

Assignment 8: Time Series Analysis

Damon L. Daniels

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

```
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(trend)
library(zoo)
```

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

```
library(Kendall)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##      method      from
##      as.zoo.data.frame zoo
```

```
library(here)
```

```
## here() starts at /home/guest/EDA_Spring2024
```

```
mytheme <- theme_classic(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.

```
#1
```

```
getwd()
```

```
## [1] "/home/guest/EDA_Spring2024/Data/Raw/Ozone_TimeSeries"
```

```
setwd("/home/guest/EDA_Spring2024/Data/Raw/Ozone_TimeSeries/")
```

```
filefolder <- here("Data", "Raw", "Ozone_TimeSeries")
```

```
files <- dir(filefolder, pattern = "*.csv")
files
```

```
## [1] "EPAair_03_GaringerNC2010_raw.csv" "EPAair_03_GaringerNC2011_raw.csv"
## [3] "EPAair_03_GaringerNC2012_raw.csv" "EPAair_03_GaringerNC2013_raw.csv"
## [5] "EPAair_03_GaringerNC2014_raw.csv" "EPAair_03_GaringerNC2015_raw.csv"
## [7] "EPAair_03_GaringerNC2016_raw.csv" "EPAair_03_GaringerNC2017_raw.csv"
## [9] "EPAair_03_GaringerNC2018_raw.csv" "EPAair_03_GaringerNC2019_raw.csv"
```

```
Ozone_TimeSeries <- files %>%
  map(read_csv) %>%
  reduce(rbind, deparse.level = 0)
```

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

```
class(Ozone_TimeSeries$Date)
```

```
## [1] "character"
```

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

4

```
Ozone_TimeSeries.subset <- Ozone_TimeSeries %>%
  select(c(Date, `Daily Max 8-hour Ozone Concentration`, DAILY_AQI_VALUE))
```

5

```
Days <- as.data.frame(seq(as.Date("2010-01-01"), as.Date("2019-12-31"), by = "day"))
names(Days) <- "Date"
```

6

```
GaringerOzone <- left_join(Days, Ozone_TimeSeries.subset)
```

```
## 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?

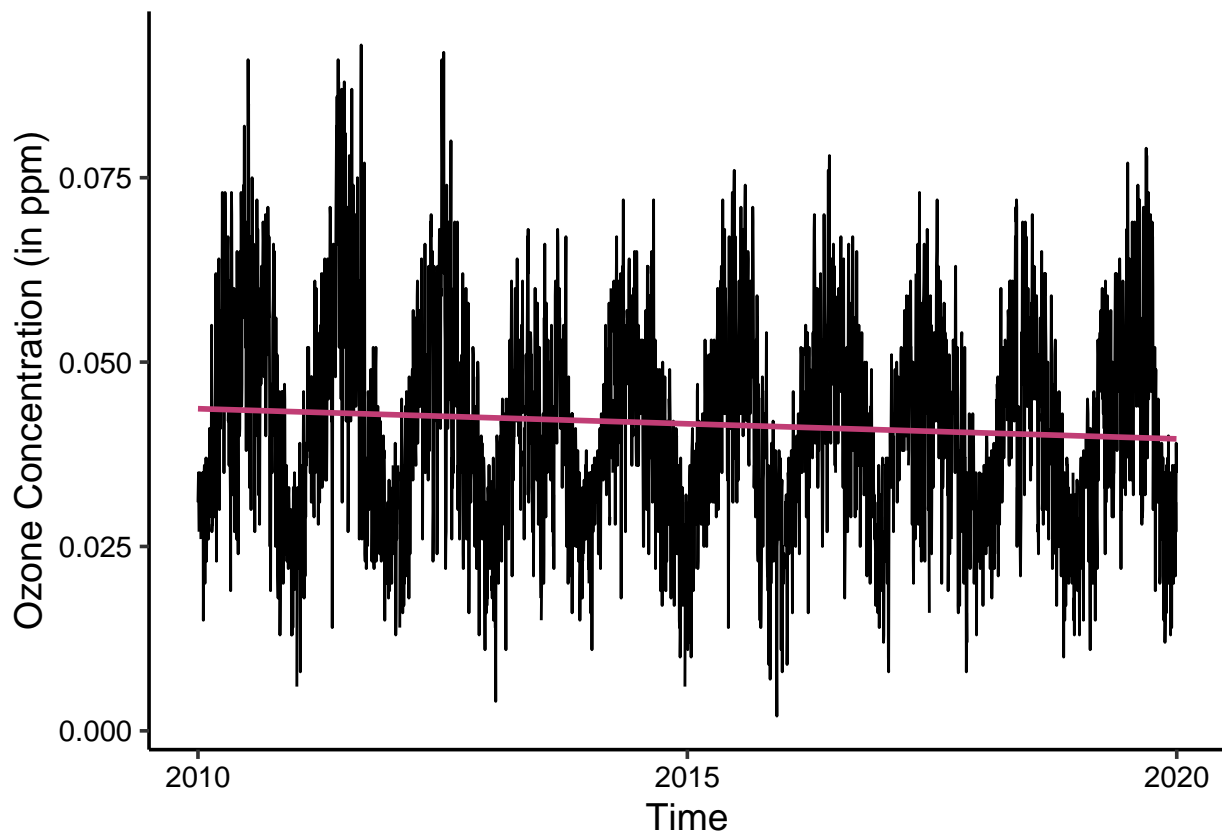
#7

```
ppm.by.time_lineplot <- GaringerOzone %>%
  ggplot(aes(x = Date, y = `Daily Max 8-hour Ozone Concentration`)) +
```

```
geom_line() +
geom_smooth(method = lm, se = FALSE, color = "#c13d75ff") +
labs(x = "Time", y = "Ozone Concentration (in ppm)")
print(ppm.by.time_lineplot)
```

