Assignment 8: Time Series Analysis

Jack Eynon

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., "Salk_A06_GLMs_Week1.Rmd") prior to submission.

The completed exercise is due on Tuesday, March 3 at 1:00 pm.

Set up

1. Set up your session:

- Check your working directory
- Load the tidyverse, lubridate, zoo, and trend packages
- Set your ggplot theme
- 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). Call these GaringerOzone201*, with the star filled in with the appropriate year in each of ten cases.

```
getwd()
```

```
## [1] "/Users/jackeynon/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Assignments"
```

```
library(tidyverse)
library(lubridate)
library(zoo)
library(trend)

mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"))
theme_set(mytheme)

GaringerOzone2010 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2011 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2012 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2013 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2014 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2015 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2016 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
```

```
GaringerOzone2017 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2018 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
GaringerOzone2019 <- read.csv("~/Courses/EnvDataAnalytics/Environmental_Data_Analytics_2020/Data/Raw/Oz
```

Wrangle

2. Combine your ten datasets into one dataset called GaringerOzone. Think about whether you should use a join or a row bind.
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 2019-12-13 (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 comine 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.

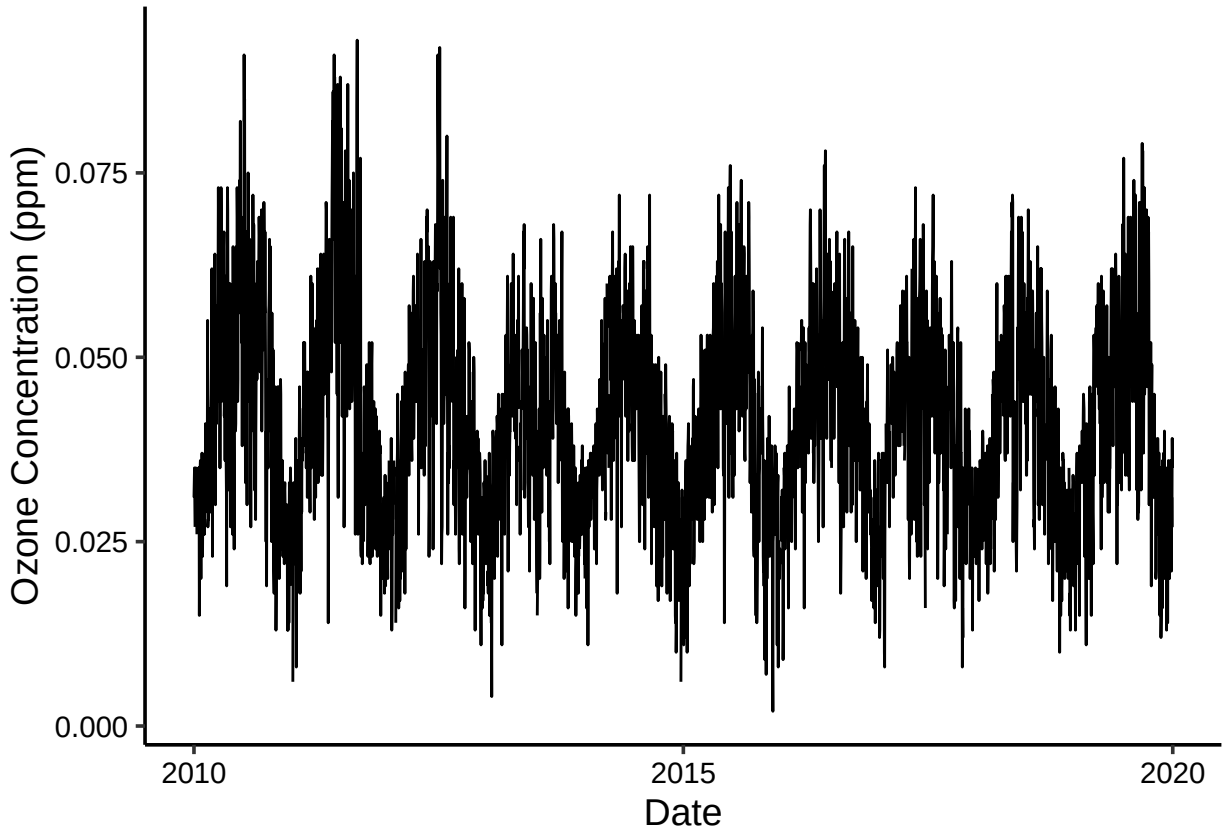
```
# 2
GaringerOzone <- rbind(GaringerOzone2010, GaringerOzone2011, GaringerOzone2012,
                       GaringerOzone2013, GaringerOzone2014, GaringerOzone2015,
                       GaringerOzone2016, GaringerOzone2017, GaringerOzone2018,
                       GaringerOzone2019)

# 3
GaringerOzone$Date <- as.Date(GaringerOzone$Date, format = "%m/%d/%Y")

# 4
GaringerOzone.new <- GaringerOzone %>% select(Date, Daily.Max.8.hour.Ozone.Concentration,
                                              DAILY_AQI_VALUE)

# 5
Days <- as.data.frame(seq(min(GaringerOzone.new$Date), max(GaringerOzone.new$Date), by = "day"))
colnames(Days)[colnames(Days)==
      "seq(min(GaringerOzone.new$Date), max(GaringerOzone.new$Date), by = \"day\")"] <- "Date"
# 6
GaringerOzone <- left_join(Days, GaringerOzone.new)
```

```
## Joining, by = "Date"
```

Visualize

7. Create a ggplot depicting ozone concentrations over time. In this case, we will plot actual concentrations in ppm, not AQI values. Format your axes accordingly.

```
ggplot(data = GaringerOzone, aes(x= Date, y= Daily.Max.8.hour.Ozone.Concentration)) +
  geom_line() +
  labs(y = "Ozone Concentration (ppm)")
```



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?

   Answer: We would expect ozone concentrations on any given day to be related to ozone concentrations the previous and next days, so piecewise constant makes less sense as interpolated values are only a function of the last concentration. There's no reason to believe concentrations are a quadratic function of prior and future concentrations, so linear interpolation is the best choice.

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)

10. Generate a time series called GaringerOzone.monthly.ts, with a monthly frequency that specifies the correct start and end dates.

11. Run a time series analysis. In this case the seasonal Mann-Kendall is most appropriate; why is this?

    Answer: A non-parametric test is more appropriate in this case since we can't assume an underlying distribution of the data. For this reason, a Mann-Kendall seems most appropriate. Also, from looking at the graph above, it seems clear that there are seasonal trends in the data, so a seasonal Mann-Kendall is the best choice.

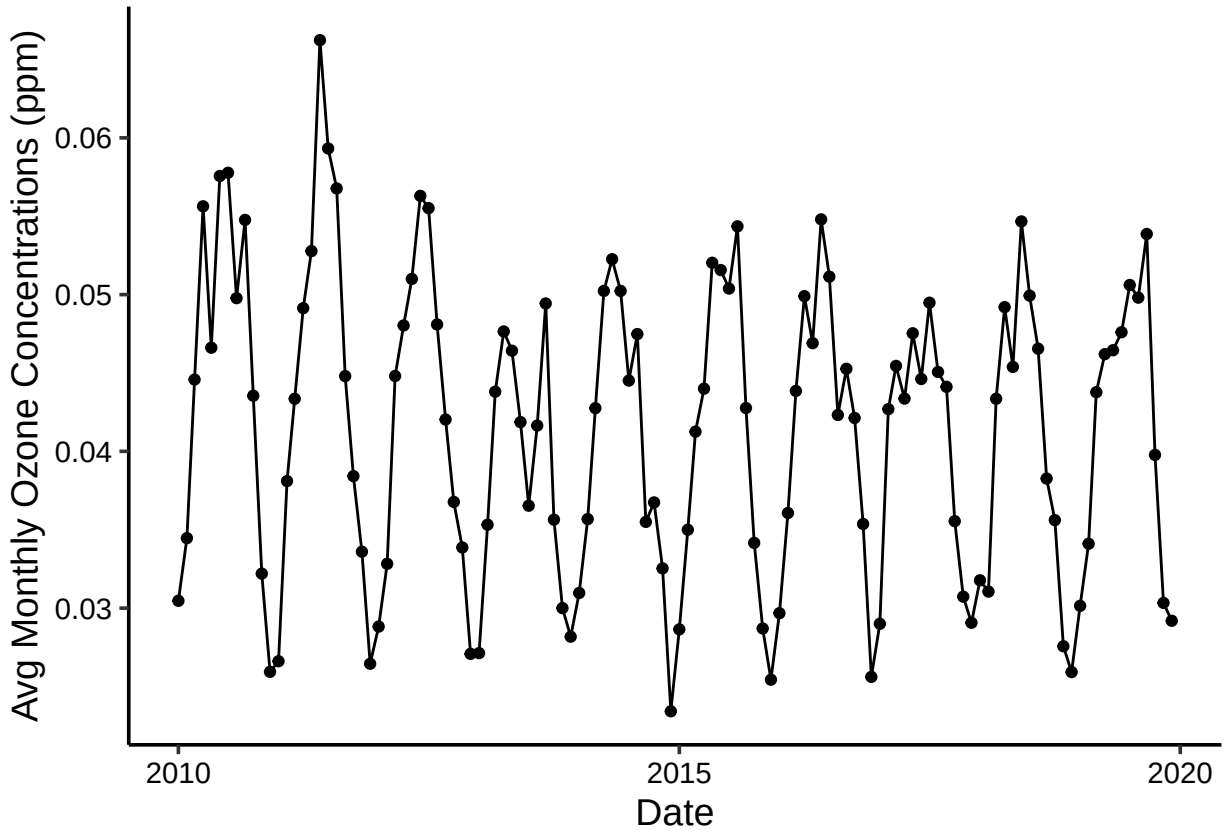
12. To figure out the slope of the trend, run the function `sea.sens.slope` on the time series dataset.

13. Create a plot depicting mean monthly ozone concentrations over time, with both a geom_point and a geom_line layer. No need to add a line for the seasonal Sen's slope; this is difficult to apply to a graph with time as the x axis. Edit your axis labels accordingly.

