14: Time Series Analysis

Environmental Data Analytics | Kateri Salk

Spring 2019

LESSON OBJECTIVES

1. Describe the aspects of hierarchical models, fixed effects, and random effects
2. Choose and justify appropriate statistical models when time is an explanatory variable
3. Apply Mann-Kendall and Seasonal Mann-Kendall to datasets with temporal components

FRA: Random effect. we are not interested in the direct effect but we want to take account the variance. We can compare AIC values between the fix and random. Also if you dont see big diff between the coeff. or its impact in the predictions over time.

SET UP YOUR DATA ANALYSIS SESSION

```r
getwd()
```

```
## [1] "C:/Users/Felipe/OneDrive - Duke University/1. DUKE/1. Ramos 2 Semestre/EOS-872 Env. Data Analyti
```

```r
library(tidyverse)
#install.packages("trend")
library(trend)


PeterPaul.nutrients <- read.csv("./Data/Processed/NTL-LTER_Lake_Nutrients_PeterPaul_Processed.csv")
USGS.flow.data <- read.csv("./Data/Raw/USGS_Site02085000_Flow_Raw.csv")

# Rename columns
colnames(USGS.flow.data) <- c("agency_cd", "site_no", "datetime",
                              "discharge.max", "discharge.max.approval",
                              "discharge.min", "discharge.min.approval",
                              "discharge.mean", "discharge.mean.approval",
                              "gage.height.max", "gage.height.max.approval",
                              "gage.height.min", "gage.height.min.approval",
                              "gage.height.mean", "gage.height.mean.approval")

# Set date to date format
PeterPaul.nutrients$sampledate <- as.Date(PeterPaul.nutrients$sampledate,
                                          format = "%Y-%m-%d")
USGS.flow.data$datetime <- as.Date(USGS.flow.data$datetime,
                              format = "%m/%d/%y")

mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

NONPARAMETRIC TREND TESTS

In many environmental datasets (especially climate and hydrology), we might not expect a linear trend in the response variable over time. In this case, we will need to employ a nonparametric test to determine whether there is a monotonic trend (i.e., consistent increase or decrease but not necessarily linear) over time. We will illustrate a few examples of nonparametric trend tests today with the `trend` package.

A vignette for the `trend` package can be found here: https://cran.r-project.org/web/packages/trend/vignettes/trend.pdf. More details here: https://cran.r-project.org/web/packages/trend/trend.pdf.

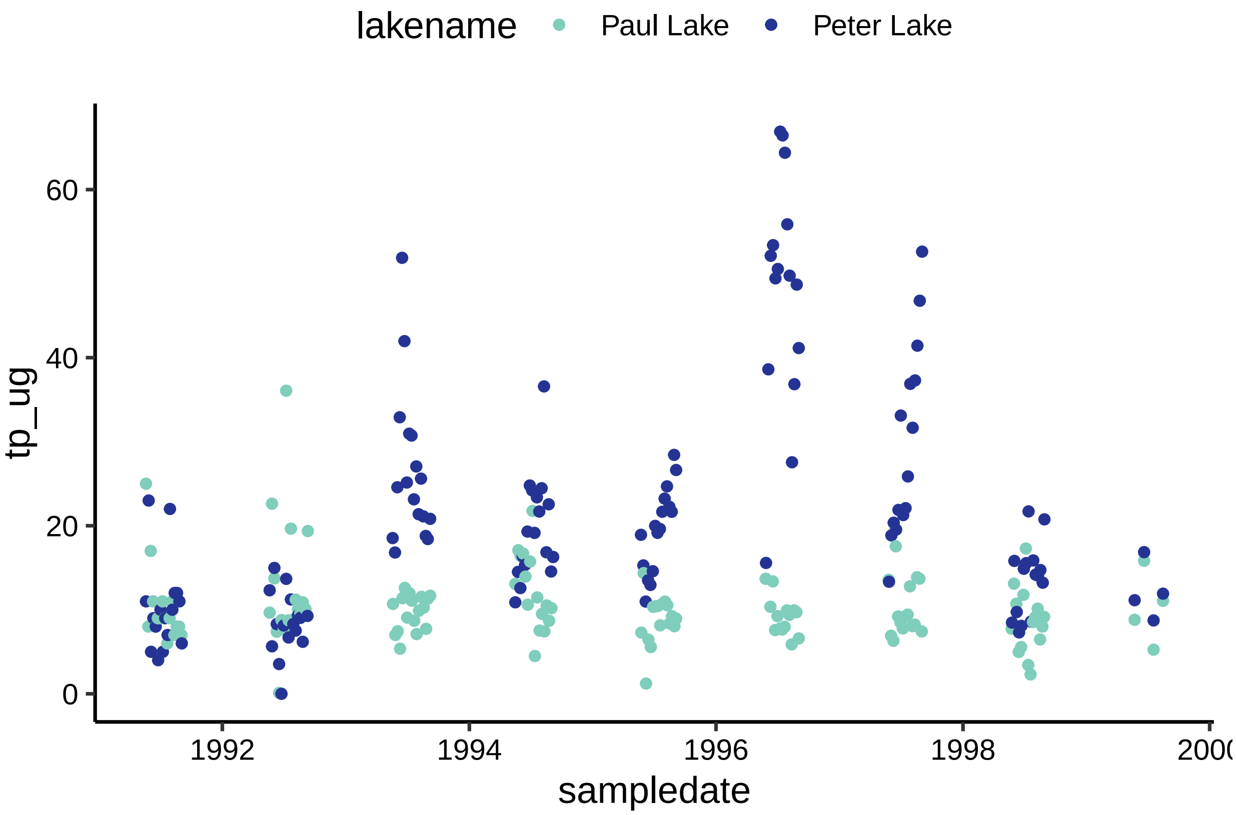
We will run a Mann-Kendall and a Seasonal Mann-Kendall test today, but there are additional variants of these tests within the package including a correlated Seasonal Mann-Kendall test, a multivariate Mann-Kendall test, a partial Mann-Kendall test, a partial correlation trend test, and a Cox and Stuart trend test. Look into the documentation for these tests to determine which one is appropriate for your purposes.

Mann-Kendall Test

A Mann-Kendall test will analyze whether there is a monotonic trend in the response variable over time. Let's use the Mann-Kendall test to investigate whether there is a trend in total phosphorus concentrations in Peter Lake over time.

