# Assignment 7: High Frequency Data

*Gaby Garcia*

## OVERVIEW

This exercise accompanies the lessons in Hydrologic Data Analysis on high frequency data

### Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single pdf file.
5. After Knitting, submit the completed exercise (pdf file) to the dropbox in Sakai. Add your last name into the file name (e.g., "A07_Chamberlin.pdf") prior to submission.

The completed exercise is due on 16 October 2019 at 9:00 am.

# Setup

1. Verify your working directory is set to the R project file,
2. Load the StreamPULSE, streamMetabolizer and tidyverse packages.
3. Set your ggplot theme (can be theme_classic or something else)

```
setwd("/Users/gabrielagarcia/Desktop/Hydrologic Data Analysis/Hydrologic_Data_Analysis/Assignments")
packages <- c(
  "tidyverse",
  "StreamPULSE",
  "streamMetabolizer",
  "lubridate",
  "EcoHydRology",
  "xts")
invisible(
  suppressPackageStartupMessages(
    lapply(packages, library, character.only = TRUE)
    )
  )

gabytheme <- theme_bw(base_size = 22) +
  theme(plot.title=element_text(face="bold", size="29", color="IndianRed3", hjust=0.5),
        axis.title=element_text(size=22, color="black"),
axis.text = element_text(face="bold", size=18, color = "black"),
panel.background=element_rect(fill="white", color="darkblue"),
panel.border = element_rect(color = "black", size = 2),
legend.position = "top", legend.background = element_rect(fill="white", color="black"),
            legend.key = element_rect(fill="transparent", color="NA"))

theme_set(gabytheme)
```

4. Download data from the Stream Pulse portal using `request_data()` for the Kansas River, ("KS_KANSASR"). Download the discharge (`Discharge_m3s`), dissolved oxygen (`DO_mgL`) and nitrate data (`Nitrate_mgL`) for the entire period of record

### Pull in Kansas Data from StreamPulse

```r
KansasRiverData<- request_data(
  sitecode = "KS_KANSASR",
  variables = c('Discharge_m3s', 'DO_mgL', 'Nitrate_mgL'),
  startdate="2018-02-01",
  enddate="2018-05-31"
  )
```

```
##
## API call: https://data.streampulse.org/api?sitecode=KS_KANSASR&startdate=2018-02-01&enddate=2018-05-3
##
## Retrieved the following variables:
##   DO_mgL, Discharge_m3s, Nitrate_mgL
```

5. Reformat the data into one dataframe with columns DateTime_UTC, DateTime_Solar (using `convert_UTC_to_solartime()`), SiteName, DO_mgL, Discharge_m3s, and Nitrate_mgL.

#### Create Longitude Value and Index the first data frame

```r
Kansas.lon <- KansasRiverData[[2]]$lon

Kansas.DO<-KansasRiverData[[1]]
class(Kansas.DO$DateTime_UTC)
```

