Assignment 8: Time Series Analysis

Aishwarya Patankar

Spring 2025

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

```
#Installing 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 1.9.4     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)
#install.packages("trend")
library(trend)
#install.packages("zoo")
library(zoo)
```

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

```
library(here)
```

```
## here() starts at /Users/aishwaryapatankar/Documents/Duke University/Spring2025/ENERGY872/EDA_Spring2
```

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

```
## [1] "/Users/aishwaryapatankar/Documents/Duke University/Spring2025/ENERGY872/EDA_Spring2025"
```

```
#Setting the theme
mytheme<- theme_grey(base_size = 14)+
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
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.

```
#2
#Getting Datasets from Raw folder
O3_Gar_2010 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2010_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2011 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2011_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2012 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2012_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2013 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2013_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2014 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2014_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2015 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2015_raw.csv"),
  stringsAsFactors = TRUE)
```

```
O3_Gar_2016 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2016_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2017 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2017_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2018 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2018_raw.csv"),
  stringsAsFactors = TRUE)

O3_Gar_2019 <- read.csv(
  file = here("./Data/Raw/Ozone_TimeSeries/EPAair_O3_GaringerNC2019_raw.csv"),
  stringsAsFactors = TRUE)

#Combining into Single Dataframe
OzoneDataList <- list(O3_Gar_2010,O3_Gar_2011, O3_Gar_2012, O3_Gar_2013, O3_Gar_2014, O3_Gar_2015, O3_Ga

GaringerOzone <- bind_rows(OzoneDataList)
```

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
#Checking class of Date column
class(GaringerOzone$Date)
```

```
## [1] "factor"
```

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

class(GaringerOzone$Date)
```

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

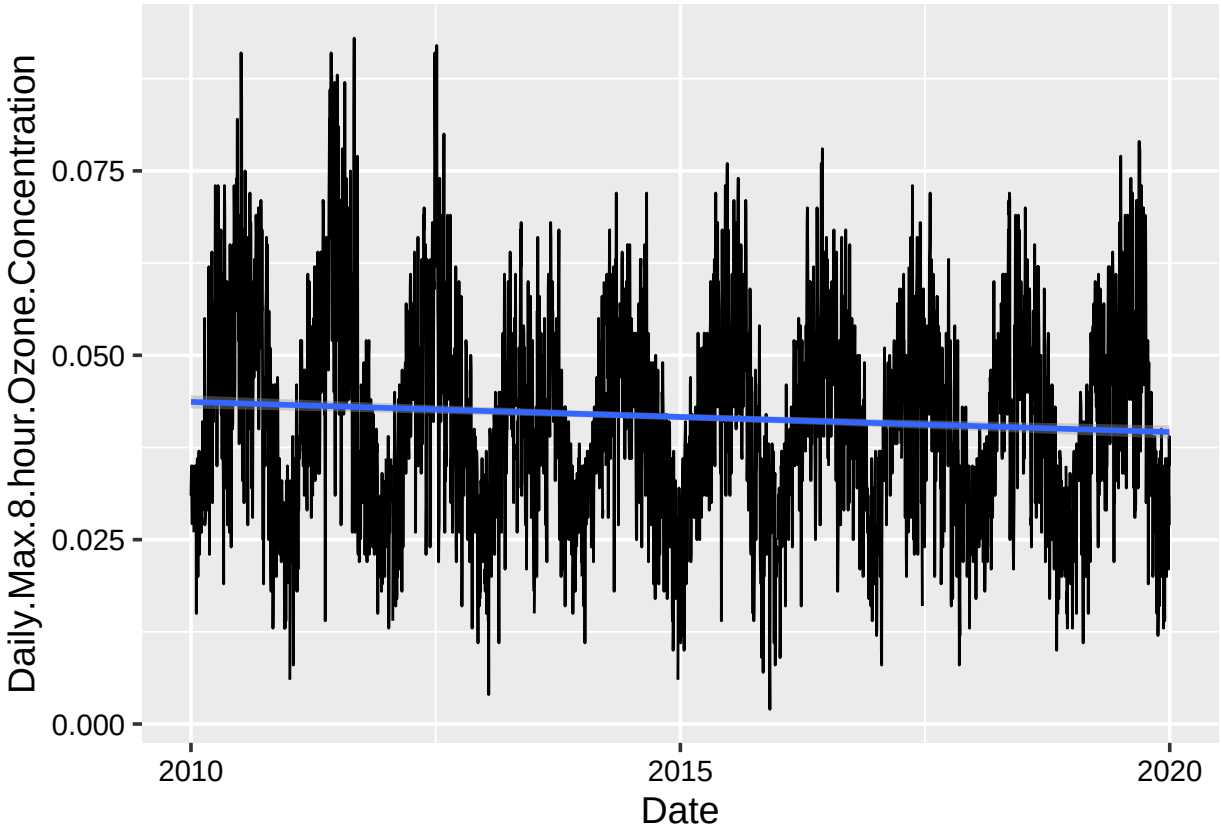
```
# 4
#Subsetting
GaringerOzone_subset <- GaringerOzone %>%
  select(Date, Daily.Max.8.hour.Ozone.Concentration, DAILY_AQI_VALUE)

