Assignment 8: Time Series Analysis

Njeri Kara

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics (ENV872L) on time series analysis.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Use the lesson as a guide. It contains code that can be modified to complete the assignment.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure to **answer the questions** in this assignment document. Space for your answers is provided in this document and is indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">". You should notice that the answer is highlighted in green by RStudio.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file. You will need to have the correct software installed to do this (see Software Installation Guide) Press the `Knit` button in the RStudio scripting panel. This will save the PDF output in your Assignments folder.
6. After Knitting, please submit the completed exercise (PDF file) to the dropbox in Sakai. Please add your last name into the file name (e.g., "Salk_A08_TimeSeries.pdf") prior to submission.

The completed exercise is due on Tuesday, 19 March, 2019 before class begins.

Brainstorm a project topic

1. Spend 15 minutes brainstorming ideas for a project topic, and look for a dataset if you are choosing your own rather than using a class dataset. Remember your topic choices are due by the end of March, and you should post your choice ASAP to the forum on Sakai.

Question: Did you do this?

> ANSWER: Yes

Set up your session

2. Set up your session. Upload the EPA air quality raw dataset for PM2.5 in 2018, and the processed NTL-LTER dataset for nutrients in Peter and Paul lakes. Build a ggplot theme and set it as your default theme. Make sure date variables are set to a date format.

```
#1
#Setting the working directory
setwd("C:/Users/jerik/OneDrive/Documents/Spring 2019 Semenster/Environmental Data Analytics/EDA_R_Work/I

#Loading necessary packages
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts ----------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```r
library(nlme)
```

```
##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse
```

```r
library(lsmeans)
```

```
## Loading required package: emmeans

## The 'lsmeans' package is now basically a front end for 'emmeans'.
## Users are encouraged to switch the rest of the way.
## See help('transition') for more information, including how to
## convert old 'lsmeans' objects and scripts to work with 'emmeans'.
```

```r
library(multcompView)
library(trend)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor
```

```r
#Importing datasets
EPA.PM25.2018.raw <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv")
NTL.Nutrients.PP.processed <-
  read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv")

#Changing date variable of PM2.5 data to date format
str(EPA.PM25.2018.raw)
```

```
## 'data.frame':    7611 obs. of  20 variables:
##  $ Date                          : Factor w/ 343 levels "1/1/18","1/10/18",..: 12 27 30 3 6 9 13 16 
##  $ Source                        : Factor w/ 2 levels "AirNow","AQS": 2 2 2 2 2 2 2 2 2 2 ...
##  $ Site.ID                       : int  370110002 370110002 370110002 370110002 370110002 370110002 3
##  $ POC                           : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Daily.Mean.PM2.5.Concentration: num  2.9 3.7 5.3 0.8 2.5 4.5 1.8 2.5 4.2 1.7 ...
```

```
##  $ UNITS                         : Factor w/ 1 level "ug/m3 LC": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DAILY_AQI_VALUE               : int  12 15 22 3 10 19 8 10 18 7 ...
##  $ Site.Name                     : Factor w/ 24 levels "","Blackstone",..: 14 14 14 14 14 14 14 14 14
##  $ DAILY_OBS_COUNT               : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ PERCENT_COMPLETE              : int  100 100 100 100 100 100 100 100 100 100 ...
##  $ AQS_PARAMETER_CODE            : int  88502 88502 88502 88502 88502 88502 88502 88502 88502 88502
##  $ AQS_PARAMETER_DESC            : Factor w/ 2 levels "Acceptable PM2.5 AQI & Speciation Mass",..: 1
##  $ CBSA_CODE                     : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ CBSA_NAME                     : Factor w/ 14 levels "","Asheville, NC",..: 1 1 1 1 1 1 1 1 1 1 ..
##  $ STATE_CODE                    : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ STATE                         : Factor w/ 1 level "North Carolina": 1 1 1 1 1 1 1 1 1 1 ...
##  $ COUNTY_CODE                   : int  11 11 11 11 11 11 11 11 11 11 ...
##  $ COUNTY                        : Factor w/ 21 levels "Avery","Buncombe",..: 1 1 1 1 1 1 1 1 1 1 ..
##  $ SITE_LATITUDE                 : num  36 36 36 36 36 ...
##  $ SITE_LONGITUDE                : num  -81.9 -81.9 -81.9 -81.9 -81.9 ...
```

```
EPA.PM25.2018.raw$Date <- as.Date(EPA.PM25.2018.raw$Date,
                                  format = "%m/%d/%y")
class(EPA.PM25.2018.raw$Date) #confirming date change
```

