2022-11-16 Update

## Catchments with negative discharge estimates - Resolved

* Three sites had certain timeframes where stage predicted discharge to be negative for those hours. This, up until now, had proved problematic for calculating mass fluxes, because I had manually removed those days with negative discharge from the study period - ultimately, having less days increases uncertainty of the total amount of DOC leaving the catchment.
* **Research decision:** for each of the three catchments, I have decided to assign their lowest estimated discharge value to each hourly stage measurement that predicted a negative value. The reason this is more accurate than omitting days is because (1) discharge is very close to zero at my lowest measurement causing the model predictions to become more inaccurate (negative) and (2) the water level data proves the streams are not dry on those days. However, in some cases, the stage predictions with negative discharge values were higher than the one with the lowest positive predicted discharge which leads to an underestimate of DOC mass flux. Ultimately, this correction allows for every day within the study period to have a positive discharge value and therefore a daily mass flux of DOC, which is the best I can do.

## Slide 1 - Preliminary seasonal DOC mass loads

* I have completed two out of three methods for computing seasonal DOC mass loads for an arbitrarily set timeframe (consistent for the 15 catchments that have high quality discharge estimates) and plotted them.
* Load range method: maximum and minimum DOC concentrations used for every day, multiplied by daily discharge and divided by catchment area to get mg/s/km^2. Then, multiplied by the entire amount of seconds in the timeframe (141 days) and divided by 1,000,000 to get kg/km^2 for the period.
* Linear interpolation: used the na.approx function to estimate concentrations between days. **Research decision:** there are a number of days (<10) prior to the first water sample taken at each site where concentrations needed to be estimated. I decided to apply each site’s first sample concentration for those days due to not having a prior point to interpolate from, which additionally causes the issue of not knowing whether the concentration was higher or lower than the first sample day. Without sampling on those days, this seems to be a sound, conservative estimate (for a period that is <10% of the timeframe).
* I’ll further assess this plot when I finish up the linear regression method. I’ve attached the load range code below if you want to quickly glance at the process Jason.

## Slide 2 - CVC and CVQ

* Here is the plot for the variance method Jim suggested. At first glance, the majority of the sites fall between ratio values of 0.2 and 0.6; two sites fall > 0.6 indicating that relative C variability is greater than relative Q variability and vice versa for the four sites that fall < 0.2. I’ll get to thinking about these outliers (catchment size, land characteristics, etc) as I wait for the eFRI data to be cleaned up by Erika. Again, simple code is attached below.

## Appendix - R code

rm(list = ls())  
options(stringsASfactors = FALSE, scipen = 999, encoding = "UTF-8")  
  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.5 ✓ purrr 0.3.4  
## ✓ tibble 3.1.6 ✓ dplyr 1.0.8  
## ✓ tidyr 1.2.0 ✓ stringr 1.4.0  
## ✓ readr 2.1.2 ✓ forcats 0.5.1

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'readr' was built under R version 4.0.5

## Warning: package 'dplyr' was built under R version 4.0.5

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(readxl)

## Warning: package 'readxl' was built under R version 4.0.5

library(zoo)

##   
## Attaching package: 'zoo'

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

#### 2. DOC seasonal mass flux

# 2.01 - Bring in streamflow, dailyQ, water chemistry ----  
  
strflow <- readRDS("streamflow\_final\_v1.02.RDS")  
  
wtr\_chem <- readRDS("glfc\_chem\_cleaned\_v1.01.RDS")  
  
# 2.02 - Test merge process ----  
  
ws40\_dailyQ <- readRDS("ws40\_dailyQ.rds")  
  
ws40\_chem <- wtr\_chem %>%  
 select(-glfc.id) %>%  
 filter(site %in% "WS 40" & variable %in% "organic.carbon") %>%  
 separate(date, c("year", "month", "day"), sep = "(\\-| )") %>%  
 select(-year)  
  
ws40\_out <- left\_join(ws40\_dailyQ, ws40\_chem) # this process works

## Joining, by = c("day", "month", "site")

ws40\_timeframe <- ws40\_out[-c(1, 2, 3, 4, 5), ] # remove the rows outside of the set timeframe

#### 2.03 - Load range mass flux computation

# 2.03.1 - Check range of DOC ----  
  
range(ws40\_out$value, na.rm = TRUE)

## [1] 8.427 11.646

ws40\_load\_range <- ws40\_timeframe %>% # compute each max and min load range  
 mutate(top.mass.flux = ((dailyQ \* 11.646) / 25.5),  
 btm.mass.flux = ((dailyQ \* 8.427) / 25.5)) # now in mg/s/km^2  
  
# 2.03.2 - Summarise total load per study period ----  
  
ws40\_load\_range <- ws40\_load\_range[-142, ] # remove the days past Oct. 23rd for the remaining sites first  
  
ws40\_max <- sum(ws40\_load\_range$top.mass.flux) \* 12182400 / 1000000 # 12182400 seconds in the corrected timeframe; now in kg / km2 over the study period  
  
ws40\_min <- sum(ws40\_load\_range$btm.mass.flux) \* 12182400 / 1000000

#### 2.04 - c-Q Variance coefficients

mass\_flux <- readRDS("mass\_flux\_estimates\_cleaned\_v1.00.rds")  
  
mass\_flux\_sub <- mass\_flux %>%  
 select(group, catchment, catchment.id, sample.number, sample.date,  
 daily.discharge, organic.carbon) # remove all the necessary columns  
  
mf\_cv <- mass\_flux\_sub %>%  
 group\_by(catchment) %>%  
 mutate(mean.conc = mean(organic.carbon), mean.q = mean(daily.discharge),  
 std.conc = sd(organic.carbon), std.q = sd(daily.discharge),  
 cv.conc = std.conc/mean.conc, cv.q = std.q/mean.q, cvc.cvq.ratio = cv.conc/cv.q)