Assignment 8: Time Series Analysis

Victoria Thompson

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)
```

```
## -- 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(zoo)
```

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

```
library(trend)

# ggplot theme
mytheme <- theme_gray(base_size = 12) +
  theme(axis.text = element_text(color = "darkblue"),
        legend.position = "right",
         plot.title = element_text(face = "bold", size = 16,
                                   color = "black", hjust = 0.5))

# set as default
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.

```
#load individual data frames
EPA2019<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv",
                   stringsAsFactors = TRUE)
EPA2018<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv",
                   stringsAsFactors = TRUE)
EPA2017<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv",
                   stringsAsFactors = TRUE)
EPA2016<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv",
                   stringsAsFactors = TRUE)
EPA2015<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv",
                   stringsAsFactors = TRUE)
EPA2014<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv",
                   stringsAsFactors = TRUE)
EPA2013<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv",
                   stringsAsFactors = TRUE)
EPA2012<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv",
                   stringsAsFactors = TRUE)
EPA2011<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv",
                   stringsAsFactors = TRUE)
EPA2010<- read.csv("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv",
                   stringsAsFactors = TRUE)

#combine data frames
EPA_combined <- rbind(EPA2019,EPA2018,EPA2017,EPA2016,EPA2015,EPA2014,EPA2013,
                      EPA2012,EPA2011,EPA2010,
```

```
                    stringsAsFactors = TRUE)

#check dataframe size
EPA_combined_dim <- dim(EPA_combined)
print(EPA_combined_dim)
```

```
## [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 Set date column as a date class using lubridate
EPA_combined$Date <- mdy(EPA_combined$Date)

# 4 Wrangle dataset
EPA_combined_wrangled <- EPA_combined %>% select(Date,
                     Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)

# 5 New data frame with dates 2010-01-01 to 2019-12-32
Days <- as.data.frame((seq(as.Date("2010-01-01"),
                           as.Date("2019-12-31"), by = "day")))
#rename date column
colnames(Days) <- "Date"

# 6 Join data frames
GaringerOzone <- left_join(Days, EPA_combined_wrangled, by = "Date")

#check dimensions
garinger_ozone_dim <- dim(GaringerOzone)
print(garinger_ozone_dim)
```

