Assignment 7: High Frequency Data

Jake Greif

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] "/Users/jakegreif/Duke/Fall_2019/Hydrologic_Data_Analysis"
```

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------- tidyverse 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_conflic
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(StreamPULSE)
```

```
## Loading required package: shiny

## Loading required package: Cairo
```

```
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.
```

```r
library(EcoHydRology)
```

```
## Loading required package: operators

##
## Attaching package: 'operators'

## The following object is masked from 'package:forcats':
##
##     %>%

## The following object is masked from 'package:stringr':
##
##     %>%

## The following object is masked from 'package:dplyr':
##
##     %>%

## The following object is masked from 'package:purrr':
##
##     %>%

## The following object is masked from 'package:tidyr':
##
##     %>%

## The following objects are masked from 'package:base':
##
##     options, strrep

## Loading required package: topmodel

## Loading required package: DEoptim

## Loading required package: parallel

##
## DEoptim package
## Differential Evolution algorithm in R
## Authors: D. Ardia, K. Mullen, B. Peterson and J. Ulrich

## Loading required package: XML
```

```r
library(xts)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```

```
##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```r
library(dygraphs)
```

```
## This version of Shiny is designed to work with 'htmlwidgets' >= 1.5.
##     Please upgrade via install.packages('htmlwidgets').

##
## Attaching package: 'dygraphs'

## The following object is masked from 'package:operators':
##
##     %>%
```

```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.

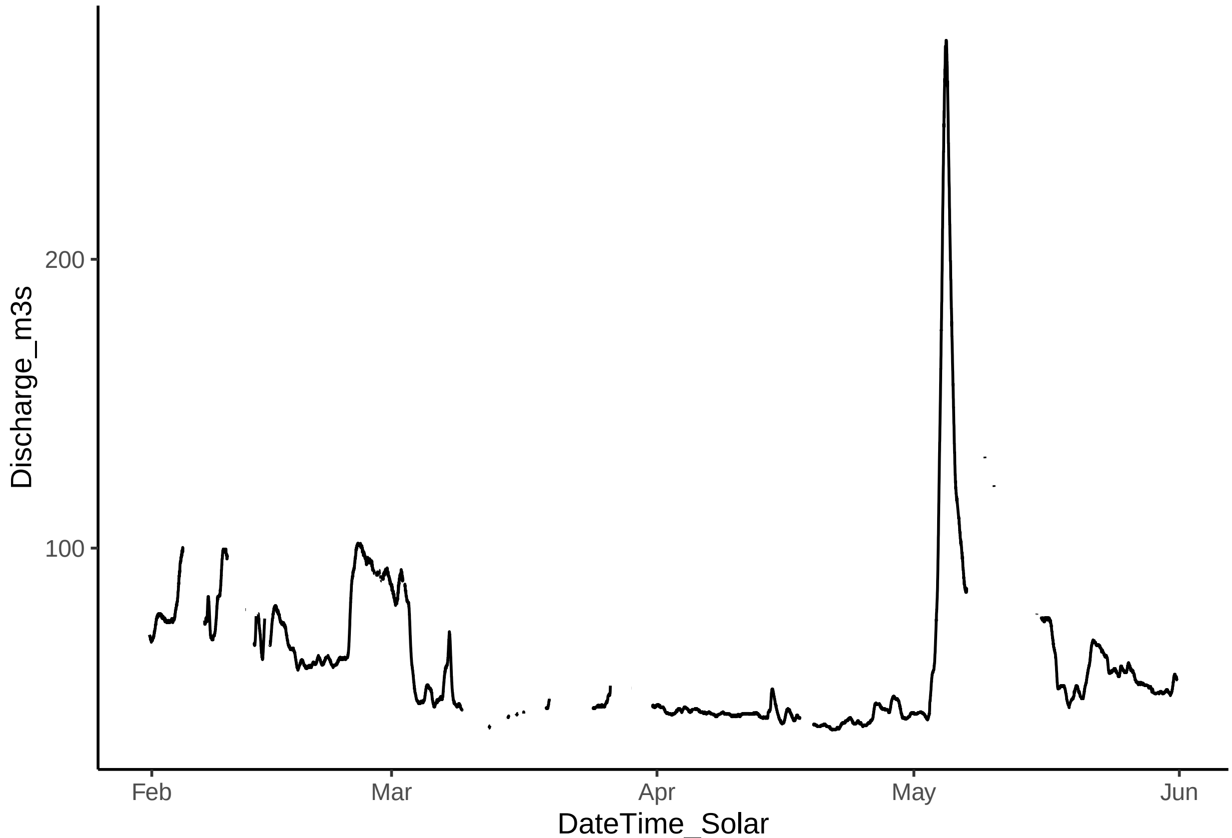
```r
KSdat <- request_data(
  sitecode = "KS_KANSASR",
  variables = c('DO_mgL', 'Discharge_m3s', 'Nitrate_mgL'))
```

```
## 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=DO_mgL,Discharge_m3s,Nitrat
##
## Retrieved the following variables:
##   DO_mgL, Discharge_m3s, Nitrate_mgL
```

