Assignment 8: Time Series Analysis

Leah Li

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

```
#Check working directory
getwd()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

```
#Load packages
library(tidyverse)
library(lubridate)
library(zoo)
library(ggplot2)
library(trend)
library(here)

# Create a ggplot theme
my_theme <- theme_light() +
  theme(
    plot.title = element_text(face = "bold", size = 14, hjust = 0.5),
```

```
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    panel.grid.minor = element_blank(),
    legend.position = "bottom"
  )

# Set the theme as the default
theme_set(my_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

# Import each file individually

GaringerNC2010 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2011 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2012 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2013 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2014 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2015 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2016 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2017 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv"),
  stringsAsFactors = TRUE)

GaringerNC2018 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv"),
  stringsAsFactors = TRUE)
```

```
GaringerNC2019 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv"),
  stringsAsFactors = TRUE)

# Combine all dataframes into a single dataframe
GaringerOzone <- bind_rows(
  GaringerNC2010, GaringerNC2011, GaringerNC2012, GaringerNC2013,
  GaringerNC2014, GaringerNC2015, GaringerNC2016, GaringerNC2017,
  GaringerNC2018, GaringerNC2019
)

# Check the dimensions
dim(GaringerOzone)
```

```
## [1] 3589   20
```

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

# Convert the Date column to Date class
GaringerOzone$Date <- as.Date(GaringerOzone$Date, format = "%m/%d/%Y")

# 4

# Select only the required columns
GaringerOzone <- GaringerOzone %>%
  select(Date, Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)


# 5

# Create a sequence of dates from 2010-01-01 to 2019-12-31
Days <- as.data.frame(seq(as.Date("2010-01-01"), as.Date("2019-12-31"), by = "day"))

# Rename the column to "Date"
colnames(Days) <- "Date"

# 6
```

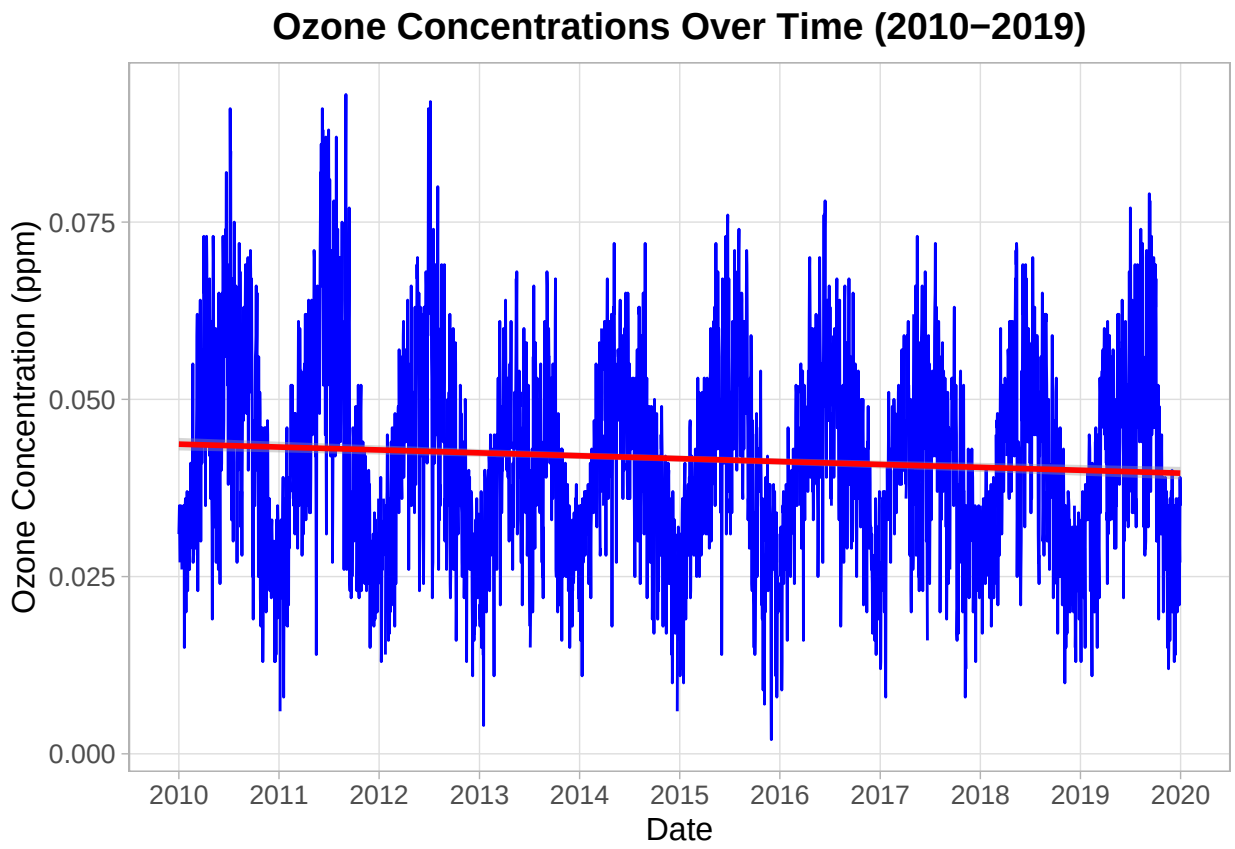
```r
# Use a left_join to combine Days and GaringerOzone
GaringerOzone <- left_join(Days, GaringerOzone, by = "Date")

# Ensure missing values are explicitly NA
GaringerOzone[is.na(GaringerOzone)] <- NA

# Check the dimensions to confirm the expected result
dim(GaringerOzone)
```

```
## [1] 3652    3
```

