Assignment 8: Time Series Analysis

Rachel Bash

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.

```
getwd()
```

```
## [1] "/Users/rachelbash/Documents/DUKE/Data Analytics/Environmental_Data_Analytics"
```

```
suppressMessages(library(tidyverse))
#install.packages("lubridate")
suppressMessages(library(lubridate))
#install.packages("nlme")
suppressMessages(library(nlme))
#install.packages("lsmeans")
suppressMessages(library(lsmeans))
#install.packages("multcompView")
```

```r
suppressMessages(library(multcompView))
suppressMessages(library(trend))

PMair <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv")
PeterPaul.nutrients <- read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv")

#Set Date
PeterPaul.nutrients$sampledate <- as.Date(PeterPaul.nutrients$sampledate,
                                          format = "%Y-%m-%d")
PMair$Date <- as.Date(PMair$Date, format = "%m/%d/%y")

Rachel_theme <- theme_bw(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "right")
theme_set(Rachel_theme)
```

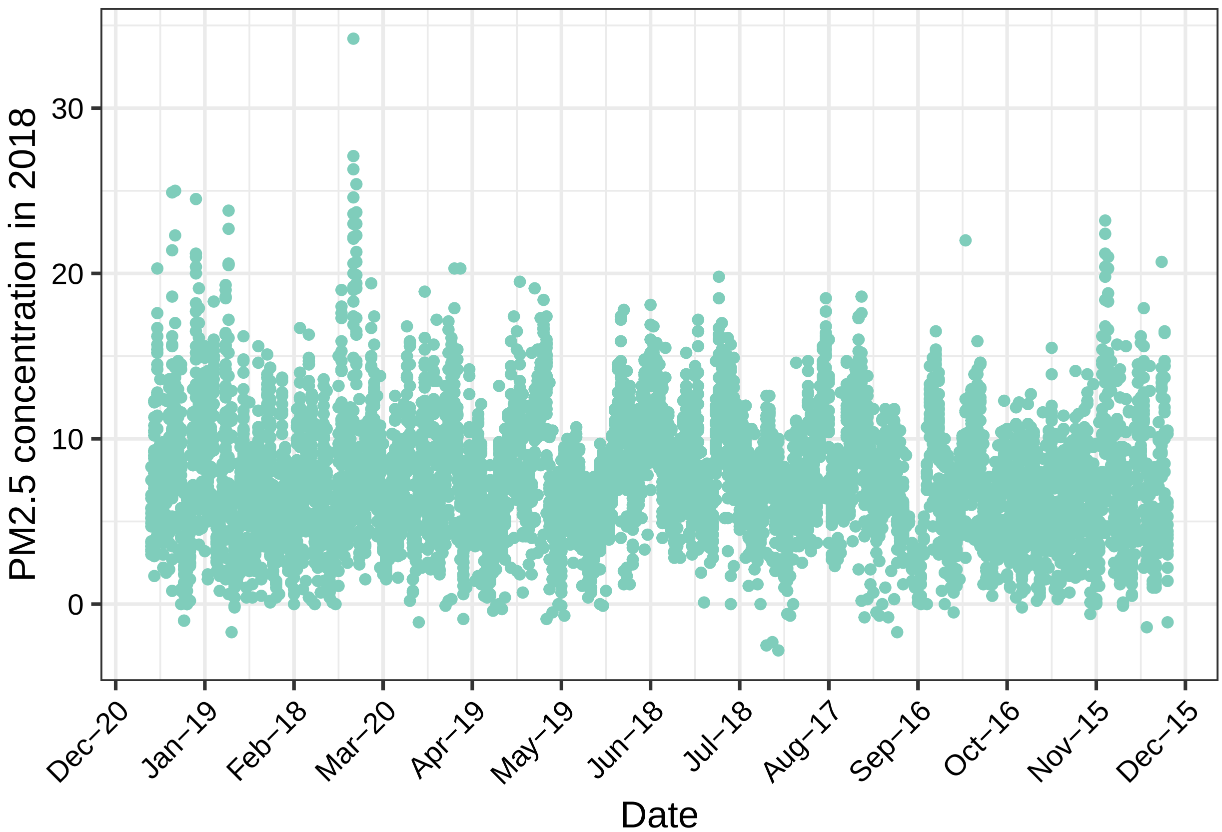
Run a hierarchical (mixed-effects) model

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.

```r
#a)
ggplot(PMair, aes(x = Date, y = Daily.Mean.PM2.5.Concentration)) +
  geom_point(color="#7fcdbb") +
  labs(x = "Date", y = "PM2.5 concentration in 2018") +
  scale_x_date(date_breaks = "30 days", date_labels = "%b-%d") +
  theme(axis.text.x = element_text(angle = 45,  hjust = 1))
```



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),]

3c. Determine the temporal autocorrelation in your model.