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

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



Answer: The plot suggests a slightly decreasing trend of Ozone concentrations over time (based on the very shallow slope of the trend line), but given the pattern of concentration increases and decreases that appear to be occurring within years, there may seasonal variations/trends also.

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

```
head(GaringerOzone)
```

```
##           Date Daily Max 8-hour Ozone Concentration DAILY_AQI_VALUE
## 1 2010-01-01                      0.031                      29
## 2 2010-01-02                      0.033                      31
## 3 2010-01-03                      0.035                      32
## 4 2010-01-04                      0.031                      29
## 5 2010-01-05                      0.027                      25
## 6 2010-01-06                      NA                       NA
```

```
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
```

```
GaringerOzone.MaxConcentration.clean <-
```

```
  GaringerOzone %>%
```

```
  mutate( x = zoo::na.approx(GaringerOzone$`Daily Max 8-hour Ozone Concentration`) )
```

```
summary(GaringerOzone.MaxConcentration.clean$x)
```

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

Answer: A piecewise constant (via the “next best neighbor” approach) or spline interpolation (via the use of a quadratic function) will both likely result in a curve, which is less helpful for this set of data. Linear interpolations, on the other hand, allow us to find the estimated median, quartiles or percentiles of a set of data, particularly when the data is presented in a group frequency table that contains class intervals.

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.MaxConcentration.clean %>%
```

```
  mutate(Year = year(Date)) %>%
```

```
  mutate(Month = month(Date)) %>%
```

```
  mutate(Date = my(paste0(Month, "-", Year))) %>%
```

```
  group_by(Year, Month, Date) %>%
```

```
  dplyr::summarise("Mean Ozone" = mean(x, na.rm = TRUE)) %>%
```

```
  select(Date, "Mean Ozone")
```

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

