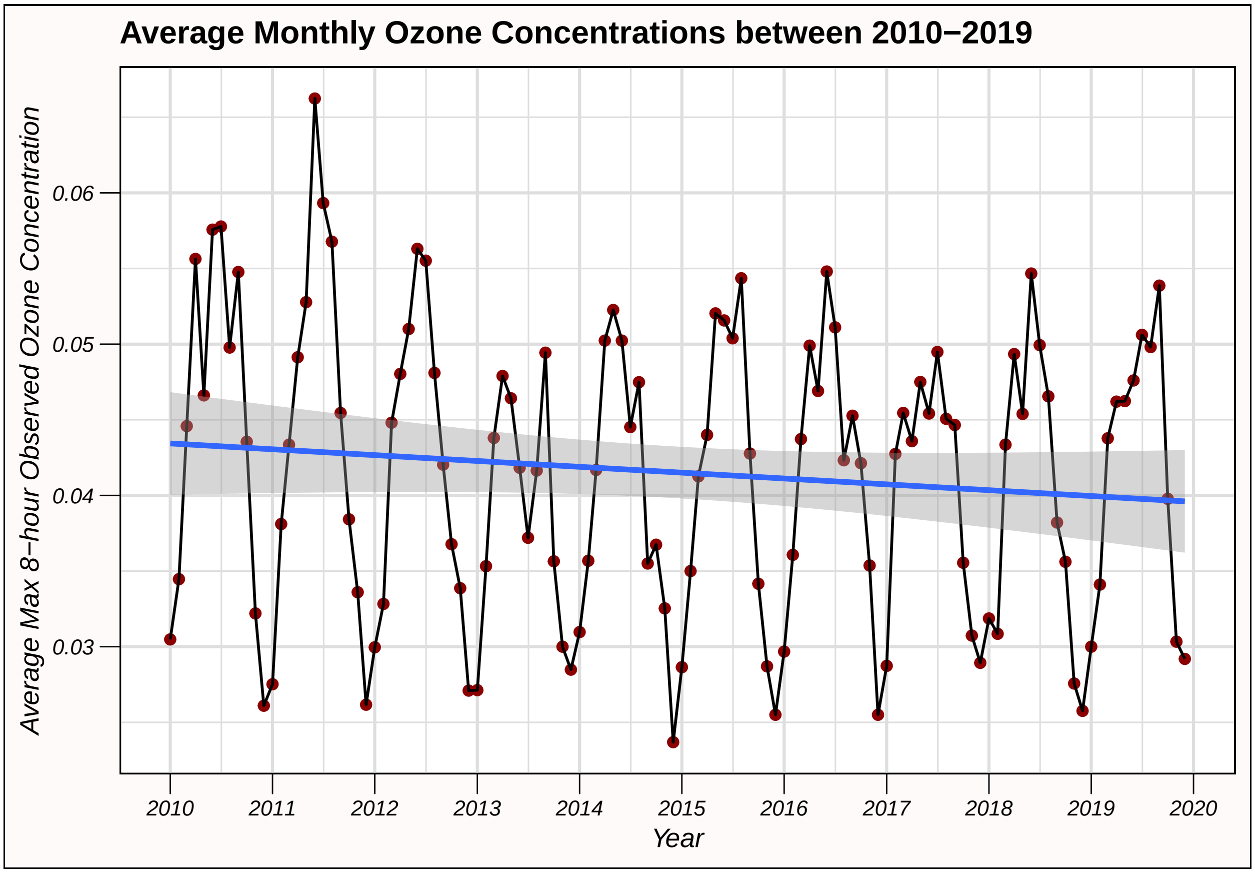
```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data: GaringerOzone.monthly.ts
## alternative hypothesis: two.sided
##
## Statistics for individual seasons
##
## H0
##                      S varS    tau      z Pr(>|z|)
## Season 1:   S = 0    9  125  0.200  0.716  0.47427
## Season 2:   S = 0   -1  125 -0.022  0.000  1.00000
## Season 3:   S = 0   -4  124 -0.090 -0.269  0.78762
## Season 4:   S = 0  -17  125 -0.378 -1.431  0.15241
## Season 5:   S = 0  -17  125 -0.378 -1.431  0.15241
## Season 6:   S = 0  -17  125 -0.378 -1.431  0.15241
## Season 7:   S = 0  -11  125 -0.244 -0.894  0.37109
## Season 8:   S = 0   -7  125 -0.156 -0.537  0.59151
## Season 9:   S = 0   -7  125 -0.156 -0.537  0.59151
## Season 10:   S = 0 -13  125 -0.289 -1.073  0.28313
## Season 11:   S = 0 -13  125 -0.289 -1.073  0.28313
## Season 12:   S = 0  10  124  0.225  0.808  0.41896
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

> Answer: The Seasonal Mann-Kendall is the most appropriate because it accounts for seasonality and is non-parametric. The other tests do not account for seasonality, which would cause potential errors in the analysis.

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
monthly_plot<-
ggplot(GaringerOzone.monthly, aes(x=new_date, y=Mean_Ozone))+
             geom_point(color="darkred")+
             geom_line()+
             geom_smooth(method="lm")+
  labs(x="Year", y="Average Max 8-hour Observed Ozone Concentration", title="Average Monthly Ozone Conc
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")

print(monthly_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: According to this figure, the research question “Have ozone concentrations changed over the 2010s at this station?”, we can say that there is not sufficient evidence to reject the null hypothesis of no changes between seasons in average monthly ozone concentration over the 2010s at this station (no p-values less than 0.15241). In the figure, this result is supported by the relatively stable oscillation of seasonality over time, as well as the only slight decrease of ozone concentration over the study period.

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.

```r
#15
seasonal_component <- ozone_monthly_decomp$time.series[, "seasonal"]
ozone_monthly_noseason <- GaringerOzone.monthly.ts - seasonal_component


#16
mk_trend_monthly<-trend::mk.test(ozone_monthly_noseason)
print(mk_trend_monthly)
```

```
##
##  Mann-Kendall trend test
##
## data:  ozone_monthly_noseason
## z = -2.8966, n = 120, p-value = 0.003773
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -1.278000e+03  1.943647e+05 -1.790167e-01
```

Answer: With the seasonality removed, there is now convicing evidence to reject the null hypothesis that there is no trend in monthly ozone concentration throughout the 2010s (p-value ~0.003). There is convincing evidence of a significant negative trend over the study period (z = -2.8966, S = -1.278000e+03). This is a different result from the Seasonal Mann-Kendall test, which presented no significant trend.