Assignment 8: Time Series Analysis

Jessalyn Chuang

Fall 2024

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
library(trend)
library(zoo)
```

```
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(Kendall)
library(here)
```

```
## here() starts at /home/guest/EDE_Fall2024
```

```
here
```

```
## function (...)
## {
##     .root_env$root$f(...)
## }
## <bytecode: 0x62d20cf3ec70>
## <environment: namespace:here>
```

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

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
file1 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv",
                  stringsAsFactors = TRUE)
file2 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv",
                  stringsAsFactors = TRUE)
file3 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv",
                  stringsAsFactors = TRUE)
file4 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv",
                  stringsAsFactors = TRUE)
file5 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv",
                  stringsAsFactors = TRUE)
file6 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv",
                  stringsAsFactors = TRUE)
file7 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv",
                  stringsAsFactors = TRUE)
file8 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv",
                  stringsAsFactors = TRUE)
file9 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv",
                  stringsAsFactors = TRUE)
file10 <- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv",
```

```
                    stringsAsFactors = TRUE)

GaringerOzone <- rbind.data.frame(file1, file2, file3, file4, file5, file6,
                                  file7, file8, file9, file10)
```

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$Date <- mdy(GaringerOzone$Date)

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

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

# 6
GaringerOzone <- left_join(Days, GaringerOzone)
```

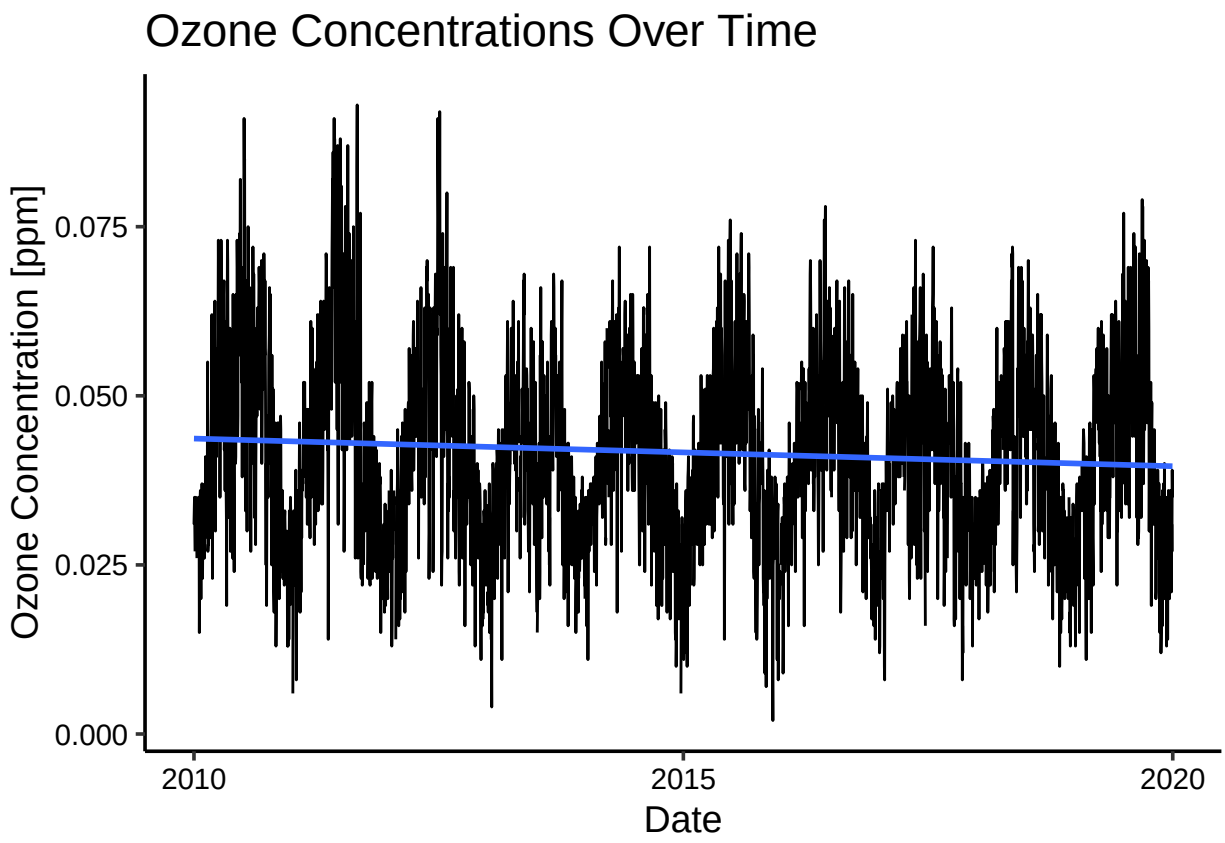
```
## Joining with `by = join_by(Date)`
```

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
ozone_concentrations <- ggplot(GaringerOzone,aes(x = Date,
                              y = Daily.Max.8.hour.Ozone.Concentration)) +
                        geom_line() +
                        labs(title = "Ozone Concentrations Over Time",
                             y = "Ozone Concentration [ppm]") +
                        geom_smooth(method = 'lm', se = FALSE)
ozone_concentrations
```