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

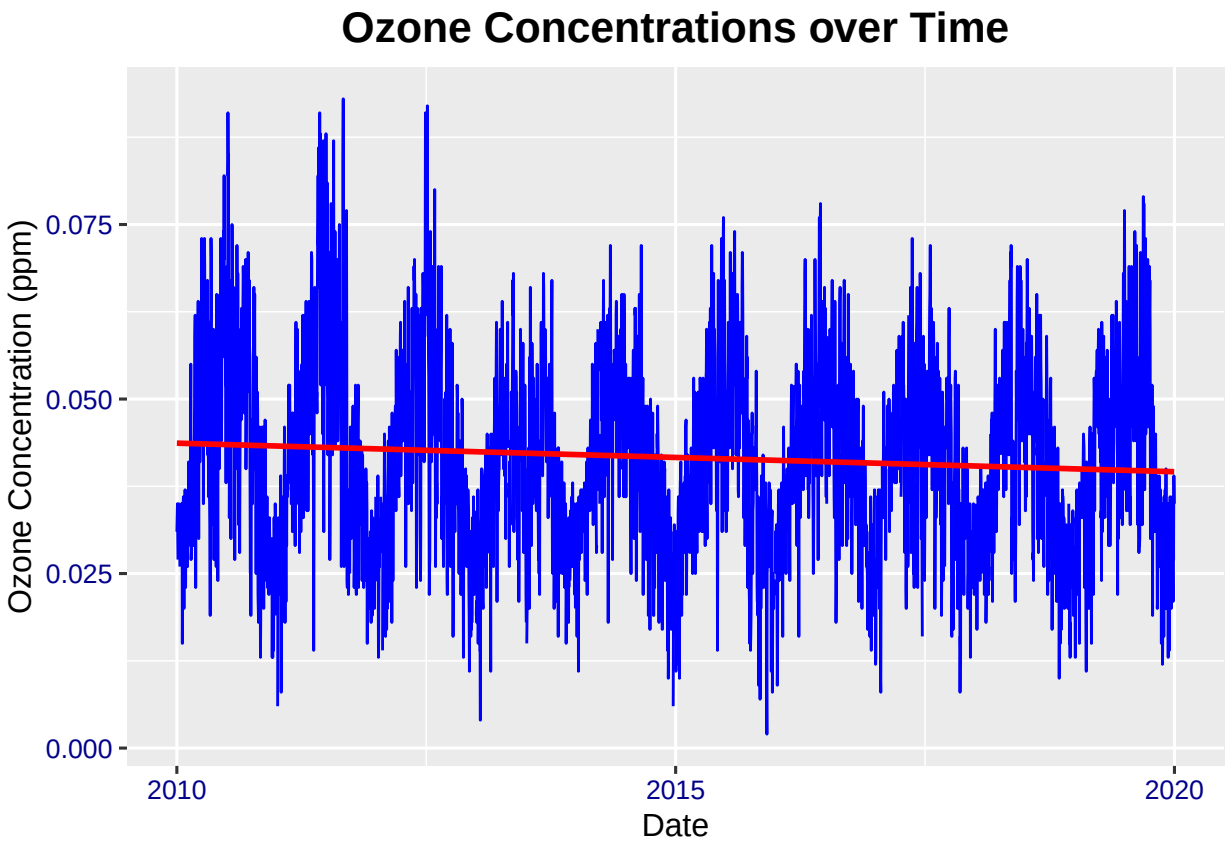
Visualize

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.vs.time <- ggplot(data = GaringerOzone, aes(x = Date,
                                    y = Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line(color = "blue") +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(
    title = "Ozone Concentrations over Time",
    x = "Date",
    y = "Ozone Concentration (ppm)"
  ) +
  mytheme

ozone.vs.time
```

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

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



Answer: Yes, the plot suggests an overall decrease in ozone concentration over time. The smoothed linear trend line (red) shows a stead decline despite the significant variation of the data (blue).

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
#adding an interpolated column
GaringerOzone.clean <-
  GaringerOzone %>%
  mutate(DMR.clean = zoo::na.approx(Daily.Max.8.hour.Ozone.Concentration))


#checking for no remaining NAs
summary(GaringerOzone.clean$DMR.clean)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00200 0.03200 0.04100 0.04151 0.05100 0.09300
```

Answer: We did not use piecewise interpolation, as that can result in values that change abruptly. Knowing that, in context, ozone concentrations do not change too much day-to-day, this would not be a helpful way to fill in gaps. We did not use a spline interpolation, as we are opertaing under the assumption that the relationship between date and concentration is mostly linear. Because of these reasons, a linear interpolation works best in the context of answering our research question.

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.clean  %>%
  # add columns for year and month
  mutate(
    year = year(Date),
    month = month(Date)
  ) %>%
  group_by(year, month) %>%
  # mean ozone conc. values per month
  summarize(
    mean_ozone_concentration = mean(DMR.clean,
                                    na.rm = TRUE)) %>%
  #adding first of month date column
  mutate(Date = as.Date(paste(year, month, "1", sep = "-")))
```

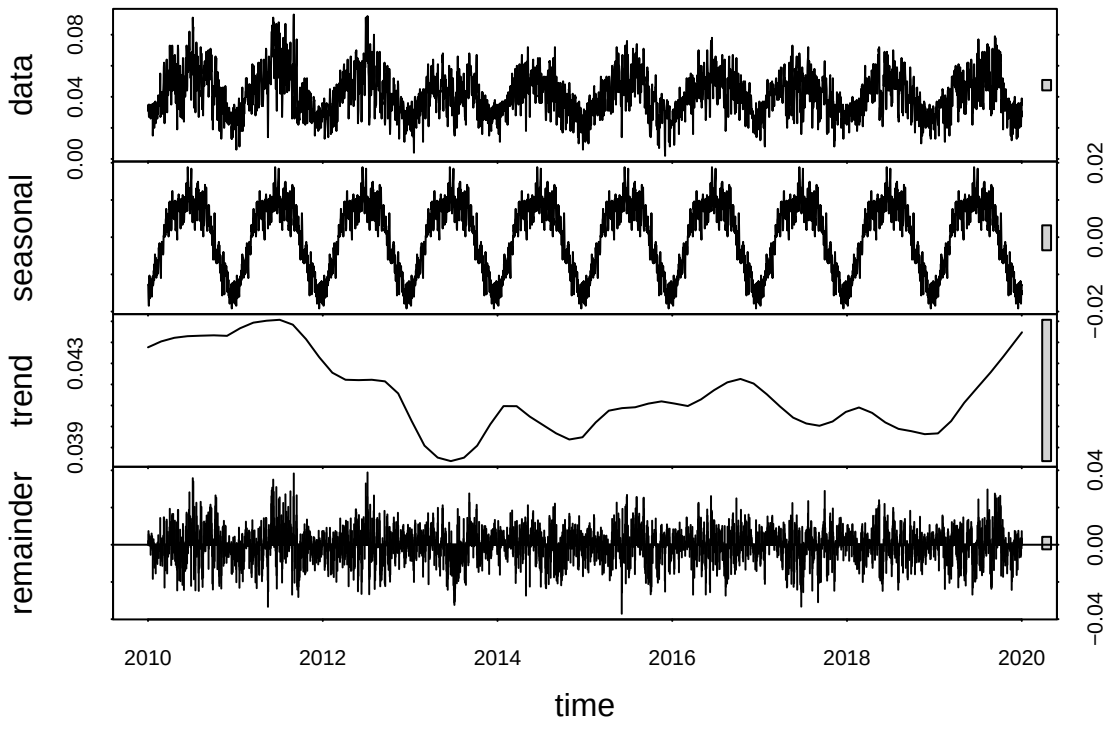
```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

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.

