Assignment 7: Time Series Analysis

Hanna Bliska

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
getwd() #checking wd
```

```
## [1] "/Users/hbliska/Desktop/EDA-Fall2022"
```

```
#loading packages
library(tidyverse)
library(lubridate)
library(zoo)
library(Kendall)
library(trend)
library(scales)
```

```
#setting my theme
mytheme <- theme_classic(base_size = 12) + theme(
  axis.text = element_text(color="black"),
  legend.position = "right") #building my theme
theme_set(mytheme) #setting my theme

#2
#importing datasets
O3_2010 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv", stringsAsFactors = TRUE)

O3_2011 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv", stringsAsFactors = TRUE)

O3_2012 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv", stringsAsFactors = TRUE)

O3_2013 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv", stringsAsFactors = TRUE)

O3_2014 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv", stringsAsFactors = TRUE)

O3_2015 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv", stringsAsFactors = TRUE)

O3_2016 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv", stringsAsFactors = TRUE)

O3_2017 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv", stringsAsFactors = TRUE)

O3_2018 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv", stringsAsFactors = TRUE)

O3_2019 <- read.csv(
  "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv", stringsAsFactors = TRUE)

#using `rbind` to combine the datasets into single dataframe
GaringerOzone <- rbind(
  O3_2010, O3_2011, O3_2012, O3_2013,
  O3_2014, O3_2015, O3_2016, O3_2017,
  O3_2018, O3_2019)

#checking dimensions of dataframe
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
#formatting date column to be a date class
GaringerOzone$Date <-as.Date(
  GaringerOzone$Date, format="%m/%d/%Y")

#checking class
class(GaringerOzone$Date)
```

```
## [1] "Date"
```

```
# 4
#using select to wrangle dataset for specific columns
GaringerOzone <- select(
  GaringerOzone, Date, Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)

# 5
#using `as.data.frame` to create a data frame
#data frame has one column with every day from 2010-01-01 to
#2019-12-31
Days <- as.data.frame(seq(as.Date("2010-01-01"), as.Date("2019-12-31"),"days"))

#renaming column
colnames(Days) <- c("Date")

# 6
#using `left_join` to join the datasets
#listing Days first to preserve the number of rows
#using by date because this is the column in common
GaringerOzone <- left_join(Days, GaringerOzone, by = c("Date"))

#checking dimensions
dim(GaringerOzone)
```

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

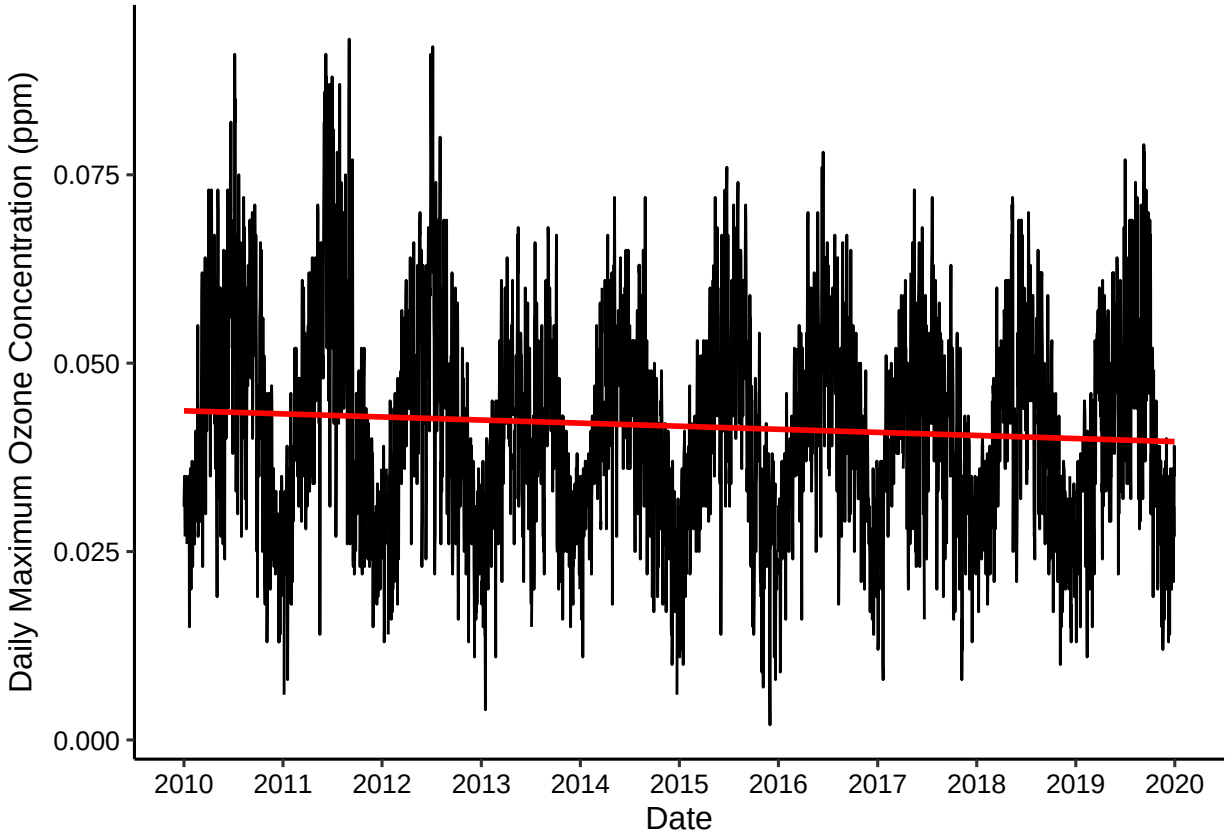
Visualize

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
Ozone_Line_Plot <- ggplot(GaringerOzone, aes(
  x=Date, y=Daily.Max.8.hour.Ozone.Concentration)) +
  #setting x and y axes
  geom_line() + #creating line plot
  scale_x_date(date_breaks="1 year", labels=date_format("%Y")) +
  #using `scale_x_date` to set a break for each year
  xlab(expression("Date")) + #setting x axis label
  ylab(expression ("Daily Maximum Ozone Concentration (ppm)")) +
  #setting y axis label
  geom_smooth(method=lm, se = FALSE, color="red") #adding linear trend
print(Ozone_Line_Plot) #printing plot
```