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

## Warning: Removed 63 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```



Answer: There is a slight downward trend in ozone concentration over time based on the fitted trend line having a slight negative slope. However, the downward trend is subtle, and the data appears to have cyclical/seasonal fluctuations, which could obscure any long-term 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
GaringerOzone_processed <- GaringerOzone %>%
                  mutate(Daily.Max.8.hour.Ozone.Concentration =
                  zoo::na.approx(Daily.Max.8.hour.Ozone.Concentration)) %>%
                  mutate(DAILY_AQI_VALUE = zoo::na.approx(DAILY_AQI_VALUE))
```

Answer: We are using a linear interpolation since there are only a few short periods of data missing and helps preserve the general trend in the data without introducing unexpected patterns. The piecewise constant interpolation, which may result in a step-like pattern, does not accurately reflect the gradual changes that happens in the ozone concentration data. Lastly, spline interpolation might introduce curvature that isn't truly present in the actual ozone trend.

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)

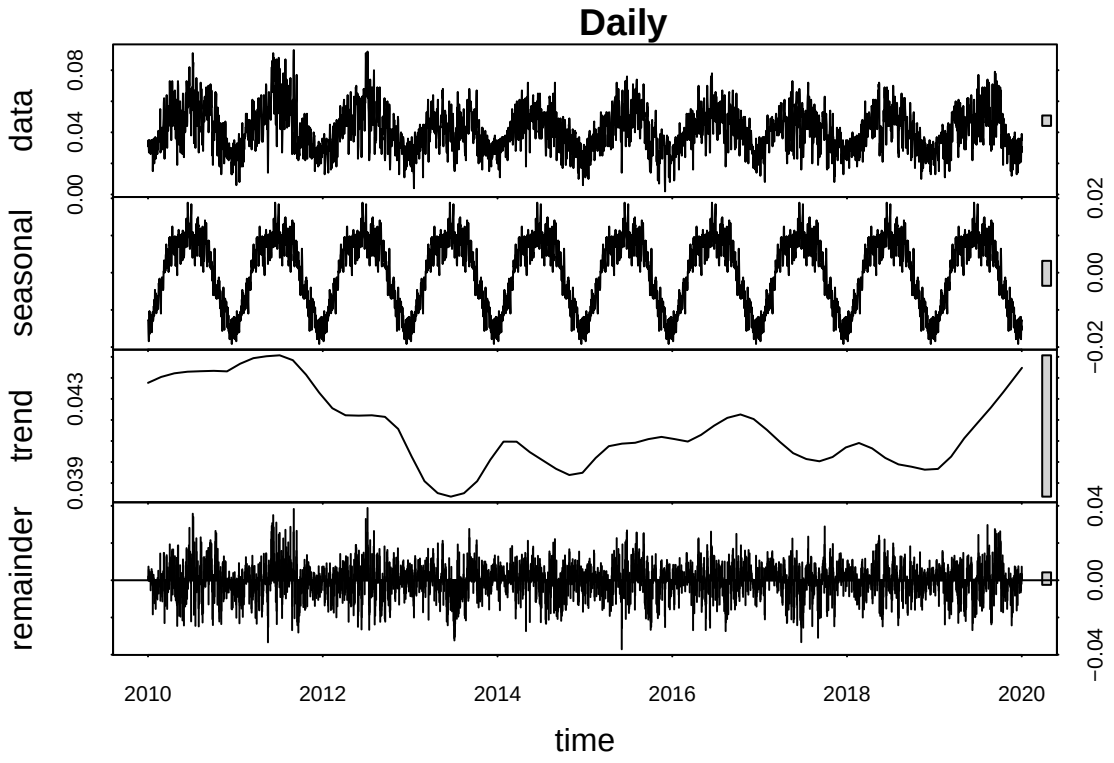
```
#9
GaringerOzone.monthly <- GaringerOzone_processed %>%
  mutate(Year = year(Date), Month = month(Date)) %>%
  group_by(Year, Month) %>%
  summarize(mean.oz = mean(Daily.Max.8.hour.Ozone.Concentration, na.rm = TRUE),
            .groups = "drop") %>%
  mutate(Date = as.Date(paste(Year, Month, "01", 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.