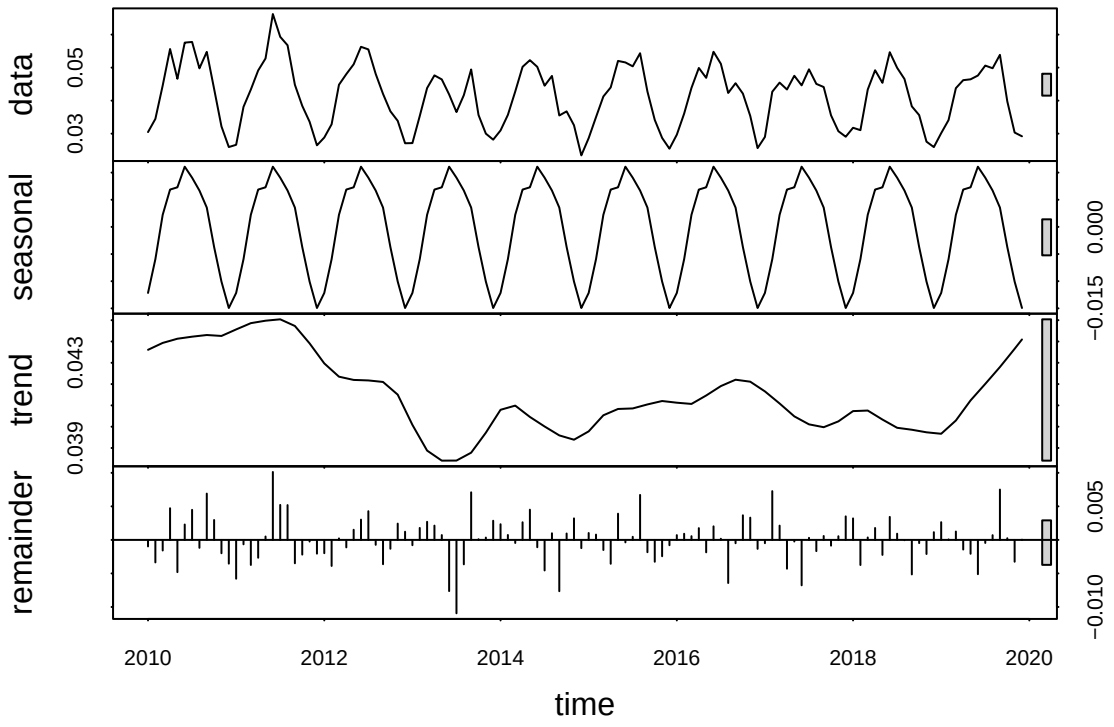
```r
#10

GaringerOzone.daily.ts <- ts(GaringerOzone.clean$DMR.clean,
                             frequency = 365,start=c(2010,1,1))
GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$mean_ozone_concentration,
                               frequency = 12,
                               start=c(2010, 1,1 ))
```

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

```r
#11
GaringerOzone.daily.decomposed <- stl(GaringerOzone.daily.ts,
                                      s.window = "periodic")
plot(GaringerOzone.daily.decomposed)
```




```r
GaringerOzone.monthly.decomposed <- stl(GaringerOzone.monthly.ts,
                                        s.window = "periodic")
plot(GaringerOzone.monthly.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
monthly_trend1 <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)

monthly_trend1
```

```
## tau = -0.143, 2-sided pvalue =0.046724
```

```
summary(monthly_trend1)
```

