Assignment 3: Physical Properties of Rivers

Lindsay Roth

OVERVIEW

This exercise accompanies the lessons in Hydrologic Data Analysis 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, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Salk_A03_RiversPhysical.Rmd") prior to submission.

The completed exercise is due on 18 September 2019 at 9:00 am.

Setup

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

```
#verify working directory
getwd()
```

```
## [1] "/Users/lindsayroth/Documents/MEM 2nd Year/HydroData/Hydrologic_Data_Analysis/Assignments"
```

```
#load packages
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------------- tidyverse 1.2.
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

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

```
library(dataRetrieval)
library(cowplot)
```

```
##
## ********************************************************
```

```
## Note: As of version 1.0.0, cowplot does not change the
```

```
##   default ggplot2 theme anymore. To recover the previous
##   behavior, execute:
##   theme_set(theme_cowplot())
## *******************************************************
```

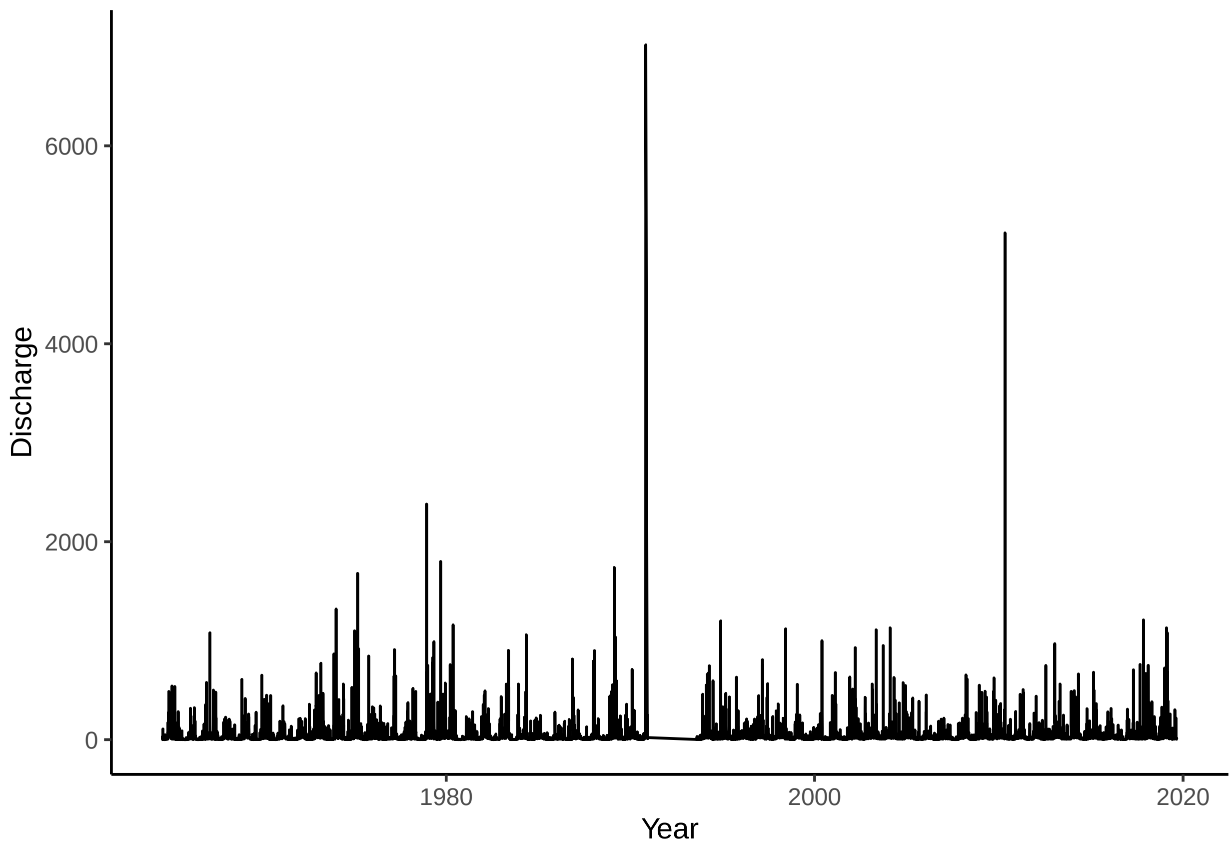
```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:cowplot':
##
##     stamp
## The following object is masked from 'package:base':
##
##     date
```

```r
#set theme
theme_set(theme_classic())

#Load data
MysterySiteDischarge <- readNWISdv(siteNumbers = "03431700",
                     parameterCd = "00060", # discharge (ft3/s)
                     startDate = "",
                     endDate = "")

