# Assignment 8: Time Series Analysis

Brian Mulu Mutua

Fall 2023

## 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

```
#Checking working directory
getwd()
```

```
## [1] "C:/Users/bmm100/Documents/EDE_Fall2023"
```

```
#Loading necessary libraries
library(tidyverse)
library(lubridate)
library(zoo)
library(trend)


#Building theme
brian_theme_A08 <- theme_minimal(base_size = 14) +
  theme(
    #Modifying background colour
    plot.background = element_rect(fill = "Honeydew",
```

```
                                    colour = NA),
    panel.background =  element_rect(fill = "Honeydew",
                                     colour = "grey80"),
    legend.key =  element_rect(fill="Honeydew"),

    #Modifying legend
    legend.position = "right",
    legend.background = element_rect(colour = "grey10"),
    legend.text = element_text(colour = "grey10",
                               size = 11))

#Setting theme
theme_set(brian_theme_A08)
```

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.

```
#2 Importing and combining datasets

#Importing files in bulk
OzoneFiles = list.files(path = "./Data/Raw/Ozone_TimeSeries/", pattern="*.csv",
↪  full.names=TRUE)
OzoneFiles
```

```
##  [1] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv"
##  [2] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv"
##  [3] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv"
##  [4] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv"
##  [5] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv"
##  [6] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv"
##  [7] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv"
##  [8] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv"
##  [9] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv"
## [10] "./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv"
```

```
#Reading and combining files in one dataframe using ldply function
GaringerOzone <- plyr::ldply(OzoneFiles,read.csv)

#Displaying dimensions of created 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 Setting the date column as a date class

glimpse(GaringerOzone$Date) #Checking date format
```

```
##  chr [1:3589] "01/01/2010" "01/02/2010" "01/03/2010" "01/04/2010" ...
```

```
GaringerOzone$Date <- mdy(GaringerOzone$Date) #Adjusting date format
glimpse(GaringerOzone$Date) #Confirming successful conversion
```

```
##  Date[1:3589], format: "2010-01-01" "2010-01-02" "2010-01-03" "2010-01-04" "2010-01-05" ...
```

```
#4 Wrangling to only include columns Date, Daily.Max.8.hour.Ozone.Concentration, and
↪  DAILY_AQI_VALUE

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

#5 Creating daily dataset

#Creating sequence for objects of class "Date"
Days <- as.data.frame(seq(as.Date("2010-01-01"),as.Date("2019-12-31"),"days"))
#Renaming column
Days <- rename(Days, Date =`seq(as.Date("2010-01-01"), as.Date("2019-12-31"), "days")`)
glimpse(Days)
```

```
## Rows: 3,652
## Columns: 1
## $ Date <date> 2010-01-01, 2010-01-02, 2010-01-03, 2010-01-04, 2010-01-05, 2010~
```

```
#The length fo the GaringerOzone dataframe is less than the Days vector implying some
↪  days are missing

#6 Using a `left_join` to combine the data frames
GaringerOzone <- left_join(Days, GaringerOzone)
```