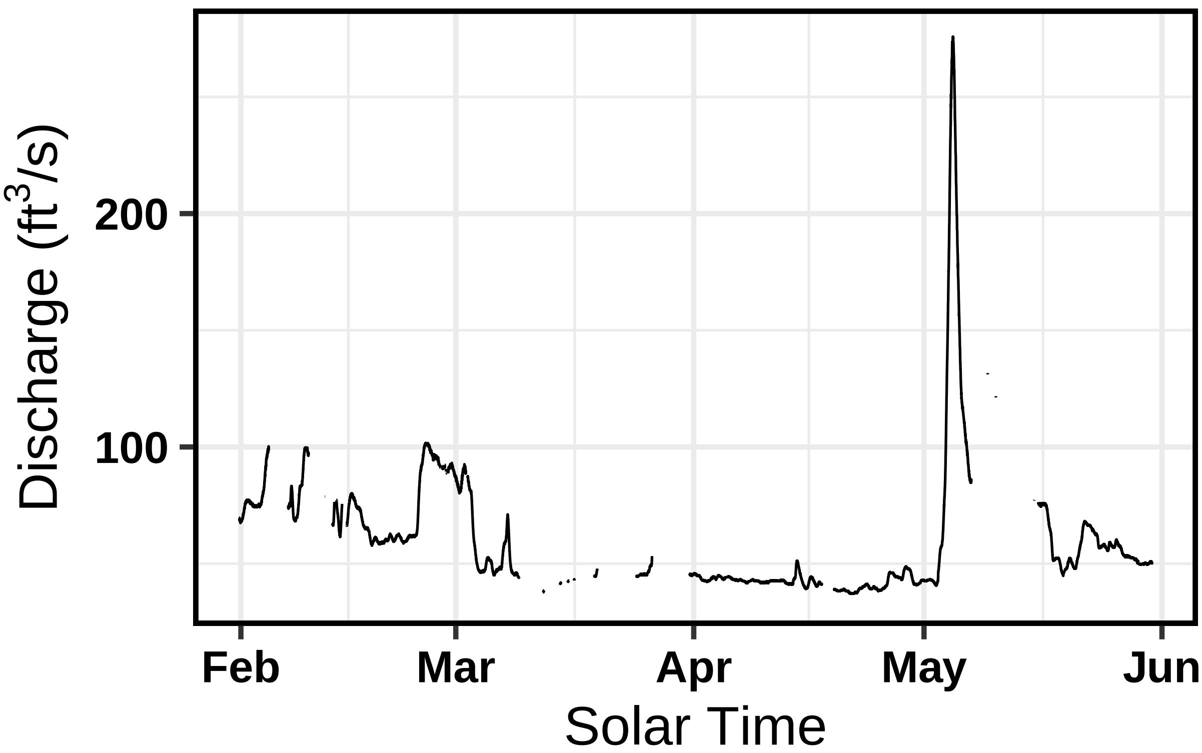
```
## [1] "POSIXct" "POSIXt"
```

```r
Kansas.DOFinal<-pivot_wider(Kansas.DO,
                            values_from=value,
                            names_from =variable)

Kansas.DOFinal<- mutate(Kansas.DOFinal,
                        DateTime_Solar=convert_UTC_to_solartime(DateTime_UTC, Kansas.lon))

   Kansas.DOFinal<-select(Kansas.DOFinal,
                          DateTime_UTC, DateTime_Solar,
                          site, DO_mgL,
                          Discharge_m3s,
                          Nitrate_mgL)
```

6. Plot each of the 3 variables against solar time for the period of record

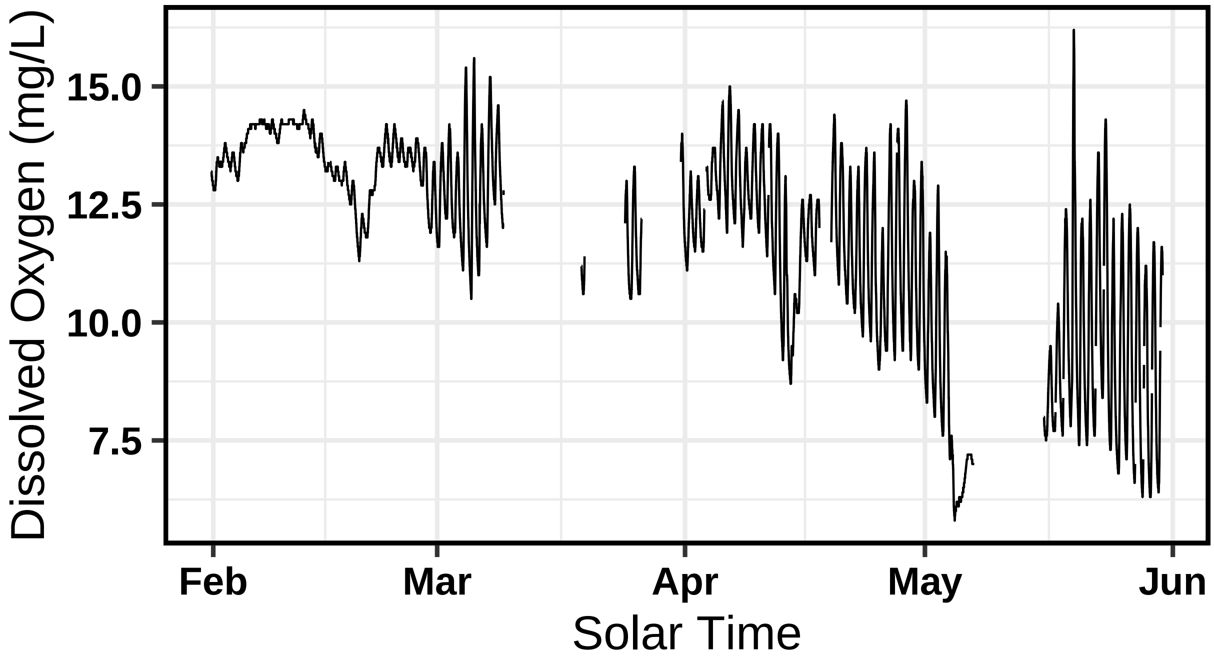
## Plot of Solar Time vs. River Discharge

```r
library(ggplot2)
ggplot(Kansas.DOFinal, aes(x = DateTime_Solar, y = Discharge_m3s)) + geom_line()+
  labs(x="Solar Time")+
    ylab(expression("Discharge (ft"^3*"/s)"))+
  gabytheme
```