# 5
Days <- as.data.frame(seq(from = as.Date("2010-01-01"), to = as.Date("2019-12-31"), by="day"))

colnames(Days) <- "Date"

# 6
GaringerOzoneNew <-left_join(Days, GaringerOzone_subset, 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?

```
#7
Ozone_v_Date<-
  ggplot(GaringerOzoneNew, aes(x = Date, y = Daily.Max.8.hour.Ozone.Concentration))+
  geom_line()+
  geom_smooth(method = "lm")+
  mytheme
  print(Ozone_v_Date)
```

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

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



Answer: Yes our plot indicates that there is seasonality to the Ozone Concentrations. Ozone concentrations are lowest in the winter season and peak during summers year on year. Also progressively over the years the concentrations have decreased.

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
#Interpolating values for the NA in Ozone Column and Updating the column with interpolated values.
GaringerOzoneNew$Daily.Max.8.hour.Ozone.Concentration <- na.approx(GaringerOzoneNew$Daily.Max.8.hour.Oz
```

Answer: Linear interpolation is a simple method which allows use of linear polynomials to fit data. Spline is used for smoother fits and piecewise contains differenr functions over different intervals. As ozone concentartions are not expected to show drastic differences between consecutive days, the linear interpolation is a good simple method to show a stepwise increase or decrease in accordance with the data of the previous and next day.

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 <- GaringerOzoneNew %>%
  mutate(Year = year(Date), Month = month(Date))%>%
  group_by(Month,Year)%>%
  summarise(MeanOzone = mean(Daily.Max.8.hour.Ozone.Concentration))%>%
  ungroup()%>%
  mutate(Date = as.Date(paste(Year,Month, "01", sep = "-" ))
  )
```

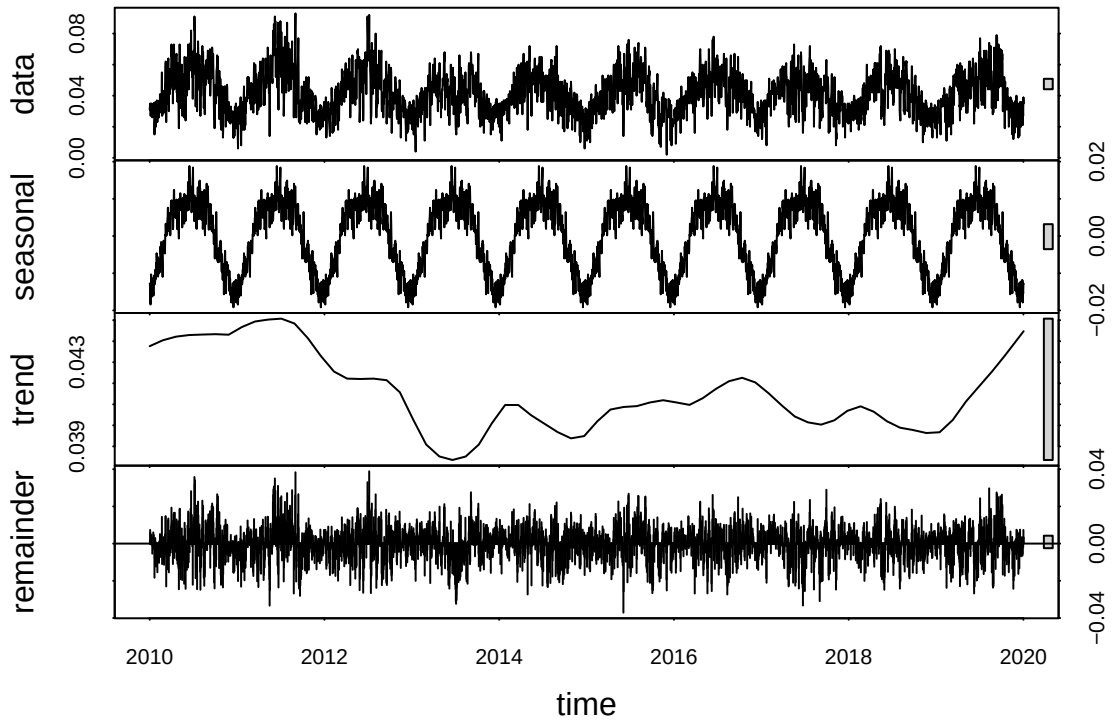
```
## `summarise()` has grouped output by 'Month'. You can override using the
## `.groups` argument.
```

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.