#check units for discharge
attr(MysterySiteDischarge, "variableInfo")
```

```
##   variableCode           variableName              variableDescription
## 1        00060 Streamflow, ft&#179;/s Discharge, cubic feet per second
##       valueType  unit options noDataValue
## 1 Derived Value ft3/s    Mean          NA
```

```r
#rename columns
names(MysterySiteDischarge)[4:5] <- c("Discharge", "Approval.Code")

#ggplot for discharge over time
MysteryPlot <-
  ggplot(MysterySiteDischarge, aes(x = Date, y = Discharge)) +
         geom_line() +
         xlab("Year")
print(MysteryPlot)
```

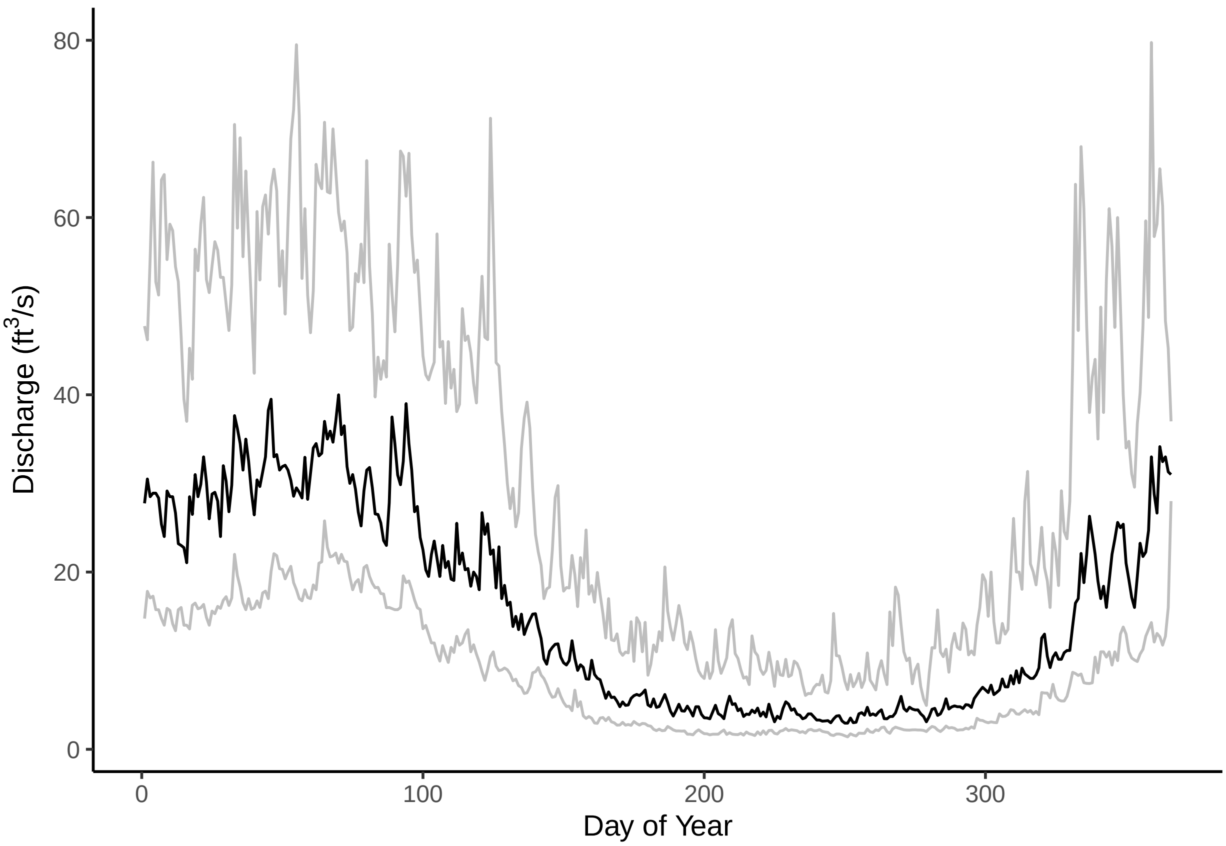


Analyze seasonal patterns in discharge

5. Add a "Year" and "Day.of.Year" column to the data frame.
6. 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.
7. Create a plot of median, 75th quantile, and 25th quantile discharges against day of year. Median should be black, other lines should be gray.