### Plot of Solar Time vs. Dissolved Oxygen Concentrations

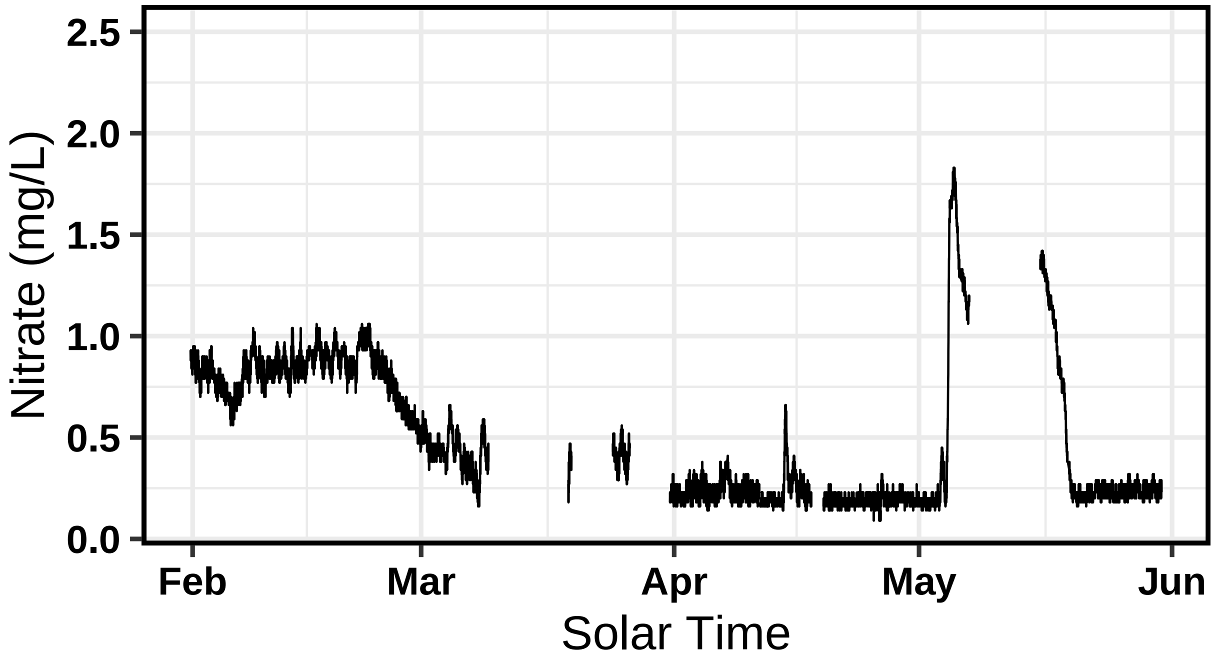
```
library(ggplot2)
ggplot(Kansas.DOFinal, aes(x = DateTime_Solar, y = DO_mgL)) +
  geom_line()+
  labs(x="Solar Time", y="Dissolved Oxygen (mg/L)")+
  gabytheme
```




>Notice that as river discharge spikes in early May, dissolved oxygen decreases dramatically while Nitrate increases dramatically.

## Plot of Solar Time vs. Nitrate Concentrations

```
library(ggplot2)
ggplot(Kansas.DOFinal, aes(x = DateTime_Solar, y = Nitrate_mgL)) +
  geom_line(color="black")+
  labs(x="Solar Time", y="Nitrate (mg/L)")+
  gabytheme
```




7. How will you address gaps in these dataseries?

   I will interpolate the data gaps using the approx() function. The arguments we specified in the function were "n" and "method". Interpolation with the approx() function takes place at n equally spaced points spanning the interval [min(x), max(x)]). I included my later calculated timestep for "n". Method specifies the interpolation method to be used-choices are "linear" or "constant", and we chose linear.