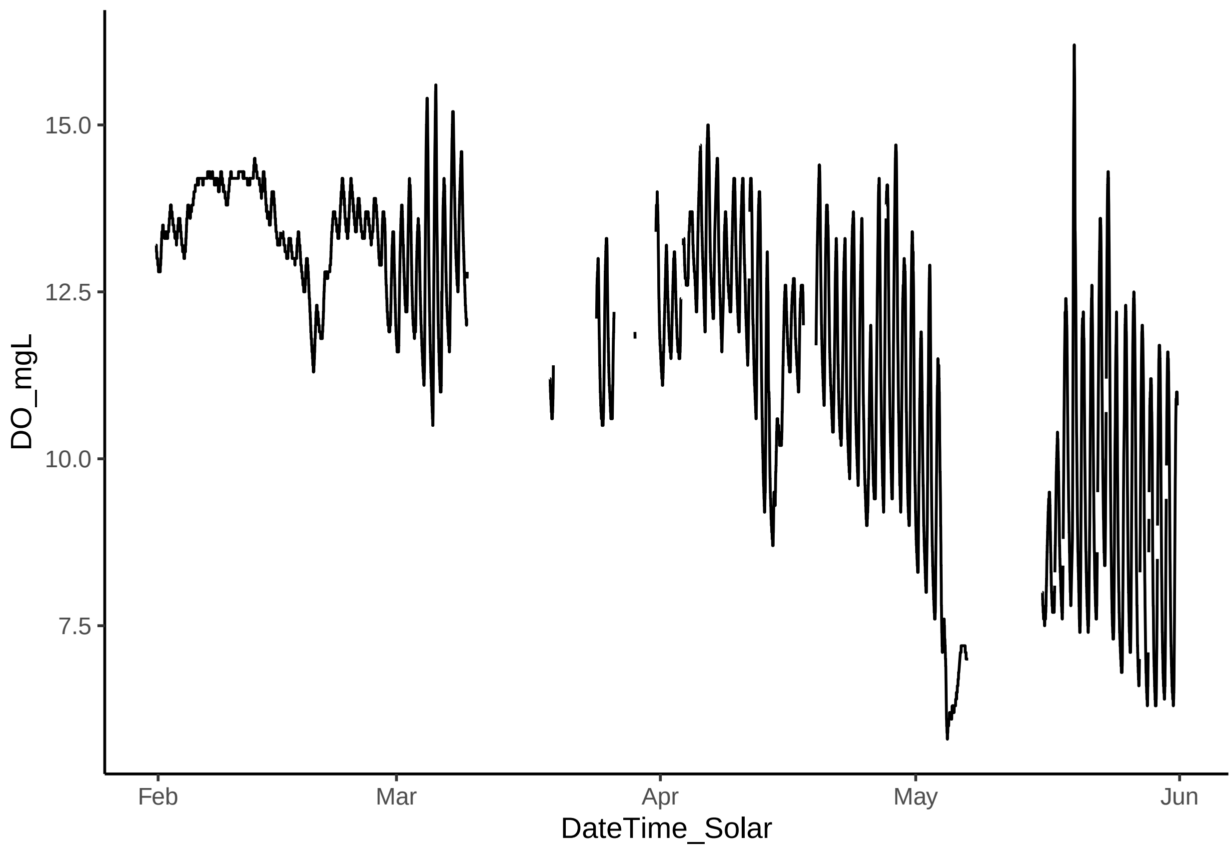
```r
KS.lon <- KSdat[[2]]$lon

KSdat.df <- KSdat[[1]] %>%
  spread(value = value, key = variable) %>%
  mutate(DateTime_Solar = convert_UTC_to_solartime(DateTime_UTC, KS.lon))

KSdat.df <- select(KSdat.df, c(-flagtype, -flagcomment))
```