```
#create year and day of year columns
MysterySiteDischarge <- MysterySiteDischarge %>%
  mutate(Year = year(Date), Day.of.Year = yday(Date))

#create new dataframe with median and quantiles
MysterySiteDischarge.Pattern <- MysterySiteDischarge %>%
  group_by(Day.of.Year) %>%
  summarise(Median.Discharge = median(Discharge),
        Percentile75 = quantile(x = Discharge, probs = 0.75),
        Percentile25 = quantile(x = Discharge, probs = 0.25))

#plot median and quantiles
MysteryPatternPlot <-
  ggplot(MysterySiteDischarge.Pattern, aes(x = Day.of.Year)) +
  geom_line(aes(y = Median.Discharge)) +
```

```r
  geom_line(aes(y = Percentile25), color = "gray") +
  geom_line(aes(y = Percentile75), color = "gray") +
  labs(x = "Day of Year", y = expression("Discharge (ft"^3*"/s)"))
print(MysteryPatternPlot)
```



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

   There is a clear seasonal pattern in this data. There is an increase in runoff volume and variability during the winter and spring months, lasting from approximately early December to late April/early May. During the summer and fall months, discharge decreases to very low volume and more consistent levels. This tells me that this watershed has more precipitation events during the winter and spring and not many during the summer and fall. The temperatures at this watershed do not reach below freezing during the winter or spring, as there is no evidence of a decrease in runoff in the winter that would normally seen as evidence of snow/ice. Usually, this kind of precipitation pattern occurs in Mediterranean climates. I would expect this watershed to be located along the western coast of the United States at a location that is not cold enough to receive precipitation as snow or have frozen rivers but that may be affected by runoff from snowmelt from surrounding mountainous areas.There are regular precipitation events from December to April that result in greater variation in runoff and a dry summer that creates the opposite effect.

Create and analyze recurrence intervals

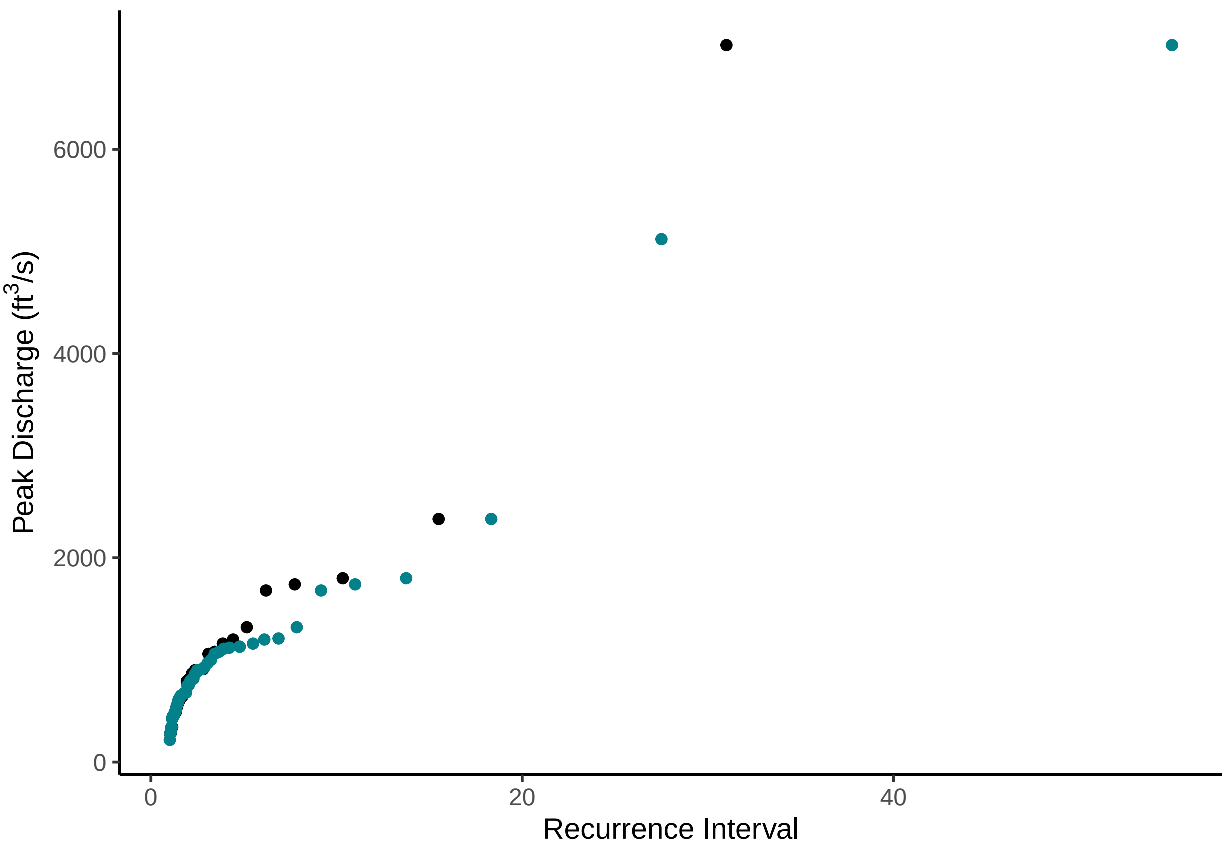
9. Create two separate data frames for MysterySite.Annual.30yr (first 30 years of record) and MysterySite.Annual.Full (all years of record). Use a pipe to create your new data frame(s) that includes the

year, the peak discharge observed in that year, a ranking of peak discharges, the recurrence interval, and the exceedende probability.

10. 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.

11. Create a model to predict the discharge for a 100-year flood for both sets of recurrence intervals.