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

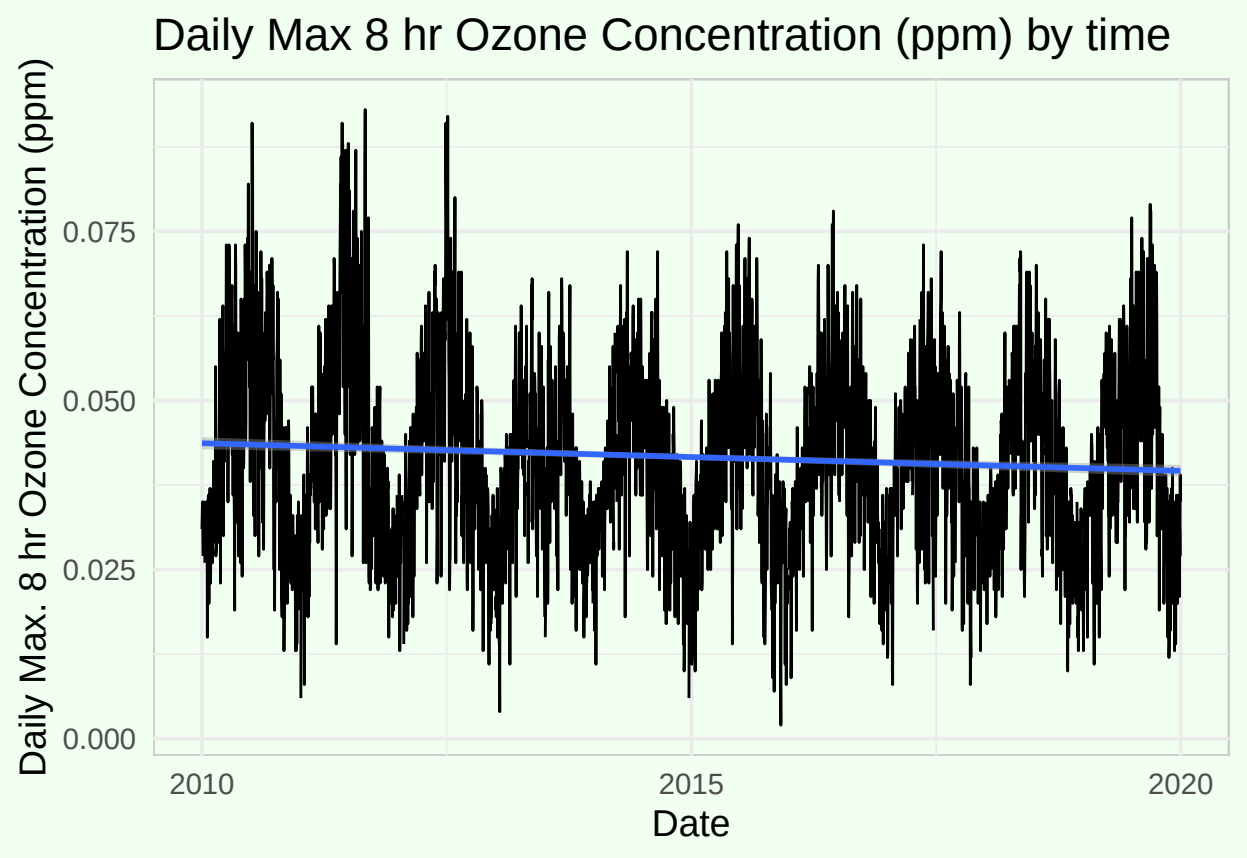
## Visualize

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?

```r
#7 Creating line plot

plot.GaringerOzone <- GaringerOzone %>%
  ggplot(aes(x=Date,y=Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line() +
  geom_smooth(method=lm) +
  labs(x="Date",
       y="Daily Max. 8 hr Ozone Concentration (ppm)",
       title="Daily Max 8 hr Ozone Concentration (ppm) by time")

print(plot.GaringerOzone)
```

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

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




Answer: The plot suggests that the daily maximum 8 hour ozone concentration in ppm is declining slightly over time as represented by the negative slope of the smoothed blue linear trend line.

### 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 Using linear interpolation to fill in missing daily data

#Checking for the NA's before interpolation
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
```

```
#Replacing NA's
GaringerOzone <-
  GaringerOzone %>%
  mutate(Daily.Max.8.hour.Ozone.Concentration =
  ↪  na.approx(Daily.Max.8.hour.Ozone.Concentration))
#Checking that the NA's have been removed after interpolation
summary(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration)
```

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

Answer: A piecewise constant interpolation was not used since the ozone concentrations values are not constant, hence changing over time. Linear interpolation allows us to match the trend of the existing dataset. Spline interpolation was not used since the pattern of varying ozone concentrations is not smooth hence does not resemble a spline.

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 Creating a new data frame called `GaringerOzone.monthly` that contains aggregated data
GaringerOzone.monthly <- GaringerOzone %>%
  mutate(Year = year(Date), Month = month(Date)) %>%
  mutate(Date = my(paste0(Month,"-",Year))) %>%
  group_by(Year, Month, Date) %>%
  summarise(Daily.Max.8.hour.Ozone.Concentration =
  ↪  mean(Daily.Max.8.hour.Ozone.Concentration))
```