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

# Plotting the ozone concentrations
ggplot(GaringerOzone, aes(x = Date, y = Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line(color = "blue", size = 0.5) +
  geom_smooth(method = "lm", color = "red") + # Add linear trend line
  labs(title = "Ozone Concentrations Over Time (2010-2019)",
       x = "Date",
       y = "Ozone Concentration (ppm)") +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")   # Format x-axis for years
```



Answer: Based on the plot I created, there appears to be a slight downward trend in ozone concentration over time, as indicated by the red linear trend line. Although the overall seasonal variations within each year are quite prominent, with higher peaks in certain years, the trend line suggests a marginal decline in ozone concentrations from 2010 to 2019. This downward trend could imply a slight improvement in air quality with respect to ozone concentrations at this location, though the change is subtle.

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

# Perform linear interpolation to fill missing values
GaringerOzone$Daily.Max.8.hour.Ozone.Concentration<-
  zoo::na.approx(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration)
```

Answer: We used linear interpolation to fill in missing daily ozone concentration data because it provides a realistic estimate by assuming a steady rate of change between observed values. For example, when filling a gap between values like 0.027 ppm on January 5 and 0.030 ppm on January 7, linear interpolation smoothly transitions between these values, reflecting the likely gradual change in ozone levels rather than abrupt shifts.

Linear interpolation avoids the artificial "steps" that would occur if we used piecewise constant interpolation, which would hold each last observed value constant until the next data point. This approach is especially important for environmental data, where continuous, gradual changes are expected. Additionally, spline interpolation, though it can create smooth curves, may introduce oscillations or exaggerated fluctuations, misrepresenting trends in ozone concentrations. Linear interpolation thus offers a balance, filling missing values realistically without introducing artifacts.

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)