```
# Wrangle our dataset
PeterPaul.nutrients.surface <-
  PeterPaul.nutrients %>%
  select(-lakeid, -depth_id, -comments) %>%
  filter(depth == 0) %>%
  filter(!is.na(tp_ug))

# Initial visualization of data
ggplot(PeterPaul.nutrients.surface, aes(x = sampledate, y = tp_ug, color = lakename)) +
  geom_point() +
  scale_color_manual(values = c("#7fcdbb", "#253494"))
```



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

# Paul lake is control. We want to see if they have diff trends.

# Run a Mann-Kendall test
mk.test(Peter.nutrients.surface$tp_ug)
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tp_ug
## z = 4.3966, n = 132, p-value = 1.099e-05
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 2.236000e+03 2.584133e+05 2.587065e-01
```

```
#Mann-Kendall trend test

#data:  Peter.nutrients.surface$tp_ug
#z = 4.3966, n = 132, p-value = 1.099e-05
#     Low pvalue. there is a trend over time. z positive. positive trend over time.
#alternative hypothesis: true S is not equal to 0      No trend over time
#sample estimates:
#           S         varS          tau
```

```
#2.236000e+03 2.584133e+05 2.587065e-01

#Is there a change point of the trend?
```

However, it looks like there might be a breakpoint in our dataset. Further, we know that Peter Lake underwent experimental fertilization starting in May 1993, a perturbation which we might expect to have induced a regime shift in the ecosystem. In this case, we might want to find out whether there is a breakpoint, or changepoint, in our dataset.

Pettitt's Test

Pettitt's test is also included in the `trend` package. This nonparametric test will determine whether there is a shift in the central tendency of the time series and will tell us at what point the changepoint occurs (if it detects one). Note: Pettitt's Test will only test for one changepoint, and further tests must be run if multiple change points are suspected.

```
# Test for change point
pettitt.test(Peter.nutrients.surface$tp_ug)
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Peter.nutrients.surface$tp_ug
## U* = 2767, p-value = 4.92e-09
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                                35
```

```
#   Pettitt's test for single change-point detection

#data:  Peter.nutrients.surface$tp_ug
#U* = 2767, p-value = 4.92e-09
#alternative hypothesis: two.sided
#sample estimates:
#probable change point at time K        # where is the change located. 1993. when the experimet started
#                               35



# Run separate Mann-Kendall for each change point  # to see if there is another change.
mk.test(Peter.nutrients.surface$tp_ug[1:34]) # no significant trend (becaue the change is balancing it)
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tp_ug[1:34]
## z = 0.14834, n = 34, p-value = 0.8821
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 1.100000e+01 4.544333e+03 1.971355e-02
```

```
mk.test(Peter.nutrients.surface$tp_ug[35:132]) # no significant trend
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tp_ug[35:132]
## z = -1.6329, n = 98, p-value = 0.1025
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -5.330000e+02  1.061503e+05 -1.121397e-01
```

```
# Is there a second change point?
pettitt.test(Peter.nutrients.surface$tp_ug[35:132]) # there is a trend change.
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Peter.nutrients.surface$tp_ug[35:132]
## U* = 1201, p-value = 0.0002228
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               79
```

```
# Run another Mann-Kendall for the second change point
mk.test(Peter.nutrients.surface$tp_ug[35:113]) #here we detect the trend
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tp_ug[35:113]
## z = 2.7432, n = 79, p-value = 0.006084
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##            S         varS          tau
## 6.490000e+02 5.580033e+04 2.106459e-01
```

```
mk.test(Peter.nutrients.surface$tp_ug[114:132]) #here no trend.
```

```
##
##  Mann-Kendall trend test
##
## data:  Peter.nutrients.surface$tp_ug[114:132]
## z = 0.62974, n = 19, p-value = 0.5289
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##           S        varS         tau
##  19.0000000 817.0000000   0.1111111
```

