Assignment 8: Time Series Analysis

Maia Griffith

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

```
# Working directory
library(here)
```

```
## here() starts at /home/guest/ENV872/maia-g/EDE_Fall2023
```

```
here()
```

```
## [1] "/home/guest/ENV872/maia-g/EDE_Fall2023"
```

```
# Loading packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## 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)

# Setting a theme
MGtheme <- theme_light(base_size = 11) +
  theme(axis.text = element_text(color = "darkgray"),
        legend.position = "right")
theme_set(MGtheme)
```

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
# Import 10 datasets
Raw.2010 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2011 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2012 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2013 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2014 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2015 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2016 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2017 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2018 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv"),
                     stringsAsFactors = TRUE)
Raw.2019 <- read.csv(here("Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv"),
                     stringsAsFactors = TRUE)

# Combine into one data frame
```

```
GaringerOzone <- rbind(Raw.2010, Raw.2011, Raw.2012, Raw.2013, Raw.2014,
                       Raw.2015, Raw.2016, Raw.2017, Raw.2018, Raw.2019)
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
# Change into date format using lubridate
GaringerOzone$Date <- mdy(GaringerOzone$Date)
class(GaringerOzone$Date)
```

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

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

# 5
# Get the start and end dates
start.date <- first(GaringerOzone.selected$Date)
end.date <- last(GaringerOzone.selected$Date)

# Create new data frame called Days
Days <- as.data.frame(seq(from = start.date, to = end.date, by = 1))

# Change column name
colnames(Days) <- "Date"

# 6
# Left join the data frames
GaringerOzone <- left_join(Days, GaringerOzone.selected)
```

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

```
dim(GaringerOzone)
```