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

```
#Changing date variable to date format
str(NTL.Nutrients.PP.processed)
```

```
## 'data.frame':    2770 obs. of  13 variables:
##  $ lakeid    : Factor w/ 2 levels "L","R": 1 1 1 1 1 1 2 2 2 2 ...
##  $ lakename  : Factor w/ 2 levels "Paul Lake","Peter Lake": 1 1 1 1 1 1 2 2 2 2 ...
##  $ year4     : int  1991 1991 1991 1991 1991 1991 1991 1991 1991 1991 ...
##  $ daynum    : int  140 140 140 140 140 140 140 140 140 140 ...
##  $ sampledate: Factor w/ 778 levels "1991-05-20","1991-05-27",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ depth_id  : int  1 2 3 4 5 6 1 2 3 4 ...
##  $ depth     : num  0 0.85 1.75 3 4 6 0 1 2.25 3.5 ...
##  $ tn_ug     : num  538 285 399 453 363 583 352 356 364 582 ...
##  $ tp_ug     : num  25 14 14 14 13 37 11 15 28 14 ...
##  $ nh34      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ no23      : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ po4       : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ comments  : logi  NA NA NA NA NA NA ...
```

```
NTL.Nutrients.PP.processed$sampledate <- as.Date(NTL.Nutrients.PP.processed$sampledate,
format = "%Y-%m-%d")
class(NTL.Nutrients.PP.processed$sampledate) #confirming date change
```

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

```
#Building a theme
NK.theme <- theme_light(base_size = 12) +
  theme(plot.background = element_rect(fill = "grey97"),
        panel.grid.major =element_line(linetype = "dotted"),
        panel.grid.minor = element_line(linetype = "dotted"), text=element_text(size = 14,
color = "black", face = "bold"),
        axis.text = element_text(color = "grey40"),
        legend.position = "right",
        legend.text = element_text(color = "grey40"))
#setting it as my default theme
theme_set(NK.theme)
```

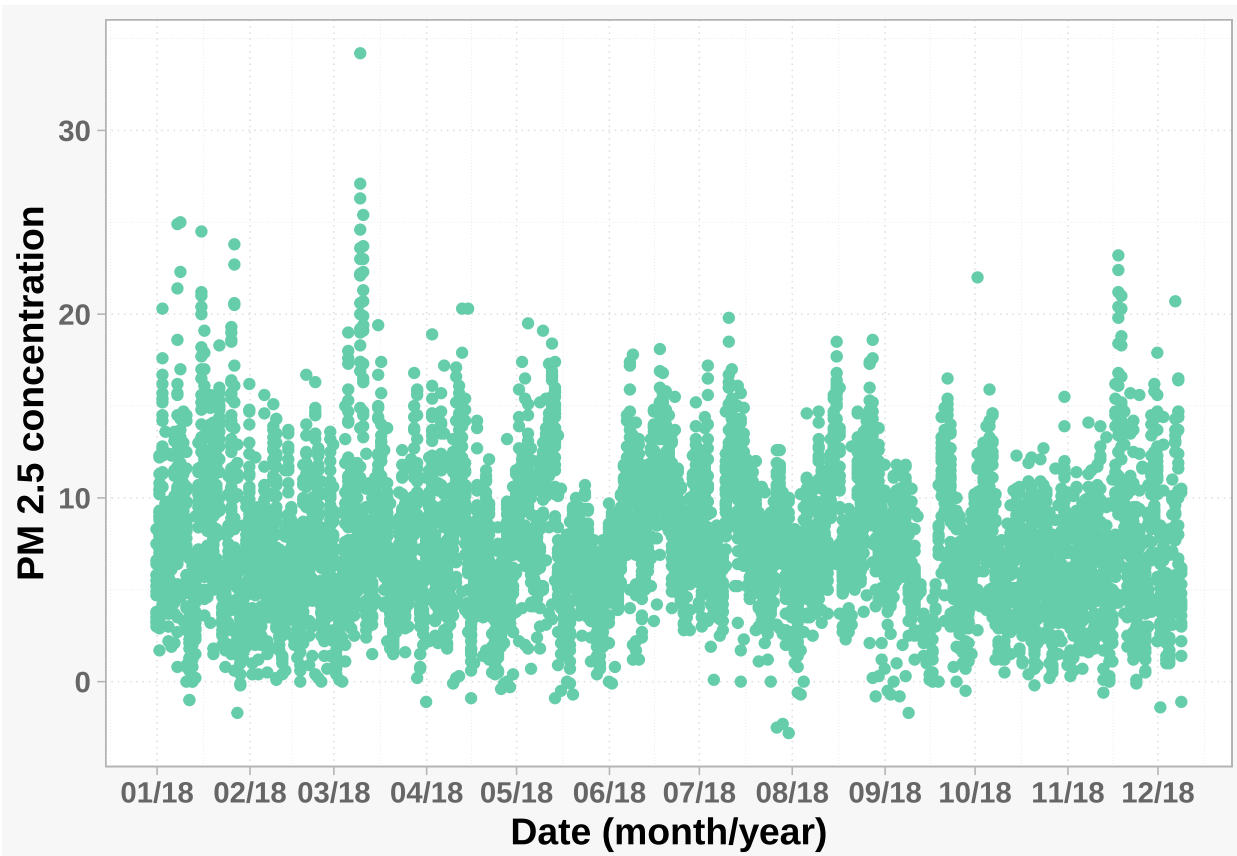
Run a hierarchical (mixed-effects) model