8. How does the daily amplitude of oxygen concentration swings change over the season? What might cause this? >It appears that as the year progressed from February to June, the range of dissolved oxygen concentrations increased in magnitude. As the season progresses to spring/summer and solar radiation increases, more biological activity (specifically plant growth) occurs, thus leading to photosynthesis and oxygen production during the day, and respiration at night.

## Baseflow separation

9. Use the `EcoHydRology::BaseflowSeparation()` function to partition discharge into baseflow and quickflow, and calculate how much water was exported as baseflow and quickflow for this time period. Use the DateTime_UTC column as your timestamps in this analysis.

The `package::function()` notation being asked here is a way to call a function without loading the library. Sometimes the EcoHydRology package can mask tidyverse functions like pipes, which will cause problems for knitting. In your script, instead of just typing `BaseflowSeparation()`, you will need to include the package and two colons as well.

**Select for DateTime_UTC and Discharge Columns to make new dataframe**

```
KansasDateDischarge<-select(Kansas.DOFinal, DateTime_UTC, Discharge_m3s)
```

### Determine number of timesteps (days * 24 hours * 4 15-minute intervals)

```
timestepnumber<-(28+31+30+31)*24*4
table(diff(KansasDateDischarge$DateTime_UTC))
```

```
##
##      0   900  1800  4500 15300
##   2788 11291     1     2     7
```

### Interpolate Discharge Values

```
DischargeInterpolation<-as.data.frame(approx(KansasDateDischarge, n=timestepnumber, method="linear"))
```

### Convert Datetime_UTC to Regular DateTime

```
library(lubridate)
DischargeInterpolation$x<-as.POSIXct(DischargeInterpolation$x, origin="1970-01-01")
names(DischargeInterpolation)<-c("Date", "Discharge")
```

## Baseflow Separation Function

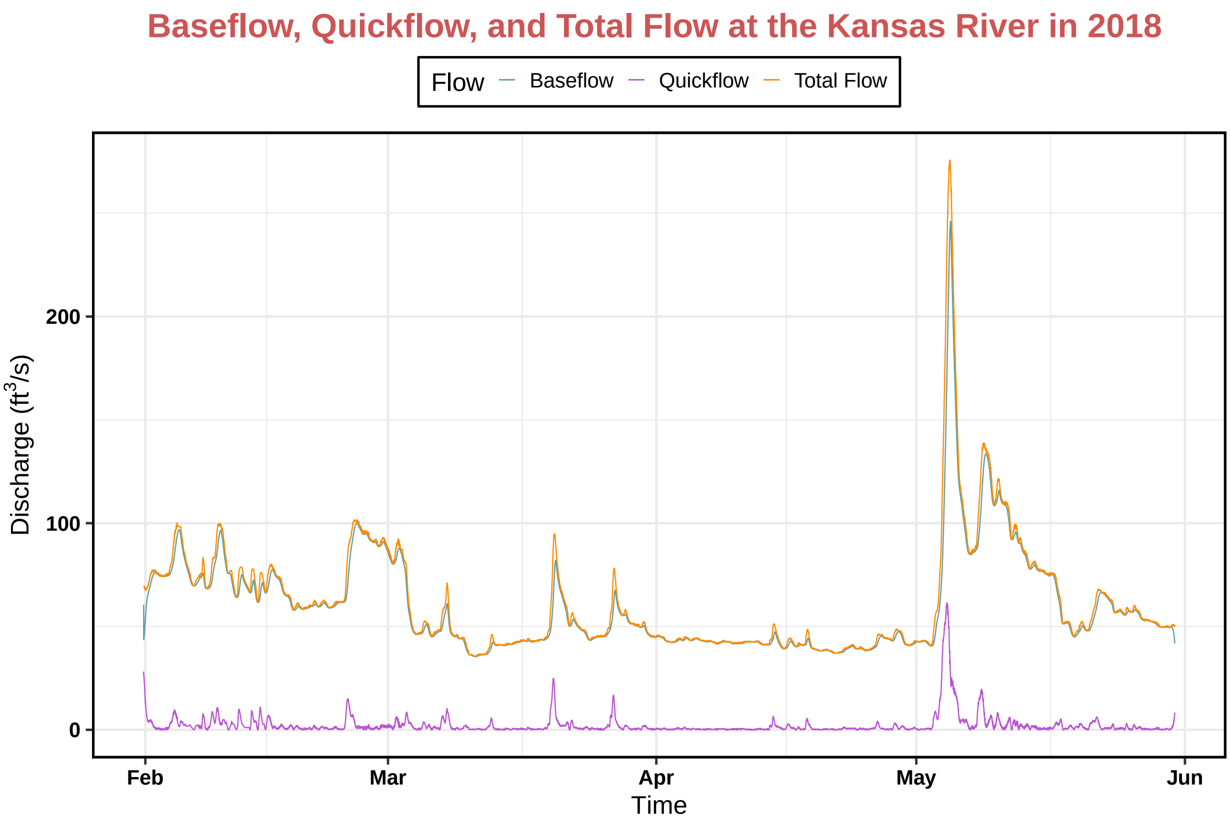
```
library(EcoHydRology)
KansasBaseflow <- EcoHydRology::BaseflowSeparation(
DischargeInterpolation$Discharge,
  filter_parameter = 0.925,
  passes = 3)
```