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

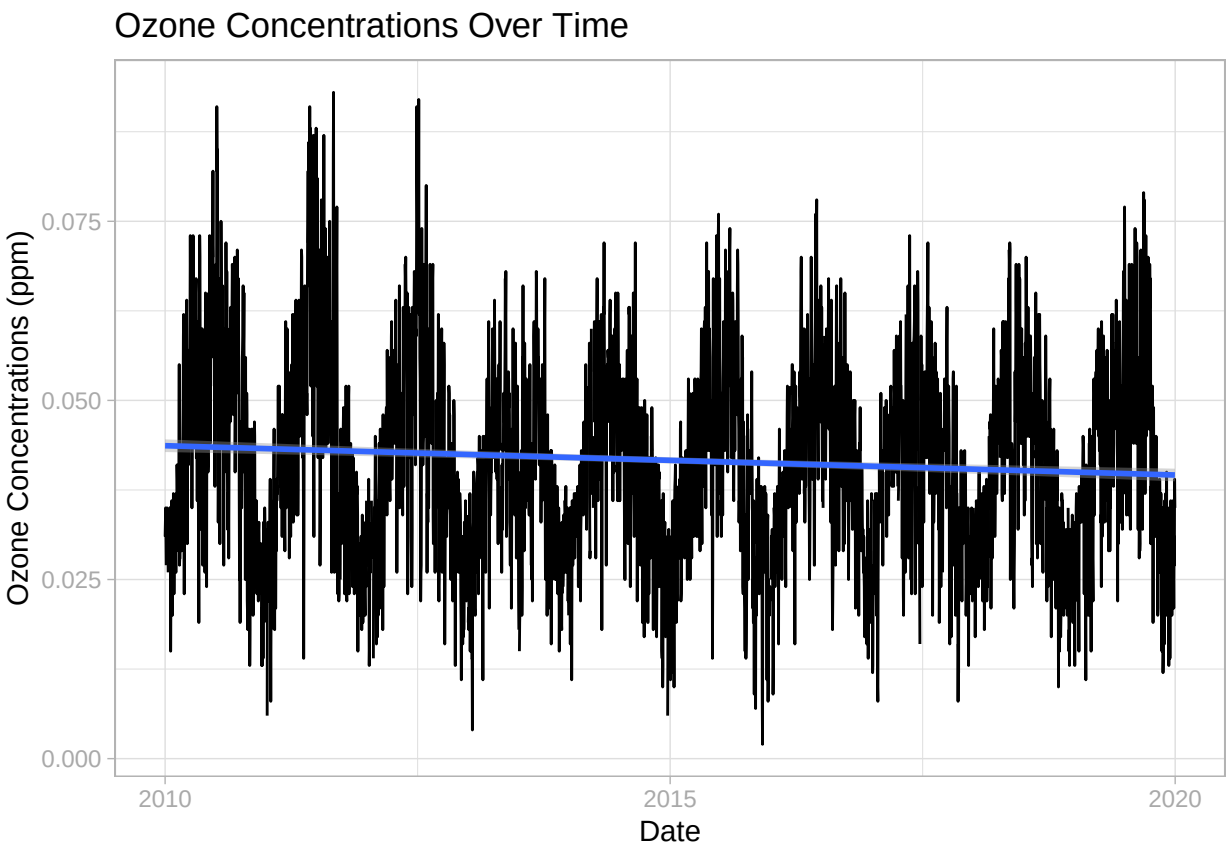
Visualize

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?

```
#7
Oz.by.time.plot <-
  ggplot(GaringerOzone,
         aes(x = Date,
             y = Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line() +
  geom_smooth(method = "lm") +
  labs(y = "Ozone Concentrations (ppm)",
       title = "Ozone Concentrations Over Time")

Oz.by.time.plot
```

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

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



Answer: The ozone concentrations appear to fluctuate seasonally, but over a long period of time have not changed much. If anything, there is a very slight negative 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

# Looking for NAs
sum(is.na(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration))
```

```
## [1] 63
```

```
# Cleaning the data using linear interpolation
GaringerOzone_clean <-
  GaringerOzone %>%
  mutate( Daily.Max.8.hour.Ozone.Concentration =
            zoo::na.approx(Daily.Max.8.hour.Ozone.Concentration) )

# Checking to make sure NAs are gone.
sum(is.na(GaringerOzone_clean$Daily.Max.8.hour.Ozone.Concentration))
```

```
## [1] 0
```

Answer: We didn't want to use a piecewise or spline because we want the interpolated data to hopefully still follow the seasonal trends of the actual data, so simply connecting the dots works best.

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
# Making a pipe
GaringerOzone.monthly <-
  GaringerOzone_clean %>%
  mutate(Month = month(Date),
         Year = year(Date)) %>% # Adding month and year cols
  group_by(Year, Month) %>% # Grouping by year and then month
  summarise(Mean_Oz = mean(Daily.Max.8.hour.Ozone.Concentration)) # Calc the mean
```

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

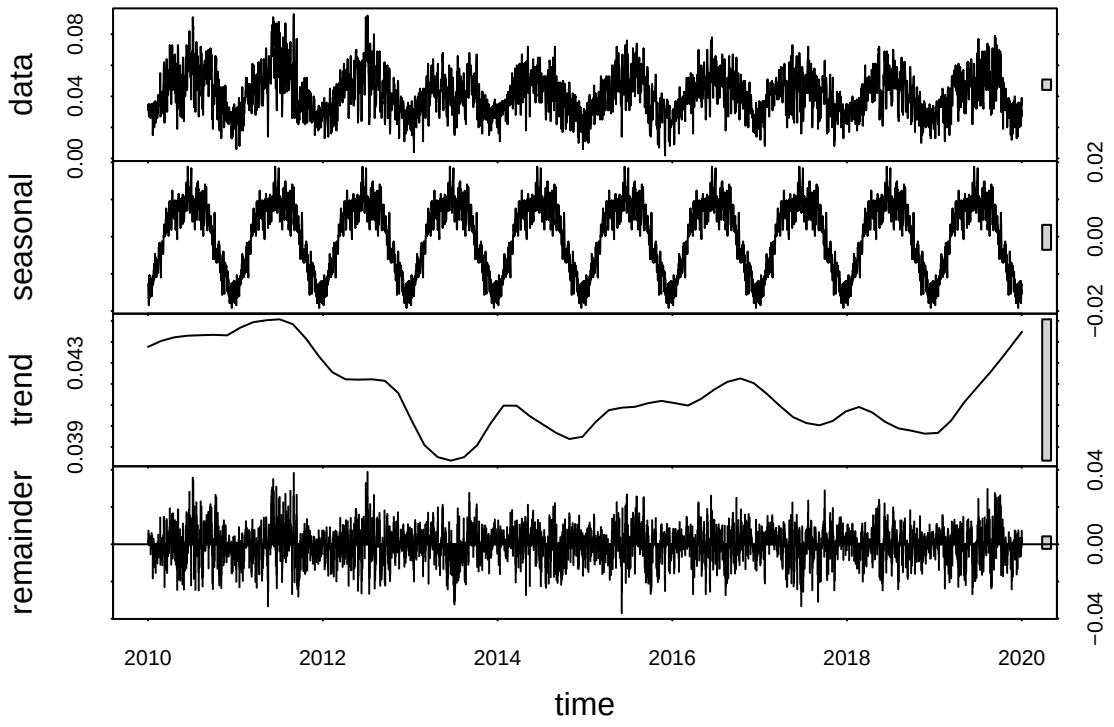
```r
# Create new Date column
GaringerOzone.monthly <-
  GaringerOzone.monthly %>%
  mutate(Date = make_date(Year, Month, 1))
```

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.

