3: Data Exploration

Environmental Data Analytics | Kateri Salk

Spring 2023

Objectives

1. Import and explore datasets in R
2. Graphically explore datasets in R
3. Apply data exploration skills to a real-world example dataset

Opening discussion: why do we explore our data?

Why is data exploration our first step in analyzing a dataset? What information do we gain? How does data exploration aid in our decision-making for data analysis steps further down the pipeline?

We need to understand our data to know its uses, limitations, assumptions, structure, and patterns in our data before we start to dig into it. Knowing the data is the first step to analyzing the data.

Import data and view summaries

```
# 1. Set up your working directory
getwd()
```

```
## [1] "/home/guest/R/EDA-Spring2023"
```

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

# 3. Import datasets
USGS.flow.data <- read.csv("./Data/Processed/USGS_Site02085000_Flow_Processed.csv",stringsAsFactors = T)

#View(USGS.flow.data)
# Alternate option: click on data frame in Environment tab

colnames(USGS.flow.data)
```

```
##  [1] "agency_cd"                "site_no"
##  [3] "datetime"                 "discharge.max"
##  [5] "discharge.max.approval"   "discharge.min"
##  [7] "discharge.min.approval"   "discharge.mean"
##  [9] "discharge.mean.approval"  "gage.height.max"
## [11] "gage.height.max.approval" "gage.height.min"
## [13] "gage.height.min.approval" "gage.height.mean"
## [15] "gage.height.mean.approval"
```

```r
str(USGS.flow.data)
```

```
## 'data.frame':    33690 obs. of  15 variables:
##  $ agency_cd                : Factor w/ 1 level "USGS": 1 1 1 1 1 1 1 1 1 1 ...
##  $ site_no                  : int  2085000 2085000 2085000 2085000 2085000 2085000 2085000 2085000 20
##  $ datetime                 : Factor w/ 33690 levels "1927-10-01","1927-10-02",..: 1 2 3 4 5 6 7 8 9
##  $ discharge.max            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ discharge.max.approval   : Factor w/ 3 levels "","A","P": 1 1 1 1 1 1 1 1 1 1 ...
##  $ discharge.min            : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ discharge.min.approval   : Factor w/ 3 levels "","A","P": 1 1 1 1 1 1 1 1 1 1 ...
##  $ discharge.mean           : num  39 39 39 39 39 39 39 39 39 39 ...
##  $ discharge.mean.approval  : Factor w/ 4 levels "","A","A:e","P": 2 2 2 2 2 2 2 2 2 2 ...
##  $ gage.height.max          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ gage.height.max.approval : Factor w/ 3 levels "","A","P": 1 1 1 1 1 1 1 1 1 1 ...
##  $ gage.height.min          : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ gage.height.min.approval : Factor w/ 3 levels "","A","P": 1 1 1 1 1 1 1 1 1 1 ...
##  $ gage.height.mean         : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ gage.height.mean.approval: Factor w/ 3 levels "","A","P": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
dim(USGS.flow.data)
```

```
## [1] 33690    15
```

```r
# Check our date column
class(USGS.flow.data$datetime)
```

```
## [1] "factor"
```

```r
USGS.flow.data$datetime <- as.Date(USGS.flow.data$datetime, format = "%Y-%m-%d")
class(USGS.flow.data$datetime)
```

