Assignment 7: High Frequency Data

Haoyu Zhang

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)

```
getwd()
```

```
## [1] "D:/William/Duke/Study/EOS 722/Hydrologic_Data_Analysis/Assignments/Assignment 7"
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

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

```
library(streamMetabolizer)
```

```
## USGS Active Research Package:
## https://owi.usgs.gov/R/packages.html#research

## This package is in development. We are using it for our own
## applications and welcome flexible, resilient users who can help us
## make the package better. Details of the user interface and model
## implementations will change. Please give us feedback at
## https://github.com/USGS-R/streamMetabolizer/issues/new.

## Can't check GitHub for new package versions just now. We'll try again next time.
```

```
library(devtools)
```

```
## Loading required package: usethis
```

```r
library(StreamPULSE)
```

```
## Loading required package: shiny

## Loading required package: Cairo
```

```r
theme_set(theme_classic())
```

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

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.

```r
kansas.raw <- request_data(
  sitecode = "KS_KANSASR",
  variables = c("Discharge_m3s", "DO_mgL", "Nitrate_mgL"))
```

```
## You may omit the "variables" parameter to automatically retrieve
##  all variables necessary for metabolism modeling.

##
## API call: https://data.streampulse.org/api?sitecode=KS_KANSASR&variables=Discharge_m3s,DO_mgL,Nitrate
##
## Retrieved the following variables:
##   DO_mgL, Discharge_m3s, Nitrate_mgL
```

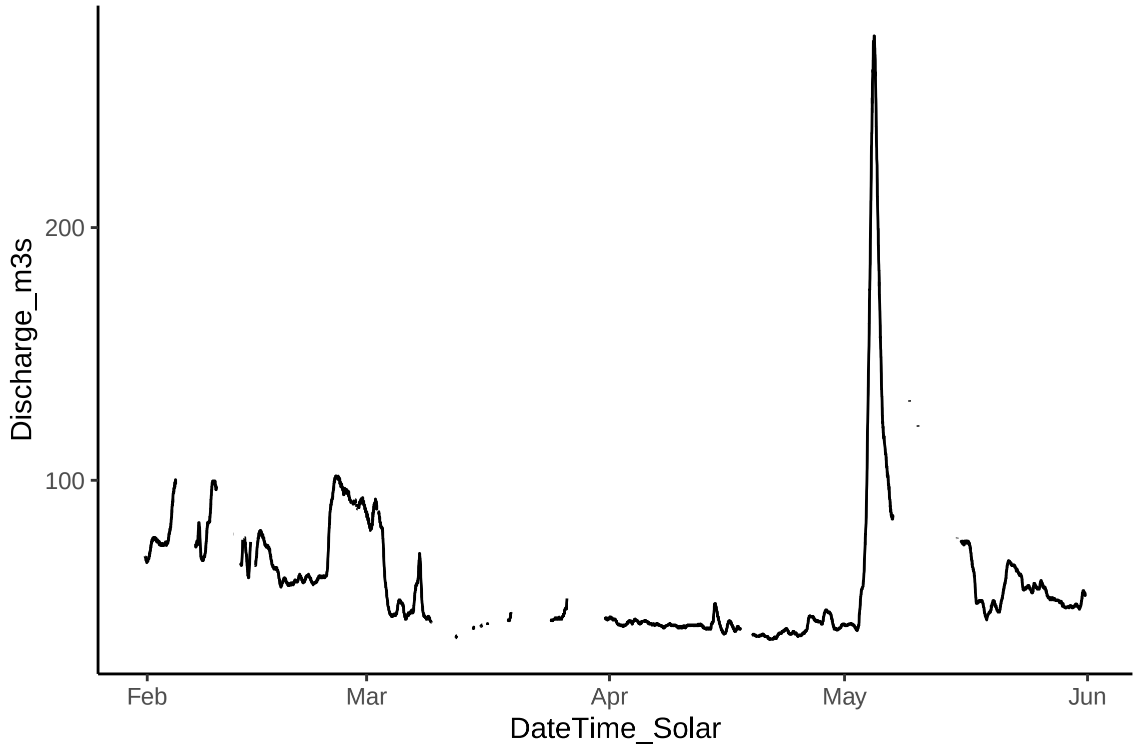
```r
kansas.lon <- kansas.raw[["sites"]]$lon