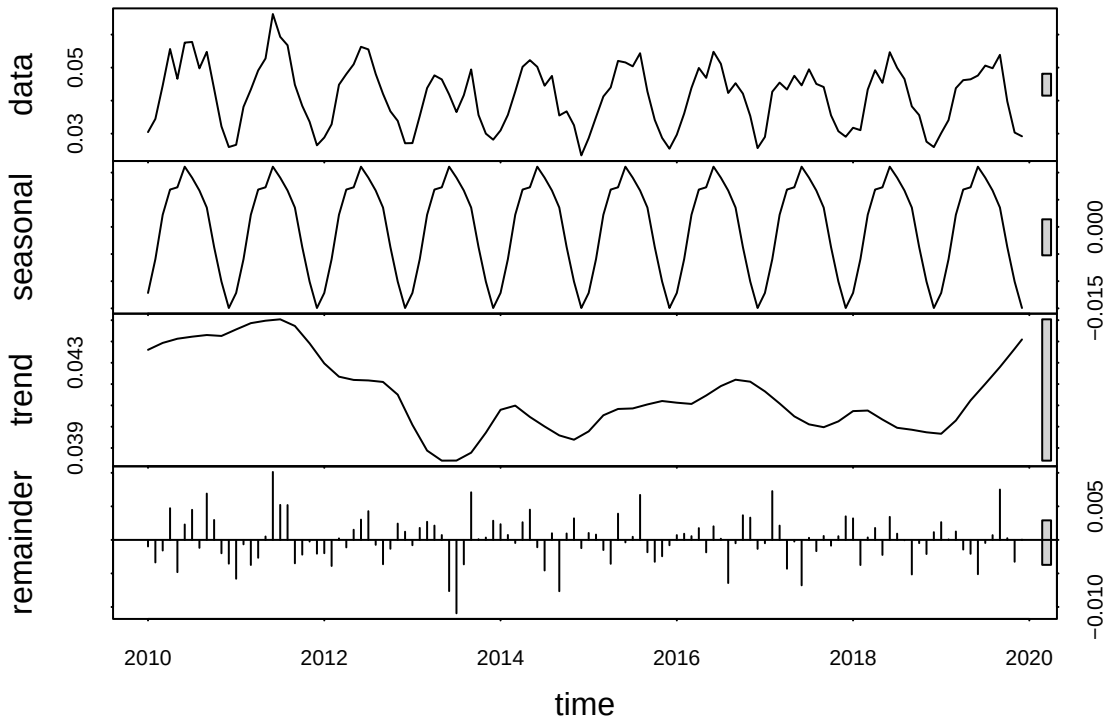
```r
#10
# Set variables with the first month and first year of the data sets
f_month <- month(first(GaringerOzone_clean$Date))
f_year <- year(first(GaringerOzone_clean$Date))

# Daily time series using cleaned (no NAs) daily data frame
GaringerOzone.daily.ts <- ts(GaringerOzone_clean$Daily.Max.8.hour.Ozone.Concentration,
                             start = c(f_year, f_month),
                             frequency = 365)

# Monthly time series using monthly data frame
GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$Mean_Oz,
                             start = c(f_year, f_month),
                             frequency = 12)
```

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

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




```
# Decomposing monthly next
monthly_data_decomp <- stl(GaringerOzone.monthly.ts,s.window = "periodic")
plot(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
monthly_Oz_trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)
monthly_Oz_trend
```