```
## [1] "Date"
```

Visualization for Data Exploration

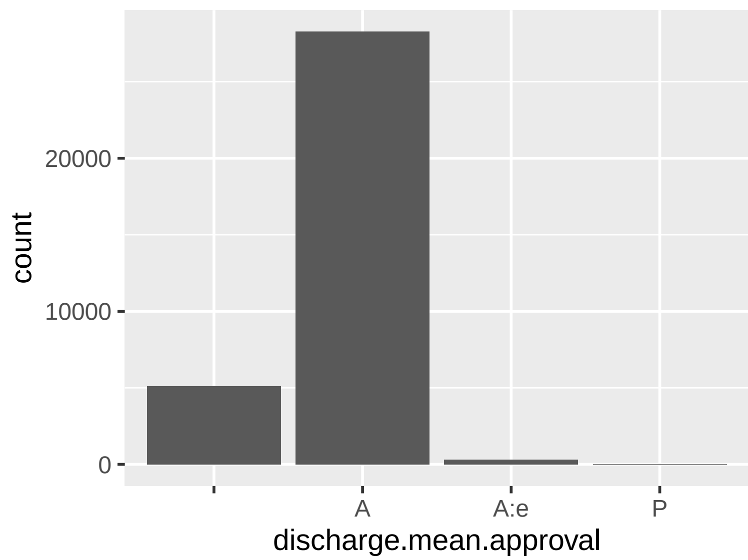
Although the `summary()` function is helpful in getting an idea of the spread of values in a numeric dataset, it can be useful to create visual representations of the data to help form hypotheses and direct downstream data analysis. Below is a summary of the useful types of graphs for data exploration.

Note: each of these approaches utilize the package "ggplot2". We will be covering the syntax of ggplot in a later lesson, but for now you should familiarize yourself with the functionality of what each command is doing.

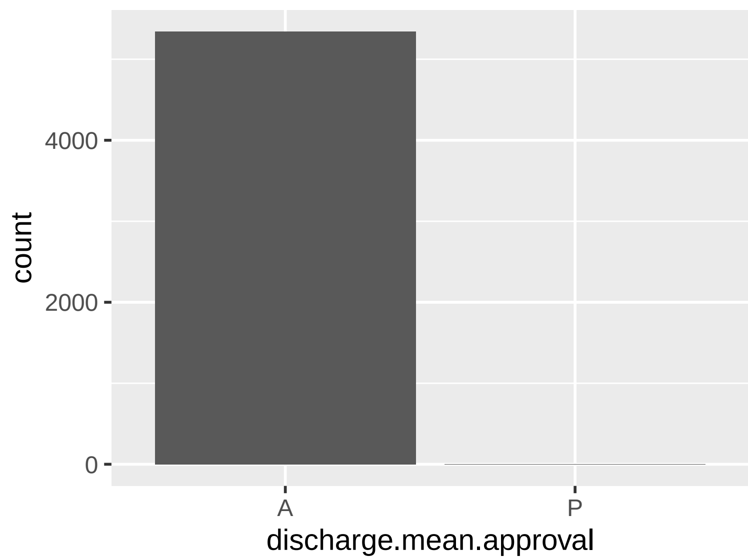
Bar Chart (function: geom_bar)

Visualize count data for categorical variables.

```r
ggplot(USGS.flow.data, aes(x = discharge.mean.approval)) +
  geom_bar()
```




```
ggplot(na.omit(USGS.flow.data), aes(x = discharge.mean.approval)) + geom_bar()
```



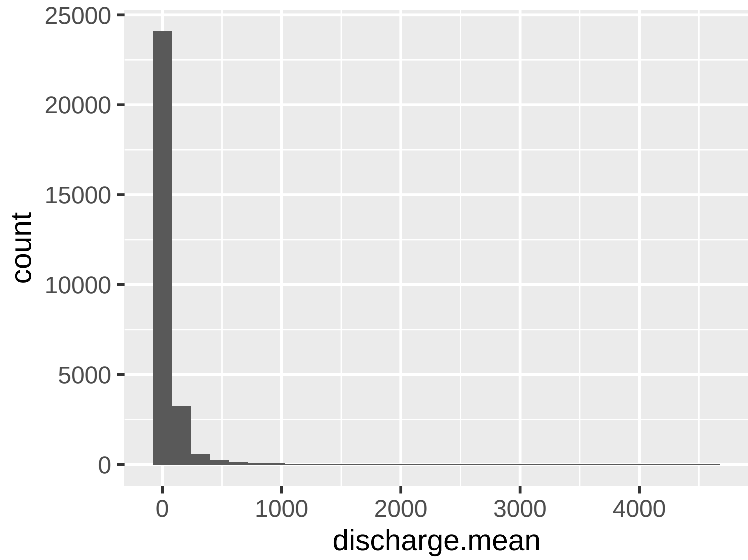
Histogram (function: geom_histogram)

Visualize distributions of values for continuous numerical variables. What is happening in each line of code? Insert a comment above each line.

