9: Time Series Analysis

Water Data Analytics | Kateri Salk

Spring 2022

Lesson Objectives

1. Discuss the purpose and application of time series analysis for hydrologic data
2. Decompose time series into individual components

Session Set Up

```
getwd()
```

```
## [1] "/Users/katerisalk/Box Sync/Courses/Water Data Analytics/Lessons"
```

```
library(tidyverse)
library(lubridate)
library(dataRetrieval)

theme_set(theme_classic())
```

Time Series Analysis

Time series are a special class of dataset, where a response variable is tracked over time. The frequency of measurement and the timespan of the dataset can vary widely. At its most simple, a time series model includes an explanatory time component and a response variable. Mixed models can include additional explanatory variables (check out the `nlme` and `lme4` R packages). We will be covering a few simple applications of time series analysis in these lessons.

Opportunities

Analysis of time series presents several opportunities. In aquatic sciences, some of the most common questions we can answer with time series modeling are:

- Has there been an increasing or decreasing **trend** in the response variable over time?
- Can we **forecast** conditions in the future?

Challenges

Time series datasets come with several caveats, which need to be addressed in order to effectively model the system. A few common challenges that arise (and can occur together within a single dataset) are:

- **Autocorrelation**: Data points are not independent from one another (i.e., the measurement at a given time point is dependent on previous time point(s))
- **Data gaps**: Data are not collected at regular intervals, necessitating *interpolation* between measurements.
- **Seasonality**: Cyclic patterns in variables occur at regular intervals, impeding clear interpretation of a monotonic (unidirectional) trend.
- **Heteroscedasticity**: The variance of the time series is not constant over time

- **Covariance**: the covariance of the time series is not constant over time

Visualizing a time series dataset

Today, we will analyze discharge data from the Neuse River in North Carolina. Let's first look at what types of data are available for this dataset.

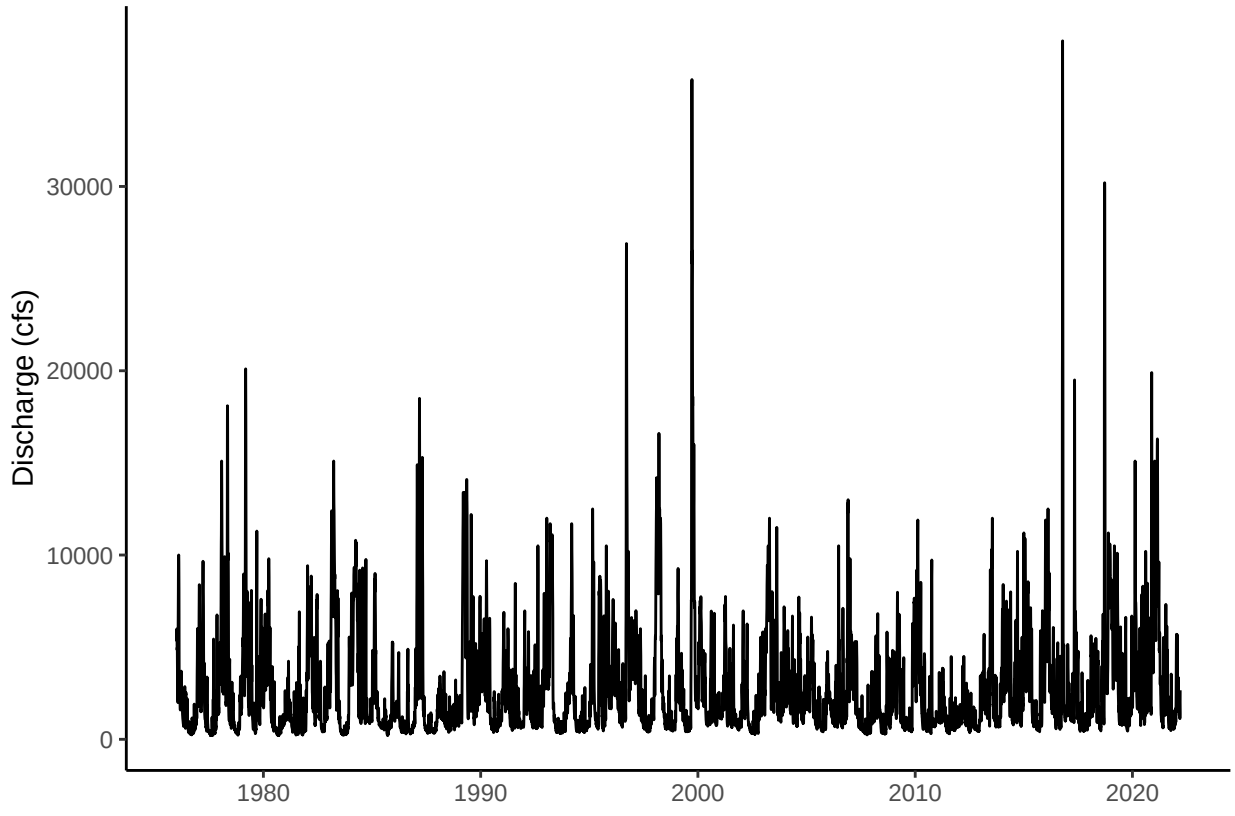
```
NeuseSummary <- whatNWISdata(siteNumbers = "06719505")
```