kansas.dat <- kansas.raw[["data"]] %>%
  pivot_wider(names_from = variable, values_from = value)%>%
  mutate(DateTime_Solar = convert_UTC_to_solartime(DateTime_UTC, kansas.lon))%>%
  select(DateTime_UTC, DateTime_Solar, Discharge_m3s, DO_mgL, Nitrate_mgL)
head(kansas.dat, 5)
```

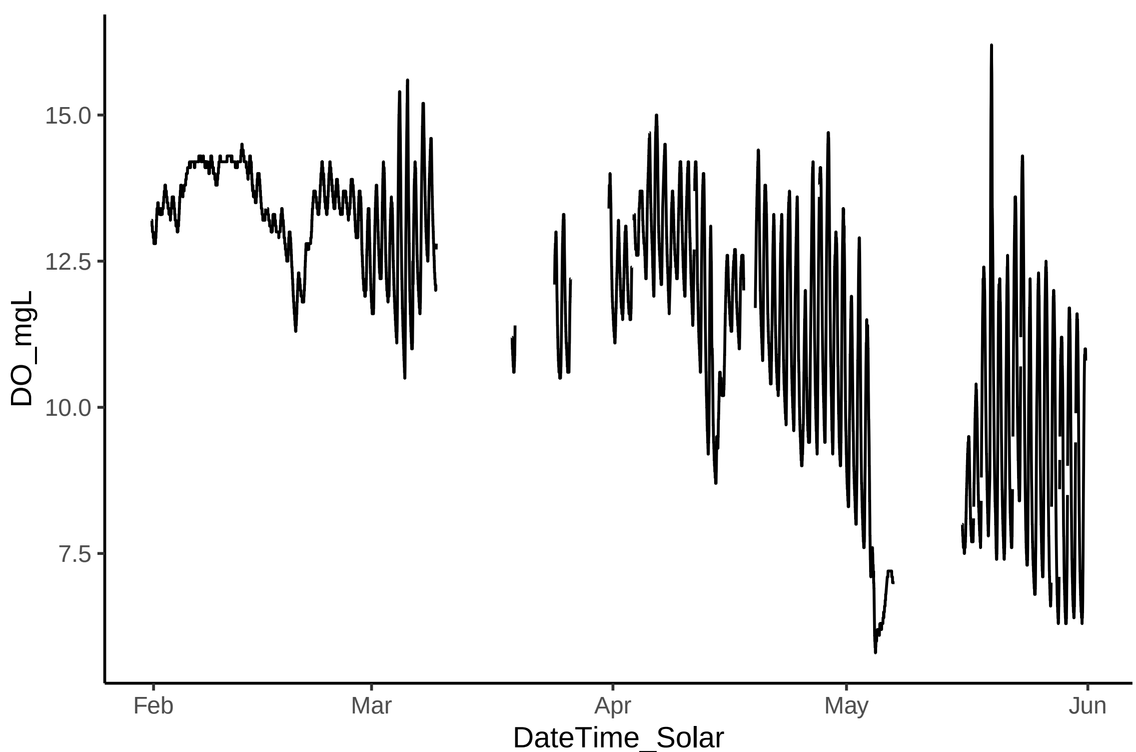
```
## # A tibble: 5 x 5
##   DateTime_UTC        DateTime_Solar      Discharge_m3s DO_mgL Nitrate_mgL
##   <dttm>              <dttm>                      <dbl>  <dbl>       <dbl>
## 1 2018-02-01 00:00:00 2018-01-31 17:27:42          69.9   13.2        0.9
## 2 2018-02-01 00:15:00 2018-01-31 17:42:41          69.7   13.2        0.93
## 3 2018-02-01 00:30:00 2018-01-31 17:57:41          69.7   13.2        0.93
## 4 2018-02-01 00:45:00 2018-01-31 18:12:41          69.7   13.2        0.9
## 5 2018-02-01 01:00:00 2018-01-31 18:27:41          69.7   13.2        0.88
```

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

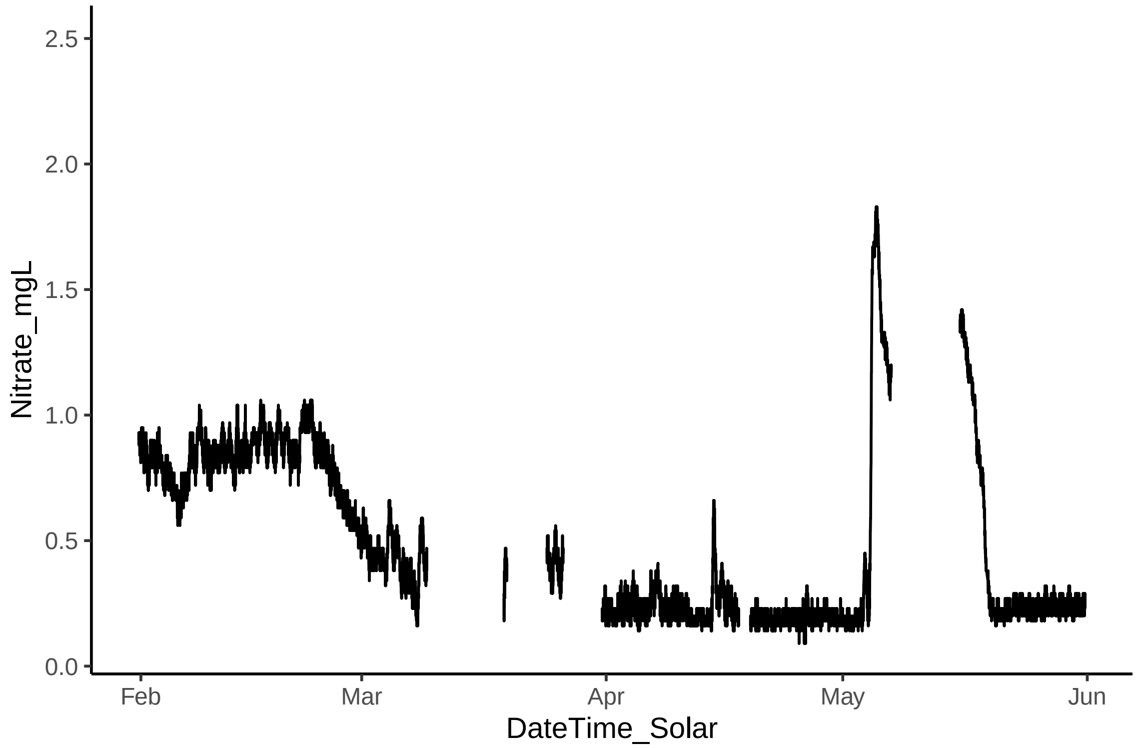
```r
ggplot(kansas.dat, aes(x = DateTime_Solar, y = Discharge_m3s)) + geom_line()
```