```
# Histogram with default 30 bins, between 5 and 20 is good
ggplot(USGS.flow.data) +
  geom_histogram(aes(x = discharge.mean))
```

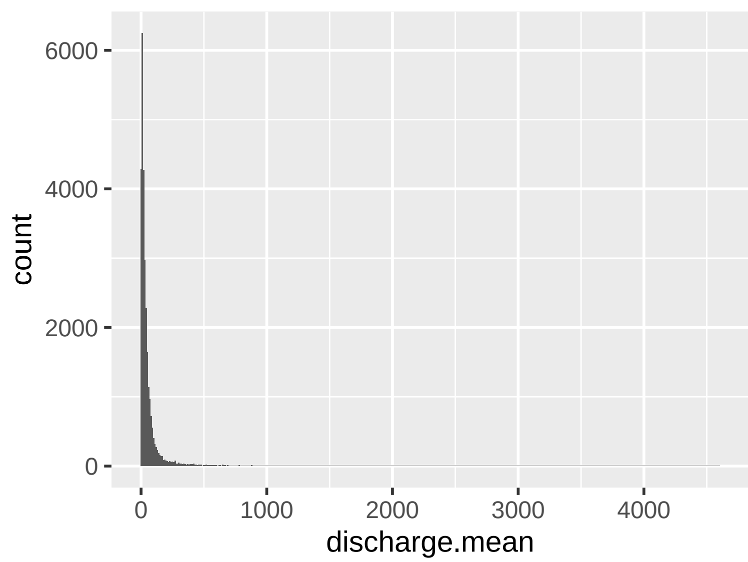
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 5108 rows containing non-finite values (`stat_bin()`).
```



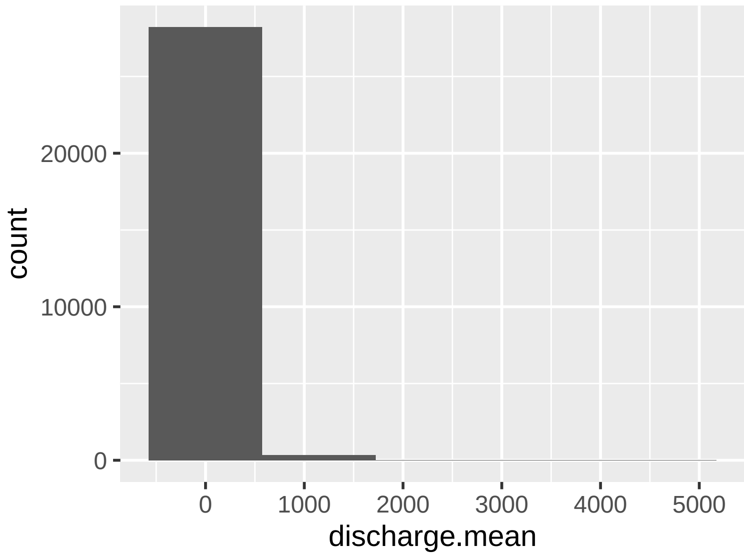

```
# Command that goes beyond default binwidth, smoother data
ggplot(USGS.flow.data) +
  geom_histogram(aes(x = discharge.mean), binwidth = 10)
```

```
## Warning: Removed 5108 rows containing non-finite values (`stat_bin()`).
```



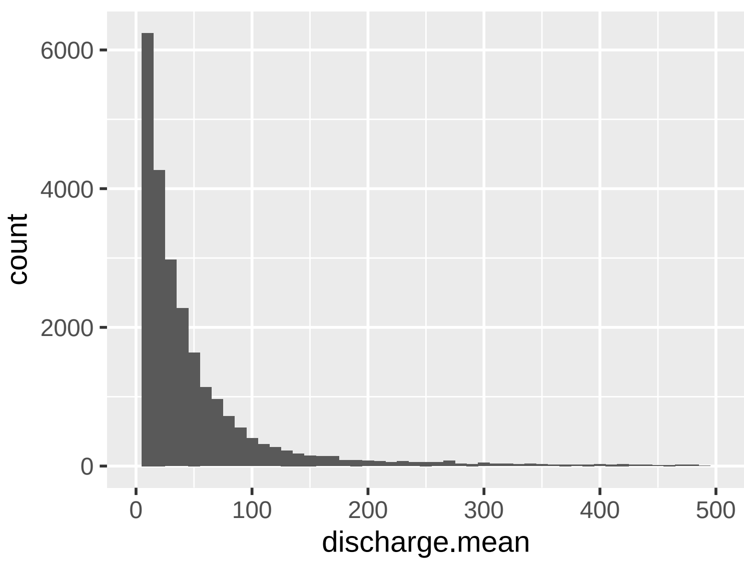

```
#Just specifying the number of bins
ggplot(USGS.flow.data) +
  geom_histogram(aes(x = discharge.mean), bins = 5)
```

```
## Warning: Removed 5108 rows containing non-finite values (`stat_bin()`).
```




```
#Exploring discharge mean, makes x go from 0 to 500, zooming in to histogram with scale_x_continuous
ggplot(USGS.flow.data, aes(x = discharge.mean)) +
  geom_histogram(binwidth = 10) +
  scale_x_continuous(limits = c(0, 500))
```

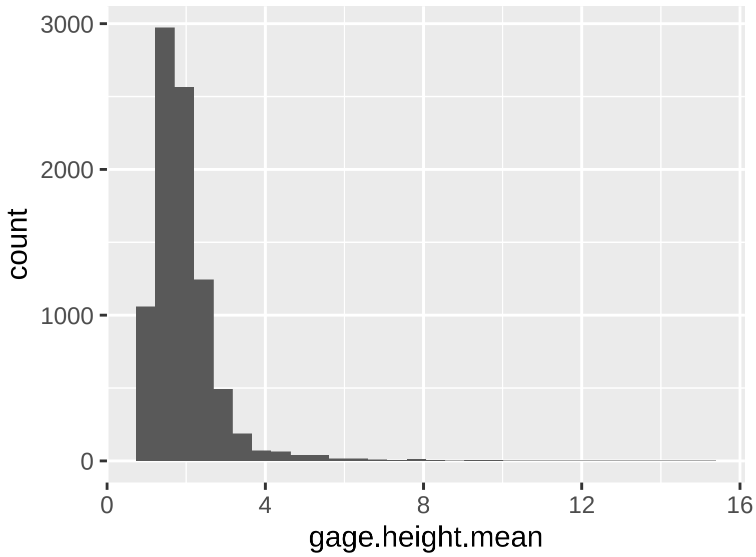
```
## Warning: Removed 5577 rows containing non-finite values (`stat_bin()`).

## Warning: Removed 2 rows containing missing values (`geom_bar()`).
```




```
#Shows gauge height that shows a peak past 0
ggplot(USGS.flow.data) +
  geom_histogram(aes(x = gage.height.mean))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 24870 rows containing non-finite values (`stat_bin()`).
```

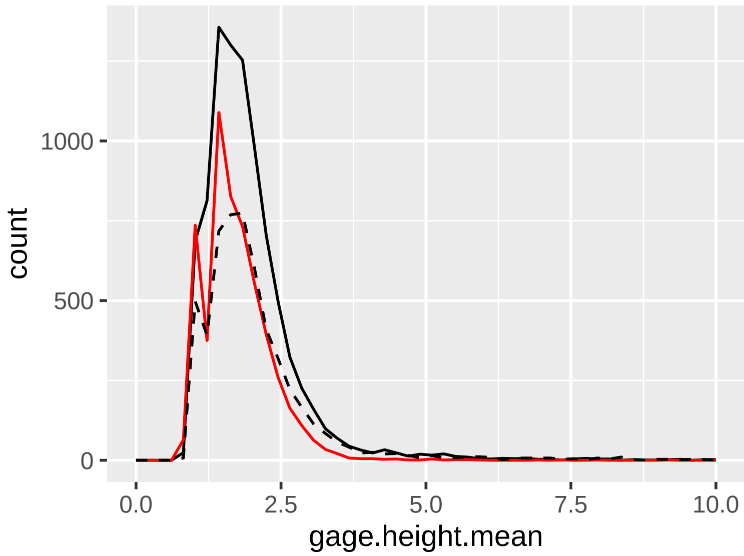


Frequency line graph (function: geom_freqpoly)

An alternate to a histogram is a frequency polygon graph (distributions of values for continuous numerical variables). Instead of displaying bars, counts of continuous variables are displayed as lines. This is advantageous if you want to display multiple variables or categories of variables at once.

```
#Line graph that shows the mean, min, and max
ggplot(USGS.flow.data) +
  geom_freqpoly(aes(x = gage.height.mean), bins = 50) +
  geom_freqpoly(aes(x = gage.height.min), bins = 50, color = "red") +
  geom_freqpoly(aes(x = gage.height.max), bins = 50,  lty = 2) +
  scale_x_continuous(limits = c(0, 10))
```