Notice that mean daily discharge has been measured at this site since 1974, with over 17,000 measurements available. This will be a robust time series dataset for us to analyze changes that have occurred over the past five decades.

```
# Import data

NeuseFlow <- readNWISdv(siteNumbers = "02089500",
                        parameterCd = "00060", # discharge (cfs)
                        startDate = "1976-01-01",
                        endDate = "")
names(NeuseFlow)[4:5] <- c("Discharge", "Approval.Code")


# Plot discharge over time
ggplot(NeuseFlow, aes(x = Date, y = Discharge)) +
  geom_line() +
  labs(x = "", y = "Discharge (cfs)")
```



How would you characterize the time series of discharge and conductivity at this location? Do you see a

linear trend over time? What factors might you need to take into account to be able to detect a trend or forecast future conditions?

Are there situations where it would be appropriate to use a linear regression to model a time series? If so, what is an example?

Decomposing a time series dataset

A given time series can be made up of several component series:

1. A **seasonal** component, which repeats over a fixed known period (e.g., seasons of the year, months, days of the week, hour of the day)
2. A **trend** component, which quantifies the upward or downward progression over time. The trend component of a time series does not have to be monotonic.
3. An **error** or **random** component, which makes up the remainder of the time series after other components have been accounted for. This component reflects the noise in the dataset.
4. (optional) A **cyclical** component, which repeats over periods greater than the seasonal component. A good example of this in hydrologic data is El Niño Southern Oscillation (ENSO) cycles, which occur over a period of 2-8 years. Cyclical behavior can be evaluated by spectral analysis.

We first need to turn the discharge data into a time series object in R. This is done using the `ts` function. Notice we can only specify one column of data and need to specify the period at which the data are sampled. The resulting time series object cannot be viewed like a regular data frame.