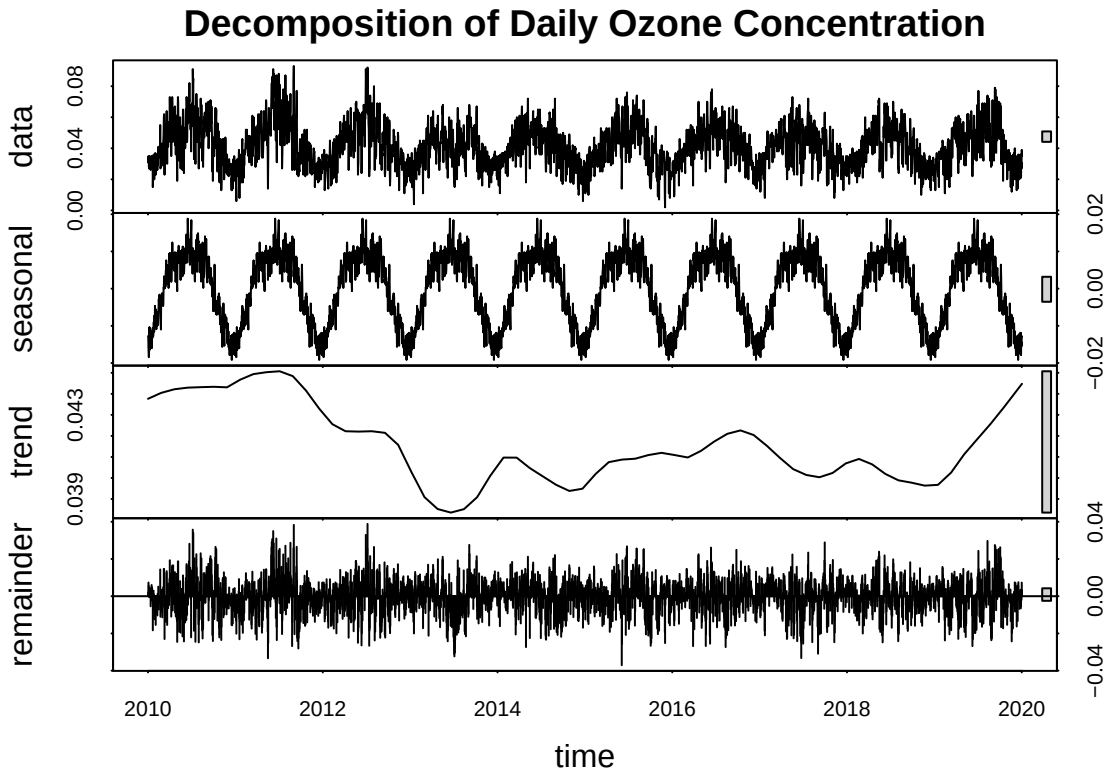
```
#9

# Create monthly aggregated data
GaringerOzone.monthly <- GaringerOzone %>%
  mutate(
    Year = year(Date),
    Month = month(Date)
  ) %>%
  group_by(Year, Month) %>%
  summarize(
    Monthly.Mean.Ozone.Concentration = mean(Daily.Max.8.hour.Ozone.Concentration, na.rm = TRUE)
  ) %>%
  ungroup()

# Create a Date column representing the first day of each month for graphing purposes
GaringerOzone.monthly <- GaringerOzone.monthly %>%
  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.

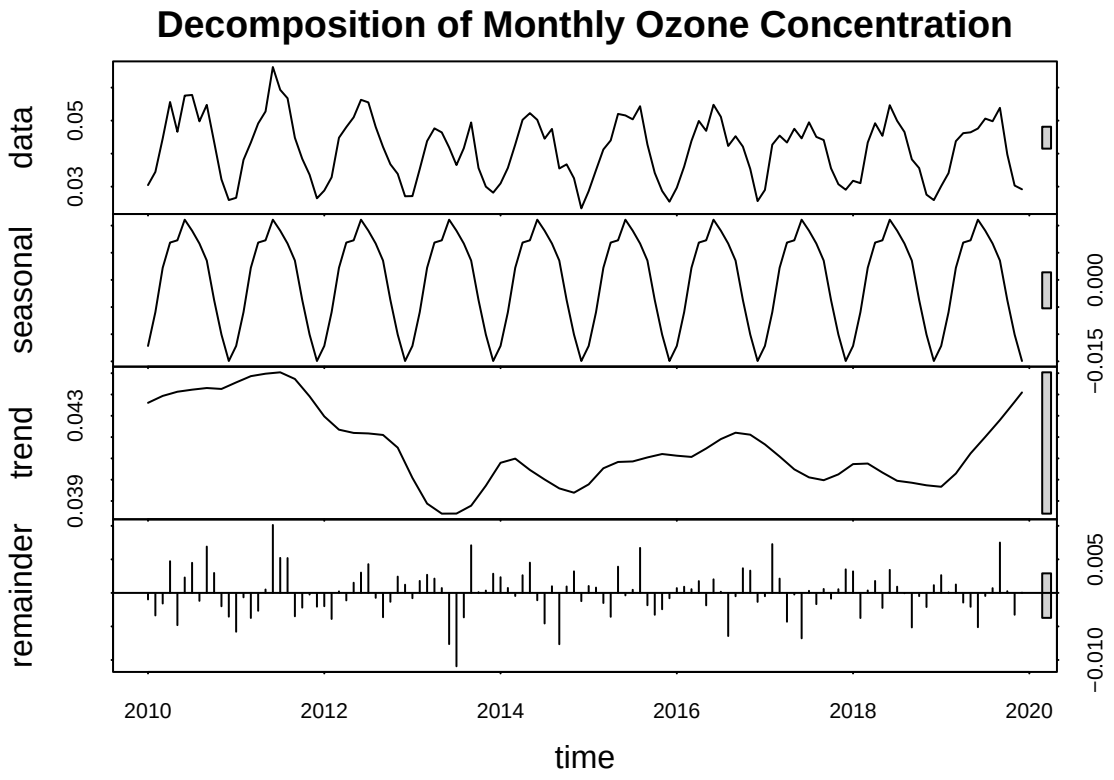
```
#10