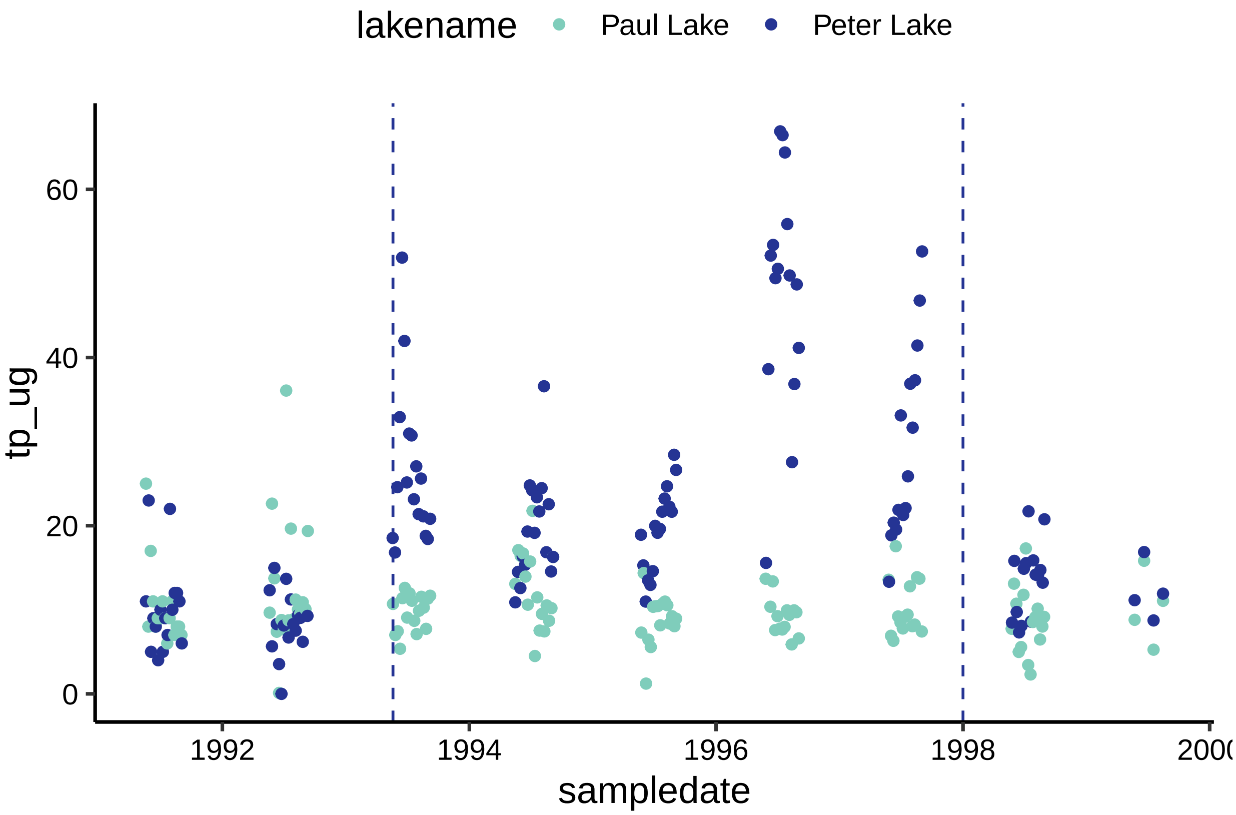
```
# Run the same test for Paul Lake.
mk.test(Paul.nutrients.surface$tp_ug) #non significant
```

```
##
##  Mann-Kendall trend test
##
## data:  Paul.nutrients.surface$tp_ug
## z = -1.4366, n = 131, p-value = 0.1508
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##             S          varS           tau
## -7.230000e+02  2.525897e+05 -8.498887e-02
```

```
pettitt.test(Paul.nutrients.surface$tp_ug) #non significant
```

```
##
##  Pettitt's test for single change-point detection
##
## data:  Paul.nutrients.surface$tp_ug
## U* = 1024, p-value = 0.1244
## alternative hypothesis: two.sided
## sample estimates:
## probable change point at time K
##                               58
```

```
# Add vertical lines to the original graph to represent change points
ggplot(PeterPaul.nutrients.surface, aes(x = sampledate, y = tp_ug, color = lakename)) +
  geom_point() +
  scale_color_manual(values = c("#7fcdbb", "#253494")) +
  geom_vline(xintercept=as.Date("1993/05/20"), color= "#253494", lty=2) +
   geom_vline(xintercept=as.Date("1998/01/01"), color= "#253494", lty=2)
```



```
  #+ scale_y_log10() use when you have realy low and high values.
#you can do sense slope to quantify the slope
```