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

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



Answer: The linear trend of my data suggests that ozone concentrations have slightly decreased over time. The distribution of my data also suggests that there is seasonal variation in my data, as shown by the up and down movement in the data that appears periodically.

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$Daily.Max.8.hour.Ozone.Concentration <-
  na.approx(
  GaringerOzone$Daily.Max.8.hour.Ozone.Concentration) #using`na.approx`
#to fill in missing daily data
```

Answer: We used a linear interpolation because we wanted to replace our NAs with ozone concentrations that fall between the ozone concentration measurements recorded immediately before and after each NA. This makes sense given that we have relatively few and infrequently spaced NAs. We did not use the piecewise constant method because this method would replace NAs with values equal to the measurement made nearest to that date, which would cause replicates in neighboring ozone concentrations rather than our desired measurement, which would fall between the ozone concentrations of the previous and next days and follow the trend of our data. Similarly, we did not use the spline interpolation because it uses a quadratic function to interpolate NAs; given that the distribution of our data linearily increases and then decreases over the seasons, it is more logical to use a linear function to interpolate NAs rather than a quadratic function.

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
#new dataframe with aggregate data
GaringerOzone.monthly <-
  GaringerOzone %>%
  mutate(month = month(Date)) %>% #adding month column
  mutate(year = year(Date)) %>% #adding a year column
  group_by(year,month) %>% #grouping by year and month
  summarise(meanOzoneConcentration = mean(Daily.Max.8.hour.Ozone.Concentration))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
#generating means for ozone
#for each month of each year

#separate line of code for month-year combinations
GaringerOzone.monthly <-
  GaringerOzone.monthly %>% #pipe with mutate to create a new column
  #combining month and year columns
  mutate("firstmonth"=my(paste0(month,"-",year)))
#paste0 allows me to ensure no spaces between month and year
```