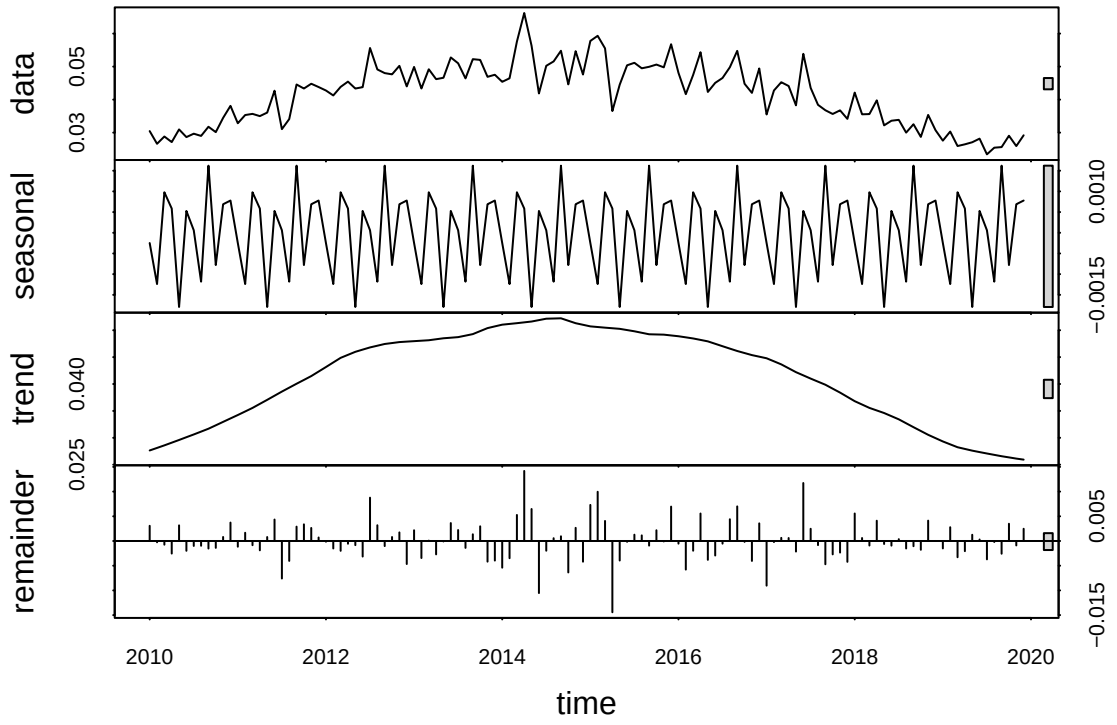
```
#10
#Creating Daily and Monthly Time Series Objects
GaringerOzone.daily.ts <- ts(GaringerOzoneNew$Daily.Max.8.hour.Ozone.Concentration, frequency = 365, sta

GaringerOzone.monthly.ts <- ts(GaringerOzone.monthly$MeanOzone,frequency = 12,start=c(2010,1))
```