# Generate daily time series
GaringerOzone.daily.ts <- ts(
  GaringerOzone$Daily.Max.8.hour.Ozone.Concentration,
  start = c(2010, 1),
  frequency = 365
)

# Generate monthly time series based on monthly averages
GaringerOzone.monthly.ts <- ts(
  GaringerOzone.monthly$Monthly.Mean.Ozone.Concentration,
  start = c(2010, 1),
  frequency = 12
)
```

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

```r
#11

# Set outer margins for the plot area
par(oma = c(0, 0, 4, 0))  # Adjust top margin for title

# Decompose and plot the daily time series
GaringerOzone.daily.decomp <- stl(GaringerOzone.daily.ts, s.window = "periodic")
plot(GaringerOzone.daily.decomp)
title("Decomposition of Daily Ozone Concentration",
      side = 3, line = 2, outer = TRUE, cex = 1.2)
```




```r
# Decompose and plot the monthly time series
GaringerOzone.monthly.decomp <- stl(GaringerOzone.monthly.ts, s.window = "periodic")
plot(GaringerOzone.monthly.decomp)
title("Decomposition of Monthly Ozone Concentration",
      side = 3, line = 2, outer = TRUE, cex = 1.2)
```



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

# Perform the Seasonal Mann-Kendall test on the monthly ozone time series
Ozone_trend <- trend::smk.test(GaringerOzone.monthly.ts)

# Print the results
print(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
```

```
summary(Ozone_trend)
```

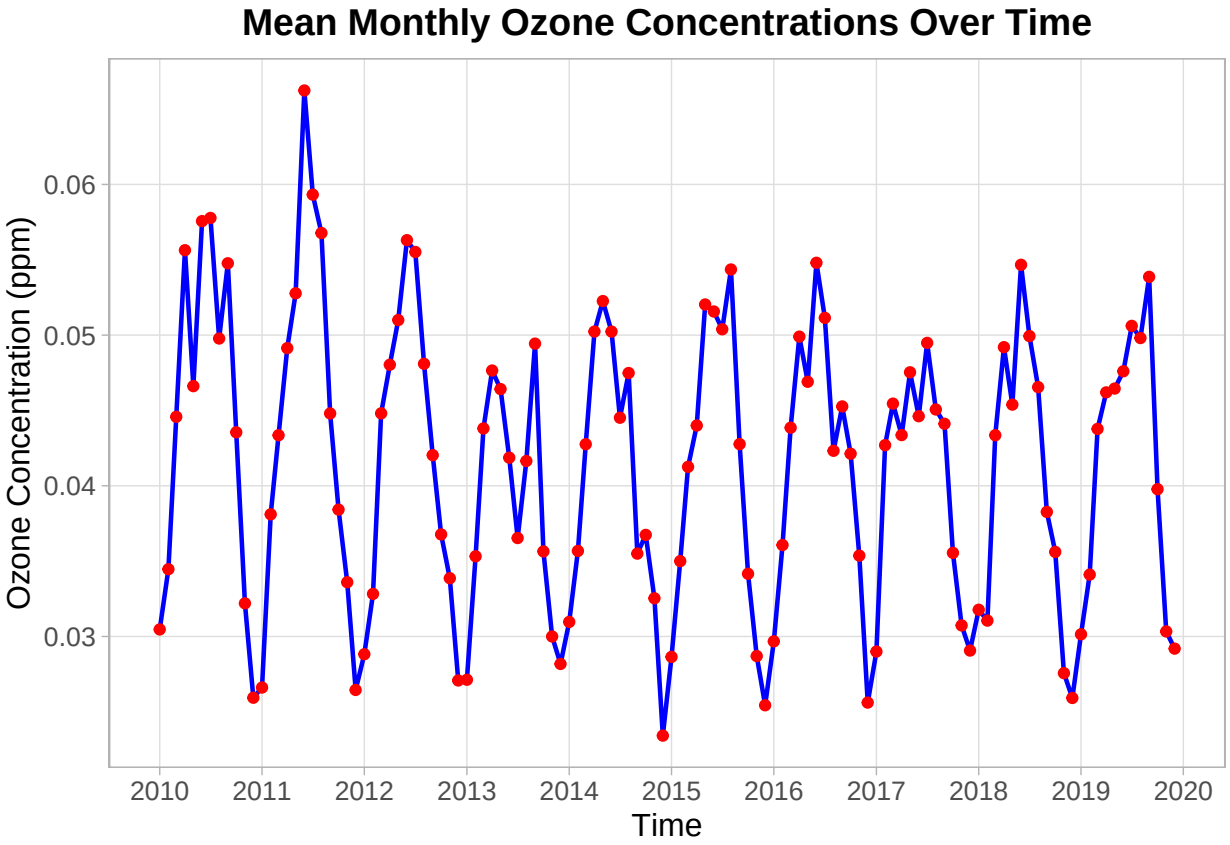
```
##
```

```
##  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   15  125  0.333  1.252  0.21050
## 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  -15  125 -0.333 -1.252  0.21050
## 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   -5  125 -0.111 -0.358  0.72051
## 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  11  125  0.244  0.894  0.37109
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Answer: The seasonal Mann-Kendall test is most appropriate for the monthly ozone series because it accounts for seasonality in the data. Ozone concentrations often show recurring seasonal patterns due to changes in weather conditions, sunlight, and atmospheric condition, which is also shown in our plot "Decomposition of Monthly Ozone Concentration" in the seasonal sub-plot. A standard trend test (like the basic Mann-Kendall test) does not account for these seasonal cycles and may incorrectly interpret seasonal fluctuations as part of a long-term trend. The seasonal Mann-Kendall test can evaluate trends within each season separately, for examplem, it compares only January values across all the years. This approach isolates the trend from the seasonal patterns and this is especially useful for monthly data where seasonality is a significant factor.

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.