```
#create dataframe for first 30 years with recurrence interval columns
MysterySite.Annual.30yr <-
  MysterySiteDischarge %>%
  filter(Year < 1996) %>%
  group_by(Year) %>%
  summarise(PeakDischarge = max(Discharge)) %>%
  mutate(Rank = rank(-PeakDischarge), #negative to make sure it's the right order
         RecurrenceInterval = (length(Year) + 1)/Rank,
         Probability = 1/RecurrenceInterval)

#datafram for full data with recurrence interval columns
MysterySite.Annual.Full <-
  MysterySiteDischarge %>%
  group_by(Year) %>%
  summarise(PeakDischarge = max(Discharge)) %>%
  mutate(Rank = rank(-PeakDischarge), #negative to make sure it's the right order
         RecurrenceInterval = (length(Year) + 1)/Rank,
         Probability = 1/RecurrenceInterval)

#plot discharge vs recurrence interval
MysteryoRecurrencePlot <-
  ggplot(MysterySite.Annual.30yr,
         mapping = aes(x = RecurrenceInterval, y = PeakDischarge)) +
  geom_point() +
  geom_point(MysterySite.Annual.Full, color = "#02818a",
             mapping = aes(x = RecurrenceInterval, y = PeakDischarge)) +
  labs(x = "Recurrence Interval", y = expression("Peak Discharge (ft"^3*"/s)"))
print(MysteryoRecurrencePlot)
```




```
#build model
Mystery.30year.model <- lm(data = MysterySite.Annual.30yr,
                           PeakDischarge ~ log(RecurrenceInterval))
summary(Mystery.30year.model)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log(RecurrenceInterval), data = MysterySite.Annual.30yr)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -974.12 -337.65   34.84  232.57 2908.00
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -69.87     185.73  -0.376     0.71
## log(RecurrenceInterval)  1217.79     147.16   8.275 5.26e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 673.9 on 28 degrees of freedom
## Multiple R-squared:  0.7098, Adjusted R-squared:  0.6994
## F-statistic: 68.48 on 1 and 28 DF,  p-value: 5.261e-09
```

```
Mystery.30year.model$coefficients[1] +
  Mystery.30year.model$coefficients[2]*log(100)
```

```
## (Intercept)
##    5538.257
```

```r
#Predicted 100 year flood 5538.257 cfs

Mystery.full.model <- lm(data = MysterySite.Annual.Full,
                         PeakDischarge ~ log(RecurrenceInterval))
summary(Mystery.full.model)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log(RecurrenceInterval), data = MysterySite.Annual.Full)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -955.95 -236.29   41.91  210.67 2805.35
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -2.001    116.322  -0.017    0.986
## log(RecurrenceInterval) 1052.234     88.834  11.845   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 578.3 on 52 degrees of freedom
## Multiple R-squared:  0.7296, Adjusted R-squared:  0.7244
## F-statistic: 140.3 on 1 and 52 DF,  p-value: < 2.2e-16
```

```r
Mystery.full.model$coefficients[1] +
  Mystery.full.model$coefficients[2]*log(100)
```

```
## (Intercept)
##    4843.717
```

```r
#Predicted 100 year flood 4843.717 cfs
```

12. 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?

    The predicted 100 year flood from the first 30 years of data is about 700 cfs larger than the predicted 100 year flood from the full set of data. This means that the stationarity of discharge in this river is not constant, and that discharge has been decreasing with time.

Reflection

13. What are 2-3 conclusions or summary points about river discharge you learned through your analysis?

    Conclusion 1: There is a seasonal precipitation pattern for this site where it rains during the winter and spring months without freezing and is dry during the summer and fall months. Conclusion 2: There were larger extreme runoff events in the first 30 years of data versus the entire dataset, as shown by the larger predicted 100 year flood, and indicating that runoff is decreasing with time.

14. What data, visualizations, and/or models supported your conclusions from 13?

    The summarized daily discharge plot with the median, 75th and 25th quantiles for each day of the year provided me with the evidence I needed to conlcude the seasonality of runoff, and the plot with Peak Discharge vs Recurrence Interval showed me the pattern of decreasing extreme events.

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

    I always learn better with graphs and visualizations, and it was easier for me to see visual patterns of discharge and trends than to learn about them theoretically.

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

    From what I have learned about climate change, I would have expected extreme events to increase with time rather than decrease. This site proved to be the opposite case. However, since I predicted that this location is somewhere along the west coast of the US, it could make sense that increasing periods of drought with a hotter drier climate could impact the intensity of large storm events.