3d. Run a mixed effects model.

```
#b)
PMair = PMair[order(PMair[,'Date'],-PMair[,'Site.ID']),]
PMair = PMair[!duplicated(PMair$Date),]

#c)
PMair.auto <- lme(data= PMair, Daily.Mean.PM2.5.Concentration ~ Date, random = ~1|Site.Name)
summary(PMair.auto) #intercept = 82.2 and Date= -0.004
```

```
## Linear mixed-effects model fit by REML
##  Data: PMair
##        AIC      BIC    logLik
##   1865.215 1880.543 -928.6076
##
## Random effects:
##  Formula: ~1 | Site.Name
##         (Intercept) Residual
## StdDev:    1.650184 3.559209
##
## Fixed effects: Daily.Mean.PM2.5.Concentration ~ Date
##                Value Std.Error  DF   t-value p-value
## (Intercept) 90.46502  34.57133 339  2.616764  0.0093
```

```
## Date        -0.00473   0.00195 339 -2.425102  0.0158
##  Correlation:
##      (Intr)
## Date -0.999
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.38072443 -0.63365107 -0.09616694  0.61426094  3.42056220
##
## Number of Observations: 343
## Number of Groups: 3
```

```
PMair.auto
```

```
## Linear mixed-effects model fit by REML
##   Data: PMair
##   Log-restricted-likelihood: -928.6076
##   Fixed: Daily.Mean.PM2.5.Concentration ~ Date
##  (Intercept)         Date
## 90.465022634 -0.004727976
##
## Random effects:
##  Formula: ~1 | Site.Name
##         (Intercept) Residual
## StdDev:    1.650184 3.559209
##
## Number of Observations: 343
## Number of Groups: 3
```

```
ACF(PMair.auto) #=0.4740630033 is second value which is the degree of autocorrelation at first level
```

```
##    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
```

```r
#d)
PMair.mixed <- lme(data= PMair, Daily.Mean.PM2.5.Concentration ~ Date, random = ~1|Site.Name, correlati
                method = "REML")
summary(PMair.mixed)
```

```
## Linear mixed-effects model fit by REML
##  Data: PMair
##        AIC      BIC   logLik
##   1756.622 1775.781 -873.311
##
## Random effects:
##  Formula: ~1 | Site.Name
##         (Intercept) Residual
## StdDev: 0.001079826 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  60.63584 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. When you account for the autocorrelation, there is not a significant change in PM2.5 concentrations over the course of 2018 (p-value = 0.36)

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.

```r
PMair.fixed <- gls(data= PMair, Daily.Mean.PM2.5.Concentration ~ Date, method = "REML")
summary(PMair.fixed)
```

```
## Generalized least squares fit by REML
##   Model: Daily.Mean.PM2.5.Concentration ~ Date
##   Data: PMair
##        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
```

```r
anova(PMair.mixed, PMair.fixed)
```

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

Which model is better?

> ANSWER: The mixed linear model (which includes Site Name as a random effects is better) because it has a lower AIC value, and the two models are significantly different from one another as shown by the p-value which is < 0.0001.

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).

```r
PeterPaul.nutrients.surface <-
  PeterPaul.nutrients %>%
  select(-lakeid, -depth_id, -comments) %>%
  filter(depth == 0) %>%
  filter(!is.na(tn_ug))

# Splitting dataset by lake
Peter.nutrients.surface <- filter(PeterPaul.nutrients.surface, lakename == "Peter Lake")
Paul.nutrients.surface <- filter(PeterPaul.nutrients.surface, lakename == "Paul Lake")

#Mann-Kendall test for Peter
mk.test(Peter.nutrients.surface$tn_ug) #there is definitely a trend over time (p-value < 0.0001)
```

```
##
##  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
```

```r
#Pettitt test for Peter
pettitt.test(Peter.nutrients.surface$tn_ug) #change point at 36
```

```
##
##  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
#Re-run separate Mann-Kendall for each change point for Peter
mk.test(Peter.nutrients.surface$tn_ug[1:36]) #no change over time for this section
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[1:36]
## z = 0.040863, n = 36, p-value = 0.9674
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 4.000000e+00 5.390000e+03 6.349206e-03
```

```r
mk.test(Peter.nutrients.surface$tn_ug[37:98]) #another change point detected within this section becaus
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[37:98]
## z = 2.9642, n = 62, p-value = 0.003035
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 4.890000e+02 2.710433e+04 2.585933e-01
```

```r
#Pettitt test for Peter to detect where second change point is
pettitt.test(Peter.nutrients.surface$tn_ug[37:98])
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Peter.nutrients.surface$tn_ug[37:98]
## U* = 522, p-value = 0.002339
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               20
```

```r
#Another separate Mann-Kendall for each change point section for Peter
mk.test(Peter.nutrients.surface$tn_ug[37:57]) #no change over time for this section
```

```
##
##  Mann-Kendall trend test
##
```

```
## data:  Peter.nutrients.surface$tn_ug[37:57]
## z = -0.9965, n = 21, p-value = 0.319
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
##  -34.0000000 1096.6666667   -0.1619048
```

```
mk.test(Peter.nutrients.surface$tn_ug[58:98]) #no change over time for this section
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tn_ug[58:98]
## z = 0.14602, n = 41, p-value = 0.8839
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 1.400000e+01 7.926667e+03 1.707317e-02
```

```
#Mann-Kendall test for Paul
mk.test(Paul.nutrients.surface$tn_ug) #no significant trend over time for Paul lake (p-value = 0.73)
```

```
##
##  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
```

```
#Pettitt test for Paul
pettitt.test(Paul.nutrients.surface$tn_ug) #change point detected at 16 but result is not significant,
```

```
##
##  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
```

What are the results of this test?

> ANSWER: Results for Peter: two change points - one at 36 and one at 57. Beyond line 36, if you split up the data even further, you see no significant positive or negative trend. Results for Paul: no change point or positive/negative trend over time. See annotations in above code for p-values.

5. Generate a graph that illustrates the TN concentrations over time, coloring by lake and adding vertical line(s) representing changepoint(s).

```
ggplot(PeterPaul.nutrients.surface, aes(x = sampledate, y = tn_ug, color = lakename)) +
  geom_point() +
  scale_color_manual(values = c("#7fcdbb", "#253494")) +
  geom_vline(xintercept = as.Date("1993-06-02"), color="#253494", lty = 2) +
```

```r
geom_vline(xintercept = as.Date("1994-06-29"), color="#253494", lty = 2) +
labs(x="Date", y="Total Nitrogen Concentration", color="Lake")
```