### Bind Kansas Data Frame to Kansas Baseflow/Quickflow dataframe

```
Kansas2018 <- cbind(DischargeInterpolation, KansasBaseflow)
```

10. Create a ggplot showing total flow, baseflow, and quickflow together.

```
library(ggplot2)
ggplot(data=Kansas2018, aes(x=Date)) +
  geom_line(mapping= aes(y = bt, color = "Baseflow")) +
  geom_line(mapping = aes(y = qft, color = "Quickflow"))+
   geom_line(mapping = aes( y = Discharge, color = "Total Flow"))+
   labs(title="Baseflow, Quickflow, and Total Flow at the Kansas River in 2018 ", x="Time")+
  ylab(expression("Discharge (ft"^3*"/s)"))+
  scale_colour_manual(values=c('#5F9EA0', '#BA55D3','#FF8C00'))+
guides(color = guide_legend(title = "Flow"))+
  gabytheme
```




11. What percentage of total water exported left as baseflow and quickflow from the Kansas River over this time period?

    96.2 of the total flow from the watershed exited as baseflow. 3.76% of the total flow from the watershed exited as quickflow.

```
library(dplyr)
ExportKansas <-mutate(Kansas2018,
                      timestep = c(diff(as.numeric(Date)), NA_real_),
                      baseflowexport = bt * timestep,
         quickflowexport = qft * timestep)


#ft^3/sec * seconds elapsed to just get the volume
 ExportKansas <-summarize(ExportKansas, BaseflowExport_cf = sum(baseflowexport, na.rm = T),
            QuickflowExport_cf = sum(quickflowexport, na.rm = T),
            TotalExport_cf = BaseflowExport_cf + QuickflowExport_cf)
```

### Baseflow Amount

```
ExportKansas$Baseflow/ExportKansas$TotalExport_cf*100
```

```
## [1] 96.24424
```

96.2%

#### Quickflow Amount

```
ExportKansas$QuickflowExport_cf/ExportKansas$TotalExport_cf*100
```

```
## [1] 3.755755
```

3.76%

12. This is a much larger river and watershed than the 2 we investigated in class. How does the size of the watershed impact how flow is partitioned into quickflow and baseflow?

    Baseflow is a proxy for groundwater discharge to rivers. The Kansas River at Desoto is a big river within a large watershed-large fast-flowing rivers require a sufficient amount of baseflow, and it would unrealistic for a large river such as this one to have quickflow as a significant proportion of its total flow-this would require significant storms in the area creating a large volume of discharge to percolate through the soil to feed baseflow. .

13. The site we are looking at is also further down in its river network (i.e. instead of being a headwater stream, this river has multiple tributaries that flow into it). How does this impact your interpretation of your results? >The discharge at a location further down the river network would not solely rely on groundwater inputs to sustain its baseflow. Precipitation events at headwater streams would also be input to baseflow.