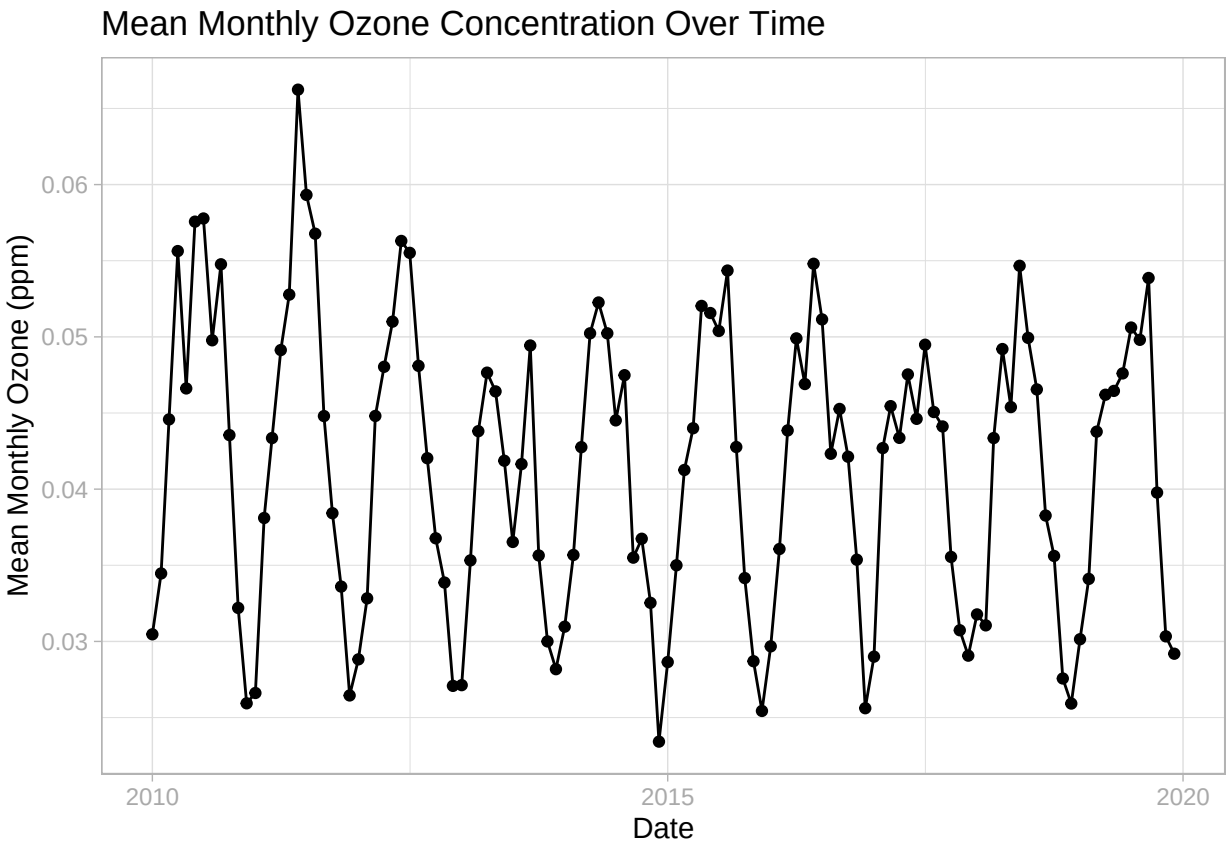
```
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: It is the only trend analysis for seasonal data.

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
monthly.Oz.time.plot <-
  ggplot(GaringerOzone.monthly,
         aes(x = Date,
             y = Mean_Oz)) +
  geom_point() +
  geom_line() +
#  geom_smooth(method = "lm") +
  labs(y = "Mean Monthly Ozone (ppm)",
       title = "Mean Monthly Ozone Concentration Over Time")
```

```
monthly.Oz.time.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: In this case, the null hypothesis would be that there was no change in ozone concentrations. The alternative would be that there was some change in ozone concentrations. The results of the Seasonal Mann-Kendall test revealed a p-value of less than 0.05, which means we can reject the null hypothesis and conclude that there is a significant decrease in ozone concentrations over the 2010s at this station (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
GaringerMonthly.Components <- as.data.frame(monthly_data_decomp$time.series[,1:3])

GaringerMonthly.nonseasonal <-
  GaringerMonthly.Components %>%
  mutate(Date = GaringerOzone.monthly$Date,
```

```
        Observed = GaringerOzone.monthly$Mean_Oz,
        Nonseasonal = Observed - seasonal)
#16
nonseasonal.monthly.ts <- ts(GaringerMonthly.nonseasonal$Nonseasonal,
                             start = c(f_year, f_month),
                             frequency = 12)
nonseasonal.monthly_Oz_trend <- Kendall::MannKendall(nonseasonal.monthly.ts)
nonseasonal.monthly_Oz_trend
```

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

Answer: After subtracting the seaosnal component, the Mann-Kendall test shows an even smaller p-value than the seasonal one. This means we can conclude with more certainty that there is a significant negative trend in ozone concentration over time (tau = -0.165, 2-sided pvalue =0.0075402).