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

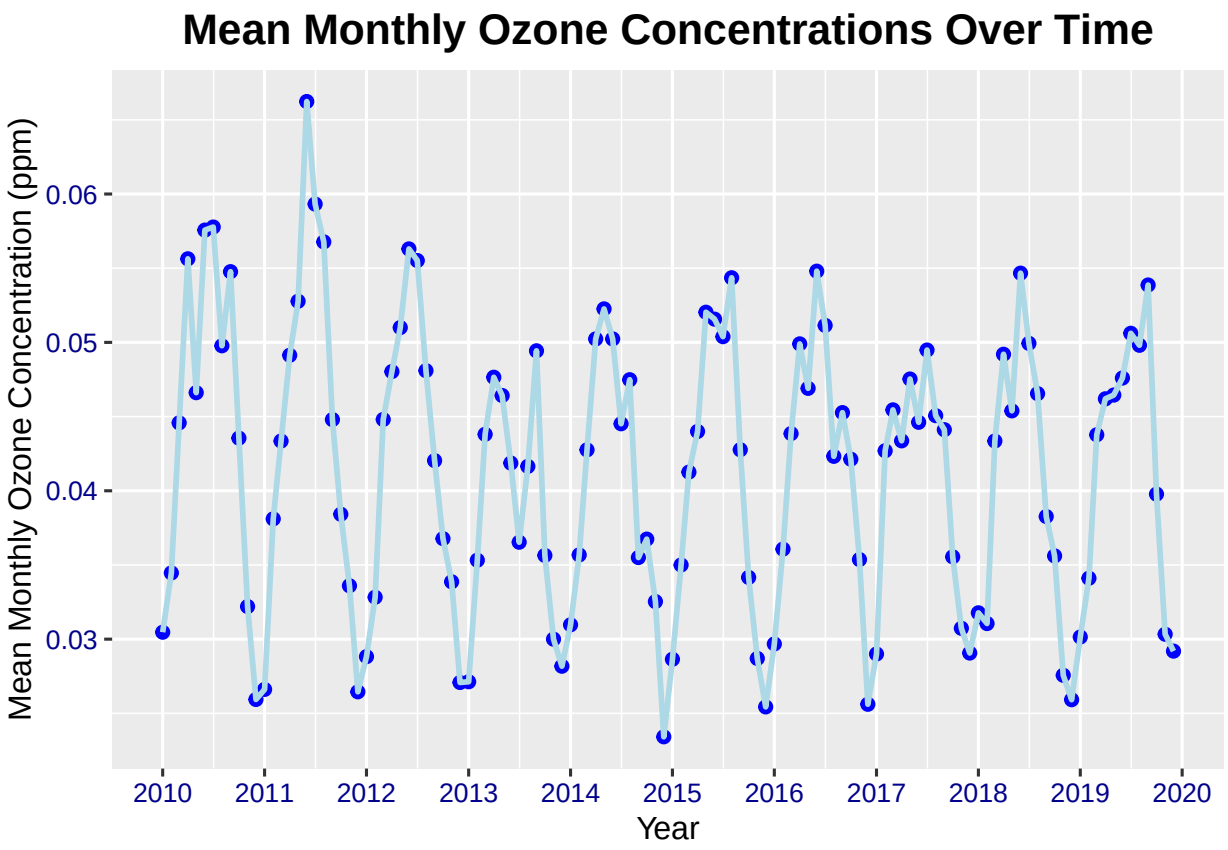
Answer: As apparent in the earlier plot of daily ozone concentrations, there is a seasonality to ozone concentrations. The seasonal Mann Kendall trend analysis takes this into account and is able to determine trends while with seasonal variability. It also does not allow for na values, which we do not have in this dataset. For these reasons, it is the most appropriate trend 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.

```r
# 13

mean.monthly.ozone <- ggplot(GaringerOzone.monthly, aes(x = Date, y = mean_ozone_concentration)) +
  geom_point(color = "blue", size = 2) +  # add points
  geom_line(color = "lightblue", size = 1) +   # add line
  scale_x_date( #adding every year to make trend more apparent
    date_breaks = "1 year",
    date_labels = "%Y") +
  labs(
    title = "Mean Monthly Ozone Concentrations Over Time",
    x = "Year",
    y = "Mean Monthly Ozone Concentration (ppm)"
  ) +
  mytheme
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```r
mean.monthly.ozone
```



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 the seasonal Mann-Kendall trend indicate a slight negative trend in ozone concentrations at the Garinger station over the 2010s (Tau = -0.143). The relationship was found to be significant (p = 0.046724, p < 0.05), meaning the null hypothesis can be rejected.

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 create new timeseries without seasonal

Ozone.Monthly.Nonseas.ts <- GaringerOzone.monthly.ts - GaringerOzone.monthly.decomposed$time.series[,1]

#16 run Mann Kendall test
monthly_nonseasonal <-
  Kendall::MannKendall(Ozone.Monthly.Nonseas.ts)

O3_Nonseas_trend <- Kendall::MannKendall(Ozone.Monthly.Nonseas.ts)
summary(O3_Nonseas_trend)
```

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

Answer: While there was still a general decreasing trend, the magnitude of the decrease was greater (tau = -0.165). Further, the results of the non-seasonal test were more significant (p = 0.007542, p > 0.05). Even though both tests resulted in different values, they both found the relationship between ozone and time to be significant.