```
ggplot(kansas.dat, aes(x = DateTime_Solar, y = DO_mgL)) + geom_line()
```




```
ggplot(kansas.dat, aes(x = DateTime_Solar, y = Nitrate_mgL)) + geom_line()
```



7. How will you address gaps in these dataseries?

   Since there are large variations in dissolved oxygen and nitrate concentration, it may be inappropriate to use interpolation to replace the missing values during mid-March and mid-May, which are the two major gaps during the whole period. Thus, we might need to remove the dates on which measurements were not taken for analyzing relationships between discharge and DO or nitrate concentration.

8. How does the daily amplitude of oxygen concentration swings change over the season? What might cause this?

   The magnitude of daily variation continuously increases from Mar to Jun. During the summer, photosynthetic organisms are abundant and perform photosynthesis during the daytime, increasing DO in the river, whereas during the night oxygen is quickly depleted due to the presence of much more organisms and thus higher respiration rate than the winter.

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.

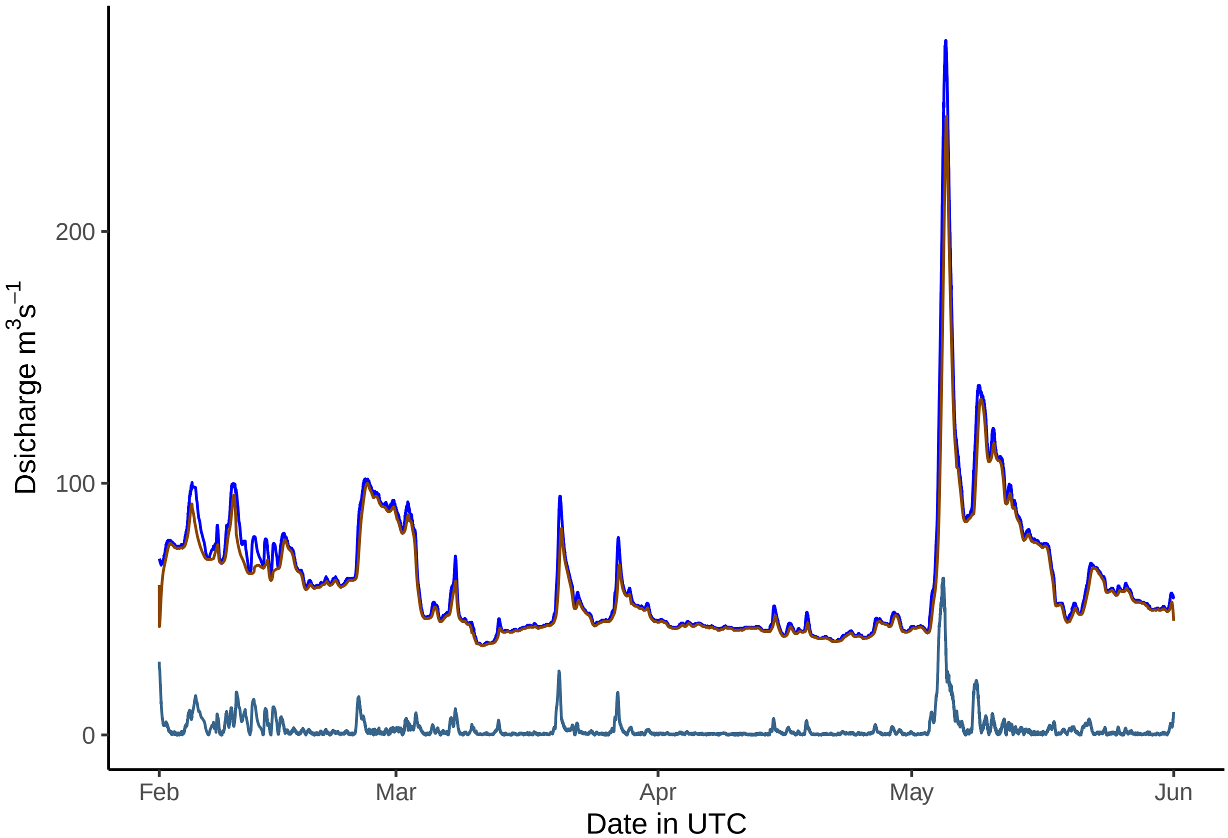
The `package::function()` notation being asked here is a way to call a function without loading the library. Sometimes the EcoHydRology package can mask tidyverse functions like pipes, which will cause problems for knitting. In your script, instead of just typing `BaseflowSeparation()`, you will need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

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