```
## Adding missing grouping variables: 'Year', 'Month'
```

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

fmonth <- month(first(GaringerOzone$Date))
fyear <- year(first(GaringerOzone$Date))

GaringerOzone.daily <- GaringerOzone.MaxConcentration.clean %>%
  select(Date, x)

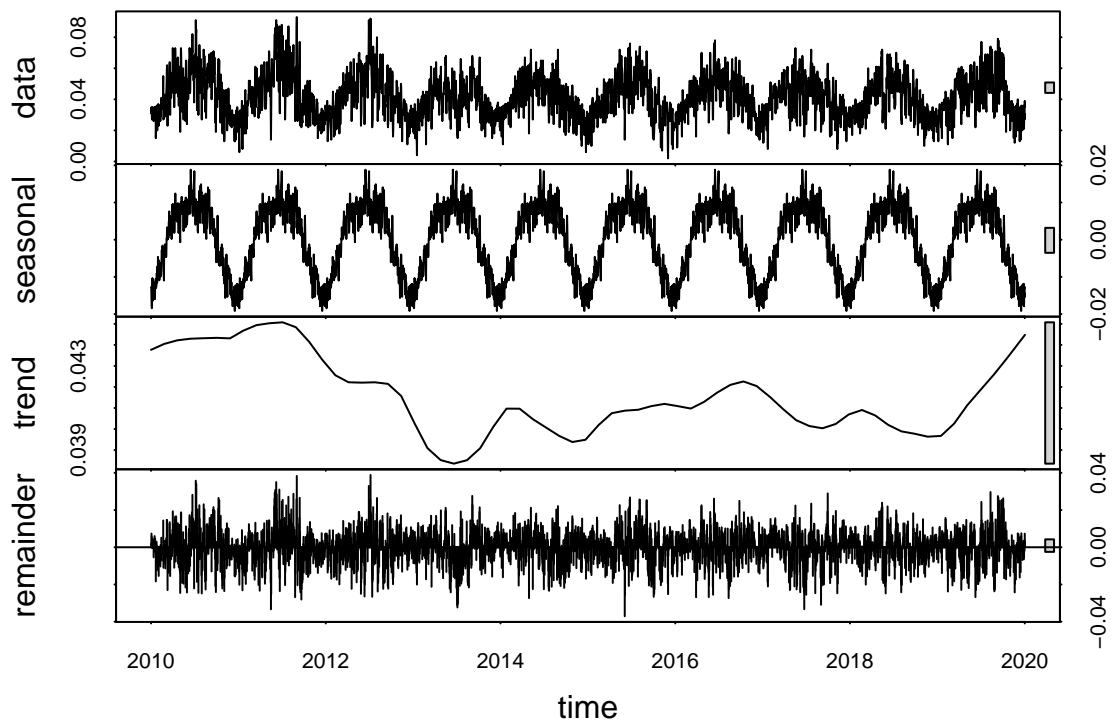
GaringerOzone.daily.ts <- ts(GaringerOzone.daily$x,
                             start = c(2010, 1),
                             frequency = 365)

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

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

```
#11

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



```
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

```
Ozone.monthly.trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)
summary(Ozone.monthly.trend)
```

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