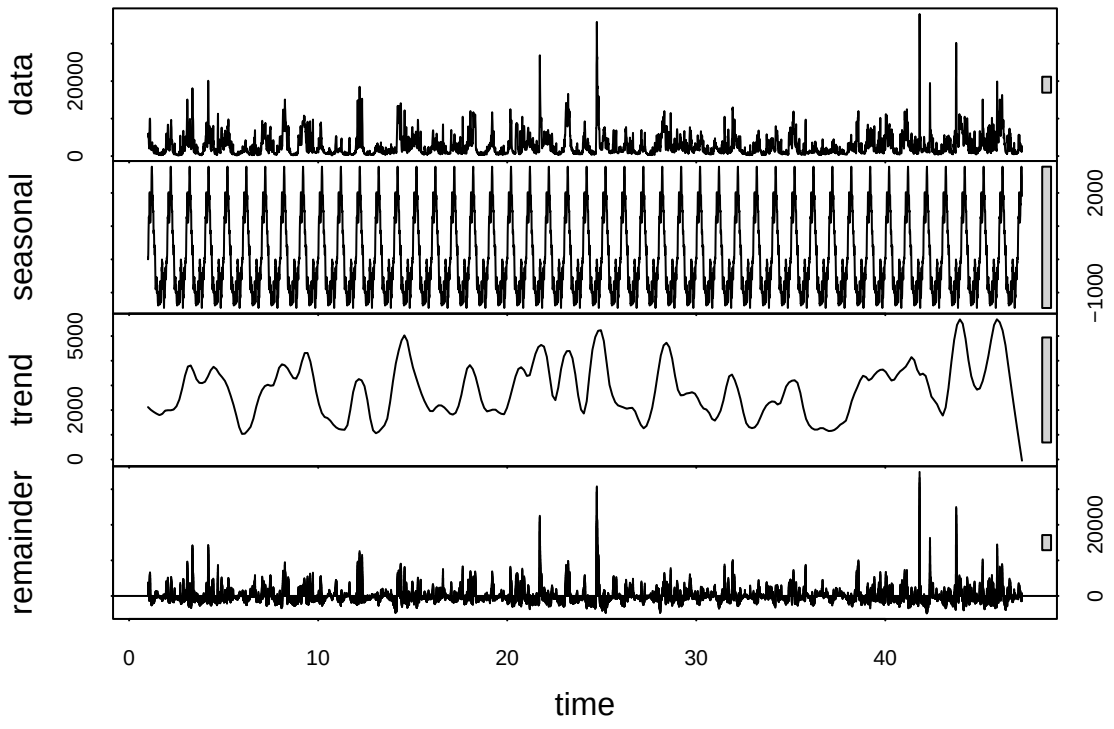
Note: time series objects must be equispaced, which requires interpolation if the data have not been collected at regular intervals. In our case, we have daily data with no NAs in the data frame, so we don't need to worry about this.

```
Neuse_ts <- ts(NeuseFlow[[4]], frequency = 365)
```

The `stl` function decomposes the time series object into its component parts. We