

# Assignment 8: Time Series Analysis

*Lindsay Roth*

## 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/lindsayroth/Documents/MEM 1st Year/Spring 2019/Env_Data_Analytics/Env_Data_Analytics"
```

```
library(tidyverse)
```

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

```
## v ggplot2 3.1.0    v purrr   0.3.0
## v tibble  2.0.1    v dplyr   0.7.8
## v tidyr   0.8.2    v stringr 1.3.1
## v readr   1.3.1    v forcats 0.3.0
```

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

library(lubridate)

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

library(nlme)

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

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'.

library(multcompView)
library(trend)

PM25.2018 <- read.csv("./Data/Raw/EPAair_PM25_NC2018_raw.csv")
Nutrients.PeterPaul <- read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv")

lindsay_theme <- theme_classic(base_size = 12) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "right")
theme_set(lindsay_theme)

Nutrients.PeterPaul$sampldate <- as.Date(Nutrients.PeterPaul$sampldate, format = "%Y-%m-%d")
class(Nutrients.PeterPaul$sampldate)

## [1] "Date"

PM25.2018$Date <- as.Date(PM25.2018$Date, format = "%m/%d/%y")
class(PM25.2018$Date)

## [1] "Date"
```

## 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.

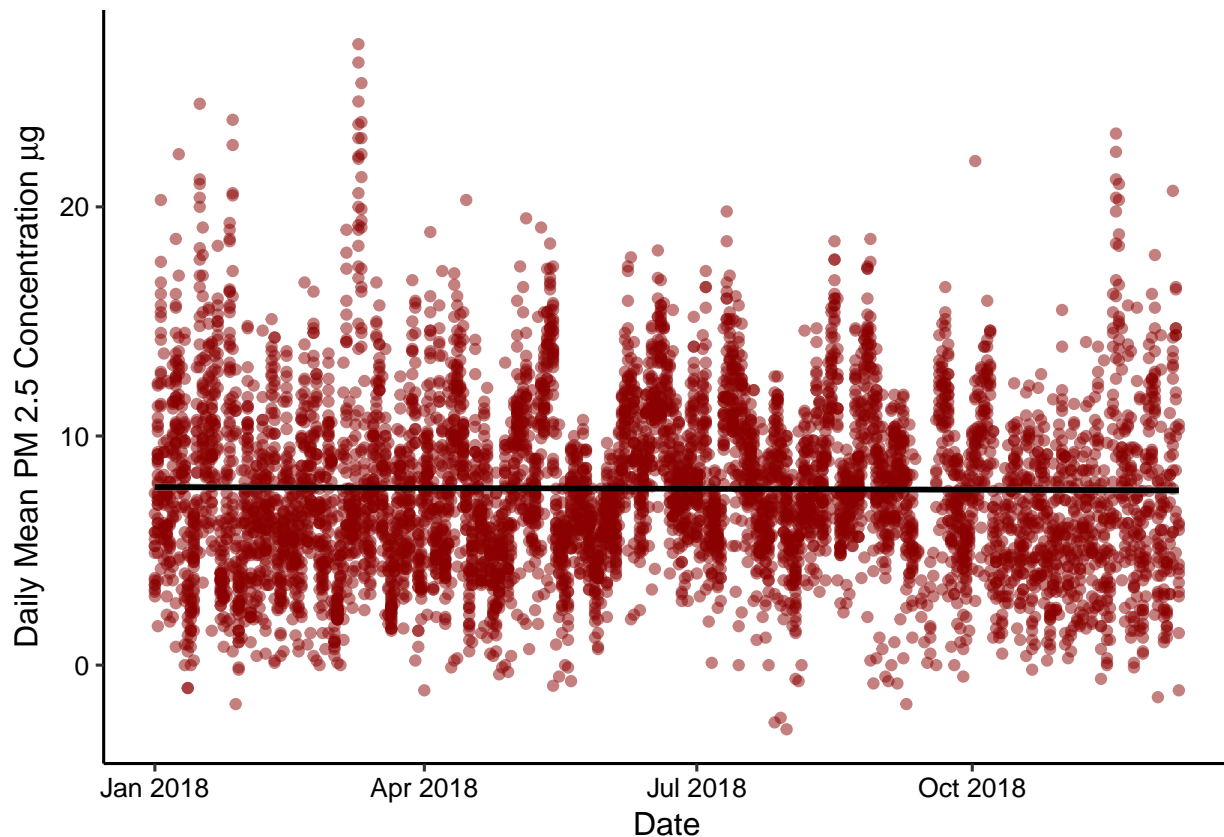
3a. Illustrate PM2.5 concentrations by date. Do not split aesthetics by site.

```
PM25.wrangled <- PM25.2018 %>% na.exclude()

ConcTest.mixed <- lme(data = PM25.wrangled,
  Daily.Mean.PM2.5.Concentration ~ Date,
  random = ~1|Site.Name)
summary(ConcTest.mixed)

## Linear mixed-effects model fit by REML
## Data: PM25.wrangled
##      AIC      BIC    logLik
## 35104.9 35132.07 -17548.45
##
## Random effects:
## Formula: ~1 | Site.Name
##      (Intercept) Residual
## StdDev:      1.728641 3.450187
##
## Fixed effects: Daily.Mean.PM2.5.Concentration ~ Date
##              Value Std.Error   DF   t-value p-value
## (Intercept) 14.297350  7.965514 6566   1.794906  0.0727
## Date        -0.000391  0.000450 6566  -0.870232  0.3842
## Correlation:
##      (Intr)
## Date -0.999
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -3.4360977 -0.6896514 -0.1274684  0.6008556  5.2500490
##
## Number of Observations: 6586
## Number of Groups: 19

ggplot(PM25.wrangled, aes(x = Date, y = Daily.Mean.PM2.5.Concentration)) +
  geom_point(color = "dark red", alpha = 0.5) +
  labs(y = expression(paste("Daily Mean PM 2.5 Concentration ", mu, 'g' ))) +
  geom_smooth(method = lm, color = "black")
```