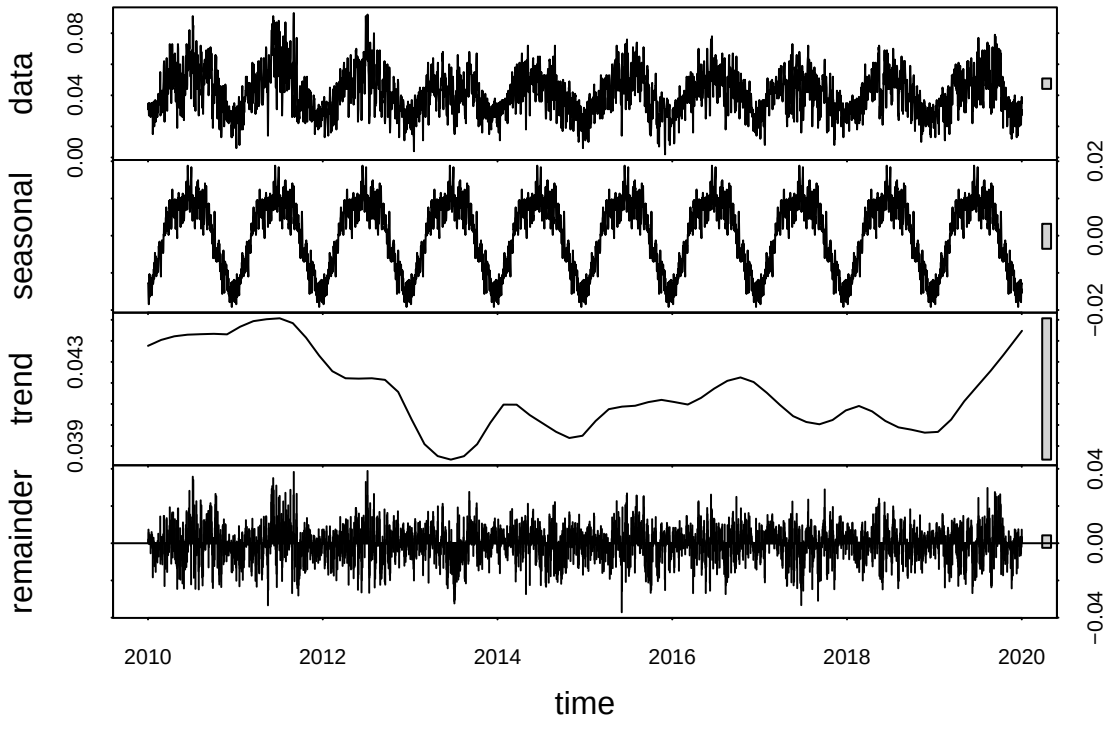
```
## `summarise()` has grouped output by 'Year', 'Month'. You can override using the
## `.groups` argument.
```

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.

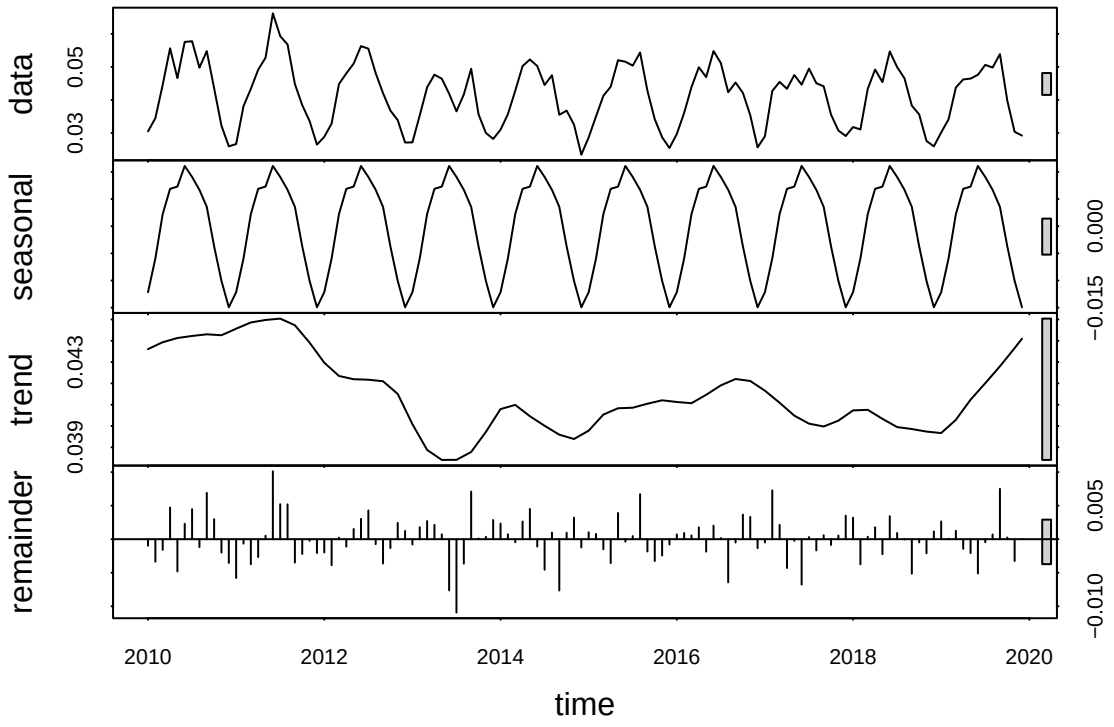
```r
#10 Generating the time series objects
first_month <- month(first(GaringerOzone$Date))
first_year <- year(first(GaringerOzone$Date))

GaringerOzone.daily.ts <- ts(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration,
                   start=c(first_year,first_month),
                   frequency=365)

GaringerOzone.monthly.ts <-
↪  ts(GaringerOzone.monthly$Daily.Max.8.hour.Ozone.Concentration,
                   start=c(first_year,first_month),
                   frequency=12)
```