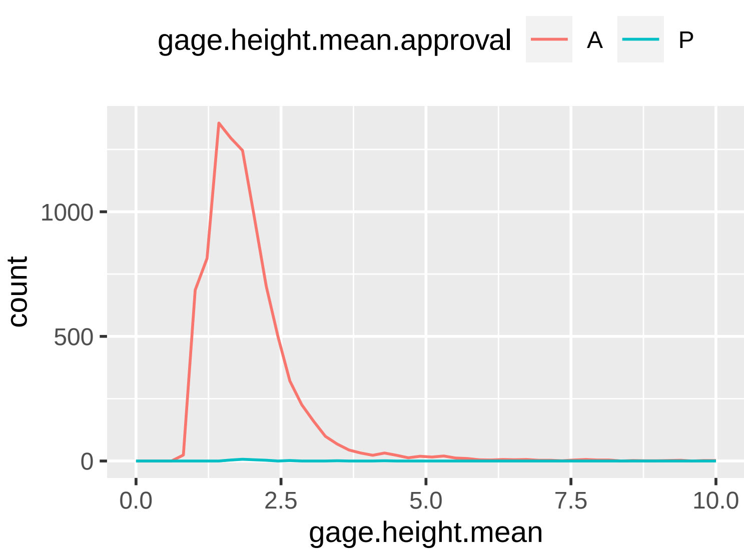
```
## Warning: Removed 24887 rows containing non-finite values (`stat_bin()`).

## Warning: Removed 28229 rows containing non-finite values (`stat_bin()`).

## Warning: Removed 28266 rows containing non-finite values (`stat_bin()`).

## Warning: Removed 2 rows containing missing values (`geom_path()`).
## Removed 2 rows containing missing values (`geom_path()`).
## Removed 2 rows containing missing values (`geom_path()`).
```




```r
#Color values by mean approval, can set color to a different categorical value
ggplot(USGS.flow.data) +
  geom_freqpoly(aes(x = gage.height.mean, color = gage.height.mean.approval), bins = 50) +
  scale_x_continuous(limits = c(0, 10)) +
  theme(legend.position = "top")
```

```
## Warning: Removed 24887 rows containing non-finite values (`stat_bin()`).
```

```
## Warning: Removed 4 rows containing missing values (`geom_path()`).
```



Box-and-whisker plots (function: geom_boxplot, geom_violin)

A box-and-whisker plot is yet another alternative to histograms (distributions of values for continuous numerical variables). These plots consist of:

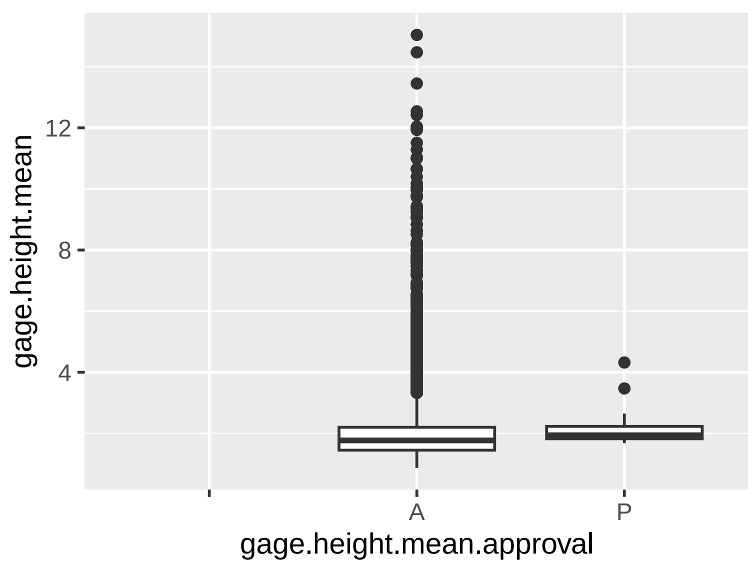
- A box from the 25th to the 75th percentile of the data, called the interquartile range (IQR).