must specify that the window for seasonal extraction is either "periodic" or a specific number of at least 7. The decomposition proceeds through a loess (locally estimated scatterplot smoothing) function.

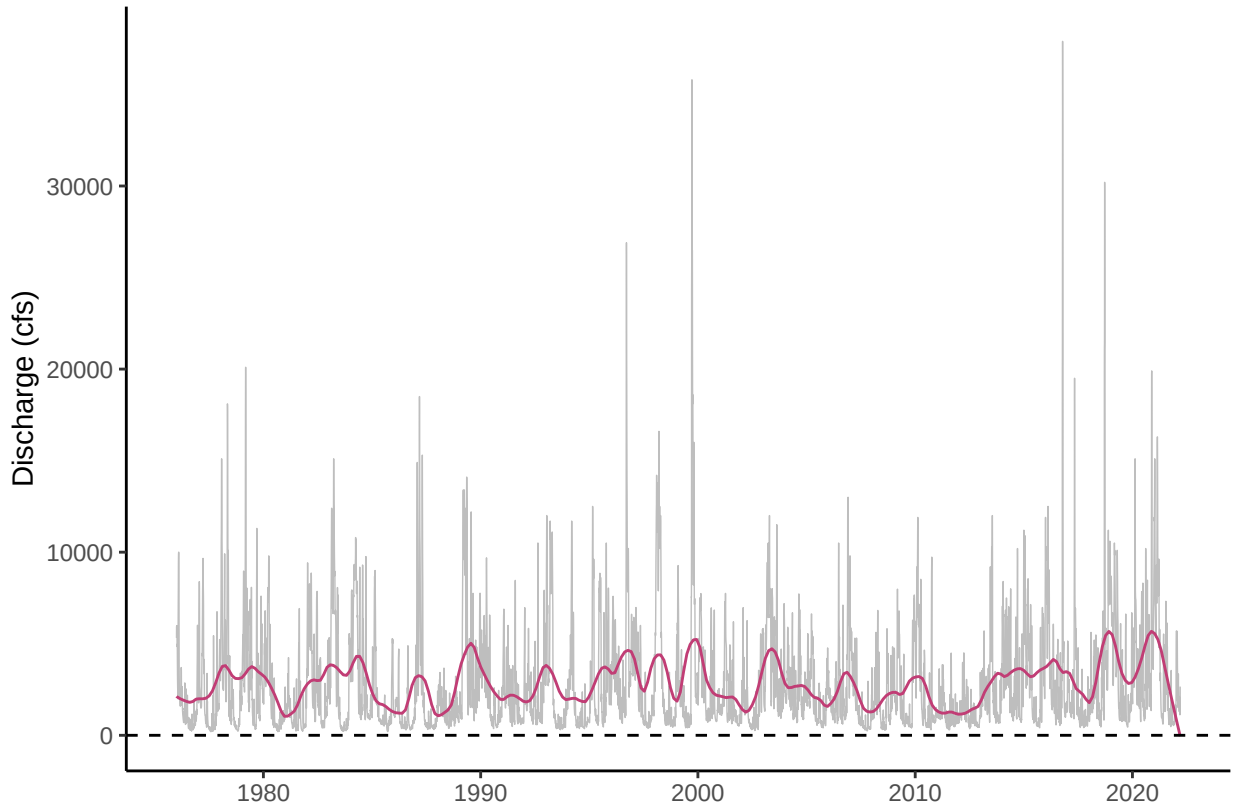
```
?stl
# Generate the decomposition
Neuse_Decomposed <- stl(Neuse_ts, s.window = "periodic")

# Visualize the decomposed series.
plot(Neuse_Decomposed)
```