### Chemical Hysteresis

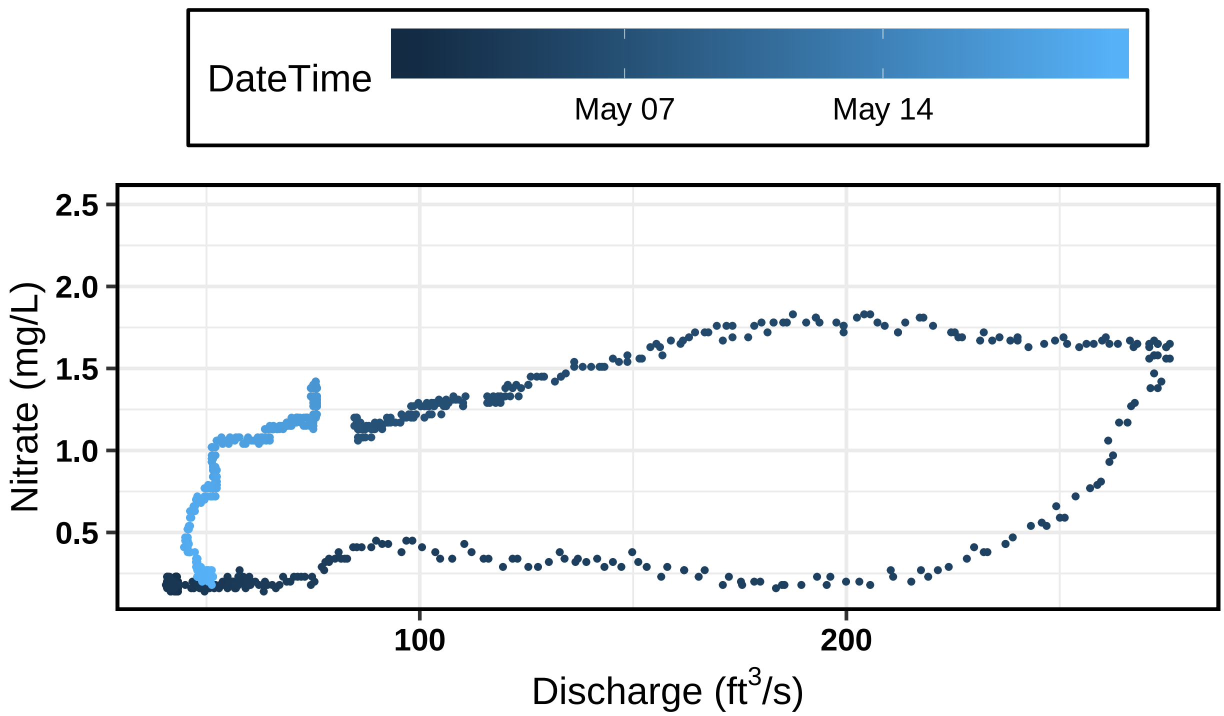
14. Create a ggplot of flow vs. nitrate for the large storm in May (~May 1 - May 20). Use color to represent Date and Time.

### Filter Kansas River data for a specific storm event

```
KansasStorm <-filter(Kansas.DOFinal,DateTime_UTC > "2018-05-01" & DateTime_UTC < "2018-05-20")
```

# Plot Chemical Hysteresis

```
ggplot(KansasStorm, aes(x = Discharge_m3s, y = Nitrate_mgL, color = DateTime_UTC)) +
  geom_point() +
   theme(legend.key.height = unit(1, "cm"),
        legend.key.width = unit(3, "cm"))+
 labs(color="DateTime", y="Nitrate (mg/L)")+
  xlab(expression("Discharge (ft"^3*"/s)"))
```




15. Does this storm show clockwise or counterclockwise hysteresis? Was this storm a flushing or diluting storm?

    Because concentrations increases as flow increases, this storm would show counterclockwise hysteresis, so this is a flushing storm.

16. What does this mean for how nitrate gets into the river from the watershed?

    Nitrate concentrations will be highest at the beginning of storm events when quickflow is a higher proportion of the total flow-nitrate originates from quickflow. This is because nitrate is often produced from agriculture and enters the watershed through overland flow.

## Reflection

17. What are 2-3 conclusions or summary points about high frequency data you learned through your analysis?

    Larger watersheds are primarily fed through baseflow, not quickflow, and have more baseflow than quickflow as a proportion of total discharge. I learned about hysterisis loops and how nitrate is a flushing nutrient; each discharge value corresponded to two nitrate concentration values.

18. What data, visualizations, and/or models supported your conclusions from 17?

    The ggplot displaying the proportions of baseflow and quickflow to total flow helped me understand my first conclusion point; the ggplot displaying my hysterisis plot helped me understand my second conclusion point.

19. Did hands-on data analysis impact your learning about high frequency data relative to a theory-based lesson? If so, how?

    Yes, visualization always helps me to see the big picture ideas of a theory, rather than getting lost in the weeds.

20. How did the real-world data compare with your expectations from theory?

I expected that quickflow and extreme storm events would be the primary input to major watersheds, so this lesson surprised me.