- A bold line inside the box representing the median value of the data. Whether the median is in the center or off to one side of the IQR will give you an idea about the skewness of your data.
- A line outside of the box representing values falling within 1.5 times the IQR.
- Points representing outliers, values that fall outside 1.5 times the IQR.

An alternate option is a violin plot, which displays density distributions, somewhat like a hybrid of the box-and-whiskers and the frequency polygon plot.

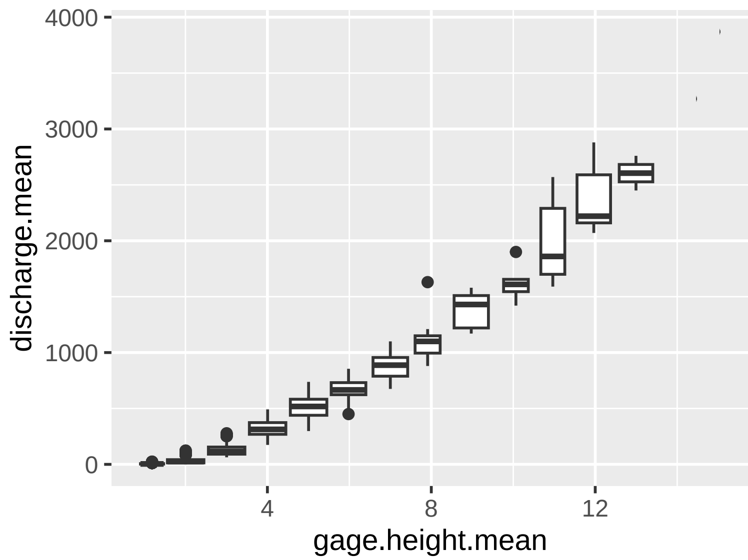
```
#Box and whisker plot of mean gage height values if approved records are subset
ggplot(USGS.flow.data) +
  geom_boxplot(aes(x = gage.height.mean.approval, y = gage.height.mean))
```

```
## Warning: Removed 24870 rows containing non-finite values (`stat_boxplot()`).
```



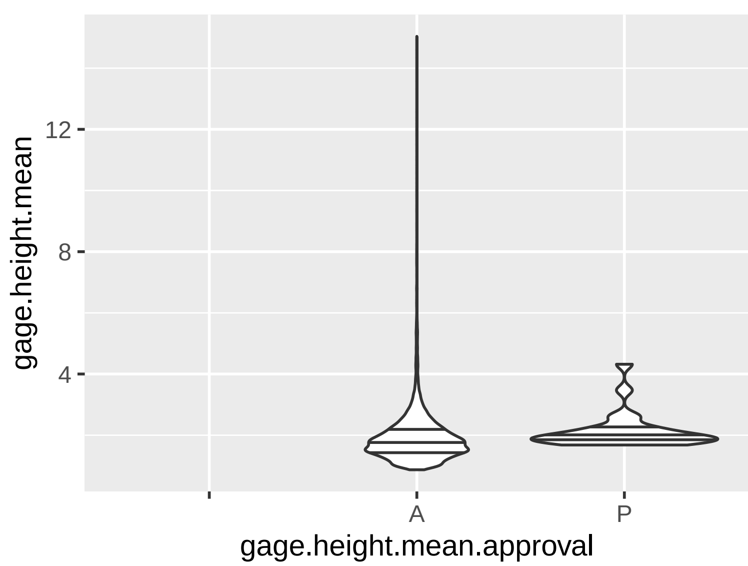

```
#Sets Y to discharge mean, slices data with cut_width command
ggplot(USGS.flow.data) +
  geom_boxplot(aes(x = gage.height.mean, y = discharge.mean, group = cut_width(gage.height.mean, 1)))
```

```
## Warning: Removed 24870 rows containing missing values (`stat_boxplot()`).
```




```
#Adds an extra distribution
ggplot(USGS.flow.data) +
  geom_violin(aes(x = gage.height.mean.approval, y = gage.height.mean),
              draw_quantiles = c(0.25, 0.5, 0.75))
```

```
## Warning: Removed 24870 rows containing non-finite values (`stat_ydensity()`).
```