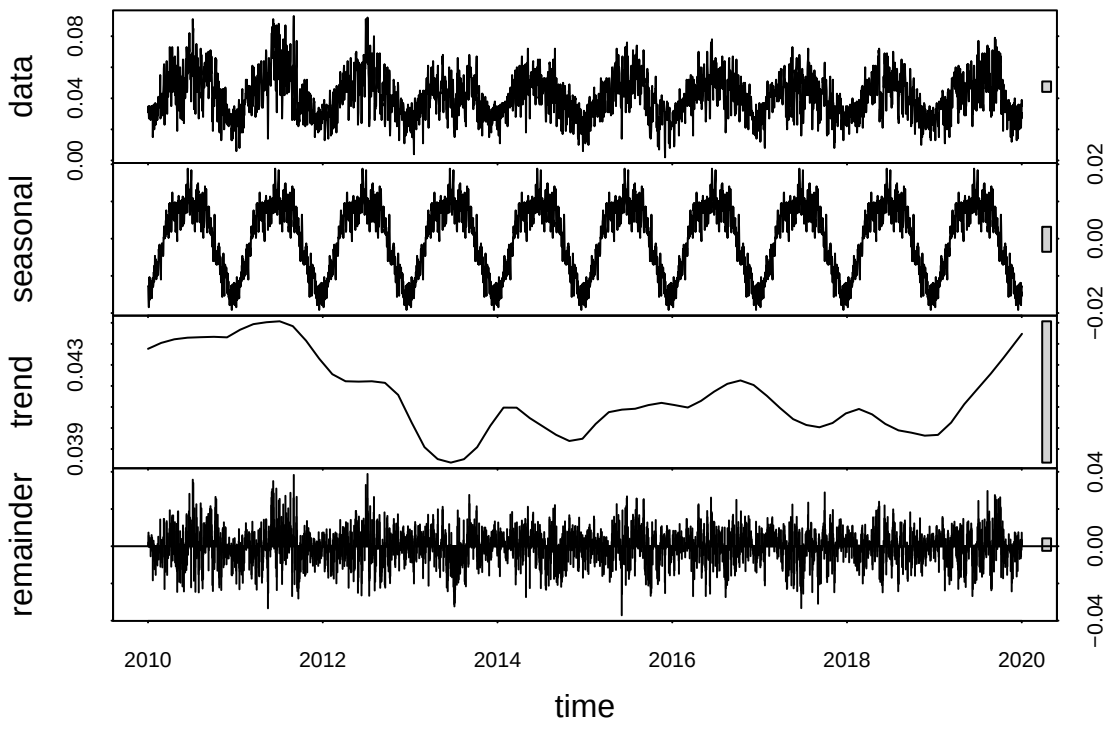
10. Generate two time series objects. Name the first `GaringerOzone.daily.ts` and base it on the dataframe of daily observations. Name the second `GaringerOzone.monthly.ts` and base it on the monthly average ozone values. Be sure that each specifies the correct start and end dates and the frequency of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

```r
#10
GaringerOzone.daily.ts <- ts(
  GaringerOzone$Daily.Max.8.hour.Ozone.Concentration,
  start=c(2010,1),frequency=365)
#generating time series object
#daily measurements, using 365

GaringerOzone.monthly.ts <- ts(
  GaringerOzone.monthly$meanOzoneConcentration,
  start=c(2010,1),frequency=12)
#generating time series object
#monthly measurements, using 12
```

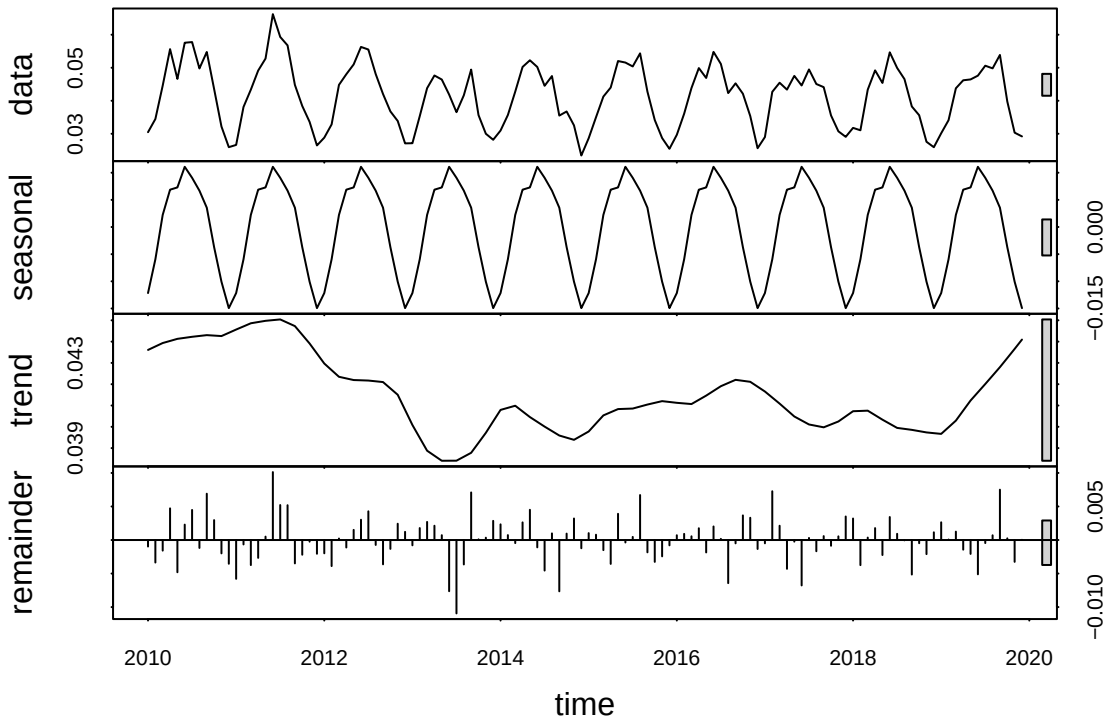
11. Decompose the daily and the monthly time series objects and plot the components using the `plot()` function.

```r
#11
GaringerOzone.daily.ts.Decomposed <- stl(
  GaringerOzone.daily.ts, s.window = "periodic")
#decomposing daily time series
plot(GaringerOzone.daily.ts.Decomposed) #plotting components
```