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.

```
PM25.wrangled2 = PM25.wrangled[order(PM25.wrangled[, 'Date'], -PM25.wrangled[, 'Site.ID']),]
PM25.wrangled2 = PM25.wrangled2[!duplicated(PM25.wrangled2$Date),]
```

```
ConcTest.auto2 <- lme(data = PM25.wrangled2,
  Daily.Mean.PM2.5.Concentration ~ Date,
  random = ~1|Site.Name)
summary(ConcTest.auto2)
```

```
## Linear mixed-effects model fit by REML
## Data: PM25.wrangled2
##      AIC      BIC    logLik
## 1866.218 1881.545 -929.1089
##
## Random effects:
## Formula: ~1 | Site.Name
##      (Intercept) Residual
## StdDev:      1.423597 3.565364
##
## Fixed effects: Daily.Mean.PM2.5.Concentration ~ Date
##              Value Std.Error   DF   t-value p-value
```

```
## (Intercept) 92.21182 34.59209 337 2.665691 0.0081
## Date -0.00484 0.00195 337 -2.480165 0.0136
## Correlation:
## (Intr)
## Date -1
##
## Standardized Within-Group Residuals:
## Min Q1 Med Q3 Max
## -2.3758299 -0.6310034 -0.1257178 0.6195616 3.4151032
##
## Number of Observations: 343
## Number of Groups: 5
```

```
ConcTest.auto2
```

```
## Linear mixed-effects model fit by REML
## Data: PM25.wrangled2
## Log-restricted-likelihood: -929.1089
## Fixed: Daily.Mean.PM2.5.Concentration ~ Date
## (Intercept) Date
## 92.211818212 -0.004837678
##
## Random effects:
## Formula: ~1 | Site.Name
## (Intercept) Residual
## StdDev: 1.423597 3.565364
##
## Number of Observations: 343
## Number of Groups: 5
```

```
ACF(ConcTest.auto2)
```

```
## lag ACF
## 1 0 1.000000000
## 2 1 0.515649380
## 3 2 0.194630440
## 4 3 0.118437238
## 5 4 0.127546199
## 6 5 0.101062520
## 7 6 0.058085196
## 8 7 -0.052907307
## 9 8 0.017656228
## 10 9 0.012029022
## 11 10 -0.003788449
## 12 11 -0.020205308
## 13 12 -0.044426855
## 14 13 -0.055382050
## 15 14 -0.065561649
## 16 15 -0.123576691
## 17 16 -0.055173547
## 18 17 0.002979207
## 19 18 0.025150933
## 20 19 -0.015170235
## 21 20 -0.143012523
## 22 21 -0.155027059
## 23 22 -0.060167417
```

```
## 24 23 0.003982492
## 25 24 0.042233417
## 26 25 0.001384018
```

```
ConcTest.mixed2 <- lme(data = PM25.wrangled2,
  Daily.Mean.PM2.5.Concentration ~ Date,
  random = ~1|Site.Name,
  correlation = corAR1(form = ~Date|Site.Name, value = 0.516),
  method = "REML")
summary(ConcTest.mixed2)
```

```
## Linear mixed-effects model fit by REML
## Data: PM25.wrangled2
##      AIC      BIC    logLik
## 1760.033 1779.192 -875.0163
##
## Random effects:
## Formula: ~1 | Site.Name
##      (Intercept) Residual
## StdDev:    0.9366661 3.586786
##
## Correlation Structure: ARMA(1,0)
## Formula: ~Date | Site.Name
## Parameter estimate(s):
##      Phi1
## 0.5324827
## Fixed effects: Daily.Mean.PM2.5.Concentration ~ Date
##              Value Std.Error  DF   t-value p-value
## (Intercept) 86.46530  59.96524 337   1.441923  0.1503
## Date       -0.00449   0.00338 337  -1.325756  0.1858
## Correlation:
##      (Intr)
## Date -1
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -2.3339408 -0.6212666 -0.1099967  0.6243141  3.4234858
##
## Number of Observations: 343
## Number of Groups: 5
```