```r
# 13
# Create the plot
ggplot(GaringerOzone.monthly, aes(x = Date, y = Monthly.Mean.Ozone.Concentration)) +
  geom_line(color = "blue", size = 0.8) +
  geom_point(color = "red", size = 1.5) +
  labs(
    title = "Mean Monthly Ozone Concentrations Over Time",
    x = "Time",
    y = "Ozone Concentration (ppm)"
  ) +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")
```



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 of mean monthly ozone concentrations over time shows fluctuations with a recurring seasonal pattern, reflecting typical variations throughout each year. However, there appears to be a slight downward trend in ozone levels over the decade. Based on the Seasonal Mann-Kendall test, there is evidence to suggest a slight downward trend in ozone concentrations over the 2010s at this station. The overall test result shows a negative trend statistic (S = -77) with a p-value of 0.04965, indicating that the trend is statistically significant at the 5% level. This suggests that, after accounting for seasonal variations, there has been a modest decrease in mean monthly ozone levels throughout the decade. However, when analyzing each season individually, none of the monthly trends are statistically significant, as shown by the higher p-values for each month. This indicates that the observed trend is consistent across seasons rather than driven by changes in any specific time of the year.

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

# Perform STL decomposition to extract components
```

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

# Subtract the seasonal component from the observed series
GaringerOzone_Nonseas <-
  GaringerOzone.monthly.ts - GaringerOzone.monthly.stl$time.series[, "seasonal"]

# Create a components data frame and add the seasonally adjusted data
GaringerOzone.monthly_Components <- as.data.frame(GaringerOzone.monthly.stl$time.series)
GaringerOzone.monthly_Components <- mutate(
  GaringerOzone.monthly_Components,
  Observed = as.numeric(GaringerOzone.monthly.ts),
  Date = time(GaringerOzone.monthly.ts),
  Nonseasonal = as.numeric(GaringerOzone_Nonseas)
)


#16
# Run the Mann-Kendall test
Ozone_Nonseas_trend <- trend::mk.test(GaringerOzone.monthly_Components$Nonseasonal)
# Print the test results
print(Ozone_Nonseas_trend)
```

```
##
##  Mann-Kendall trend test
##
## data:  GaringerOzone.monthly_Components$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: The results of the Seasonal Mann-Kendall test on the original monthly ozone series indicate a modest, statistically significant downward trend in ozone concentrations over the 2010s, with a test statistic (S) of -77 and a p-value of 0.04965. This suggests a decline in ozone levels at this station, although the trend is relatively weak. Seasonal fluctuations, evident in the time series plot, likely influenced this result, as the test adjusts for seasonality by analyzing each month separately.

When we remove seasonality from the data and run the Mann-Kendall test on the seasonally adjusted (non-seasonal) ozone series, we see a much stronger downward trend. The test statistic (S) is more negative (-1179), and the p-value is much lower (0.00754), indicating a highly significant trend at the 1% level. This result suggests that, without the seasonal variations, the underlying decline in ozone levels over time is more pronounced. In other words, seasonality may have partially masked the trend in the original data, and the seasonally adjusted analysis provides clearer evidence of a consistent decline in ozone concentrations throughout the 2010s.