Assignment 7: High Frequency Data

Lindsay Roth

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/lindsayroth/Documents/MEM 2nd Year/HydroData/Hydrologic_Data_Analysis"
```

```
library(EcoHydRology)
```

```
## Loading required package: operators

##
## Attaching package: 'operators'

## 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
```

```
library(StreamPULSE)
```

```
## Loading required package: shiny
```

```
## Loading required package: Cairo

## Loading required package: dplyr

##
## Attaching package: 'dplyr'

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

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
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(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------------------------------ tidyve

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

## -- Conflicts ----------------------------------------------------------------------------- tidyverse_c
## x forcats::%>%()  masks stringr::%>%(), purrr::%>%(), tidyr::%>%(), dplyr::%>%(), operators::%>%()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```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.

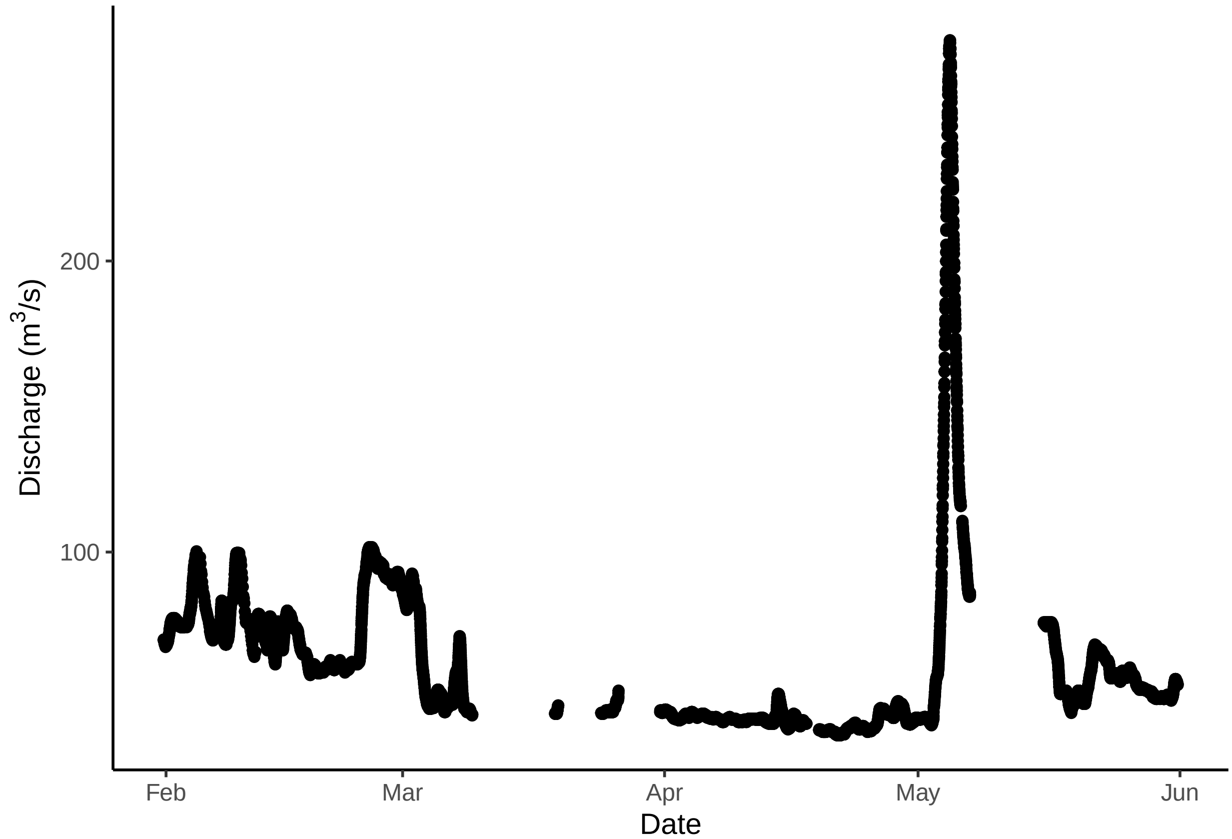
```r
Kansasdat <- request_data(
  sitecode = "KS_KANSASR",
  variables = c('Discharge_m3s', 'DO_mgL', 'Nitrate_mgL')
  )
```