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

ANSWER: There is not a significant increasing or decreasing trend in the PM2.5 concentrations in 2018 ( $p > 0.05$ ).

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.

```
ConcTest.fixed <- gls(data = PM25.wrangled2,
  Daily.Mean.PM2.5.Concentration ~ Date)
summary(ConcTest.fixed)
```

```
## Generalized least squares fit by REML
## Model: Daily.Mean.PM2.5.Concentration ~ Date
## Data: PM25.wrangled2
##      AIC      BIC    logLik
```

```
##    1865.261 1876.757 -929.6307
##
## Coefficients:
##              Value Std.Error   t-value p-value
## (Intercept) 98.66793   34.60585   2.851192  0.0046
## Date        -0.00514    0.00195  -2.627388  0.0090
##
## Correlation:
##      (Intr)
## Date -1
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -2.3528126 -0.6429287 -0.1150800  0.6385145  3.4060776
##
## Residual standard error: 3.584632
## Degrees of freedom: 343 total; 341 residual
```

```
anova(ConcTest.mixed2,ConcTest.fixed)
```

```
##              Model df      AIC      BIC    logLik    Test  L.Ratio
## ConcTest.mixed2    1  5 1760.033 1779.192 -875.0163
## ConcTest.fixed     2  3 1865.261 1876.757 -929.6307 1 vs 2 109.2288
##              p-value
## ConcTest.mixed2
## ConcTest.fixed    <.0001
```

Which model is better?

ANSWER: The mixed model is better because it has a lower AIC score, and when the models were compared using an anova, the fixed model was significantly different from the mixed model ( $p < 0.0001$ ).

