Assignment 3: Physical Properties of Rivers

Ethan Ready

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.

```
getwd()
```

```
## [1] "/Users/ethanready/Hydrologic_Data_Analysis/Assignments"
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------- tidyverse 1.2.1 --
```

```
## 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()
```

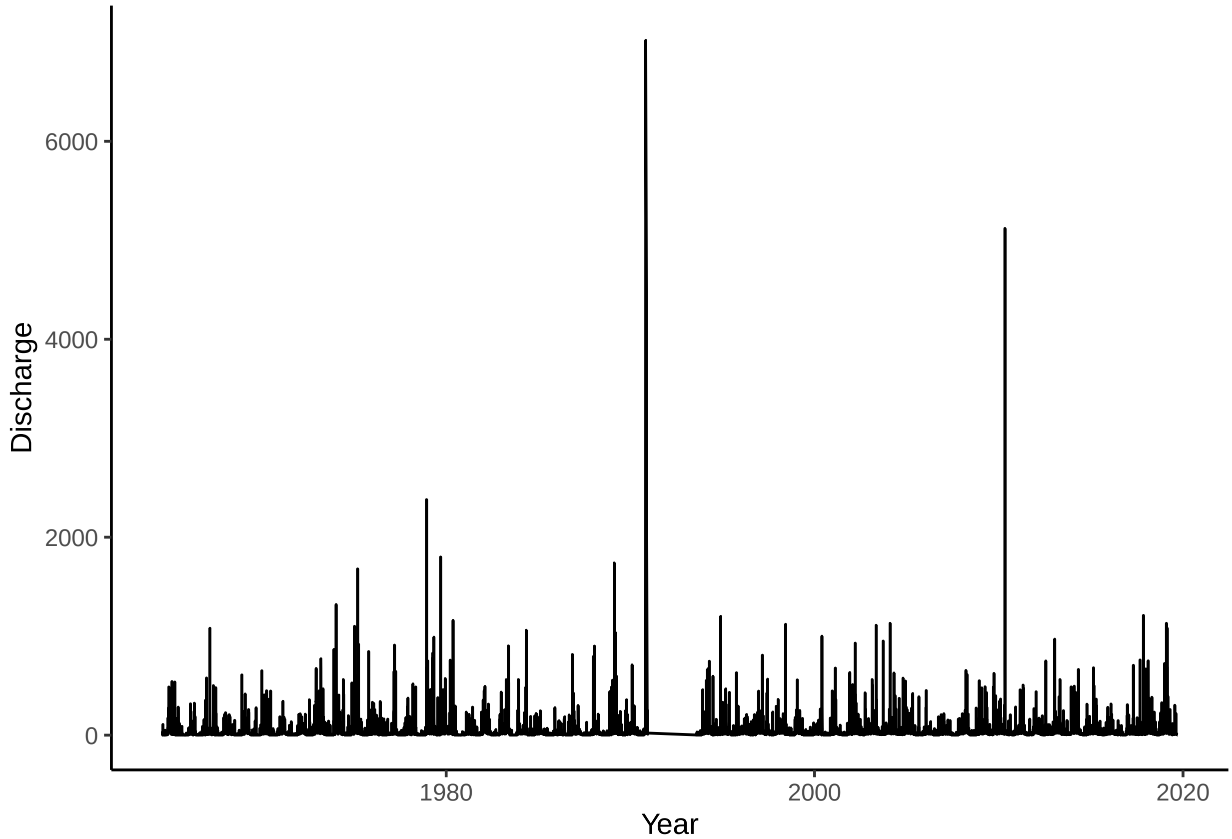
```
library(dataRetrieval)
library(cowplot)
```

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

## 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())

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

```
library(lubridate)
```