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

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



```
GaringerOzone.monthly.ts_Decomposed <- stl(GaringerOzone.monthly.ts, s.window = "periodic")
plot(GaringerOzone.monthly.ts_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?

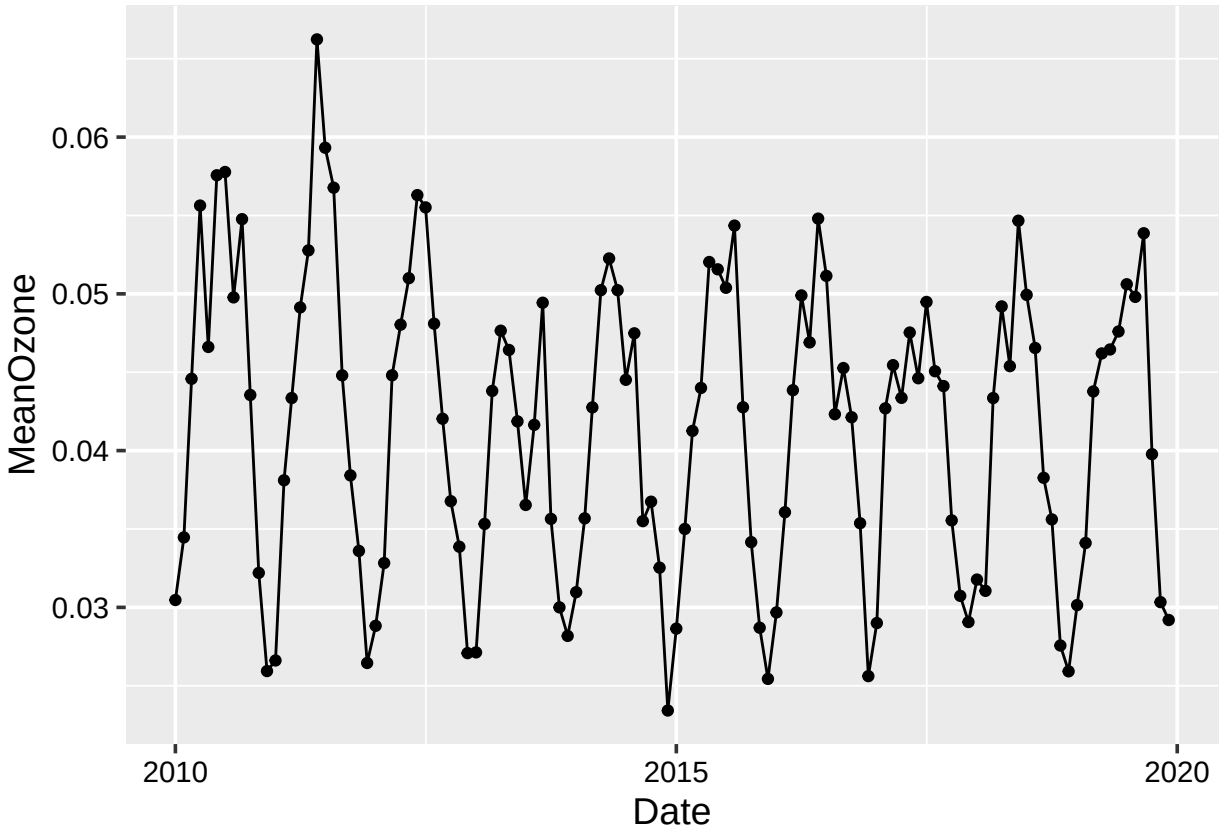
```
#12

MonthlyOzone_trend <- trend::smk.test(GaringerOzone.monthly.ts)
# Inspect results
MonthlyOzone_trend
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data:  GaringerOzone.monthly.ts
## z = -1.3685, p-value = 0.1712
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##    S varS
##  -54 1500
```

Answer: The Mann-Kendall test is used for to study deterministic observations in environmental data. The Mann-Kendall test cannot be applied to seasonal data but a modified version of this test known as sesonal Mann-Kendall can be used to study seasonality.

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

MeanO3_v_Date<-
  ggplot(GaringerOzone.monthly, aes(x = Date, y = MeanOzone))+
  geom_line()+
  geom_point()+
  mytheme
  print(MeanO3_v_Date)
```



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: Our MeanOzone Vs Date graph shows us O3 concentration trend with a seasonal component. The seasonal Mann-Kendall test shows that the S value is -54 indicating a downward monotonic trend, meaning that the ozone concentrations are decreasing over the 2010s at this station. However, the p value is 0.1712 which is greater than 0.05 indicating that our results are not statistically significant so we cannot reject Null Hypothesis that there is no monotonic trend.

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.

```r
#15
#Extracting Component of teh series and turning it into a dataframe
GaringerOzone.monthly_Components <- as.data.frame(GaringerOzone.monthly.ts_Decomposed$time.series[,1:3])
#Removing seasonality for Ozone
MeanOzone_Nonseas <- GaringerOzone.monthly.ts - GaringerOzone.monthly.ts_Decomposed$time.series[,1]
GaringerOzone.monthly_Components <- mutate(GaringerOzone.monthly_Components,
Nonseasonal = MeanOzone_Nonseas)



#16
MeanOzone_Nonseas_trend <- trend::mk.test(MeanOzone_Nonseas)
# Inspect results
MeanOzone_Nonseas_trend
```

```
##
##  Mann-Kendall trend test
##
## data:  MeanOzone_Nonseas
## z = -1.6263, n = 120, p-value = 0.1039
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -7.180000e+02  1.943667e+05 -1.005602e-01
```

Answer: In the Mann kendall test with non-seasonal component we observe that the S values is -718 indicating a downward monotonic trend, meaning that the ozone concentrations are decreasing over the 2010s at this station. However, the p value is 0.1039 which is greater than 0.05 indicating that our results are not statistically significant so we cannot reject Null Hypothesis that there is no monotonic trend.