```
print(Ozone.monthly.trend)
```

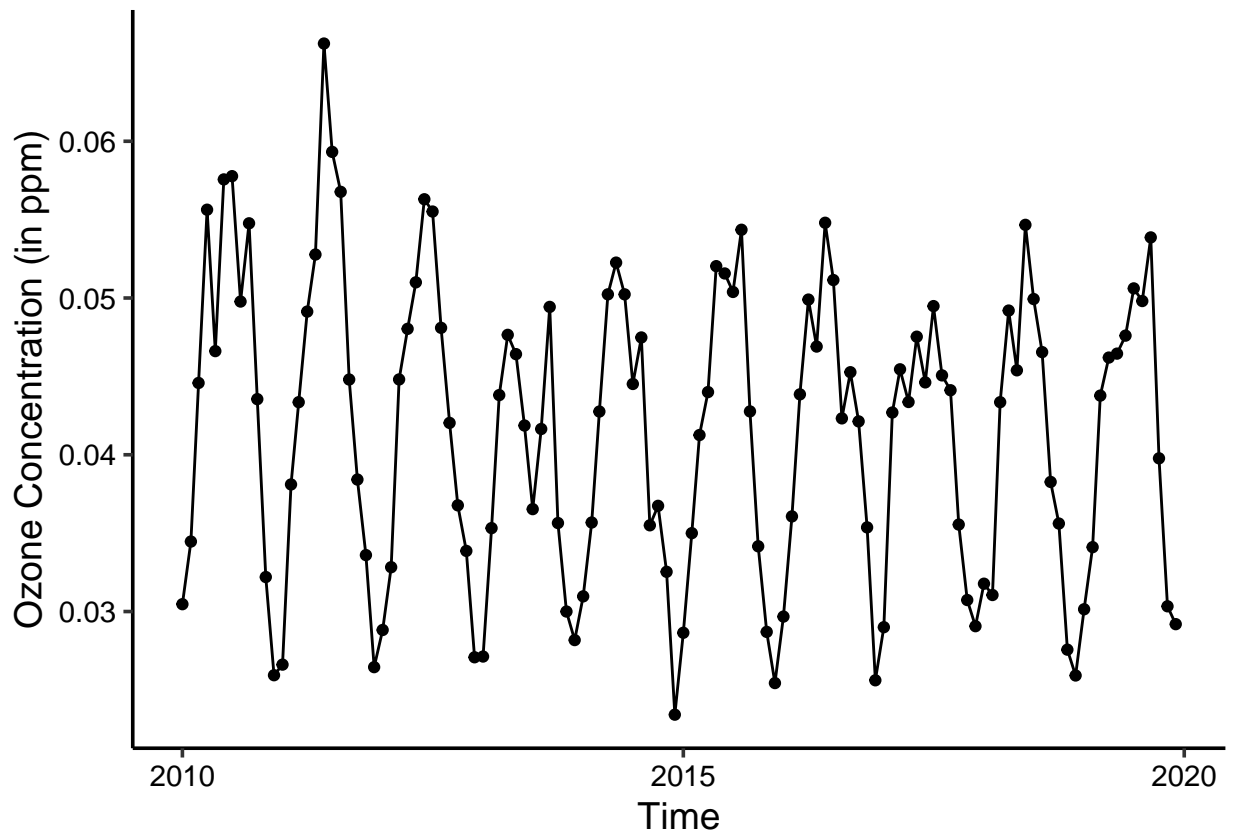
```
## tau = -0.143, 2-sided pvalue =0.046724
```

Answer: The observation of a wave pattern of results from question no. 11 suggests a seasonal trend, for which the Seasonal Mann-Kendall becomes more appropriate (the other analyses techniques are noted to occur during conditions of non-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.


```
# 13
```

```
MonthlyMeanOzone_byTime.lineplot <- GaringerOzone.monthly %>%  
  ggplot(aes(x = Date, y = `Mean Ozone`)) +  
  geom_point() +  
  geom_line() +  
  labs(x = "Time", y = "Ozone Concentration (in ppm)")  
print(MonthlyMeanOzone_byTime.lineplot)
```



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.

```
Ozone.monthly.trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)  
summary(Ozone.monthly.trend)
```

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

```
print(Ozone.monthly.trend)
```

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

Answer: To the study question, “Have ozone concentrations changed over the 2010s at this station?” –

HO: There are no changes in Ozone concentrations during the 2010s decade at Garinger Station (i.e., the differences in the ozone means = 0) H1: There are 1 or more changes in Ozone concentrations during the 2010s decade at Garinger Station. (i.e., the differences in the ozone means != 0)

Using the result from the Seasonal Mann-Kendall test – tau = -0.143; 2-sided p value = 0.046724 (which is < 0.05) Given these values, we can reject the Null Hypothesis and therefore conclude that there is a descending trend in Ozone concentrations during the 2010s decade at Garinger Station. This conclusion matches the observations derived from the line plot that was produced in Question no. 7.

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

```
GaringerOzone.monthly.components <-
  as.data.frame(GaringerOzone.monthly.decomposed$time.series[,1:3])
GaringerOzone.monthly.deducted <-
  GaringerOzone.monthly.ts - GaringerOzone.monthly.components$seasonal
print(GaringerOzone.monthly.deducted)
```

```
##           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

```
Ozone.monthly.nonseasonaltrend <-  
  Kendall::MannKendall(GaringerOzone.monthly.deducted)  
summary(Ozone.monthly.nonseasonaltrend)
```

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

```
Ozone.monthly.trend <- Kendall::SeasonalMannKendall(GaringerOzone.monthly.ts)  
summary(Ozone.monthly.trend)
```

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

Answer: Compared to the results from the Seasonal Mann-Kendall test, after deduction of the season trend: $\tau = -0.165$ (a moderate decrease from -0.143); p value = 0.0075402 (a large decrease from 0.046724) Therefore, excluding from the seasonal trends at Garinger Station, there is nonetheless a statistically significant decreasing trend in Ozone concentrations during the 2010s decade.