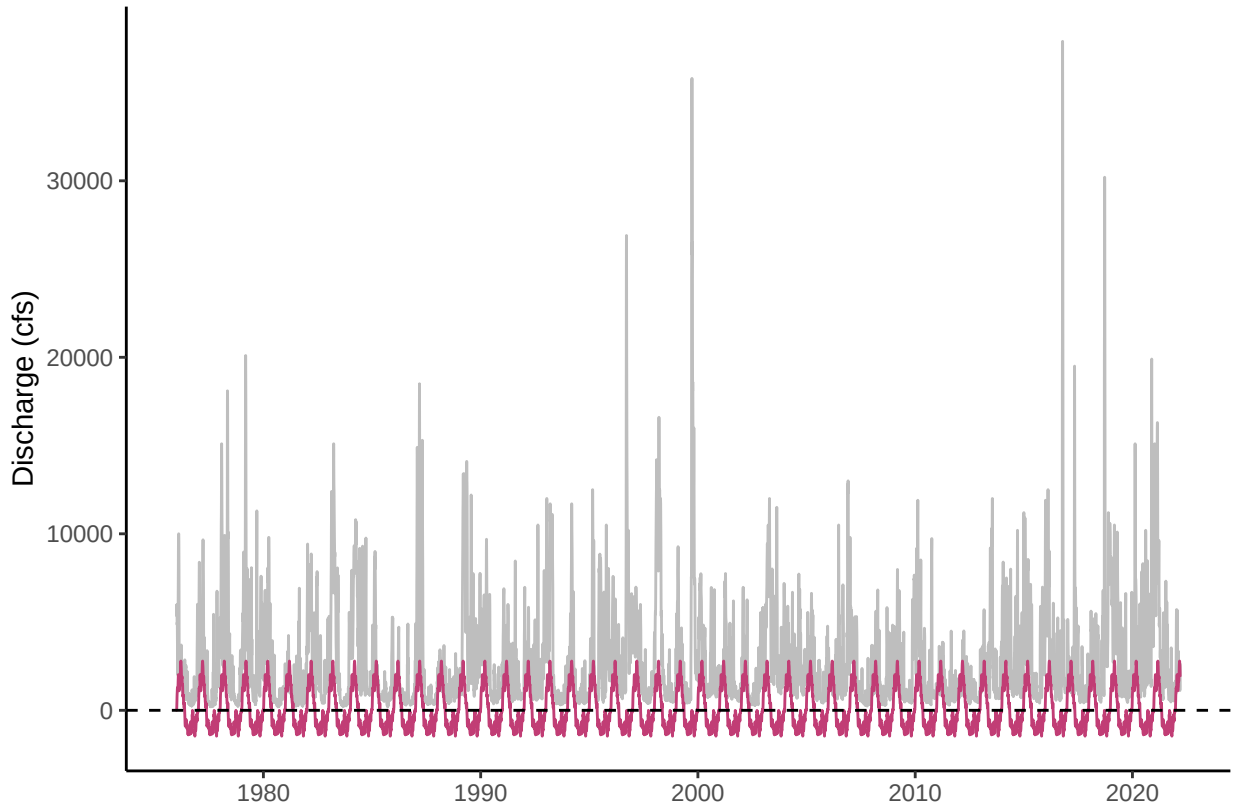



```
# We can extract the components and turn them into data frames
Neuse_Components <- as.data.frame(Neuse_Decomposed$time.series[,1:3])
Neuse_Components <- mutate(Neuse_Components,
                      Observed = NeuseFlow$Discharge,
                      Date = NeuseFlow$Date)

# Visualize how the trend maps onto the data
ggplot(Neuse_Components) +
  geom_line(aes(y = Observed, x = Date), color = "gray", size = 0.25) +
  geom_line(aes(y = trend, x = Date), color = "#c13d75ff") +
  geom_hline(yintercept = 0, lty = 2) +
  labs(x = "", y = "Discharge (cfs)")
```




```
# Visualize how the seasonal cycle maps onto the data
ggplot(Neuse_Components) +
  geom_line(aes(y = Observed, x = Date), color = "gray") +
  geom_line(aes(y = seasonal, x = Date), color = "#c13d75ff") +
  geom_hline(yintercept = 0, lty = 2) +
  labs(x = "", y = "Discharge (cfs)")
```