```
KStotalflow <- drop_na(kansas.dat, Discharge_m3s)
KSbaseflow <- EcoHydRology::BaseflowSeparation(
  KStotalflow$Discharge_m3s, filter_parameter = 0.925, passes = 3)
```

```r
names(KSbaseflow) <- c("Baseflow", "Quickflow")
KS2018 <- cbind(KStotalflow, KSbaseflow)

ggplot(data = KS2018, aes(x = DateTime_UTC))+
  geom_line(aes(y = Discharge_m3s), color = "blue") +
  geom_line(aes(y = Baseflow), color = "darkorange4") +
  geom_line(aes(y = Quickflow), color = "steelblue4") +
  labs(x = "Date in UTC", y = expression("Dsicharge "*m^3*s^-1))
```




```r
# Calculate percentages of Qf vs Bf
ExportKS <- KS2018 %>%
  mutate(timestep = c(diff(as.numeric(DateTime_UTC)), NA_real_),
         baseflowexport = Baseflow * timestep,
         quickflowexport = Quickflow * timestep) %>%
  summarise(BaseflowExport_m3 = sum(baseflowexport, na.rm = T),
            QuickflowExport_m3 = sum(quickflowexport, na.rm = T),
            TotalExport_m3 = BaseflowExport_m3 + QuickflowExport_m3)