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

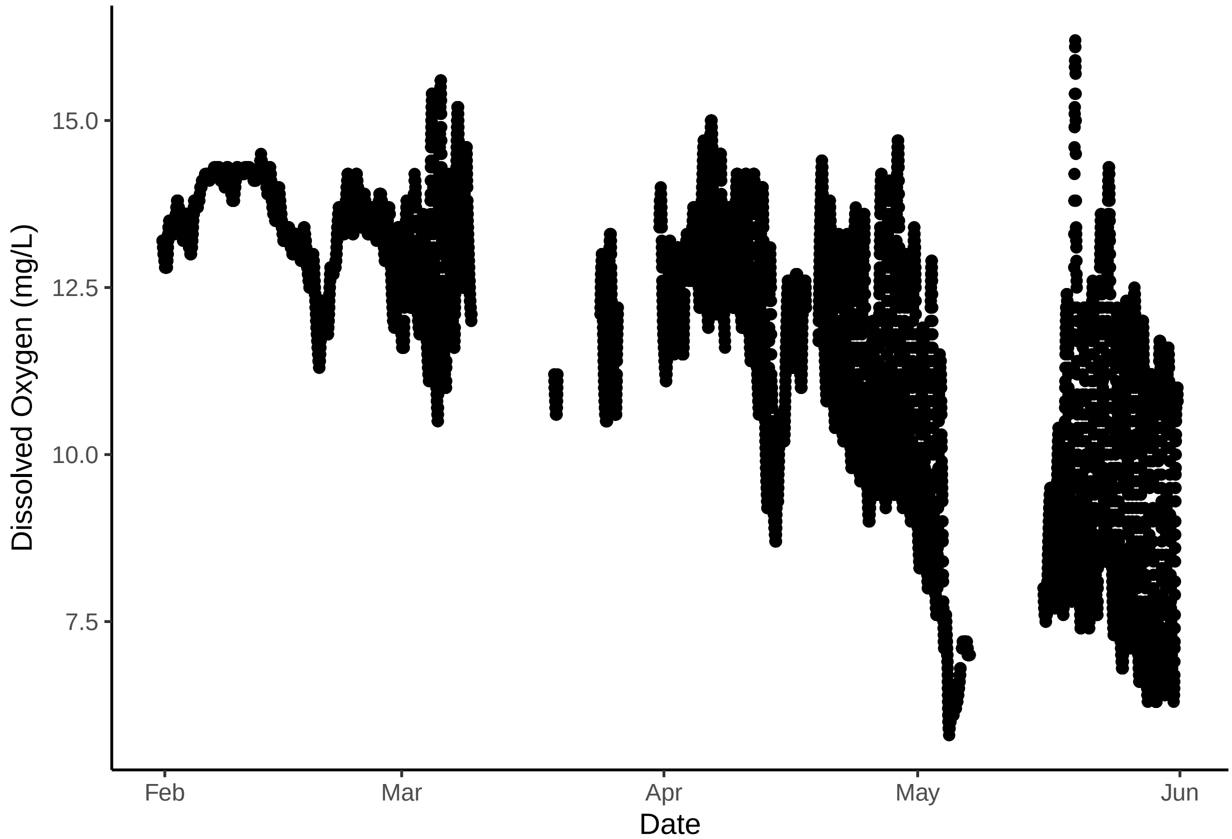
```r
Kansas.lon <- Kansasdat[[2]]$lon

Kansasdf <- Kansasdat[[1]] %>%
  spread(value = value, key = variable) %>%
  mutate(DateTime_Solar = convert_UTC_to_solartime(DateTime_UTC, Kansas.lon)) %>%
  na.omit()
```