Research question: Do PM2.5 concentrations have a significant trend in 2018?

3. Run a repeated measures ANOVA, with PM2.5 concentrations as the response, Date as a fixed effect, and Site.Name as a random effect. This will allow us to extrapolate PM2.5 concentrations across North Carolina.

3a. Illustrate PM2.5 concentrations by date. Do not split aesthetics by site.

```
ggplot(EPA.PM25.2018.raw, aes(x = as.POSIXct(Date), y = Daily.Mean.PM2.5.Concentration)) +
  geom_point(color='aquamarine3')+
  xlab("Date (month/year)") +
  ylab("PM 2.5 concentration") +
  scale_x_datetime(date_breaks = "1 month", labels = date_format("%m/%y"))
```



3b. Insert the following line of code into your R chunk. This will eliminate duplicate measurements on single dates for each site. PM2.5 = PM2.5[order(PM2.5[,'Date'],-PM2.5[,'Site.ID']),] PM2.5 = PM2.5[!duplicated(PM2.5$Date),]

```
EPA.PM25.2018.raw = EPA.PM25.2018.raw[order(EPA.PM25.2018.raw[,'Date'],
                                            -EPA.PM25.2018.raw[,'Site.ID']),]
EPA.PM25.2018.raw = EPA.PM25.2018.raw[!duplicated(EPA.PM25.2018.raw$Date),]
```

3c. Determine the temporal autocorrelation in your model.

```
# Determining temporal autocorrelation in the model
Temp.auto <- lme(data = EPA.PM25.2018.raw,
                     Daily.Mean.PM2.5.Concentration ~ Date,
```

```r
                     random = ~1|Site.Name)
ACF(Temp.auto)
```

```
##    lag          ACF
## 1    0  1.000000000
## 2    1  0.513829909
## 3    2  0.194512680
## 4    3  0.117925187
## 5    4  0.126462863
## 6    5  0.100699787
## 7    6  0.058215891
## 8    7 -0.053090104
## 9    8  0.017671857
## 10   9  0.012177847
## 11  10 -0.003699721
## 12  11 -0.020305291
## 13  12 -0.044621086
## 14  13 -0.055602646
## 15  14 -0.065787345
## 16  15 -0.123987593
## 17  16 -0.055414056
## 18  17  0.002911218
## 19  18  0.025133456
## 20  19 -0.015306468
## 21  20 -0.143472007
## 22  21 -0.155495492
## 23  22 -0.060369985
## 24  23  0.003954231
## 25  24  0.042295682
## 26  25  0.001320007
```