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

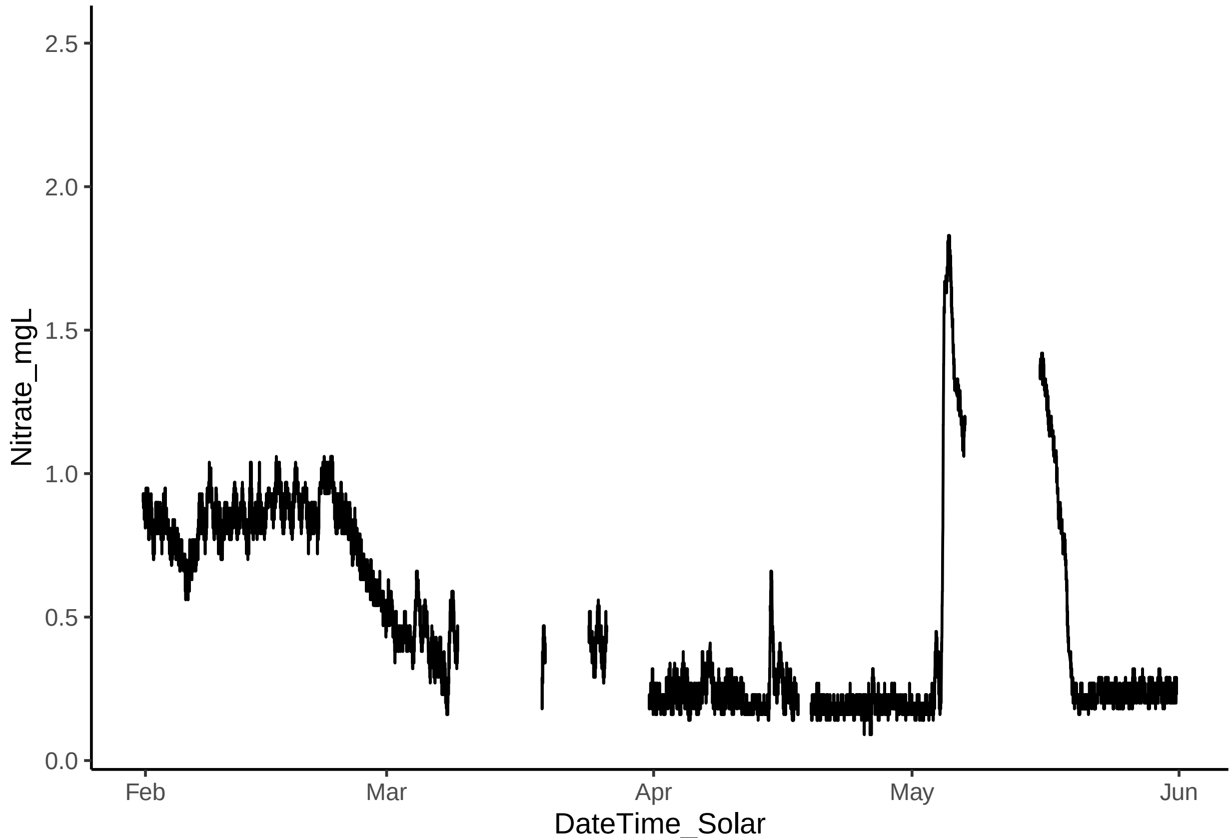
```r
ggplot(KSdat.df, aes(x = DateTime_Solar, y = Discharge_m3s)) + geom_line()
```




```
ggplot(KSdat.df, aes(x = DateTime_Solar, y = DO_mgL)) + geom_line()
```




```
ggplot(KSdat.df, aes(x = DateTime_Solar, y = Nitrate_mgL)) + geom_line()
```



7. How will you address gaps in these dataseries?

   I will remove the NAs first, and then check the the gaps in the data. If the gaps are less than 12 hours, I'll ignore them because the data spans several month.

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

   The amplitude of daily DO increases as the seasons change from winter to summer. This is likely due to the increased biological activity (and therefore respiration and photosynthesis) in the summer months.

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.