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

```
#decomposing monthly time series
plot(GaringerOzone.monthly.ts.Decomposed) #plotting components
```



12. Run a monotonic trend analysis for the monthly Ozone series. In this case the seasonal Mann-Kendall is most appropriate; why is this?

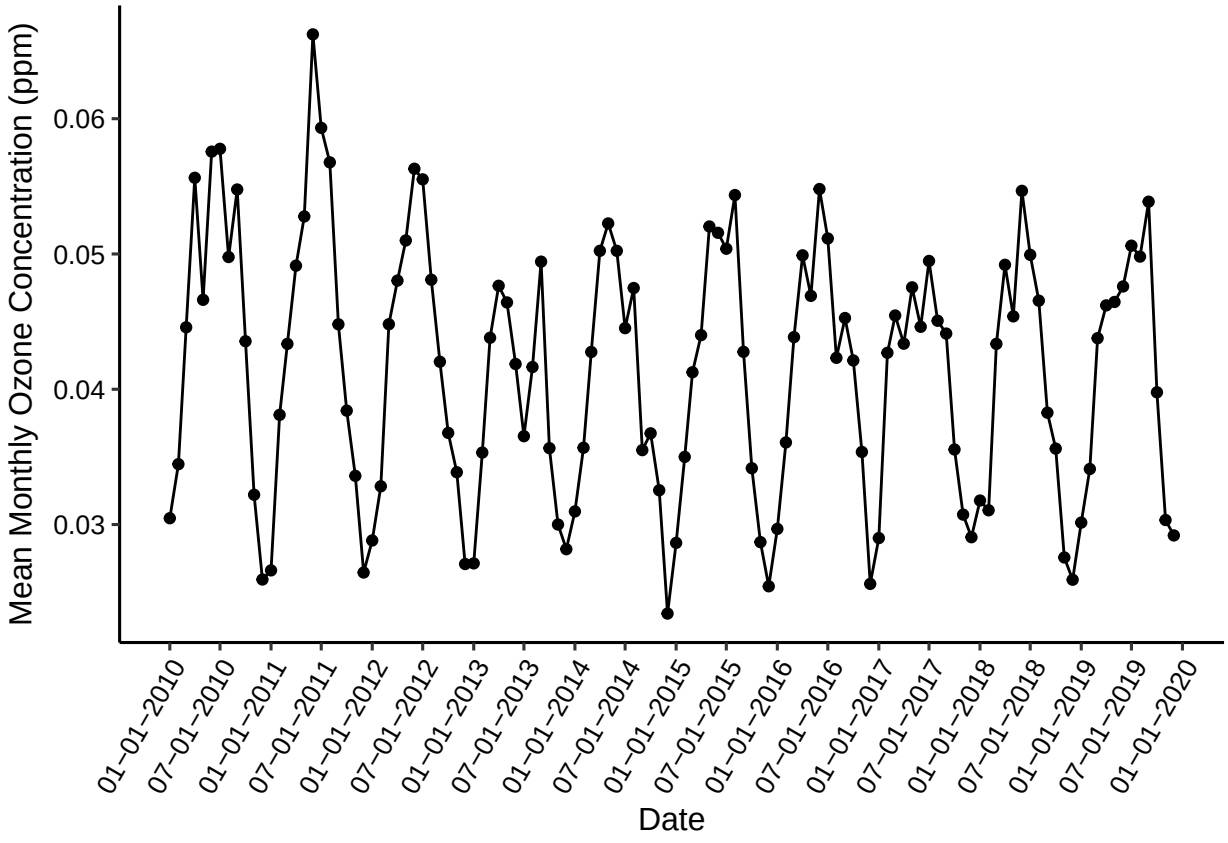
```
#12
Trend.Monthly.GaringerOzone <-
  SeasonalMannKendall(
  GaringerOzone.monthly.ts) #running seasonal Mann-Kendall
summary(Trend.Monthly.GaringerOzone) #summary to display results
```