## 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).

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

Peter.nutrients.surface <- filter(PeterPaul.nutrients.surface, lakename == "Peter Lake")
Paul.nutrients.surface <- filter(PeterPaul.nutrients.surface, lakename == "Paul 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
```

```
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
```

```
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
```

```
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
```

```
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
```

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

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



```

##
## data: Peter.nutrients.surface$tn_ug[36:98]
## z = 3.1909, n = 63, p-value = 0.001418
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## 5.390000e+02 2.842700e+04 2.759857e-01
mk.test(Paul.nutrients.surface$tn_ug[1:15])

##
## Mann-Kendall trend test
##
## data: Paul.nutrients.surface$tn_ug[1:15]
## z = -2.6723, n = 15, p-value = 0.007533
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## -55.0000000 408.3333333 -0.5238095
mk.test(Paul.nutrients.surface$tn_ug[16:99])

##
## Mann-Kendall trend test
##
## data: Paul.nutrients.surface$tn_ug[16:99]
## z = 2.2058, n = 84, p-value = 0.0274
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## 5.720000e+02 6.700867e+04 1.640849e-01
pettitt.test(Peter.nutrients.surface$tn_ug[36:98]) #change at 57

##
## Pettitt's test for single change-point detection
##
## data: Peter.nutrients.surface$tn_ug[36:98]
## U* = 560, p-value = 0.001213
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               21
pettitt.test(Paul.nutrients.surface$tn_ug[16:99]) #change at 52

##
## Pettitt's test for single change-point detection
##
## data: Paul.nutrients.surface$tn_ug[16:99]
## U* = 852, p-value = 0.001403
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               36

```

```

mk.test(Peter.nutrients.surface$tn_ug[36:56])

##
## Mann-Kendall trend test
##
## data: Peter.nutrients.surface$tn_ug[36:56]
## z = -1.0569, n = 21, p-value = 0.2906
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## -36.0000000 1096.6666667 -0.1714286

mk.test(Peter.nutrients.surface$tn_ug[57:98])

##
## Mann-Kendall trend test
##
## data: Peter.nutrients.surface$tn_ug[57:98]
## z = 0.15172, n = 42, p-value = 0.8794
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
##  15.0000000 8514.3333333  0.0174216

mk.test(Paul.nutrients.surface$tn_ug[16:51])

##
## Mann-Kendall trend test
##
## data: Paul.nutrients.surface$tn_ug[16:51]
## z = -1.8116, n = 36, p-value = 0.07005
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## -134.0000000 5390.0000000 -0.2126984

mk.test(Paul.nutrients.surface$tn_ug[52:99])

##
## Mann-Kendall trend test
##
## data: Paul.nutrients.surface$tn_ug[52:99]
## z = -1.2888, n = 48, p-value = 0.1975
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##          S          varS          tau
## -146.0000000 12658.6666667 -0.1294326

```

What are the results of this test?

ANSWER: There are breaks in the Paul Lake Nutrient data on September 2, 1991 and May 26, 1994. There are Breaks in the Peter Lake nutrient data on June 2, 1993 and June 29, 1994. In between these breaks, there is no significant trends in the data ( $p > 0.05$ ).

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 = sampleddate, y = tn_ug, color = lakename)) +
  geom_point() +
  geom_vline(xintercept = as.Date("1991-09-02"), linetype = 2, color = "#7fcdbb") + #Paul
  geom_vline(xintercept = as.Date("1994-05-26"), linetype = 2, color = "#7fcdbb") + #Paul
  geom_vline(xintercept = as.Date("1993-06-02"), linetype = 2, color = "#253494") + #Peter
  geom_vline(xintercept = as.Date("1994-06-29"), linetype = 2, color = "#253494") + #Peter
  scale_color_manual(values = c("#7fcdbb", "#253494")) +
  labs(x = "Date", y = expression(paste("Total Nitrogen ", mu, 'g' )), color = "Lake")
```