Seasonal Mann-Kendall

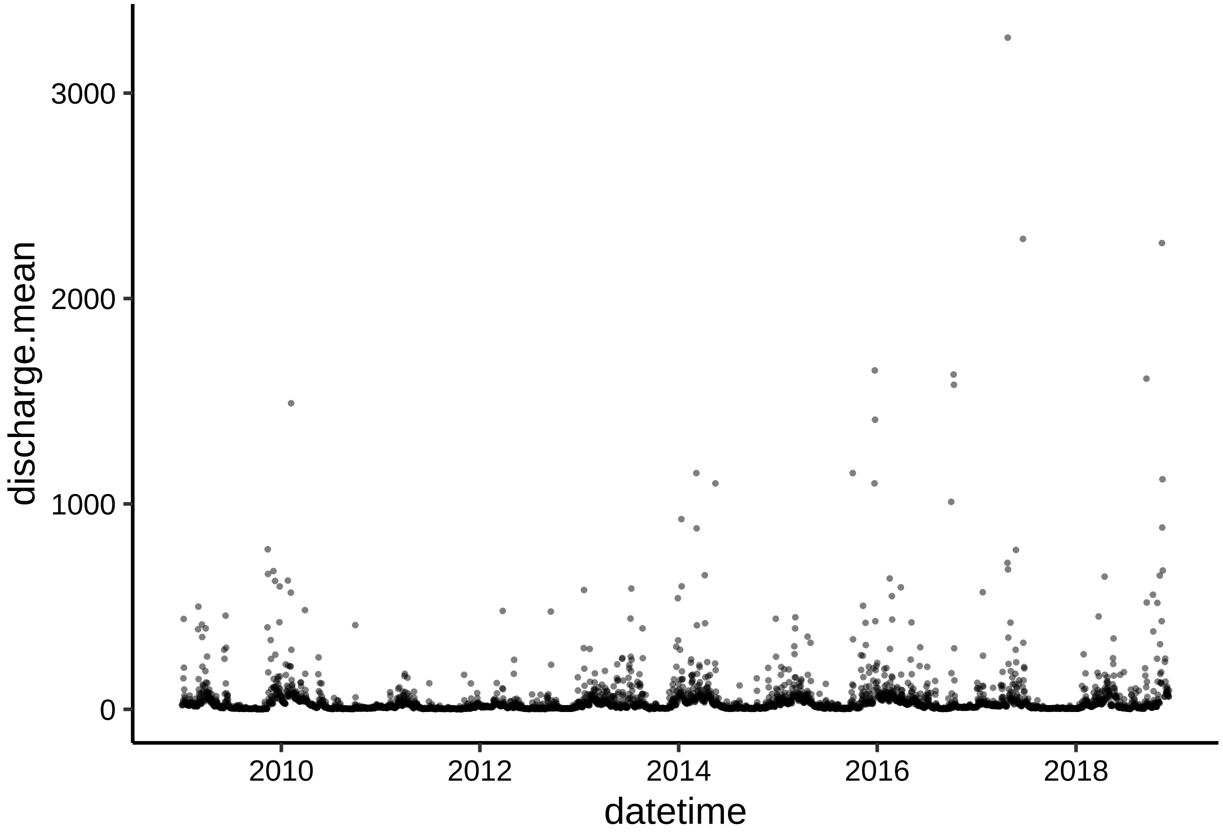
Like a **Mann-Kendall Test**, the **Seasonal Mann-Kendall Test**, or **Hirsch-Slack Test**, analyzes trends in response variables over time. It replaces the traditional Mann-Kendall when there are seasonal trends in a dataset that obscure the overall direction of the trend. It is important to note that "seasonal" does not necessarily equate to actual seasons but can represent any time period within which there are oscillating temporal trends. The test needs at least two seasons to operate.

For instance, we might want to know whether there is a change in discharge of the Eno River over the last 10 years.

```r
# Wrangle the USGS dataset
USGS.flow.data.trimmed <- USGS.flow.data %>%
  select(datetime, discharge.mean) %>%
  filter(datetime > as.Date("2008-12-31") & datetime < as.Date("2019-01-01"))

# Visualize the data
ggplot(USGS.flow.data.trimmed, aes(x = datetime, y = discharge.mean)) +
  geom_point(size = 0.5, alpha = 0.5)
```

```
## Warning: Removed 16 rows containing missing values (geom_point).
```



Interpolation

Some situations may require us to predict values for data points that fall within the time frame of our analyses but were not sampled. For instance, the `smk.test` function needs to take a time series format rather than a data frame, which cannot have any NAs. In this case, we will want to make an estimate of the missing values based on what we know about the dataset using a method called **interpolation.** There are several options for interpolation:

- **Means interpolation:** Defines values between sampled values as the mean value within a dataset. Uses the R function `aggregate`.
- **Piecewise constant interpolation:** Defines values between sampled values as the value of the nearest sampled value. Uses the R function `approx` with `method = "constant"`
- **Linear interpolation:** Defines values between sampled values based on the slope between sampled values. Uses the R function `approx` with `method = "linear"`
- **Spline interpolation:** Defines values between sampled values based on polynomial functions between sampled values and chooses the polynomials so that they fit smoothly together. Uses the R function `splinefun`.