3d. Run a mixed effects model.

```r
Test.mixed <- lme(data = EPA.PM25.2018.raw,
                     Daily.Mean.PM2.5.Concentration ~ Date,
                     random = ~1|Site.Name,
                     correlation = corAR1(form = ~ Date|Site.Name, value = 0.51383),
                     method = "REML")
summary(Test.mixed)
```

```
## Linear mixed-effects model fit by REML
##  Data: EPA.PM25.2018.raw
##        AIC      BIC   logLik
##   1756.622 1775.781 -873.311
##
## Random effects:
##  Formula: ~1 | Site.Name
##         (Intercept) Residual
## StdDev: 0.001024989 3.597269
##
## Correlation Structure: ARMA(1,0)
##  Formula: ~Date | Site.Name
##  Parameter estimate(s):
##      Phi1
## 0.5384349
```

```
## Fixed effects: Daily.Mean.PM2.5.Concentration ~ Date
##                Value Std.Error  DF   t-value p-value
## (Intercept) 83.14801  60.63585 339  1.371268  0.1712
## Date        -0.00426   0.00342 339 -1.244145  0.2143
##  Correlation:
##      (Intr)
## Date -1
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -2.3220745 -0.6187194 -0.1116751  0.6164257  3.4192603
##
## Number of Observations: 343
## Number of Groups: 3
```

Is there a significant increasing or decreasing trend in PM2.5 concentrations in 2018?

> ANSWER: No there is no significant trend because the p-value is high (>0.05).

3e. Run a fixed effects model with Date as the only explanatory variable. Then test whether the mixed effects model is a better fit than the fixed effect model.

```
#fixed effect model
Test.fixed <- gls(data = EPA.PM25.2018.raw,
                      Daily.Mean.PM2.5.Concentration ~ Date,
                      method = "REML")
summary(Test.fixed)
```

```
## Generalized least squares fit by REML
##   Model: Daily.Mean.PM2.5.Concentration ~ Date
##   Data: EPA.PM25.2018.raw
##        AIC      BIC    logLik
##   1865.202 1876.698 -929.6011
##
## Coefficients:
##                Value Std.Error   t-value p-value
## (Intercept) 98.57796  34.60285  2.848840  0.0047
## Date        -0.00513   0.00195 -2.624999  0.0091
##
##  Correlation:
##      (Intr)
## Date -1
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -2.3531000 -0.6348100 -0.1153454  0.6383004  3.4063068
##
## Residual standard error: 3.584321
## Degrees of freedom: 343 total; 341 residual
```

```
#comparing mixed effects and fixed effects models
anova(Test.mixed, Test.fixed)
```

```
##            Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## Test.mixed     1  5 1756.622 1775.781 -873.3110
## Test.fixed     2  3 1865.202 1876.698 -929.6011 1 vs 2 112.5802  <.0001
```

Which model is better?

ANSWER: The models are significantly different. The mixed effects model is better because it has a lower AIC value.

Run a Mann-Kendall test

Research question: Is there a trend in total N surface concentrations in Peter and Paul lakes?

4. Duplicate the Mann-Kendall test we ran for total P in class, this time with total N for both lakes. Make sure to run a test for changepoints in the datasets (and run a second one if a second change point is likely).

```
# Wrangling the our dataset
Nutrients.PP.surface <-
  NTL.Nutrients.PP.processed %>%
  select(-lakeid, -depth_id, -comments) %>%
  filter(depth == 0) %>%
  filter(!is.na(tn_ug))

#splitting by lake
Peter.nutrients.surface <- filter(Nutrients.PP.surface, lakename == "Peter Lake")
Paul.nutrients.surface <- filter(Nutrients.PP.surface, lakename == "Paul Lake")

#Mann-Kendall test for Peter lake
mk.test(Peter.nutrients.surface$tn_ug)
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug
## z = 7.2927, n = 98, p-value = 3.039e-13
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 2.377000e+03 1.061503e+05 5.001052e-01
```

```
##Mann-Kendall test for Paul lake
mk.test(Paul.nutrients.surface$tn_ug)
```

```
##
##  Mann-Kendall trend test
##
## data:  Paul.nutrients.surface$tn_ug
## z = -0.35068, n = 99, p-value = 0.7258
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -1.170000e+02  1.094170e+05 -2.411874e-02
```