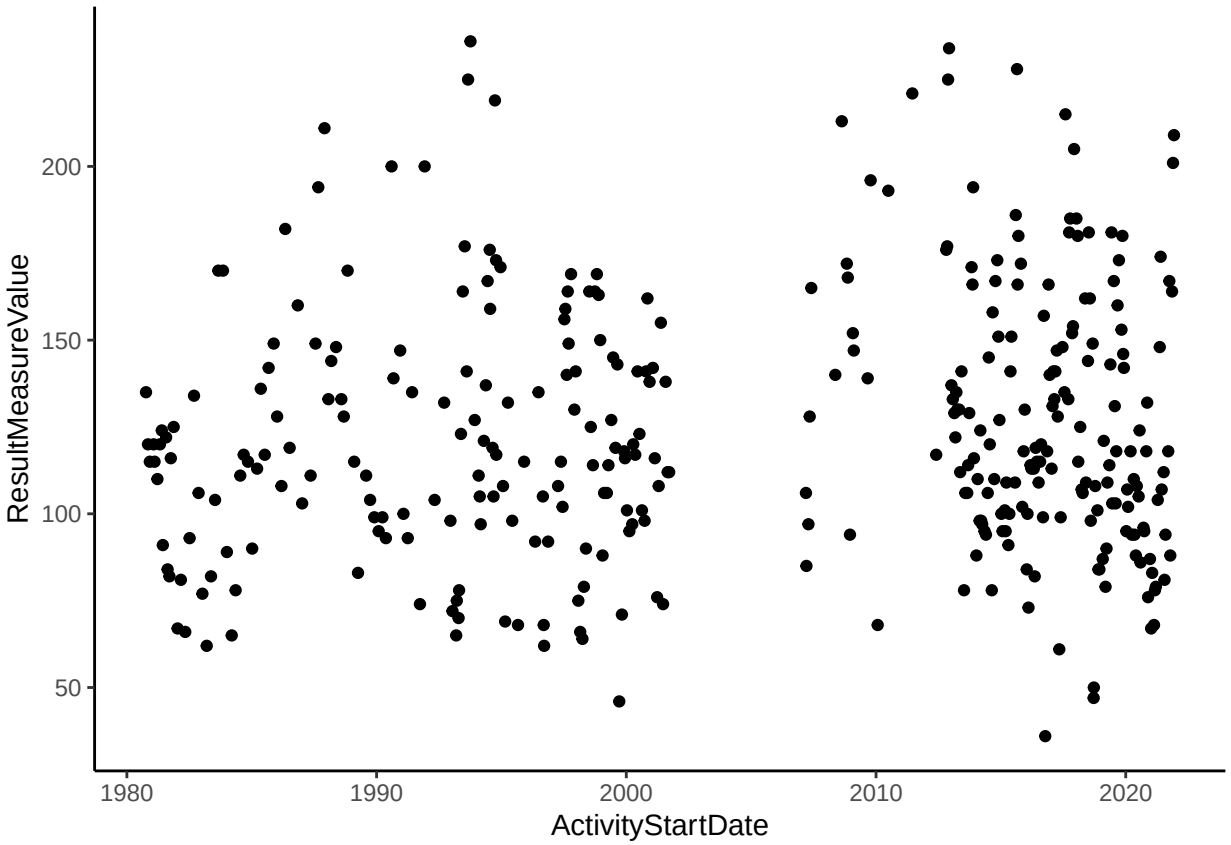


Note that the decomposition can yield negative values when we apply a seasonal adjustment or a trend adjustment to the data. The decomposition is not constrained by a lower bound of zero as discharge is in real life. Make sure to interpret with caution!

Additional application: Conductivity

We talked last time about the phenomenon of salinization in freshwaters and how this has increased over time in many places. Let's look into conductivity in the Neuse to analyze this phenomenon.

```
NeuseCond <- readWQPqw(siteNumbers = "USGS-02089500", # Neuse River at Kinston, NC
                       parameterCd = "90095",  # Specific conductance, uS/cm
                       startDate = "1976-01-01",
                       endDate = "")
# Plot conductivity over time
ggplot(NeuseCond, aes(x = ActivityStartDate, y = ResultMeasureValue)) +
  geom_point()
```



It is important to visualize your time series before moving forward with any test. In this case, we notice a few things:

1. Conductivity is measured approximately monthly, whereas discharge is measured daily.
2. There is a gap in conductivity measurements from October 2001 through February 2007.

Recall that conductivity is negatively correlated with discharge in the Neuse. Before figuring out whether conductivity is increasing over time, we will need to determine how much seasonality impacts this parameter and whether we need to account for that in the trend analysis. Let's decompose the time series of conductivity.

We see that conductivity data were collected approximately monthly across the sampling period. However, most trend tests require identically distributed data. We will therefore interpolate the data to generate monthly values for conductivity

Common interpolation methods:

- **Piecewise constant**:
- **Linear**:
- **Spline**:

Linear interpolation is most common for water quality data, and fits with our understanding about how conductivity might change over time in this system.

```
# create a data frame of months
Months <- data.frame(Date_monthrounded = seq.Date(from = as.Date("1980-10-01"), to = as.Date("2021-12-0

NeuseCond_processed <- NeuseCond %>%
  select(Date = ActivityStartDate,
         Conductivity = ResultMeasureValue) %>%
  mutate(Year = year(Date),
```

```
         Month = month(Date),
         Date_monthrounded = floor_date(Date, "month")) %>%
  arrange(Date)

NeuseCond_monthly <- left_join(Months, NeuseCond_processed)
```

```
## Joining, by = "Date_monthrounded"
```

```
# Generate monthly values from October 1980 to December 2021
linearinterpolation <- as.data.frame(approx(NeuseCond_monthly$Conductivity, n = 566, method = "linear"))
NeuseCond_monthly$Conductivity <- linearinterpolation$y
```

Exercise: decompose the conductivity time series

1. Create a time series of monthly conductivity.
2. Decompose and plot the time series.
3. Analyze the decomposed time series. Is there distinct seasonality? How does the magnitude of seasonality compare to the trend and random components of the time series? What are some caveats that need to be considered for the gap between October 2001 through February 2007?