```
# 8
GaringerOzone$Daily.Max.8.hour.Ozone.Concentration <-
  na.approx(GaringerOzone$Daily.Max.8.hour.Ozone.Concentration)

# 9
GaringerOzone.monthly <- GaringerOzone %>% mutate(Year = year(Date), Month = month(Date)) %>%
  group_by(Year, Month) %>%
  summarise(mean.monthly.ozone.concentration = mean(Daily.Max.8.hour.Ozone.Concentration))

GaringerOzone.monthly$Date <- as.Date(paste(GaringerOzone.monthly$Year,
                                            GaringerOzone.monthly$Month,
                                            1, sep="-"),
                                      format = "%Y-%m-%d")
# 10
GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$mean.monthly.ozone.concentration,
                               frequency = 12,
                               start = c(2010, 01, 01), end = c(2019, 12, 1))

# 11
 GaringerOzone.monthly.trend <- smk.test(GaringerOzone.monthly.ts)

# 12
sea.sens.slope(GaringerOzone.monthly.ts)
```

```
## [1] -0.0002044163
```

```
## Slope = -0.0002044163; very slight negative slope
# 13
ggplot(data = GaringerOzone.monthly, aes(x= Date, y = mean.monthly.ozone.concentration)) +
  geom_point() +
  geom_line() +
  labs(y = "Avg Monthly Ozone Concentrations (ppm)")
```



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: There has been a very weak decreasing trend in ozone concentrations from January 2010 to December 2020. The decrease is very slight relative to seasonal variations in ozone concentrations. There was a significant decrease in ozone concentrations of about 0.0002 ppm per year over the time range (Seasonal Mann-Kendall, $p < 0.05$).