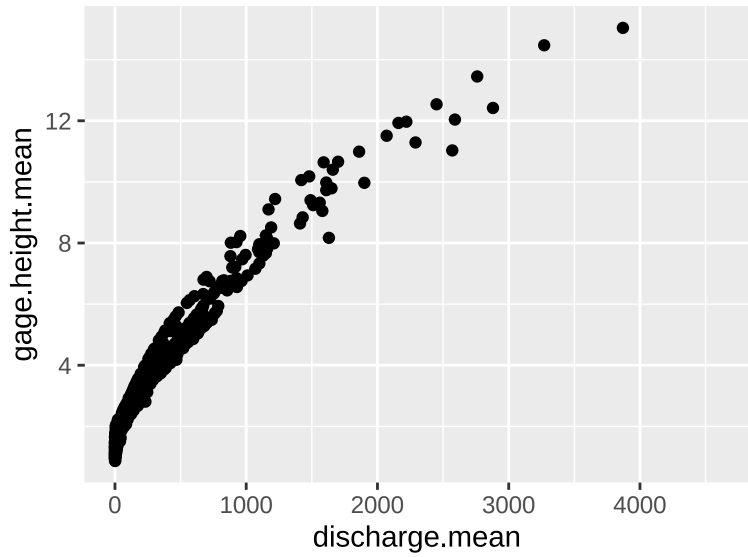


Scatterplot (function: geom_point)

Visualize relationships between continuous numerical variables.

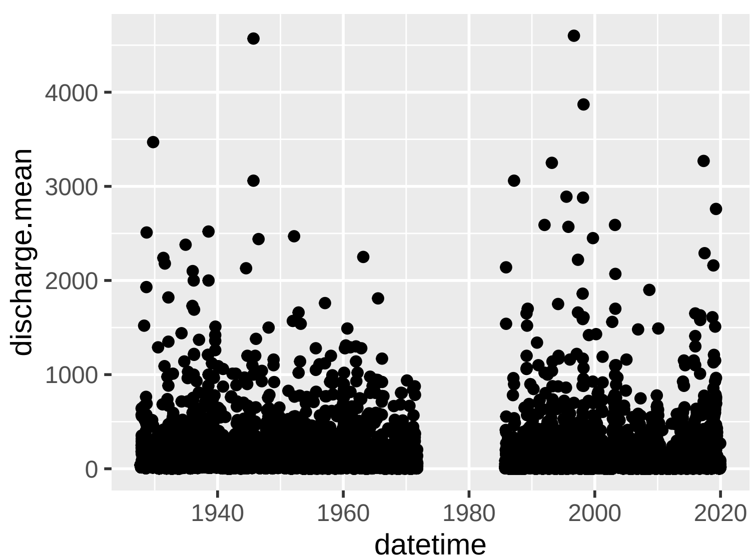
```
ggplot(USGS.flow.data) +
  geom_point(aes(x = discharge.mean, y = gage.height.mean))
```

```
## Warning: Removed 24870 rows containing missing values (`geom_point()`).
```




```r
ggplot(USGS.flow.data) +
  geom_point(aes(x = datetime, y = discharge.mean))
```

```
## Warning: Removed 5108 rows containing missing values (`geom_point()`).
```



Question: under what circumstances would it be beneficial to use each of these graph types (bar plot, histogram, frequency polygon, box-and whisker, violin, scatterplot)?

> Answer: Scatterplots would be useful for showing each of the rows in the data as a point shows a trend. The second scatterplot shows the mean discharge at different times.

Ending discussion

What did you learn about the USGS discharge dataset today? What separate insights did the different graph types offer? > Answer: I learned how far back the USGS discharge dataset goes–I didn't know that they've been collecting data for so long! The different graphs represented the data in such different ways–some were more intuitive to interpret and relevant to our data than others!

How can multiple options for data exploration inform our understanding of our data?

> Answer: The options for data exploration and variety of charts were facinating to explore! Some of the were easier to interpret and more inuitive than others, and some of them seemed more relevant for our data than others. It made me think more about how data is represented, and how this representation can completely shape how the data is interpreted and what the audience takes away from it.

Do you see any patterns in the USGS data for the Eno River? What might be responsible for those patterns and/or relationships?

> Answer: It seemed like the gage height mean and the discharge mean were positively correlated. Based on the scatter plot, it also seemed like the discharge mean has remained similiar throughout the years, although there is a gap in the data stretching from the early 1970s to mid 1980s.The gage height mean and discharge mean correlation makes sense, as a gage that has a greater amount of discharge would also likely be higher.