ExportKS$BaseflowExport_m3 / ExportKS$TotalExport_m3
```

```
## [1] 0.9568351
```

```r
ExportKS$QuickflowExport_m3 / ExportKS$TotalExport_m3
```

```
## [1] 0.04316493
```

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

time period?

95.7% of the total flow during this period was accounted for by baseflow, while only 4.3% by quickflow.

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?

    In larger rivers, the relative amount of baseflow compared to quickflow induced by precipitation is higher than small rivers. Thus, in Kansas River baseflow is the the predominant source for discharge.

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?

    Since Kansas River is downstream to many other tributaries, a considerable proportion of its discharge is not determined by groundwater flow and overground flow in its own watershed. We therefore may not expect substantial impacts of quickflow that is caused by precipitation on its own watershed on the total flow in the river.

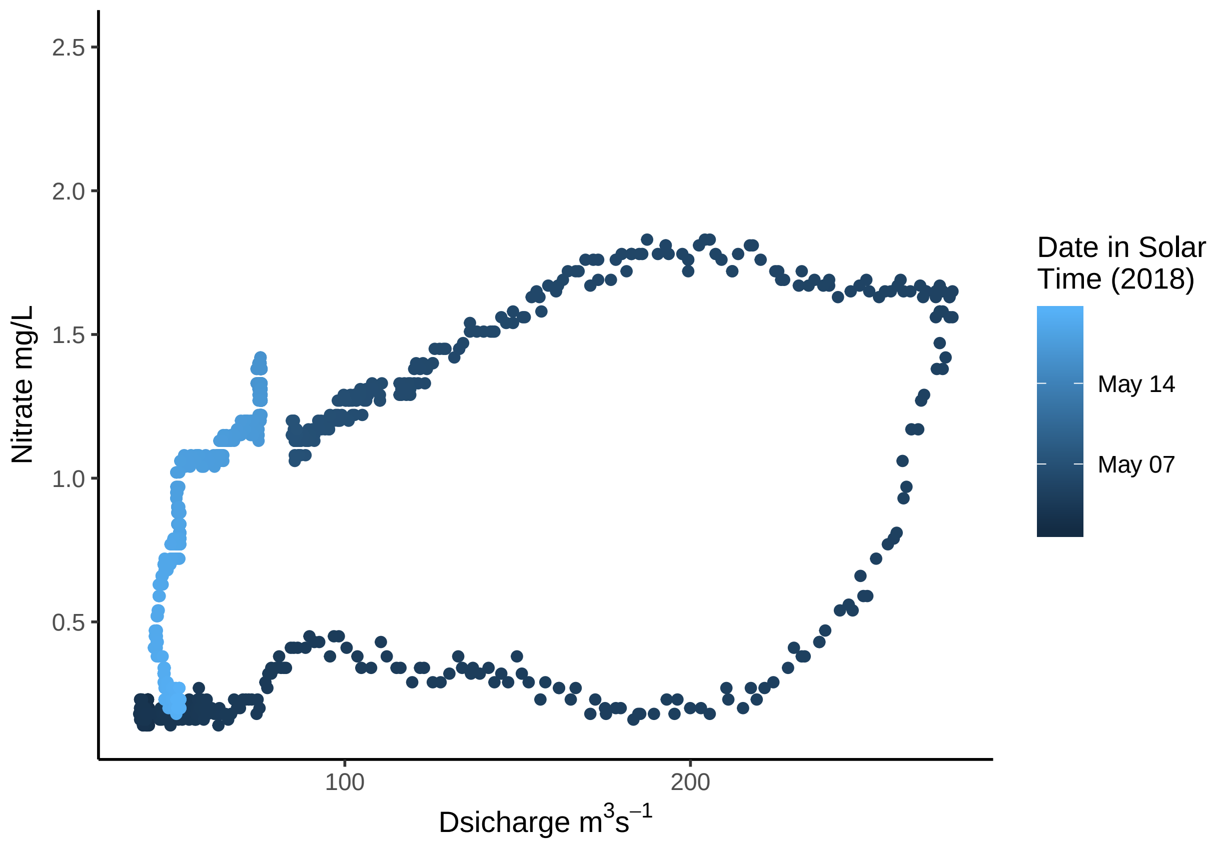
Chemical Hysteresis

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.

```
KS2018May <- kansas.dat %>%
  filter( DateTime_Solar > "2018-05-01" & DateTime_Solar < "2018-05-20")

ggplot(KS2018May, aes(x = Discharge_m3s, y = Nitrate_mgL, colour = DateTime_Solar)) +
  geom_point() +
  labs(x = expression("Dsicharge "*m^3*s^-1), y = expression("Nitrate mg/L"),
       color = "Date in Solar \nTime (2018)")
```

```
## Warning: Removed 1638 rows containing missing values (geom_point).
```



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

    This storm even shows counterclockwise hysteresis. The storm is a flushing one, because the slope is positive.

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

    A counterclockwise hysteresis indicates that the quickflow mostly occuring in the rising limb has lower nitrate concentration than the baseflow dominating the falling limb. It is likely that nitrate enters the river primariy through groundwater flow or from upstream tributaries, both of which constitute the baseflow of Kansas River.

Reflection

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

    We should be careful when handling missing values/gaps in high frequency dataset, as the data could have very large variations within a day, constraining our ability to predict the missing values with interpolations. We can take advantage of the data to perform baseflow separation to gain knowledge on the relative contribution of baseflow versus quickflow in a river. Moreover, high-frequency data can be used to analyze the hysteresis type of a river and allow us to make inference on the main pathway for nutrients of interest to enter the water body.

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

    Baseflow-Quickflow partition can be accomplished by `EcoHydRology::BaseflowSeparation()` and visualized by plotting baseflow, quickflow, and total flow over the same time period on one diagram.

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

    Yes. Some real-world data help understand the concept of hysteresis as well as how results (plots) we may get when we are actually analyzing a river rather than the theoretical patterns.

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

    Generally the hysteresis plots constructed based on real-world data are consistent with the theoretical ones.