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

Answer: We ran a sesonal Mann-Kendall test because monthly ozone concentrations have a seasonal cycle as demonstrated by the plot of the seasonal component produced in #11. In the components plot, a relatively smaller bar is present in the seasonal plot compared to the trend plot, indicating that the seasonal component explains more of the variability in the data than the trend. Because a seasonal Mann-Kendall test is most appropriate for determining monotonic trends in data with seasonality, which is the case for our data, we will use this test for our 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
Mean.Monthly.Ozone.Time.Plot <- ggplot(
  GaringerOzone.monthly, aes(
    x=firstmonth, y=meanOzoneConcentration)) + #setting axes
  geom_point() + #generating scatter plot
  geom_line() + #generating line plot
  scale_x_date(date_breaks="6 months", labels=date_format("%m-%d-%Y")) +
  #using `scale_x_date` to generate a break every 6 months
  #labels to show month-date-year
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  #angle for the day-month-year labels
  xlab(expression("Date")) + #setting x axis label
  ylab(expression ("Mean Monthly Ozone Concentration (ppm)"))
#setting y axis label
print(Mean.Monthly.Ozone.Time.Plot) #printing 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: The results of my trend analysis for mean monthly ozone concentrations indicates that there is a trend in the data over time (p=0.0467). In the plot in #13, we can see that the

trend in mean monthly ozone concentrations is decreasing over time. This decreasing trend is confirmed by the negative s statistic produced by the seasonal Mann-Kendall test (s=-77). In the plot from #13, the decreasing trend is particularly visible when comparing mean monthly ozone concentrations between the summer seasons over time; we can see that in 2010 and 2011, the mean monthly ozone concentrations in the summer months were higher than those observed in the summer months of 2017-2019. The results of our analysis allows us to reject the null hypothesis that the data are stationary and accept the alternative hypothesis that there is a significant trend in the data.

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
Monthly.GaringerOzone.NonseasonalComponents <- as.data.frame(
  GaringerOzone.monthly.ts.Decomposed$time.series[,2:3])
#extracting the trend and remainders, subtracting out seasonal
#this will allow me to have a dataframe without seasonal data

Monthly.GaringerOzone.NonseasonalComponents <-
  Monthly.GaringerOzone.NonseasonalComponents %>%
  mutate(data=trend + remainder) %>% #creating a column with
  #trend and remainder summed together
  mutate(date=GaringerOzone.monthly$firstmonth) %>%
  #adding date column
  select(data,date) #selecting columns of interest

Nonseasonal.GaringerOzone.monthly.ts <- ts(
  Monthly.GaringerOzone.NonseasonalComponents$data,
  start=c(2010,1),frequency=12) #generating time series object
#from the new dataframe we made of nonseasonal data
#frequency of 12 for 12 months per year

#16
Nonseasonal.Trend.Monthly.GaringerOzone <-
  MannKendall(Nonseasonal.GaringerOzone.monthly.ts)
#using `MannKendall` for trend analysis
summary(Nonseasonal.Trend.Monthly.GaringerOzone) #summary
```

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

Answer: The results of my trend analysis for non-seasonal mean monthly ozone concentrations indicates that there is a trend in the data over time (p=0.0075). Since p<0.05, we can reject the null hypothesis that the data are stationary and accept the alternative hypothesis that there is a trend in the data. We know that the trend is decreasing because the s statistic produced by the Mann-Kendall test is negative (s=-1179). Overall, without considering seasonality in the data, there is a significant decline in mean monthly ozone concentrations over time. We can also see that by comparing the results of the Mann-Kendall test on the non-seasonal data (p=0.0075) to the results of the Seasonal Mann-Kendall test on the seasonal data (p=0.0467) that

the decreasing trend in mean monthly ozone concentrations over time is much more significant when the seasonality is removed. This is likely because the seasonality of the data (with up and down cycles) has more noise and diminishes the strength of the trend in the seasonal data; once removed, this trend is determined to be stronger.