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

```r
#11 Decomposing the daily and monthly time series objects and plotting

Ozone.daily.data.decomp <- stl(GaringerOzone.daily.ts,s.window = "periodic")
Ozone.monthly.data.decomp <- stl(GaringerOzone.monthly.ts,s.window = "periodic")

plot(Ozone.daily.data.decomp)
```



```
plot(Ozone.monthly.data.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 Running a monotonic trend analysis

# Running the seasonal Mann-Kendall test
Ozone.data.trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)

# Inspecting results
Ozone.data.trend
```

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

```
summary(Ozone.data.trend)
```

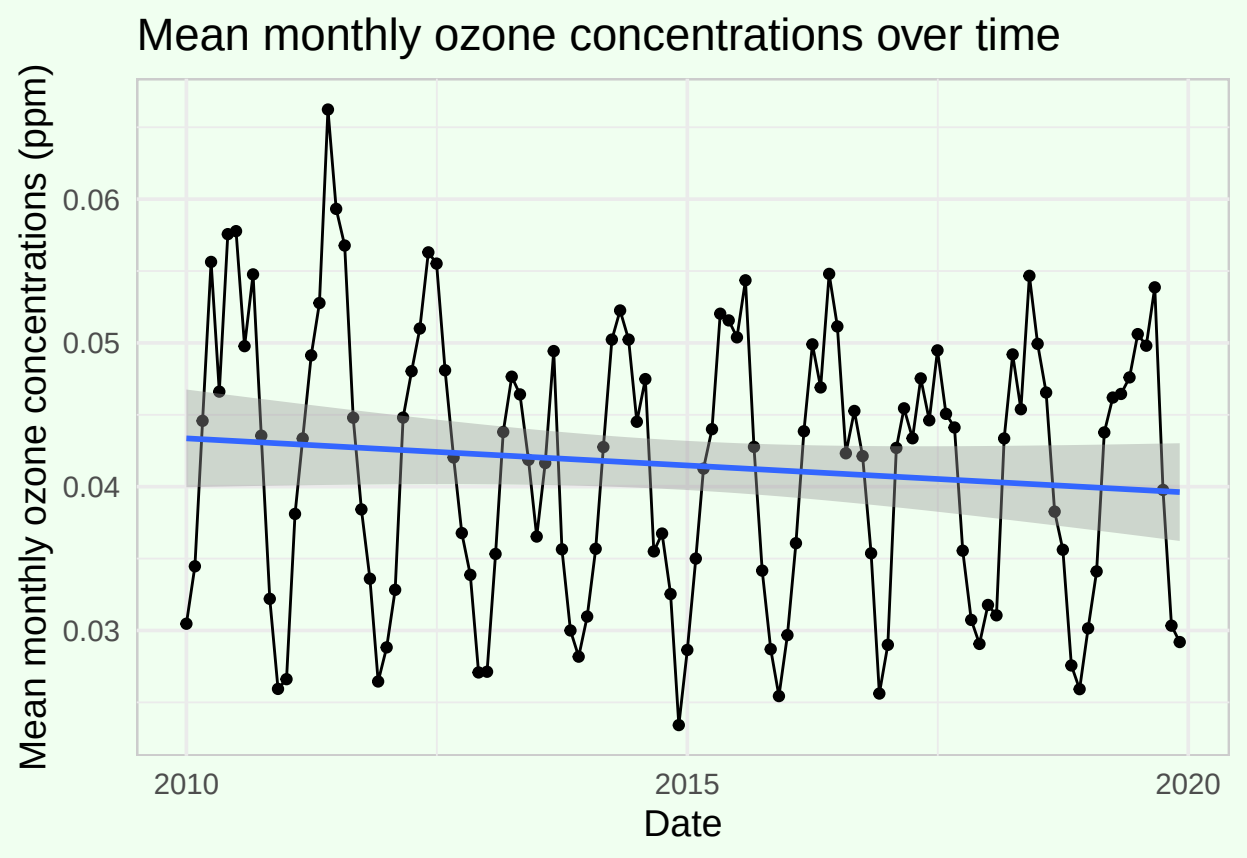
```
## Score =  -77 , Var(Score) = 1499
## denominator =  539.4972
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: The seasonal Mann-Kendall is most appropriate since the data has a seasonal component as observed in the time series plots.

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 Creating a plot depicting mean monthly ozone concentrations
Ozone.monthly.data.plot <- GaringerOzone.monthly %>%
  ggplot(aes(x = Date, y = Daily.Max.8.hour.Ozone.Concentration)) +
  geom_point() +
  geom_line() +
  labs(x = "Date",
  y = "Mean monthly ozone concentrations (ppm)",
  title = "Mean monthly ozone concentrations over time") +
  geom_smooth(method = lm)
print(Ozone.monthly.data.plot)
```