The first Mann-Kendall test for Peter lake has a small p-value (3.039e-13) therefore we reject the Ho hypothesis that there is no monotonic trend. The Z value is also positive therefore meaning the trend is increasing over time

The first Mann-Kendall test for Paul lake has a large p-value (0.7258) so we accept the Ho hypothesis that there is a monotonic trend.

```r
#Finding out if there is a change point in the data
#change point in Peter lake data
pettitt.test(Peter.nutrients.surface$tn_ug)
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Peter.nutrients.surface$tn_ug
## U* = 1884, p-value = 3.744e-10
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               36
```

```r
#change point in Paul lake data
pettitt.test(Paul.nutrients.surface$tn_ug)
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Paul.nutrients.surface$tn_ug
## U* = 704, p-value = 0.09624
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               16
```

> The Pettitt's test for Peter Lake data has a small p-value (3.744e-10) therefore we reject the Ho hypothesis that there is no change point in the data. The test has detected a change point at observation 36.
>
> The Pettitt's test for Paul Lake data has a large p-value (0.09624) therefore we accept the Ho hypothesis that there is no change point in the data.

```r
#Carrying out Mann Kendall tests of subseted Peter lake data at the change point
mk.test(Peter.nutrients.surface$tn_ug[1:35])
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[1:35]
## z = -0.22722, n = 35, p-value = 0.8203
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
##  -17.00000000 4958.33333333  -0.02857143
```

```r
mk.test(Peter.nutrients.surface$tn_ug[36:98])
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[36:98]
## z = 3.1909, n = 63, p-value = 0.001418
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##           S         varS          tau
```

```
## 5.390000e+02 2.842700e+04 2.759857e-01
```

What are the results of this test?

> ANSWER: On the first group of data, between observation 1-35, the Mann Kendall test has a large p-value (0.8203) therefore we accept the Ho hypothesis that this group of data has no significant monotonic trend.
>
> On the second group of data, between observation 36-98 after the change point, the Mann Kendall test has a small p-value (0.001418) therefore we reject the Ho hypothesis that this group of data does not have a significant monotonic trend. The positive Z value indicates that the trend is increasing over test.

```r
#Checking for a second change point in Peter Lake data
pettitt.test(Peter.nutrients.surface$tn_ug[36:98])
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Peter.nutrients.surface$tn_ug[36:98]
## U* = 560, p-value = 0.001213
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                              21
```

> The pettitt test has a small p value (0.001213) therefore we reject the Ho hypothesis of the test that there is no significant change point in the data.The change point is at observation 56.

```r
#Carrying out Mann Kendall tests of subseted Peter lake data at the 2nd change point
mk.test(Peter.nutrients.surface$tn_ug[36:55])
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[36:55]
## z = -1.2004, n = 20, p-value = 0.23
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##     S  varS   tau
## -38.0 950.0  -0.2
```

```r
mk.test(Peter.nutrients.surface$tn_ug[56:98])
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[56:98]
## z = 0.48141, n = 43, p-value = 0.6302
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 4.700000e+01 9.130333e+03 5.204873e-02
```

> The Mann Kendall tests for both groups of data have a large p-value (0.23 and 0.6302 respectively) therefore we accept the Ho hypothesis that these groups of data have no significant monotonic trend.

5. Generate a graph that illustrates the TN concentrations over time, coloring by lake and adding vertical

line(s) representing changepoint(s).

```
ggplot(Nutrients.PP.surface, aes(x = as.POSIXct(sampledate), y = tn_ug, color = lakename)) +
  geom_point() +
  scale_color_manual(values = c("Violet Red 3", "Medium Orchid 3")) +
  geom_vline(xintercept = as.POSIXct("1993-06-02"), color = "Medium Orchid 3", lty = 2) +
   geom_vline(xintercept = as.POSIXct("1994-06-22"), color = "Medium Orchid 3", lty = 2) +
  xlab("Date (year)") +
  ylab("Total Nitrogen (\U003BCg/L)") +
  scale_x_datetime(date_breaks = "1 year", labels = date_format("%Y"))
```