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

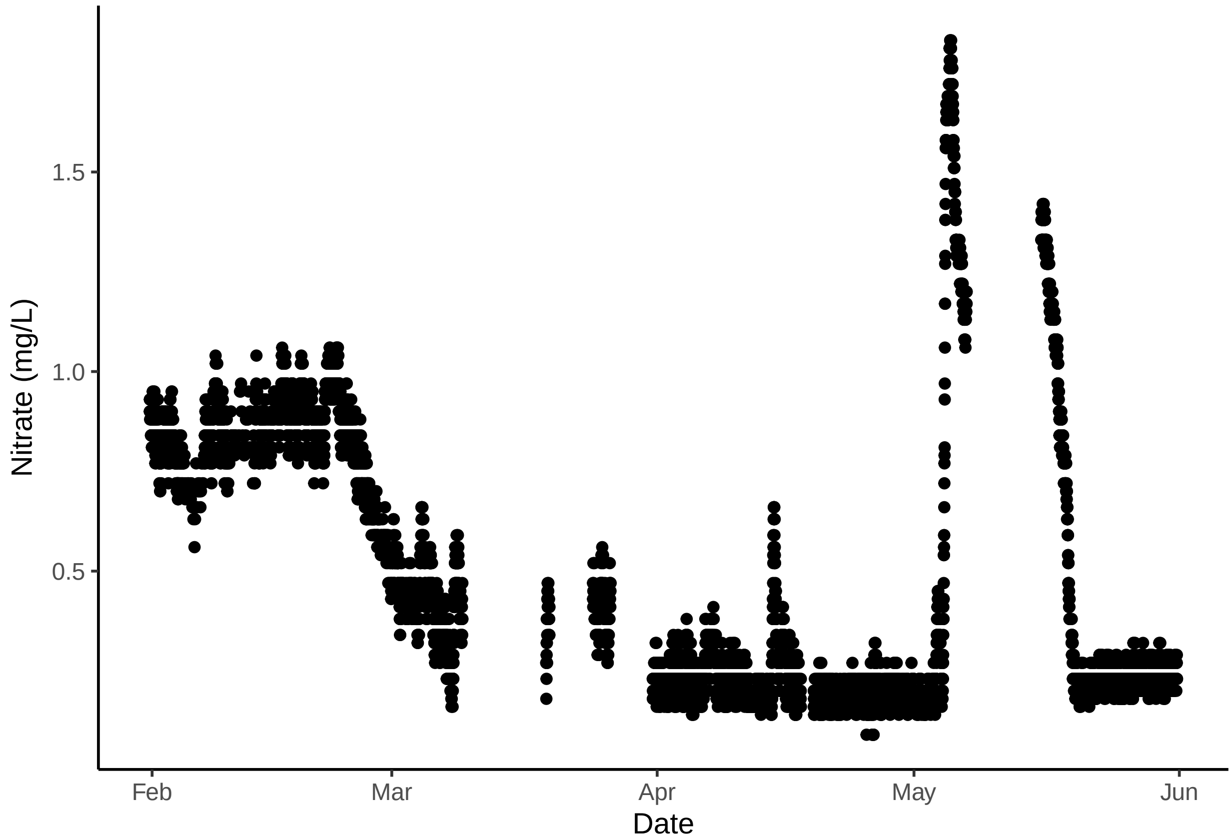
```r
Dischargeplot <- ggplot(Kansasdf) +
  geom_point(aes(x = DateTime_Solar, y = Discharge_m3s)) +
  labs(x = "Date", y = expression("Discharge (m"^3*"/s)"))
print(Dischargeplot)
```




```
DOplot <- ggplot(Kansasdf) +
  geom_point(aes(x = DateTime_Solar, y = DO_mgL)) +
  labs(x = "Date", y = expression("Dissolved Oxygen (mg/L)"))
print(DOplot)
```




```
Nitrateplot <- ggplot(Kansasdf) +
  geom_point(aes(x = DateTime_Solar, y = Nitrate_mgL)) +
  labs(x = "Date", y = expression("Nitrate (mg/L)"))
print(Nitrateplot)
```



7. How will you address gaps in these dataseries?

   I will remove the NAs using since there are double the Nitrate points, there are half of the DO and Discharge rows without measurements. Since there are so few gaps (a few hours over 5 months) I'm going to ignore them.

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

   The swings get larger and the lows and highs are lower in the warmer months than in the colder months. THis may be because