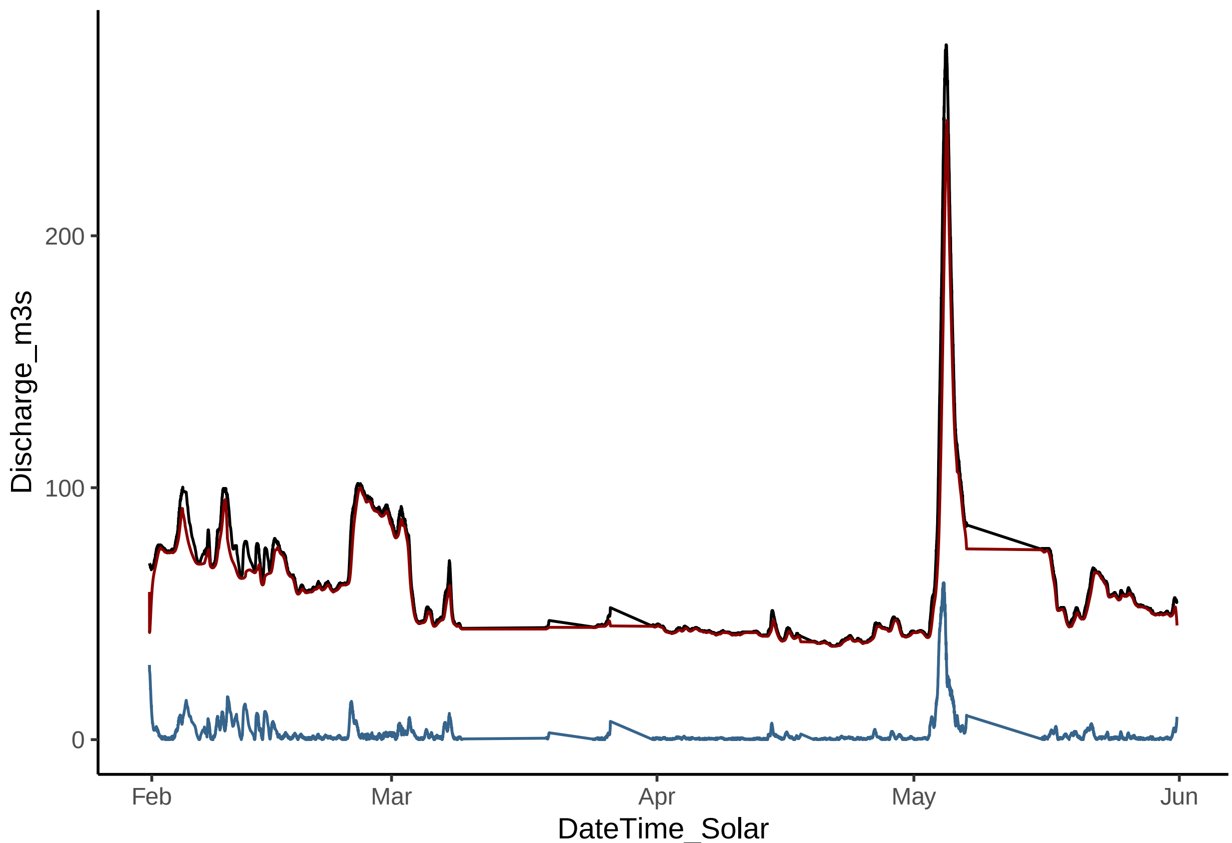
10. Create a ggplot showing total flow, baseflow, and quickflow together.

```
table(diff(KSdat.df$DateTime_UTC))
```

```
##
##     0   900  1800  4500 15300
```

```
## 2788 11388 1 2 7
```

```r
KSdat.df <- na.omit(KSdat.df)

KSbaseflow <- BaseflowSeparation(
  KSdat.df$Discharge_m3s,
  filter_parameter = 0.925,
  passes = 3)

KSdat.flow <- cbind(KSdat.df, KSbaseflow)

ggplot(KSdat.flow, aes(x = DateTime_Solar, y = Discharge_m3s)) +
  geom_line() +
  geom_line(mapping = aes(x = DateTime_Solar, y = bt), color = "red4") +
  geom_line(mapping = aes(x = DateTime_Solar, y = qft), color = "steelblue4")
```




```r
Export <- KSdat.flow %>%
  mutate(timestep = c(diff(as.numeric(DateTime_Solar)), NA_real_),
         baseflowexport = bt * timestep,
         quickflowexport = qft * timestep) %>%
  summarize(BaseflowExport_cf = sum(baseflowexport, na.rm = T),
            QuickflowExport_cf = sum(quickflowexport, na.rm = T),
            TotalExport_cf = BaseflowExport_cf + QuickflowExport_cf)

# Percent Baseflow
595844142/629764592
```

```
## [1] 0.9461379
```

```
# Percent Quickflow
33920450/629764592
```

```
## [1] 0.05386211
```

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

    94.6% is baseflow, 5.4% is quick flow

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 a very large river that resides in a large watershed, the channel is full and flowing year round. Surface flows from storm events are very small compared to the entire area of the watershed, therefore they have little influence on the discharge of the entire watershed. For example, the Mississippi River is always flowing and the stage height is around 15-17 feet (baseflow). A storm event may add quick flow to the river, increasing the stage to 20 feet. In this scenario, baseflow is three times as large as quickflow, which is expected in a large river/watershed, similar to the Kansas River.