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

## The following object is masked from 'package:cowplot':
##
##     stamp

## The following object is masked from 'package:base':
##
##     date
```

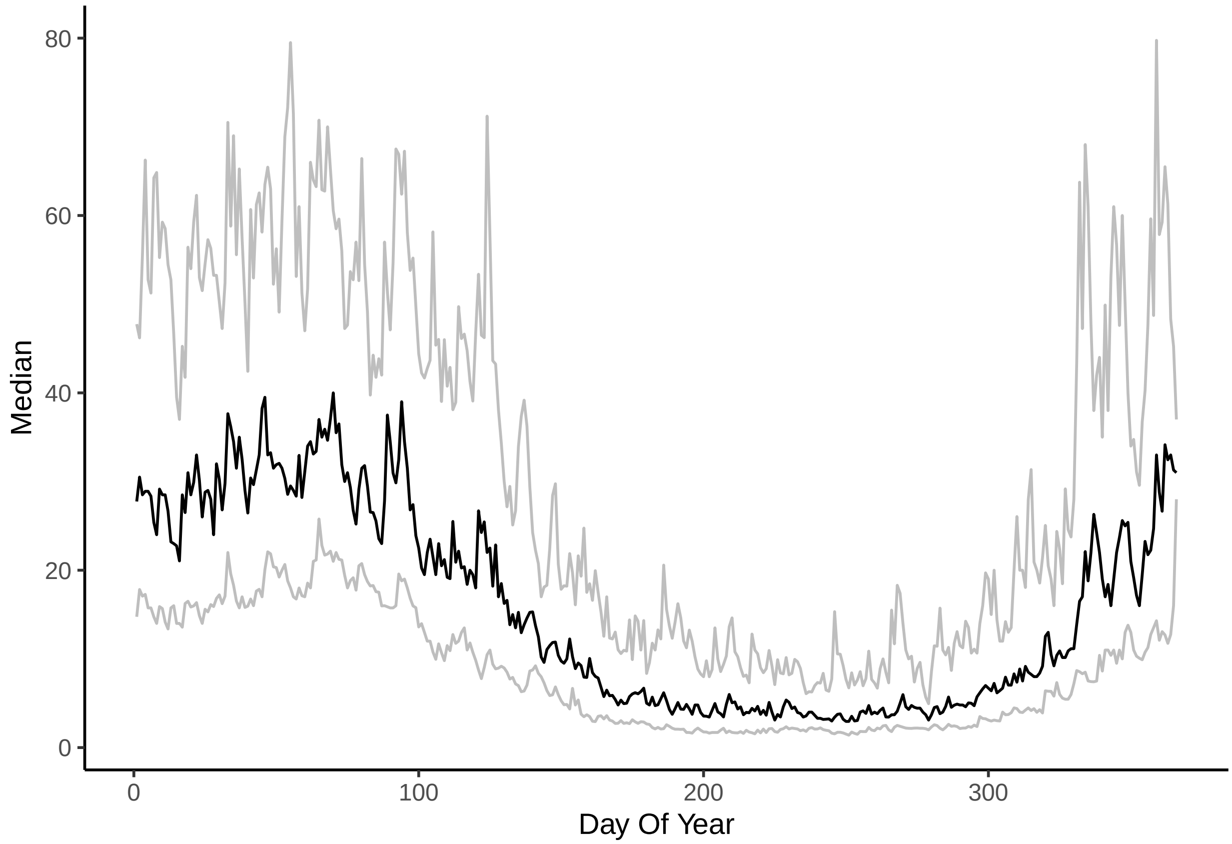
```
theme_set(theme_classic())

MysterySiteDischarge<-readNWISdv(siteNumbers="03431700",
                                 parameterCd = "00060",
                                 startDate = "",
                                 endDate = ""
                                 )

names(MysterySiteDischarge)[4:5]<-c("Discharge", "Approval.Code")

Mystery.Discharge.Plot<-ggplot(MysterySiteDischarge, aes(x=Date, y=Discharge ))+
  geom_line()+
  xlab("Year")
Mystery.Discharge.Plot
```



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.

```
MysterySiteDischarge<-mutate(MysterySiteDischarge, Year = year(Date), Day.of.Year= yday(Date))

#plot 25th percentile, median, and 75th percentile of discharge on 1 plot
MysterySiteDischarge.Pattern<-MysterySiteDischarge%>%
  group_by(Day.of.Year)%>%
  summarise(median= median(Discharge), Percentile.25=quantile(Discharge, 0.25), Percentile.75=quantile(I

MysteryDischarge.Plot.Quantile<-ggplot(MysterySiteDischarge.Pattern, aes(x=Day.of.Year))+
  geom_line(aes(y=median))+
  geom_line(aes(y=Percentile.25), color="gray")+
  geom_line(aes(y=Percentile.75), color="gray")+
  labs(x="Day Of Year", y="Median")