Baseflow separation

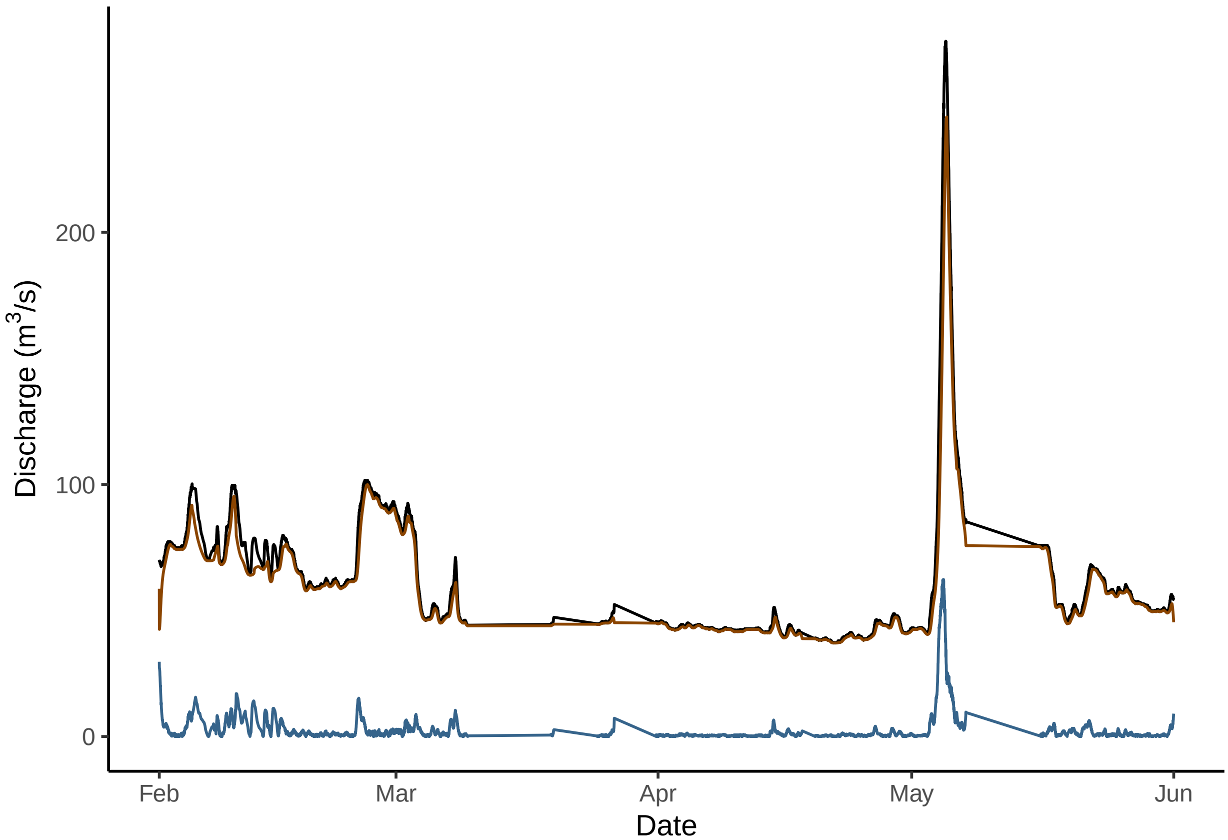
9. Use the `EcoHydRology::BaseflowSeparation()` function to partition discharge into baseflow and quickflow, and calculate how much water was exported as baseflow and quickflow for this time period. Use the DateTime_UTC column as your timestamps in this analysis.

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.

```
Kansas.baseflow <- BaseflowSeparation(
  Kansasdf$Discharge_m3s,
  filter_parameter = 0.925,
```

```
  passes = 3
  )

Kansas.flow <- cbind(Kansasdf, Kansas.baseflow)

ggplot(Kansas.flow, aes(x = DateTime_UTC, y = Discharge_m3s)) +
  geom_line() +
  geom_line(mapping = aes(x = DateTime_UTC, y = bt), color = "darkorange4") +
  geom_line(mapping = aes(x = DateTime_UTC, y = qft), color = "steelblue4") +
  labs(x = "Date", y = expression("Discharge (m"^3*"/s)"))
```




```
#dygraph(
#  cbind(
#    Flow = with(Kansas.flow, xts(Discharge_m3s, order.by = DateTime_UTC)),
#    Baseflow = with(Kansas.flow, xts(bt, order.by = DateTime_UTC)),
#    Quickflow = with(Kansas.flow, xts(qft, order.by = DateTime_UTC))
#  )
#  ) %>%
#  dyRangeSelector()

Export <- Kansas.flow %>%
  mutate(timestep = c(diff(as.numeric(DateTime_UTC)), 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)

#baseflow percent
(5.96e+08)/(6.3e+08)
```

```
## [1] 0.9460317
```

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

    Baseflow is 94.6% of total flow and quickflow is 5.4% of total 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?

    Because it is such a large river and watershed, there is a lot of baseflow coming from the tributaries that supply most of the flow of the river, with storms having a minimal effect on quickflow

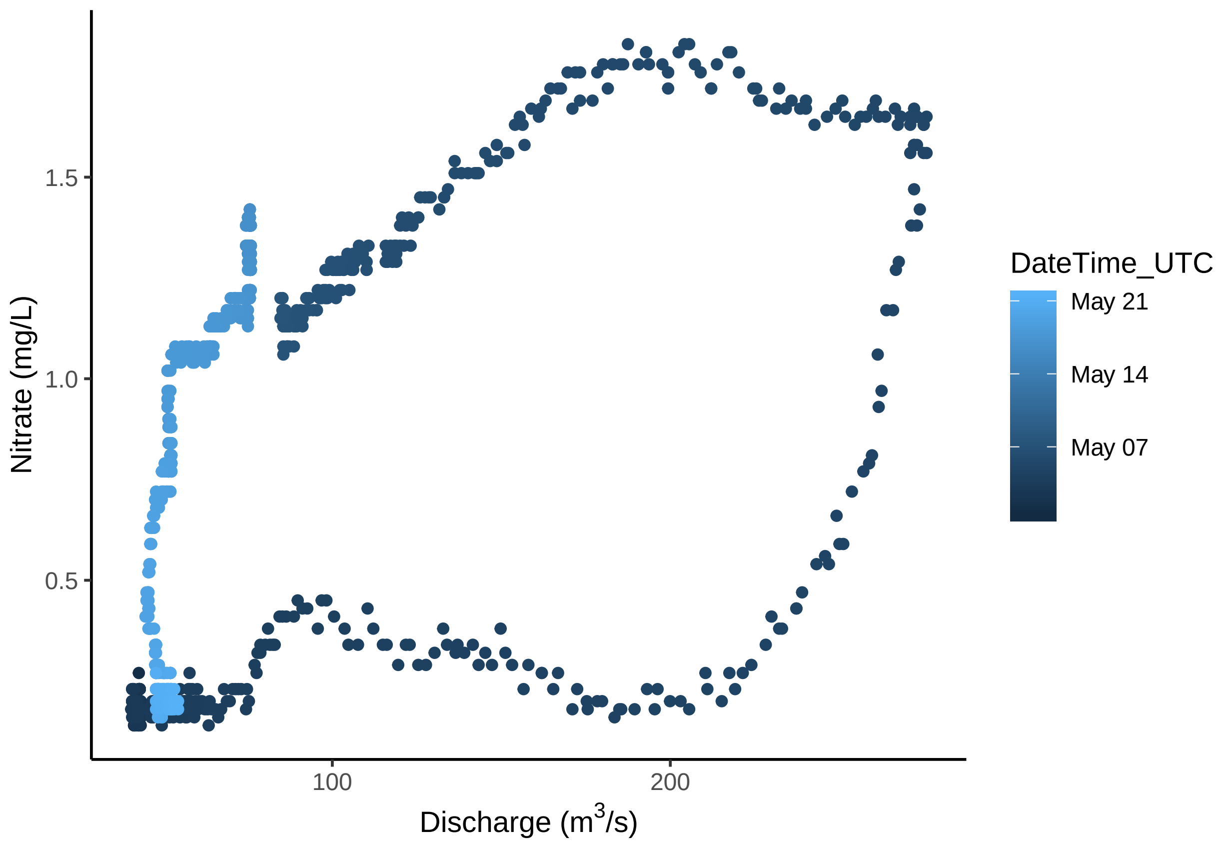
13. The site we are looking at is also further down in its river network (i.e. instead of being a headwater stream, this river has multiple tributaries that flow into it). How does this impact your interpretation of your results?

    This supports my assumption that because most of the flow is coming from tributaries rather than inputs from overland flow and groundwater recharge, the flow of the river is dominated by baseflow rather than quickflow.

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.

```
Kansas.Storm <- Kansasdf %>%
  filter(DateTime_Solar > "2018-04-30" & DateTime_Solar < "2018-05-21")

Nitrate.flow.plot <- ggplot(Kansas.Storm, aes(x = Discharge_m3s, y = Nitrate_mgL, color = DateTime_UTC))
  geom_point() +
  labs(x = expression("Discharge (m"^3*"/s)"), y = "Nitrate (mg/L)")
print(Nitrate.flow.plot)
```



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

    This storm is showing a counterclockwise hysteresis. This storm is a flushing storm.

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

    Nitrate gets into the watershed from overland flow during rain events.

Reflection

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

    Larger watersheds are dominated by baseflow, and nitrate is a flushing nutrient.

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

    The ggplot showing overal discharge, baseflow, and quickflow supported my first conclusion, and the hysteresis plot supported my second conclusion.

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

    Yes, I always learn better by visualizations rather than theory.

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

    I expected that nitrate would be a flushing nutrient, but I did not realize how much the size of a watershed influenced the dominance of baseflow on dishcarge.