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




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 research question seeks to determine if ozone concentrations changed over the 2010s at this station. The null hypothesis is that there has been no change in the recorded ozone concentrations at this station over the 2010s. The alternative hypothesis is that there has been a change in the recorded ozone concentrations at this station over the 2010s. Based on the

seasonal Mann-Kendall (SMK) test, the p-value is less than 0.05 (<0.05), which implies that the results are statistically significant and we reject the null hypothesis. Therefore there has been a change in the recorded ozone concentrations at this station over the 2010s. The negative tau value suggests a negative correlation which implies that this change has been a decrease. (tau = -0.143, 2-sided pvalue =0.046724)

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 Subtracting the seasonal component
GaringerOzone.monthly.nonseasonalts <- GaringerOzone.monthly.ts -
↪  Ozone.monthly.data.decomp$time.series[,1]

GaringerOzone.monthly.nonseasonalts
```

```
##             Jan        Feb        Mar        Apr        May        Jun
## 2010 0.04263190 0.04041003 0.04234881 0.04875492 0.03932081 0.04647348
## 2011 0.03877706 0.04405289 0.04112300 0.04225492 0.04548211 0.05514015
## 2012 0.04098674 0.03877333 0.04257462 0.04115492 0.04370791 0.04520681
## 2013 0.03929319 0.04126717 0.04157462 0.04077159 0.03912727 0.03077348
## 2014 0.04313190 0.04162432 0.04052623 0.04335492 0.04496598 0.03914015
## 2015 0.04080932 0.04094575 0.03902623 0.03712159 0.04474017 0.04047348
## 2016 0.04184158 0.04201471 0.04162300 0.04302159 0.03961114 0.04370681
## 2017 0.04116416 0.04864217 0.04321978 0.03648826 0.04024017 0.03352348
## 2018 0.04393835 0.03699932 0.04112300 0.04232159 0.03809501 0.04357348
## 2019 0.04230932 0.04005289 0.04154236 0.03932159 0.03915952 0.03650681
##             Jul        Aug        Sep        Oct        Nov        Dec
## 2010 0.04871023 0.04307797 0.05120815 0.04725211 0.04226527 0.04087082
## 2011 0.05025862 0.05007797 0.04124148 0.04212308 0.04366527 0.04138695
## 2012 0.04645216 0.04140056 0.03847481 0.04047792 0.04393193 0.04201598
## 2013 0.02746829 0.03494894 0.04587481 0.03934888 0.04006527 0.04311275
## 2014 0.03545216 0.04078765 0.03194148 0.04044566 0.04259860 0.03835469
## 2015 0.04132313 0.04765862 0.03920815 0.03786501 0.03876527 0.04037082
## 2016 0.04208120 0.03562636 0.04170815 0.04583275 0.04543193 0.04054824
## 2017 0.04041991 0.03836830 0.04055815 0.03925211 0.04079860 0.04399985
## 2018 0.04087152 0.03985217 0.03470815 0.03931662 0.03763193 0.04085469
## 2019 0.04154894 0.04311023 0.05030815 0.04347792 0.04039860 0.04412888
```

```
#16 Running the Mann Kendall test on the non-seasonal Ozone monthly series
Ozone.data.trend.q16 <- Kendall::MannKendall(GaringerOzone.monthly.nonseasonalts)

# Inspecting results
Ozone.data.trend.q16
```

```
## tau = -0.165, 2-sided pvalue =0.0075402
```

```
summary(Ozone.data.trend.q16)
```

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

Answer: Based on the Mann-Kendall (SMK) test on the non-seasonal ozone monthly series, the p-value is less than 0.05 (<0.05) with a negative correlation which is consistent with the earlier output from the seasonal Mann-Kendall (SMK) test. The results are still statistically significant and we reject the null hypothesis. Omitting the seasonal component, there has still been a decrease in the recorded ozone concentrations at this station over the 2010s. (tau = -0.165, 2-sided pvalue =0.0075402)