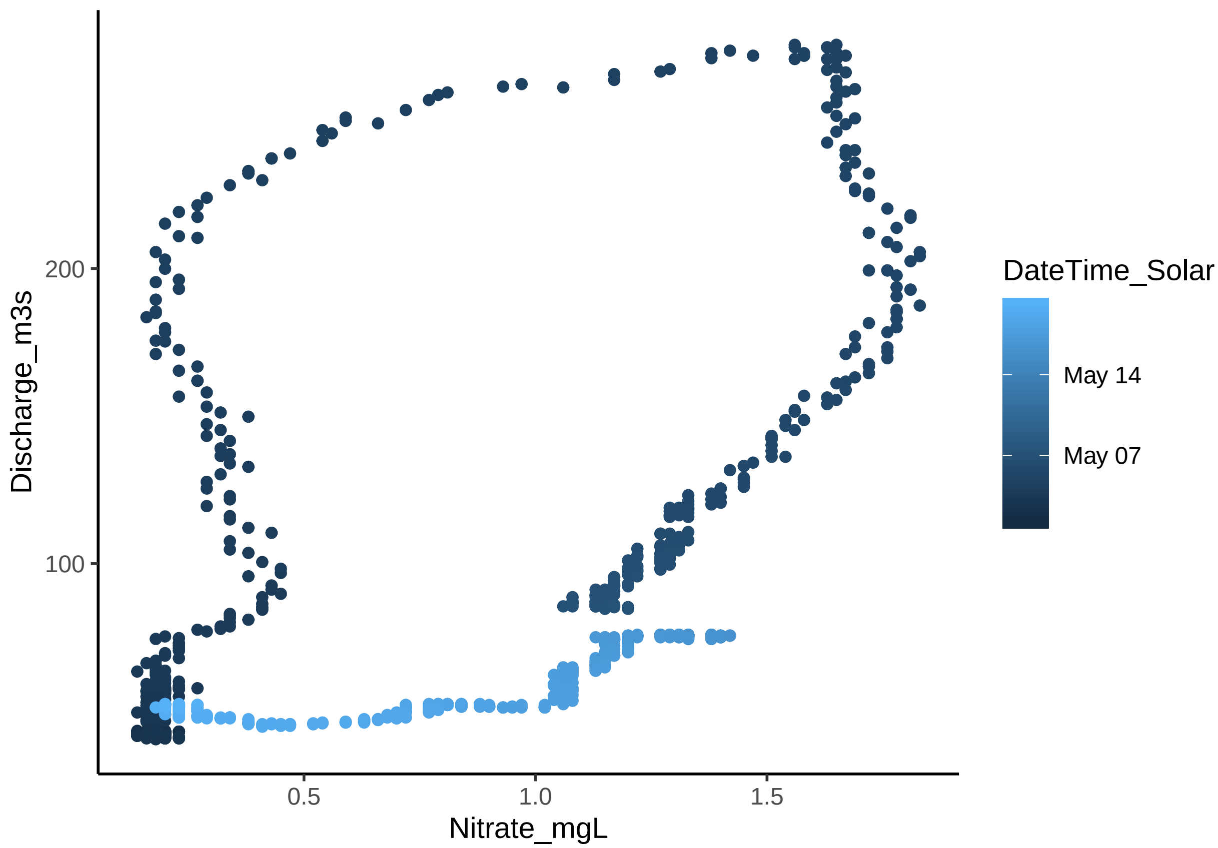
13. The site we are looking at is also further down in its river network (i.e. instead of being a headwater stream, this river has multiple tributaries that flow into it). How does this impact your interpretation of your results?

    The further down we are on the river network, the greater the difference between baseflow and quick flow. Understanding this concept is important when considering what results to expect.

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.

```
KSdat.storm <- filter(KSdat.flow,
  DateTime_Solar > "2018-05-01" & DateTime_Solar < "2018-05-20")

ggplot(KSdat.storm, aes(x = Nitrate_mgL, y = Discharge_m3s,
  color = DateTime_Solar)) +
  geom_point()
```



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

    This storm shows counterclockwise hysteresis. This was a flushing storm.

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

    Nitrate enters the river through surface flow in this watershed.

Reflection

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

    High frequency data allows us to learn about processes that occur on short timescales, like respiration and precipitation. It can help us learn about where contaminant inputs come from by pairing baseflow separation with other high frequency data.

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

    Hysteresis plots and baseflow separation dygrpahs.

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

    Yes, it allowed me to see play with the data and see how systems are differ based on size, location, etc.

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

It met my expections from theory.