MysteryDischarge.Plot.Quantile
```



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

   Discharge is comparatively low, with little variation, from around day 180 to around day 300, which would be from the beginning of July to the end of October. The rest of the year has higher discharge in the 25th percentile, the median, and the 75th percentile, and also more variation. This suggests two distinct precipitation seasons in the watershed.

Create and analyze recurrence intervals

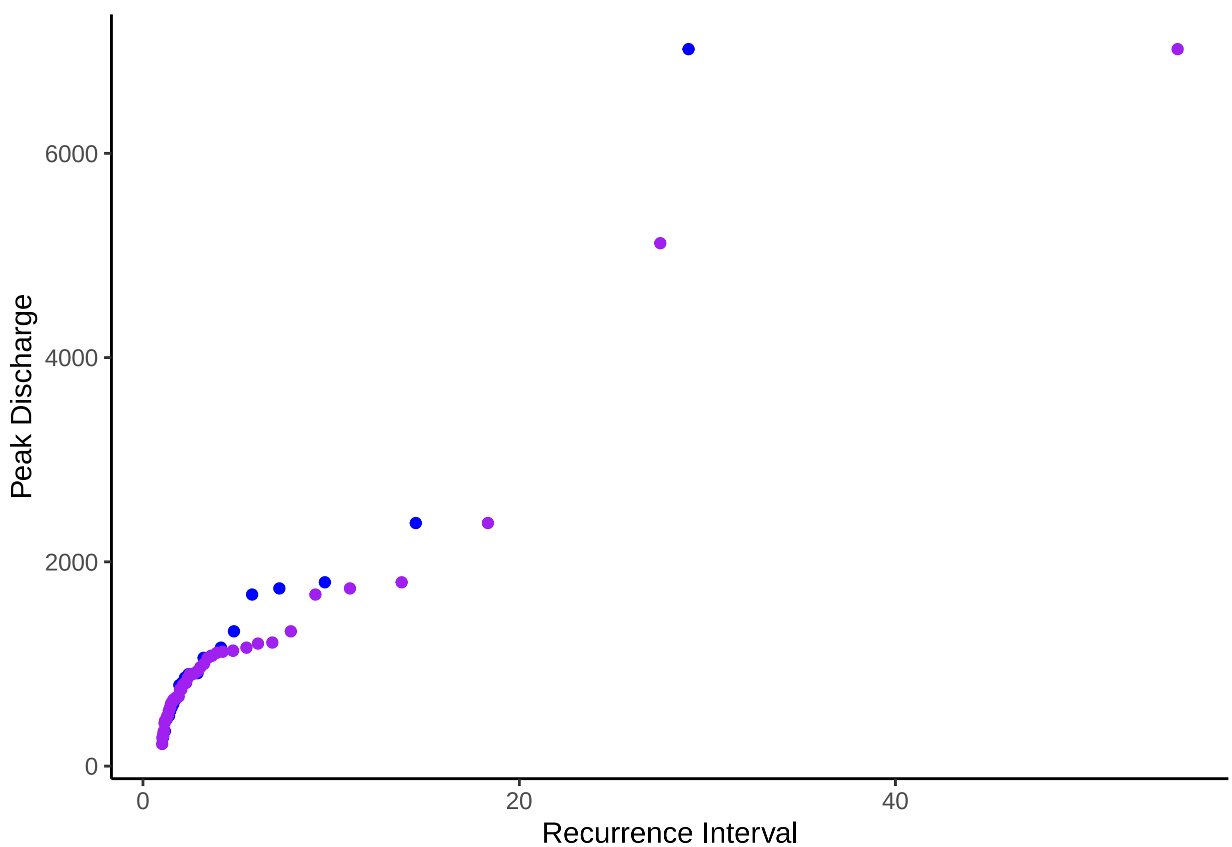
9. Create two separate data frames for MysterySite.Annual.30yr (first 30 years of record) and MysterySite.Annual.Full (all years of record). Use a pipe to create your new data frame(s) that includes the year, the peak discharge observed in that year, a ranking of peak discharges, the recurrence interval, and the exceedende probability.

```
#create two new datasets, one of the first 30 years and one with all the years
MysterySite.Annual.30yr<-MysterySiteDischarge%>%
  filter(Year<1994)%>%
  group_by(Year)%>%
  summarise(PeakDischarge=max(Discharge))%>%
  mutate(Rank=rank(-PeakDischarge),
         Recurrence=(length(Year)+1)/Rank,
         Probability=1/Recurrence)
```

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

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.

```r
# plot peak discharge vs. recurrence interval for each dataset
Discharge.Recurrence<-ggplot()+
  geom_point(data=MysterySite.Annual.30yr, aes(x=Recurrence, y=PeakDischarge), color="blue")+
  geom_point(data=MysterySite.Annual.Full, aes(x=Recurrence, y=PeakDischarge), color="purple")+
  labs(x="Recurrence Interval", y="Peak Discharge")

Discharge.Recurrence
```



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

```r
#create log fit models for both datasets
Model.100flood.30y<-lm(data=MysterySite.Annual.30yr, PeakDischarge~log(Recurrence))
Model.100flood.Full<-lm(data=MysterySite.Annual.Full,PeakDischarge~log(Recurrence))
```

```r
summary(Model.100flood.30y)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log(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
## log(Recurrence)   1273.8      157.1   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
summary(Model.100flood.Full)
```

```
##
## Call:
## lm(formula = PeakDischarge ~ log(Recurrence), 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(Recurrence) 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
#predict 100-year flood based on the two different data sets
Model.100flood.30y$coefficients[1]+Model.100flood.30y$coefficients[2]*log(100)
```

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

```r
Model.100flood.Full$coefficients[1]+Model.100flood.Full$coefficients[2]*log(100)
```

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

Both models had significant linear models with P-values below 0.01

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 first 30 year dataframe predicted a much higher 100-year flood level than the full dataframe, showing that different periods have significantly different discharge yearly maximums in this river. Since we base so much on 100-year floods, like FEMA building codes, the datasets that we choose to use for our recurrence intervals is extremely important.

Reflection

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

    1. River discharge can vary a lot between seasons and between years. While there are some patterns, especially seasonal patterns, it can be hard to predict river discharge.
    2. "100-year flood" Is a complicated statistic, both in terms of understanding what it means and in the variability based on size and scope of the dataset.

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

    Seeing how changing the dataset could change the 100-year flood calculation in the graph with the log distributions. Also seeing the seasonal variation in the day of year plot, and the long dry spell in the first basic discharge by date plot.

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

    Yes definitely. It got me thinking from the perspective of someone managing a floodplain and how they would have to analyze the data.

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

    I'm surprised at how much large discharge events vary even within relatively large windows of time.