

# Assignment 1: Introduction

Student Name

## OVERVIEW

This exercise accompanies the lessons in Water Data Analytics on introductory material.

## 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 (marked with >).
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After completing your assignment, fill out the assignment completion survey in Sakai.

Having trouble? See the assignment’s answer key if you need a hint. Please try to complete the assignment without the key as much as possible - this is where the learning happens!

Target due date: 2022-01-18

## Course Setup

1. Post the link to your forked GitHub repository below. Your repo should include one or more commits and an edited README file.

Link:

## Data Visualization Exercises

2. Set up your work session. Check your working directory, load packages `tidyverse`, `dataRetrieval`, and `zoo`. Set your ggplot theme as `theme_classic` (you may need to look up how to set your theme).

```
getwd()
```

```
## [1] "/Users/katerisalk/Box Sync/Courses/Water Data Analytics"
```

```
# install.packages("tidyverse")
# install.packages("dataRetrieval")
# install.packages("zoo")
# install.packages("ggrepel")
```

```
library(tidyverse)
library(dataRetrieval)
library(zoo)
```

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

3. Upload discharge data for the Eno River at site 02096500 for the same dates as we studied in class (2012-01-01 through 2021-12-31). Obtain data for discharge. Rename the columns with informative titles, as we did in class.

```
# Import data
EnoDischarge <- readNWISdv(siteNumbers = "02096500",
                           parameterCd = "00060", # discharge (ft3/s)
                           startDate = "2012-01-01",
                           endDate = "2021-12-31")

# Renaming columns (one method of multiple)
names(EnoDischarge)[4:5] <- c("Discharge_cfs", "Approval.Code")
```

4. Build a plot called `EnoPlot2`. Use the base plot we made in class and make the following changes:

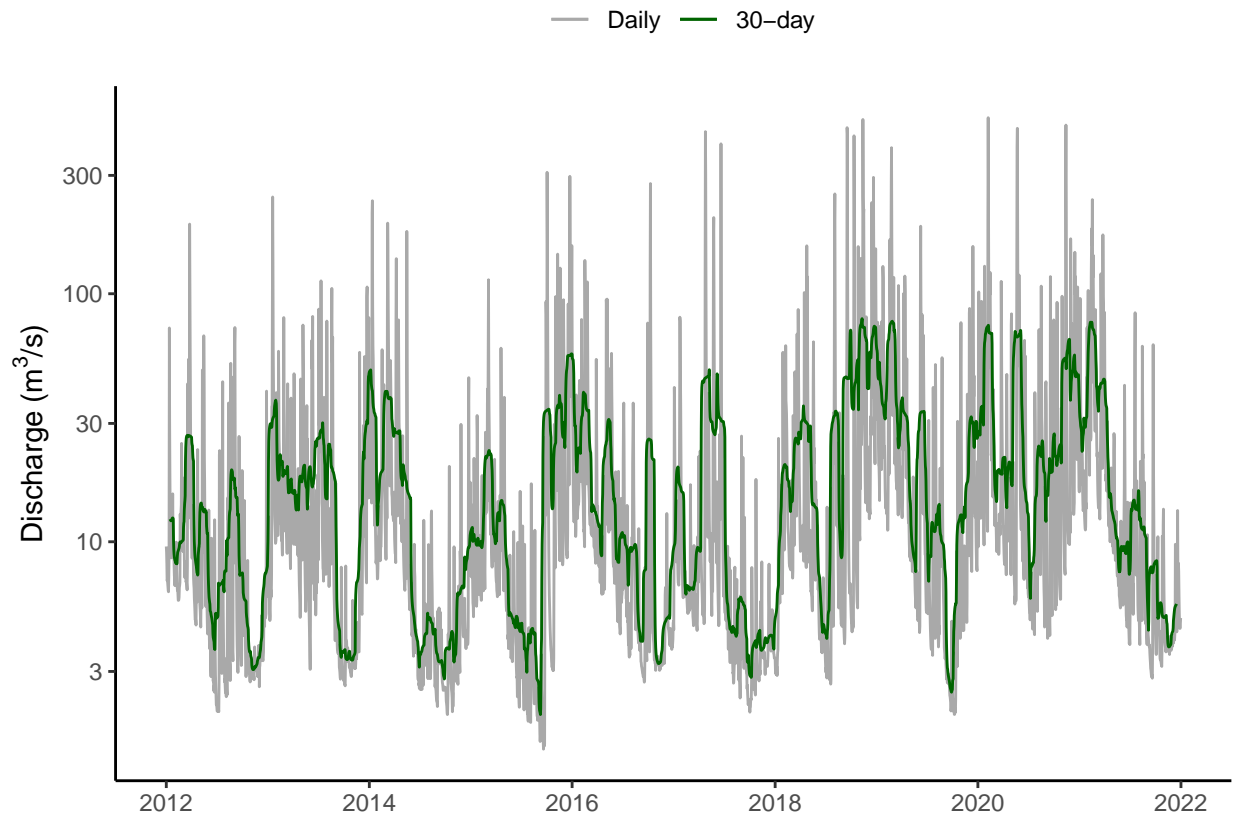
- Add a column to your data frame for discharge in meters cubed per second. hint: package `dplyr` in `tidyverse` includes a `mutate` function
- Add a column in your data frame for a 30-day rolling mean of the metric discharge. (hint: package `dplyr` in `tidyverse` includes a `mutate` function. hint: package `zoo` includes a `rollmean` function)
- Create two `geom_line` aesthetics, one for daily discharge (meters cubed per second) and one for rolling mean of discharge. Color these differently.
- Update your `ggplot` theme. I suggest “classic.” (hint: <https://ggplot2.tidyverse.org/reference/ggtheme.html>)
- Update axis names
- Change the y axis from a linear to a `log10` axis (hint: google “ggplot logged axis”)
- Add a legend. (hint: Google “add legend two geom layers ggplot”)

```
EnoDischarge <- EnoDischarge %>%
  mutate(Discharge_metric = Discharge_cfs/35.3147,
         Discharge_RollMean = rollmean(Discharge_metric, 30, fill = NA))

EnoPlot2 <-
  ggplot(EnoDischarge, aes(x = Date)) +
  geom_line(aes(y = Discharge_metric, color = "Daily")) +
  geom_line(aes(y = Discharge_RollMean, color = "30-day")) +
  scale_color_manual(values = c("Daily" = "darkgray", "30-day" = "darkgreen")) +
  scale_y_log10(name = expression("Discharge (m"~3~/s)")) +
  theme_classic() +
  theme(axis.title.x = element_blank(), legend.title = element_blank(),
        legend.position = "top")

EnoPlot2
```

```
## Warning: Removed 29 row(s) containing missing values (geom_path).
```



5. In what ways was the second plot a more effective visualization than the first?

ANSWER:

6. What portions of the coding were challenging for you?

ANSWER:

7. Interpret the graph you made. What are the things you notice about within- and across-year variability, as well as the differences between daily values and 30-day rolling mean?

ANSWER: