Assignment 3: Physical Properties of Rivers

Lydie Costes

OVERVIEW

This exercise accompanies the lessons in Water Data Analytics on the physical properties of rivers.

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, check your PDF against the key and then submit your assignment completion survey at https://forms.gle/ydeD5axzCnaNzgss9

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: 2022-02-08

Setup

1. Verify your working directory is set to the R project file. Load the tidyverse, dataRetrieval, lubridate, and lfstat packages. Set your ggplot theme (can be theme_classic or something else).
2. Import a data frame called "MysterySiteDischarge" from USGS gage site 03431700. Import discharge data starting on 1964-10-01 and ending on 2021-09-30. Rename columns 4 and 5 as "Discharge" and "Approval.Code". DO NOT LOOK UP WHERE THIS SITE IS LOCATED.
3. Build a ggplot of discharge over the entire period of record.

```
# 1. Check Project working directory
getwd()
```

```
## [1] "/Users/lydiecostes/Documents/Duke/WaterDataAnalytics/Water_Data_Analytics_2022/Assignments"
```

```
# Load packages
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

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

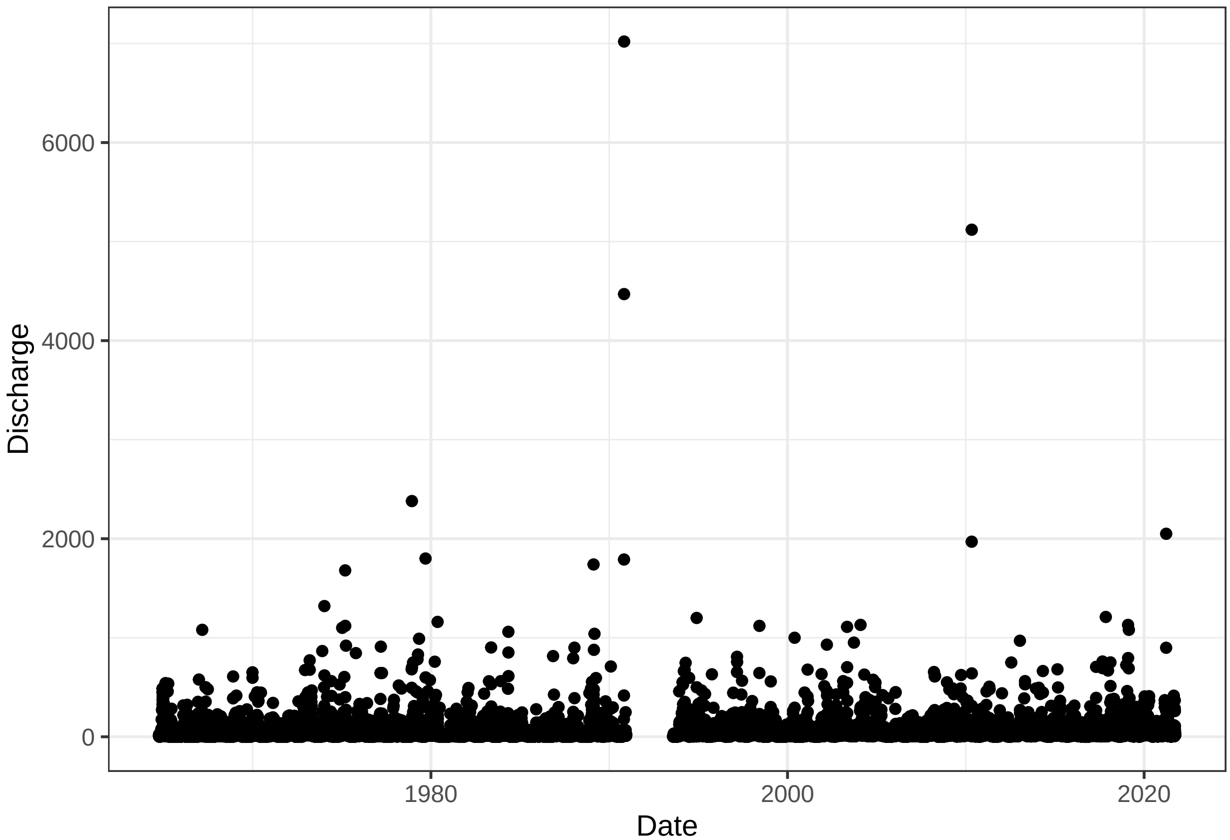
```r
library(dataRetrieval)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```r
library(lfstat)
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

##
## Attaching package: 'xts'

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

## Loading required package: lmom

## Loading required package: lattice
```

```r
# Set theme
theme_set(theme_bw())

# 2. Import dataframe
MysterySiteDischarge <- readNWISdv(siteNumbers = "03431700",
                                   parameterCd = "00060",
                                   startDate = "1964-10-01",
                                   endDate = "2021-09-30")
# Rename columns
colnames(MysterySiteDischarge)[4:5] <- c("Discharge", "Approval.Code")

# 3. Build ggplot
ggplot(MysterySiteDischarge, aes(x=Date, y=Discharge)) +
  geom_point()
```

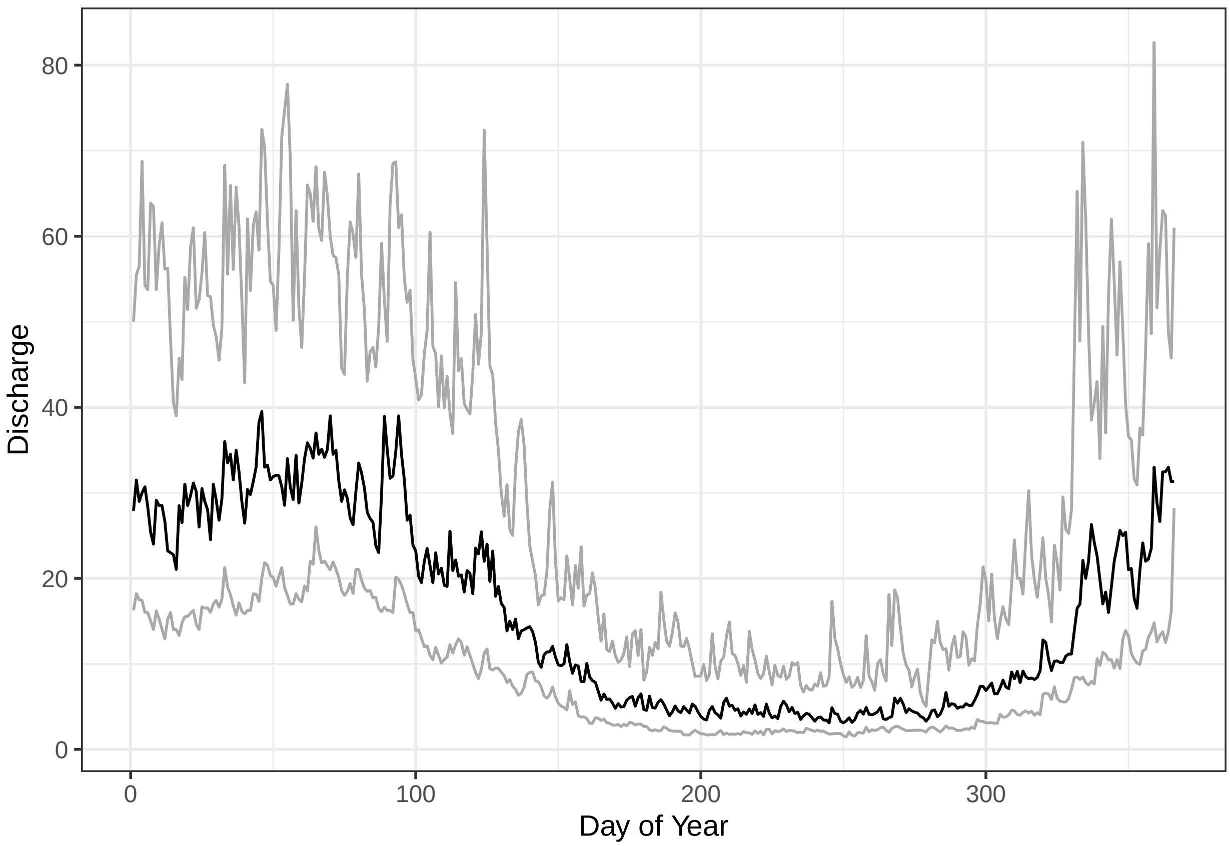


Analyze seasonal patterns in discharge

4. Add a "WaterYear" and "DayOfYear" column to the data frame. Hint: Use a pipe, and you will need both the lubridate and lfstat packages. Set WaterYear to numeric.
5. Create a new data frame called "MysterySiteDischarge.Pattern" that has columns for Day.of.Year, median discharge for a given day of year, 75th percentile discharge for a given day of year, and 25th percentile discharge for a given day of year. Hint: the summarise function includes `quantile`, wherein you must specify `probs` as a value between 0 and 1.
6. Create a plot of median, 75th quantile, and 25th quantile discharges against day of year. Median should be black, other lines should be gray.

```r
# 4. Add year and day of year columns
MysterySiteDischarge <- MysterySiteDischarge %>%
  mutate(MysterySiteDischarge, WaterYear = year(Date),
         DayOfYear = yday(Date))

# 5. Create summary data frame
MysterySiteDischarge.Pattern <- MysterySiteDischarge %>%
  group_by(DayOfYear) %>%
  summarise(Median = median(Discharge),
            Pct25 = quantile(Discharge, probs = 0.25),
            Pct75 = quantile(Discharge, probs = 0.75))

# 6. Plot results
ggplot(MysterySiteDischarge.Pattern, aes(x=DayOfYear)) +
  geom_line(aes(y=Median)) +
```

```
  geom_line(aes(y=Pct25), color="darkgray") +
  geom_line(aes(y=Pct75), color="darkgray") +
  labs(x = "Day of Year", y = "Discharge")
```



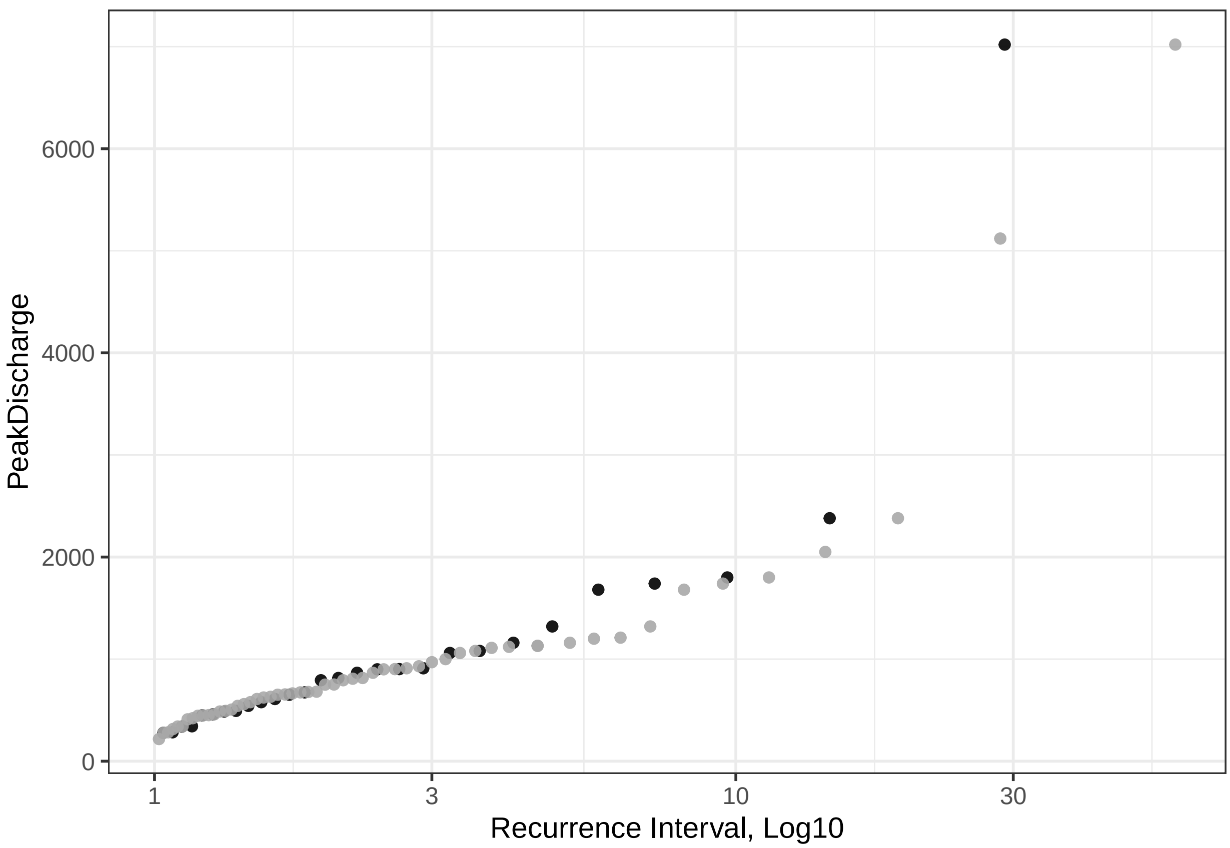
7. What seasonal patterns do you see? What does this tell you about precipitation patterns and climate in the watershed?

   Discharge is lowest in the summer-early fall. It is highest and more variable in the winter and spring. Precipitation appears to be lowest in the warmest months. There doesn't appear to be a strong snowpack effect, so it's likely not a particularly cold climate.

Create and analyze recurrence intervals

8. Create two separate data frames for MysterySite.Annual.30yr (first 30 years of record) and MysterySite.Annual.Full (all years of record). Use a pipe to create your new data frame(s) that includes the water year, the peak discharge observed in that year, a ranking of peak discharges, the recurrence interval, and the exceedende probability.
9. Create a plot that displays the discharge vs. recurrence interval relationship for the two separate data frames (one set of points includes the values computed from the first 30 years of the record and the other set of points includes the values computed for all years of the record.
10. Create a model to predict the discharge for a 100-year flood for both sets of recurrence intervals.

```r
# 8. Create two separate dataframes:
MysterySite.Annual.30yr <- MysterySiteDischarge %>%
  filter(WaterYear < 1994) %>%
  group_by(WaterYear)  %>%
  summarise(PeakDischarge = max(Discharge)) %>%
  mutate(Rank = rank(-PeakDischarge),
         Recurrence = (length(WaterYear) + 1)/Rank,
         Probability = 1/Recurrence)

MysterySite.Annual.Full <- MysterySiteDischarge %>%
  group_by(WaterYear)  %>%
  summarise(PeakDischarge = max(Discharge)) %>%
  mutate(Rank = rank(-PeakDischarge),
         Recurrence = (length(WaterYear) + 1)/Rank,
         Probability = 1/Recurrence)

# 9. Plots to display discharge vs. recurrence interval
ggplot(MysterySite.Annual.30yr, aes(x=Recurrence, y=PeakDischarge)) +
  geom_point(alpha = 0.9) +
  geom_point(data = MysterySite.Annual.Full, aes(x=Recurrence, y=PeakDischarge),
             color = "darkgrey", alpha = 0.9) +
  scale_x_log10() +
  labs(x = "Recurrence Interval, Log10")
```



```r
# 10. Create models to predict, first with just 30 years of data
MysteryModel.30yr <- lm(data = MysterySite.Annual.30yr, PeakDischarge ~ log10(Recurrence))
summary(MysteryModel.30yr)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log10(Recurrence), data = MysterySite.Annual.30yr)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -982.36 -366.95   42.38  247.09 2838.21
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -107.5      197.3  -0.545     0.59
## log10(Recurrence)   2933.1      361.8   8.107 1.38e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 689.4 on 26 degrees of freedom
## Multiple R-squared:  0.7166, Adjusted R-squared:  0.7057
## F-statistic: 65.73 on 1 and 26 DF,  p-value: 1.377e-08
```

```r
MysteryModel.30yr$coefficients[1] + MysteryModel.30yr$coefficients[2]*log10(100)
```

```
## (Intercept)
##    5758.621
```

```r
# 5759 cms

# Repeat with all data
MysteryModel.Full <- lm(data = MysterySite.Annual.Full, PeakDischarge ~ log10(Recurrence))
summary(MysteryModel.Full)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log10(Recurrence), data = MysterySite.Annual.Full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -773.70 -230.18   35.09  201.60 2743.79
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -0.6529   108.3381  -0.006    0.995
## log10(Recurrence) 2435.7437   190.1486  12.810   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 549 on 54 degrees of freedom
## Multiple R-squared:  0.7524, Adjusted R-squared:  0.7478
## F-statistic: 164.1 on 1 and 54 DF,  p-value: < 2.2e-16
```

```r
MysteryModel.Full$coefficients[1] + MysteryModel.Full$coefficients[2]*log10(100)
```

```
## (Intercept)
##    4870.835
```

```r
# 4871 cms
```

11. How did the recurrence interval plots and predictions of a 100-year flood differ among the two data frames? What does this tell you about the stationarity of discharge in this river?

    With more data, the estimation of the 100-year flood discharge was lower. From looking at the graph, it appears that one extreme year in the first 30-year period may have driven the effect to be more extreme. However, the difference is not very large, suggesting that the river discharge is fairly stable over time.