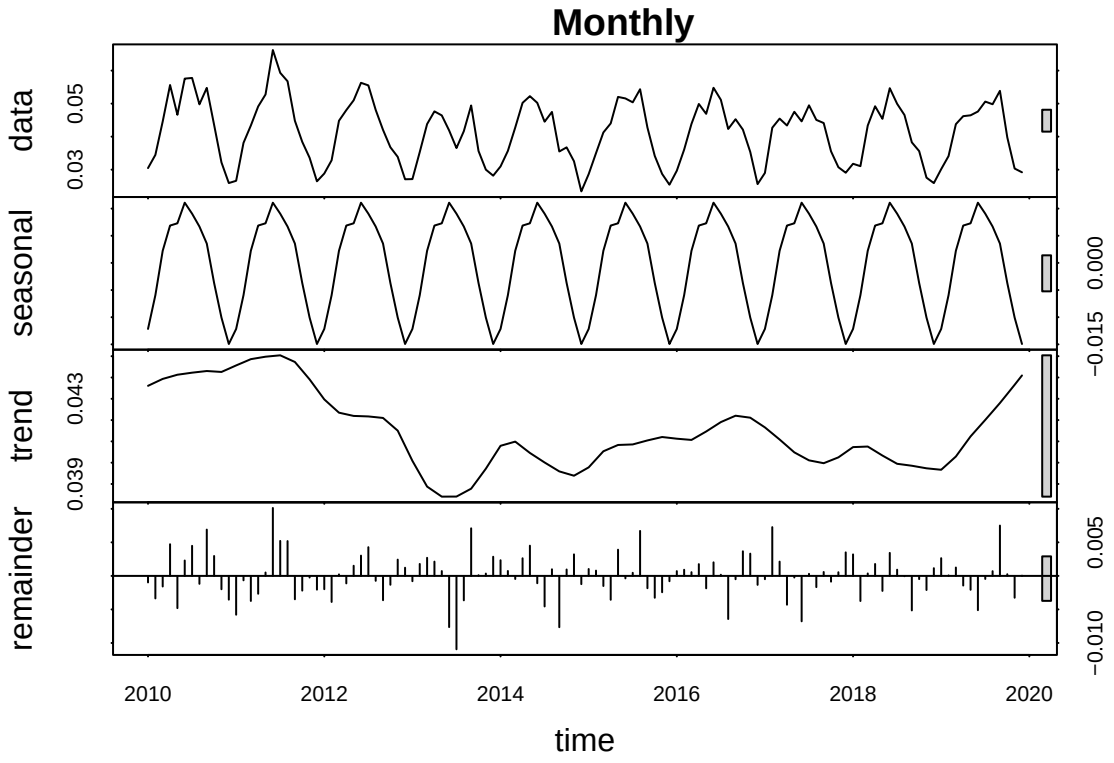
```
#10
f_month <- month(first(GaringerOzone.monthly$Date))
f_year <- year(first(GaringerOzone.monthly$Date))

f_month_daily <- month(first(GaringerOzone$Date))
f_year_daily <- year(first(GaringerOzone$Date))

GaringerOzone.daily.ts <- ts(
  GaringerOzone_processed$Daily.Max.8.hour.Ozone.Concentration,
                       frequency = 365, start = c(f_year_daily, f_month_daily))
GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$mean.oz,
                               frequency = 12, start = c(f_year, f_month))
```

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)
title("Daily")
```




```
plot(GaringerOzone.monthly_Decomposed)
title("Monthly")
```



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_ozone_trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)
summary(monthly_ozone_trend)
```

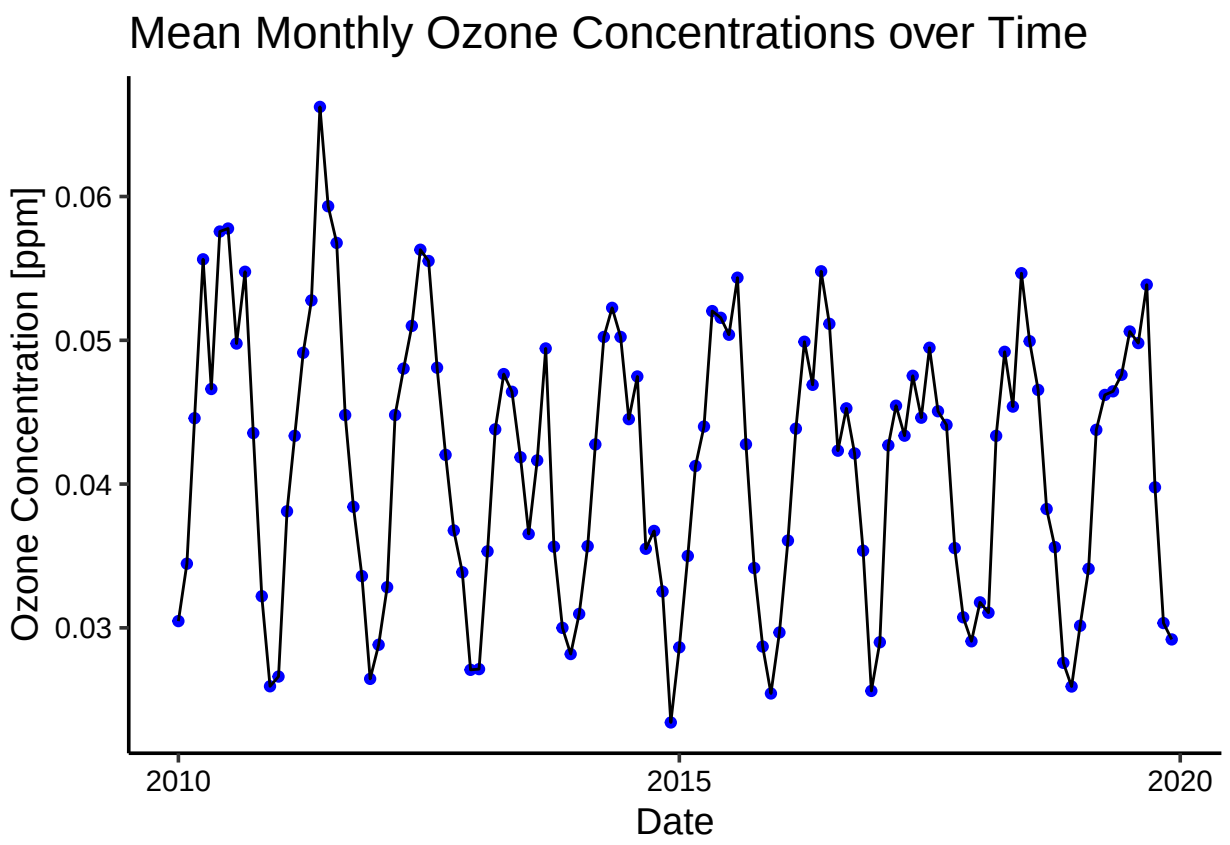
```
## Score =  -77 , Var(Score) = 1499
## denominator =  539.4972
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: The seasonal Mann-Kendall analysis is the most approporiate because there is seasonality to the ozone concentration data. Also, this test being non-parametric (not requiring the data to fit any specific distribution) helps because ozone data often do not follow a normal distribution due to factors like pollution events, temperature anomalies, or regional meteorological conditions, which can result in skewed distributions.

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
ggplot(GaringerOzone.monthly, aes(x = Date,
                                  y = mean.oz)) +
  geom_point(color = "Blue") +
```

```
  geom_line()+
  labs(title = "Mean Monthly Ozone Concentrations over Time",
       x = "Date",
       y = "Ozone Concentration [ppm]")
```



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: Ozone concentrations have changed over the 2010s at this station. The p-value, which tests the null hypothesis that there is no trend in ozone concentrations, was calculated to be less than 0.05 (2-sided pvalue = 0.046724). This indicates that the decreasing trend is statistically significant. Thus, the null hypothesis can be rejected for the alternate hypothesis that there is indeed a significant downward trend in ozone concentrations over time.

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
seasonal_component <- GaringerOzone.monthly_Decomposed$time.series[, "seasonal"]
GaringerOzone.monthly.nonseasonal <- GaringerOzone.monthly.ts -
```

```
  seasonal_component

#16
monthly_ozone_trend2 <- Kendall::MannKendall(GaringerOzone.monthly.nonseasonal)
summary(monthly_ozone_trend2)
```

```
## Score =  -1179 , Var(Score) = 194365.7
## denominator =  7139.5
## tau = -0.165, 2-sided pvalue =0.0075402
```

Answer: Significance increased since the p value became even smaller after removing seasonality, suggesting that the trend in ozone concentrations is more likely to reflect a true underlying change, rather than seasonal or short-term variations (new p value = 0.0075402). Additionally, tau became even more negative (previous -0.143, now -0.165) suggests that the inverse relationship between the variables is stronger without the seasonality included.