Question: Under what circumstances would you consider each of these options for interpolation?

> ANSWER: Linear. because continues data.

Tip: Check your dataset to see if there is an NA value in the first row. You may need to add a value for that first row or trim the dataset so that the new first row corresponds to the first measurement.

```
# Run a linear interpolation of the dataset to fill in gaps
USGS.flow.data.interpolated <- approx(USGS.flow.data.trimmed$datetime,
                                      USGS.flow.data.trimmed$discharge.mean,
                                      method = "linear", n = 3630)
# so you dont generate new points. I think this was to fill rows with NAs. It give us a list.

# Turn the interpolated dataset into a proper dataframe
USGS.flow.data.interpolated <- do.call(cbind.data.frame, USGS.flow.data.interpolated)
names(USGS.flow.data.interpolated) <- c("Date", "Discharge")
USGS.flow.data.interpolated$Date <- as.Date(USGS.flow.data.interpolated$Date,
                                            origin = "1970/01/01")

# Create a time series object FRA: we need this for the seasonal test.
USGS.flow.data.timeseries <- ts(USGS.flow.data.interpolated$Discharge,
                                start = c(2009, 1) ,frequency = 12)
#12 beacuse is monthly. you can out 365.

# Run a Seasonal Mann-Kendall test
USGS.smktest <- smk.test(USGS.flow.data.timeseries)
USGS.smktest # overall trend. similar to the first test that we did. pvalue low, there is a trend.
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
## data:  USGS.flow.data.timeseries
## z = 7.6477, p-value = 2.047e-14
## alternative hypothesis: true S is not equal to 0
## sample estimates:
##        S     varS
##    46576 37089370
```

```
summary(USGS.smktest) #gives as each month. the trends each season. Season 11 biggest trend.
```

```
##
##  Seasonal Mann-Kendall trend test (Hirsch-Slack test)
##
```

```
## data: USGS.flow.data.timeseries
## alternative hypothesis: two.sided
##
## Statistics for individual seasons
##
## H0
##                       S    varS   tau     z   Pr(>|z|)
## Season 1:   S = 0  4158 3106080 0.091 2.359 0.01833881   *
## Season 2:   S = 0  3495 3106087 0.076 1.983 0.04742183   *
## Season 3:   S = 0  2907 3106086 0.064 1.649 0.09917234   .
## Season 4:   S = 0  2614 3106083 0.057 1.483 0.13817263
## Season 5:   S = 0  2762 3106073 0.060 1.567 0.11720613
## Season 6:   S = 0  3381 3106074 0.074 1.918 0.05513217   .
## Season 7:   S = 0  2982 3075479 0.066 1.700 0.08916281   .
## Season 8:   S = 0  4417 3075480 0.097 2.518 0.01179905   *
## Season 9:   S = 0  3984 3075481 0.088 2.271 0.02313537   *
## Season 10:   S = 0 5431 3075484 0.120 3.096 0.00195952  **
## Season 11:   S = 0 5921 3075481 0.130 3.376 0.00073625 ***
## Season 12:   S = 0 4524 3075485 0.100 2.579 0.00990553  **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# No trend in season 5 for example
```

Interpreting results of the Seasonal Mann-Kendall Test:

- Overall z score and p-value: test the alternative hypothesis that the true change in response variable over time is not equal to zero
- Monthly z score and p-value: test the alternative hypothesis that the true change in response variable over time for a given month is not equal to zero
- S: reports trend. A positive value indicates response variable increased over time, and a negative value indicates response variable decreased over time

Question: How would you interpret the results of the Seasonal Mann-Kendall test for this example?

ANSWER: