PairProgramming_Exploratory_Analysis_Distributions

HD Sheets

2024-08-07

Pair Programming: Exploratory Data Analysis, Distributions

HD Sheets August 9, 2024 Checked 01/03/2025

Student Name: Ryan Waterman Teammate Name: Nicholas Perry

See, Wickham chapter 3

Yakir, chapter 3 https://eleuven.github.io/statthink/ChapDescriptiveStat.html#displaying-data (https://eleuven.github.io/statthink/ChapDescriptiveStat.html#displaying-data)

Diez, Chapter 4

Things to watch for in this exercise

-how do we get basic statistics on univariate data

-what do these statistics tell us about the shape of the data

-what are some basic plots we can use in exploratory data analysis to develop an understanding of our data

Four simple distributions

These are four simple models, generated using random values

These are *synthetic* or *simulated* data sets, we will use them to develop some understanding of the summary statistics and graphics. We are using R's ability to create simulations as a way to develop our understanding.

The values we will look at are from classic algebraically derived models of distributions. There are 20-20 such models in common use, you will see a dozen of them regularly. Always read the Wikipedia article on them if you run into a new one.

The ones we will look at today are all continuous, we'll look at discrete distributions later

Today's menu of distributions

-Continuous Uniform -Gaussian -Exponential -Weibull

Uniform Distribution

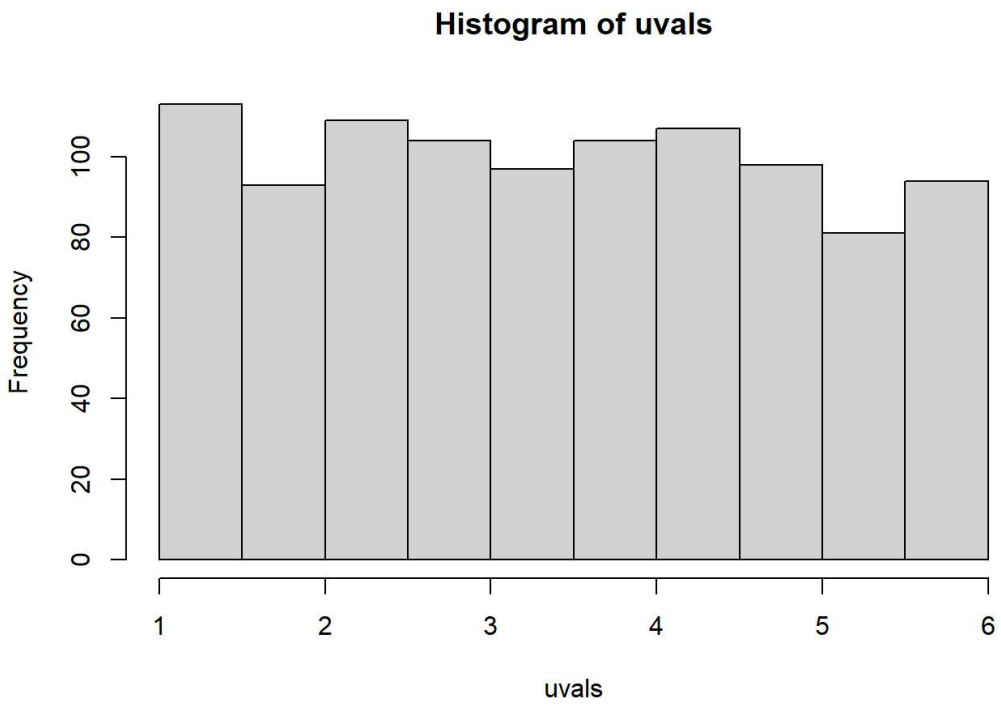
All values in the range are equally likely to occur

-rolling dice, picking cards from a deck, where along a hallway my cat drops a toy- no values are more probable than others

```
#create a uniform distribution
uvals=runif(1000,1,6)

# graph this using the simple histogram plot function
# We'll see the better one in ggplot later

hist(uvals)
```



Summary statistics for the Uniform distribution data

Here is how to calculate these values

We can call r functions for each of our univariate variables

```
library("parameters")
library("moments")
```

```
##
## Attaching package: 'moments'
```

```
## The following objects are masked from 'package:parameters':
##
##     kurtosis, skewness
```

```
mean(uvals)
```

```
## [1] 3.423628
```

```
median(uvals)
```

```
## [1] 3.442089
```

```
var(uvals)
```

```
## [1] 2.047253
```

```
sd(uvals)
```

```
## [1] 1.430823
```

```
skewness(uvals)
```

```
## [1] 0.0536558
```

```
kurtosis(uvals)
```

```
## [1] 1.848063
```

The Gaussian Distribution

aka “normal”, “bell curve”

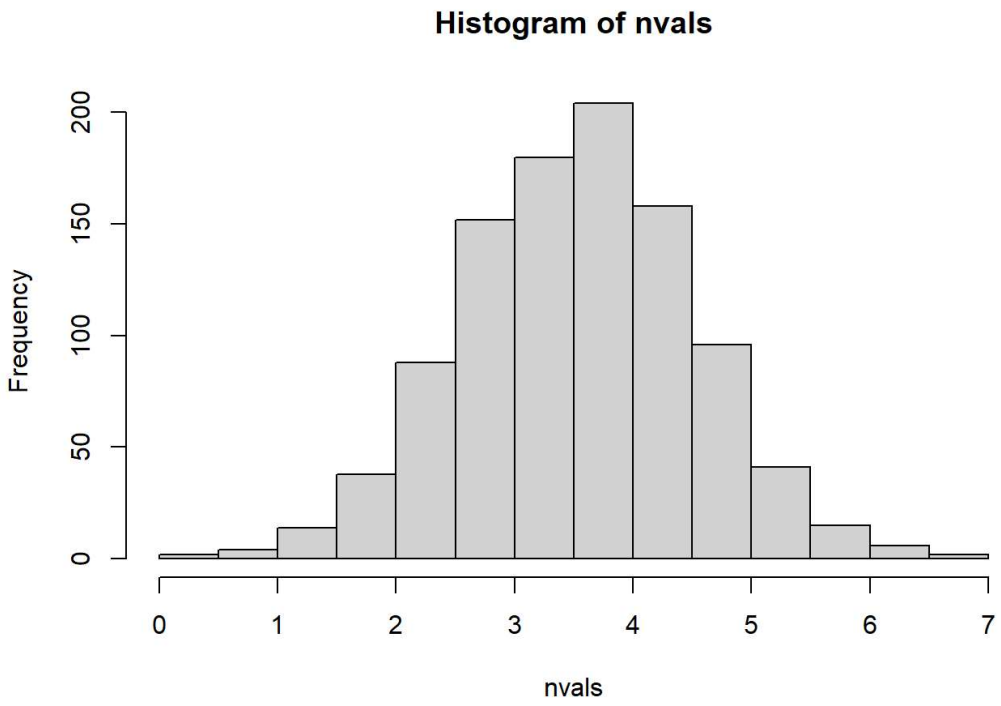
-symmetric about the mean -width described well by the standard deviation

-Mathematically derived as a approximation for any symmetric distribution with a strong central peak at the mean

-Useful for variables where many other variables determine the outcome -human heights and weights, other biological responses -averages of composite variables

-Often used when we don’t know what else to do

```
nvals=rnorm(1000,3.5,1)
hist(nvals)
```




```
mean(nvals)
```

```
## [1] 3.535113
```

```
median(nvals)
```

```
## [1] 3.566058
```

```
var(nvals)
```

```
## [1] 0.9910326
```

```
sd(nvals)
```

```
## [1] 0.9955062
```

```
skewness(nvals)
```

```
## [1] -0.05095665
```

```
kurtosis(nvals)
```

```
## [1] 3.12716
```

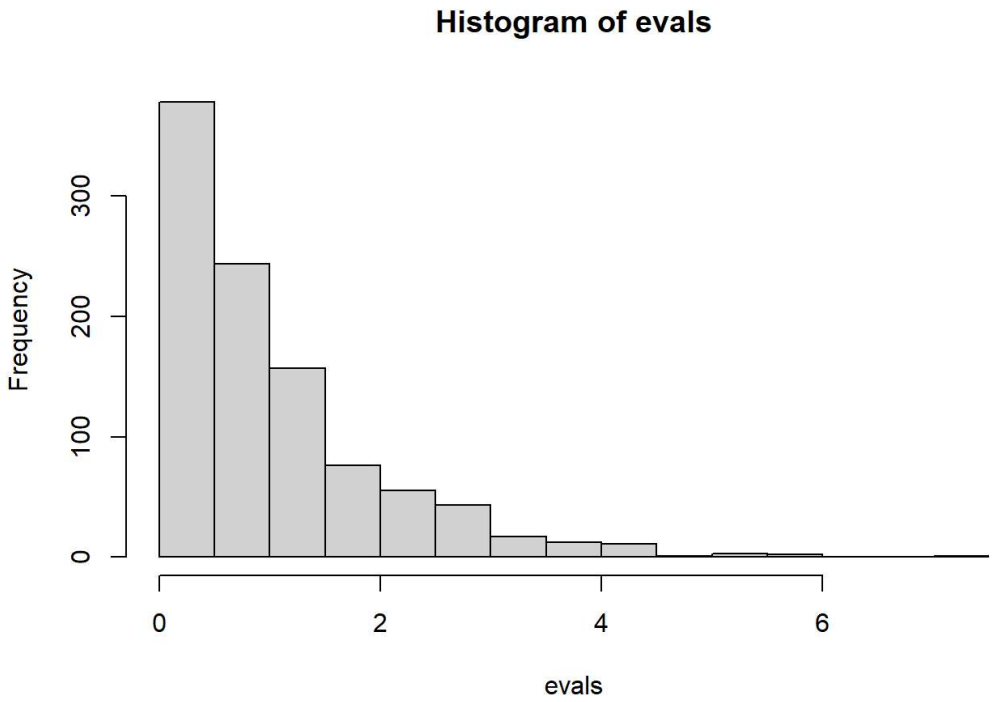
Exponential distributions

Used when the number of instances decays exponential relative to the starting value.

It has a long positive or right tail, meaning a few cases at very high values and less at low values

-Many economic variables show this type of trend, bank account balances, house prices family net income, many values below the mean, less above, but very large values above the mean

```
evals=rexp(1000,1)
hist(evals)
```




```
mean(evals)
```

```
## [1] 1.018875
```

```
median(evals)
```

```
## [1] 0.7250316
```

```
var(evals)
```

```
## [1] 0.9629382
```

```
sd(evals)
```

```
## [1] 0.9812941
```

```
skewness(evals)
```

```
## [1] 1.761232
```

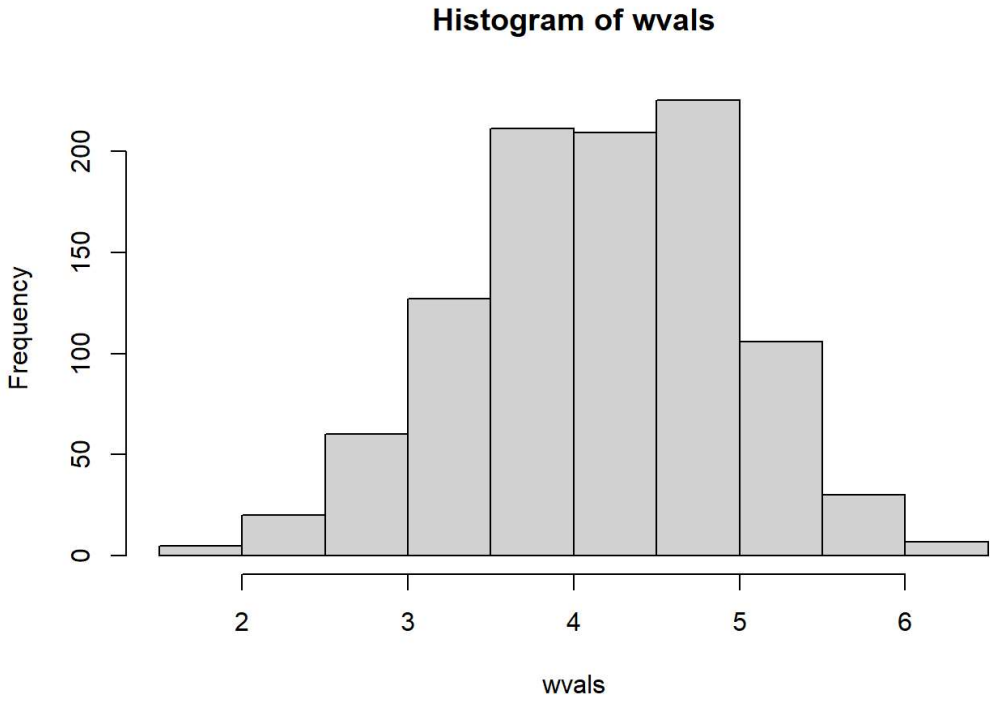
```
kurtosis(evals)
```

```
## [1] 7.145858
```

Weibull

The Weibull distribution can take on many shapes, depending on the parameter values chosen

```
wvals=rweibull(1000,6,4.5)
hist(wvals)
```



```
mean(wvals)
```

```
## [1] 4.155038
```

```
median(wvals)
```

```
## [1] 4.165076
```

```
var(wvals)
```

```
## [1] 0.6551561
```

```
sd(wvals)
```

```
## [1] 0.8094171
```

```
skewness(wvals)
```

```
## [1] -0.2262465
```

```
kurtosis(wvals)
```

```
## [1] 2.795525
```

Putting our data into a Data Frame

This will allow for easier plotting

We want to use ggplot for high quality plots with a lot of formatting options

ggplot requires that the data be in a data frame. So we will put our four sets of simulated data into a dataframe

```
distrib_df=data.frame(uniform=uvals, normal=nvals,exponential=evals, weibull=wvals)
```

Check on the data frame, to see if it is what we meant to create

our tools to do this are head(), str() and summary()

```
head(distrib_df)
```

```
##    uniform   normal exponential  weibull
## 1 1.665171 4.807140   1.6356369 4.497493
## 2 5.062274 2.457588   0.4468564 4.707275
## 3 5.605332 5.479367   1.9128594 3.886404
## 4 1.545346 1.654011   1.6979683 3.204187
## 5 1.395065 5.227305   0.6557815 4.607905
## 6 2.329592 4.853766   2.3287444 3.141982
```

```
str(distrib_df)
```

```
## 'data.frame':    1000 obs. of  4 variables:
##  $ uniform    : num  1.67 5.06 5.61 1.55 1.4 ...
##  $ normal     : num  4.81 2.46 5.48 1.65 5.23 ...
##  $ exponential: num  1.636 0.447 1.913 1.698 0.656 ...
##  $ weibull    : num  4.5 4.71 3.89 3.2 4.61 ...
```

Summary shows us a lot of measures

Notice we get the mean, median and range (min, max) and also the 1st quartile (the value at which 25% is less that the quartile) and the 3rd quartile (where 25% are greater than this point)

“Quartile” refers to spliting data up into 4 groups, with boundaries at the 1st quartile (25%), the 2nd or median (50%) and the 3rd quartile (75%)

Quartile is one form of a “quantile”. Quantiles can be at any percentage, 5%,10% 90%, 95%, 99% etc

```
summary(distrib_df)
```

```
##     uniform          normal       exponential          weibull      
##  Min.   :1.014   Min.   :0.425   Min.   :0.000333   Min.   :1.512  
##  1st Qu.:2.188   1st Qu.:2.868   1st Qu.:0.304671   1st Qu.:3.620  
##  Median :3.442   Median :3.566   Median :0.725032   Median :4.165  
##  Mean   :3.424   Mean   :3.535   Mean   :1.018875   Mean   :4.155  
##  3rd Qu.:4.607   3rd Qu.:4.188   3rd Qu.:1.438562   3rd Qu.:4.738  
##  Max.   :5.991   Max.   :6.933   Max.   :7.398248   Max.   :6.107
```

So summary gives us a lot of information about the 4 univariate distributions currently in our data frame

There is a fancier version of summary in the package called skimr

Here’s what it looks like

```
library('skimr')
skim(distrib_df)
```

Data summary

Name	distrib_df
Number of rows	1000
Number of columns	4
Column type frequency:	
numeric	4
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
uniform	0	1	3.42	1.43	1.01	2.19	3.44	4.61	5.99	▇▇▇▇▇
normal	0	1	3.54	1.00	0.43	2.87	3.57	4.19	6.93	▁▃▇▂▁
exponential	0	1	1.02	0.98	0.00	0.30	0.73	1.44	7.40	▇▂▁▁▁
weibull	0	1	4.16	0.81	1.51	3.62	4.17	4.74	6.11	▁▃▇▂▁

Using summary() or skim() is a quick way to get many of the “standard” descriptive statistics

It is probably safe to say you should always use both head() and summary() on a data set you haven’t seen before.

Plotting with GGPlot

Histograms

In ggplot, data must always be in a data frame to plot it

the aes() section specifies which data from the data frame should be used for the x and y axes on a plot and for coloring and setting symbols

We will look at some ggplots for the uniform distribution

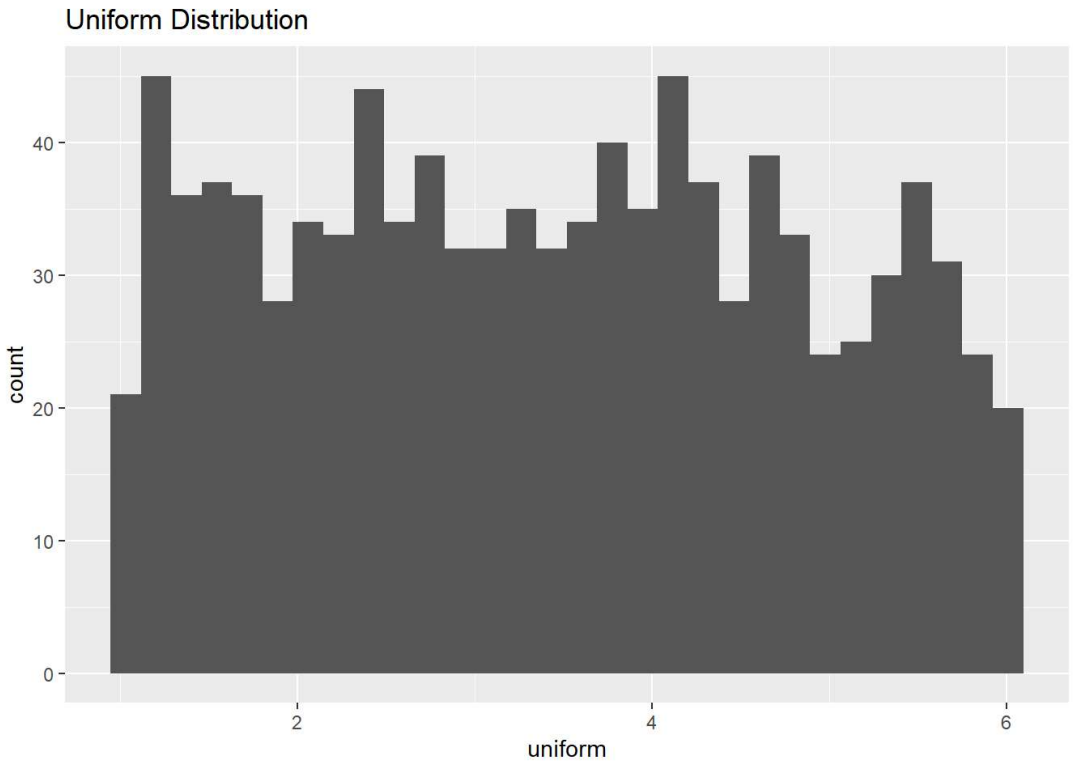
Note, I say ggplot, but the package is named ggplot2

the term +geom_histogram adds an “item” to the plot, in this case a histogram

ggtitle- adds a title

```
library('ggplot2')
ggplot(distrib_df, aes(x=uniform))+geom_histogram()+ggtitle('Uniform Distribution')
```

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



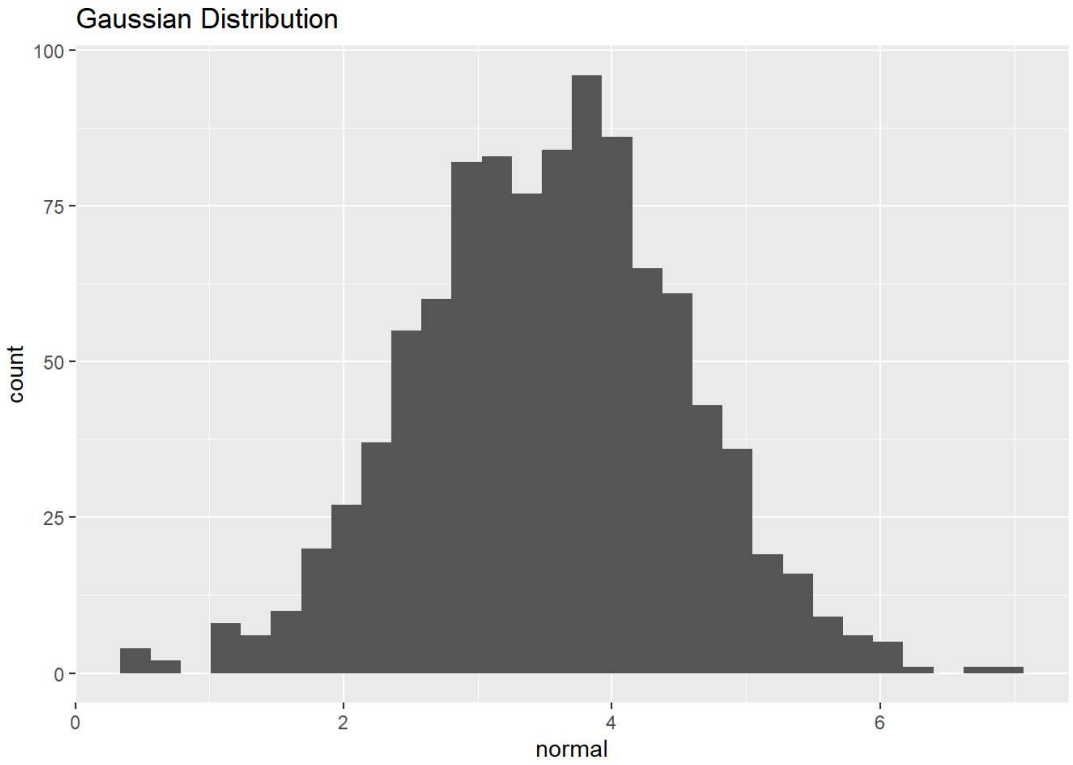
Question/Action

Cut and paste to create a ggplot histogram of the normal or Gaussian data

Remember to change the title

```
ggplot(distrib_df, aes(x=normal))+geom_histogram()+ggtitle('Gaussian Distribution')
```

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



What is very different about the normal or gaussian relative to the Uniform? Are both symmetric?

The Gaussian Distribution forms a bell curve, where a majority of the values fall on or near the mean, and exponentially decreases as the value moves away from the mean. They are both symmetric about the mean.

BoxPlot

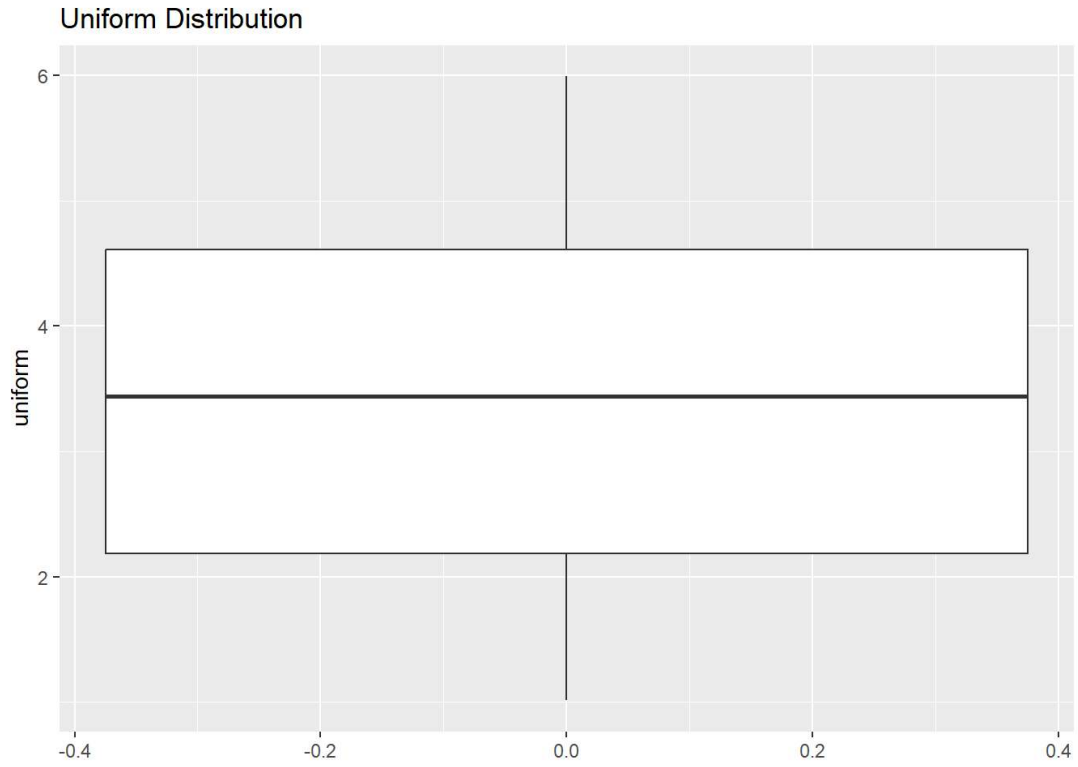
This is a super handy plot, it shows a box that is the 25%-75% quartile, with whiskers representing the 5% lower and 95% upper quantiles

“Outliers” or extreme values are shown as dots

https://en.wikipedia.org/wiki/Box_plot (https://en.wikipedia.org/wiki/Box_plot)

Notice- I just cut and pasted and changed to geom_boxplot also the data is now y=uniform to make the plot vertical

```
library('ggplot2')
ggplot(distrib_df, aes(y=uniform))+geom_boxplot()+ggtitle('Uniform Distribution')
```

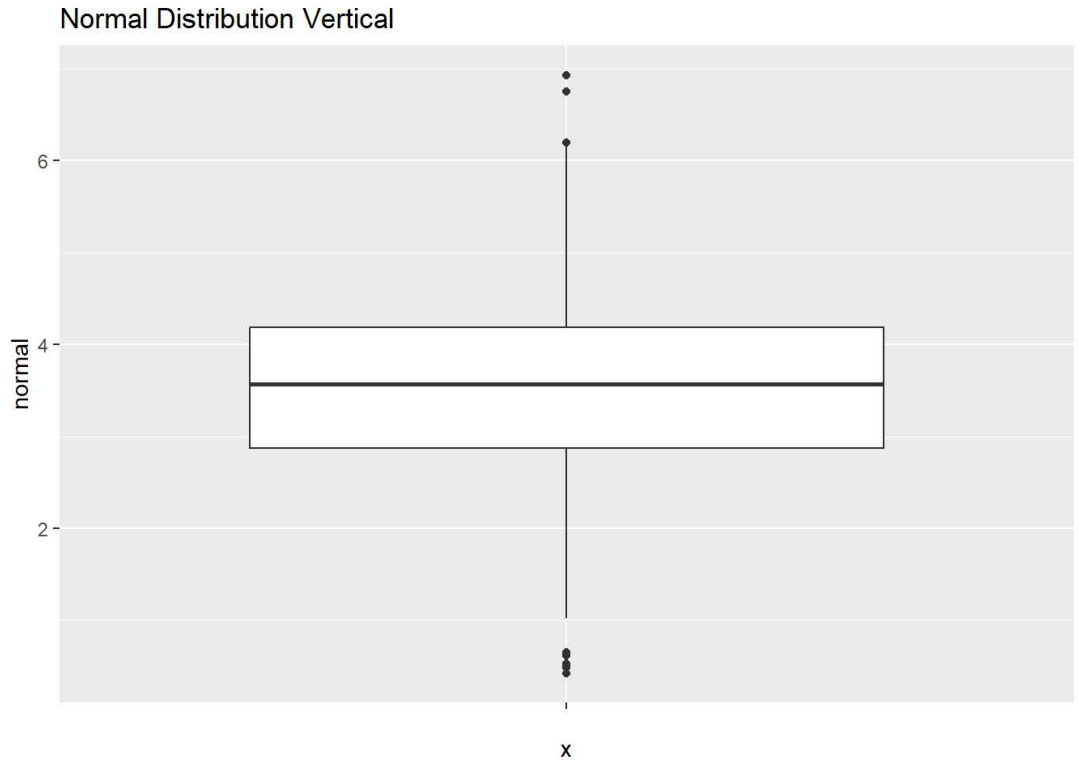


 # Question/Action

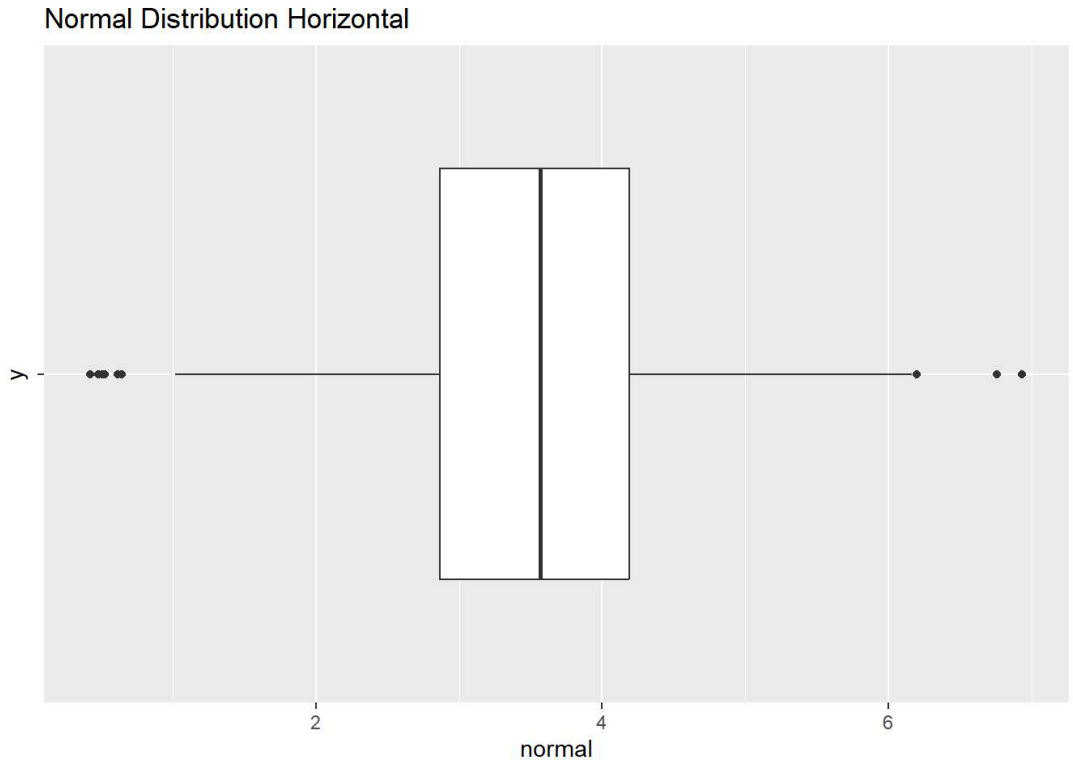
Create a box plot for some other of the 4 distributions, label it, produce both vertical (y=) and horizontal (x=) versions of it

Discuss how the other distributions look different from the uniform when we display them with a boxplot

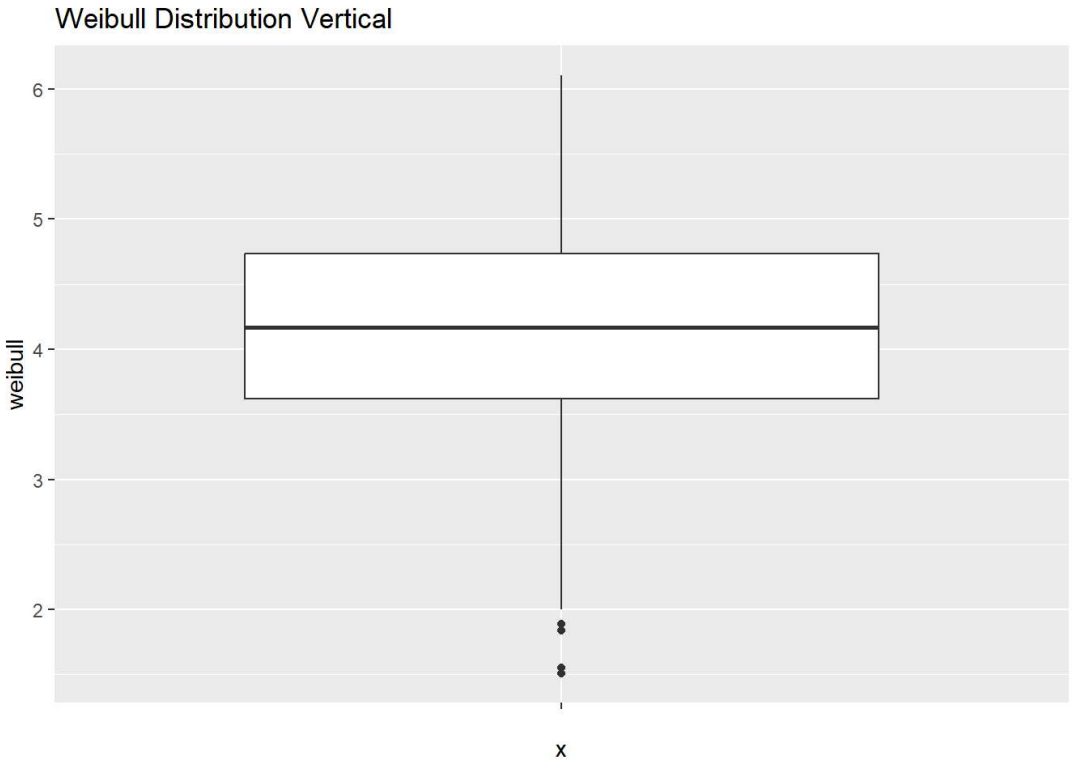
```
ggplot(distrib_df, aes(x="",y=normal))+geom_boxplot()+ggtitle('Normal Distribution Vertical')
```



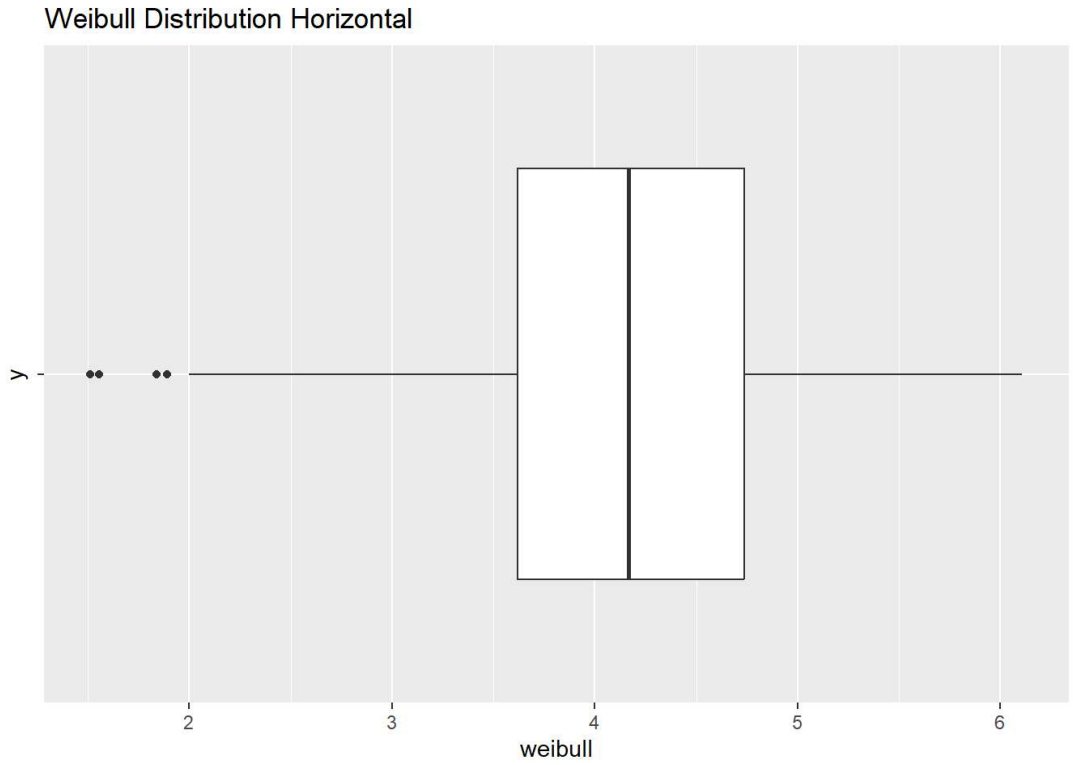

```
ggplot(distrib_df, aes(x=normal,y=""))+geom_boxplot()+ggtitle('Normal Distribution Horizontal')
```




```
ggplot(distrib_df, aes(x="",y=weibull))+geom_boxplot()+ggtitle('Weibull Distribution Vertical')
```




```
ggplot(distrib_df, aes(x=weibull,y=""))+geom_boxplot()+ggtitle('Weibull Distribution Horizontal')
```



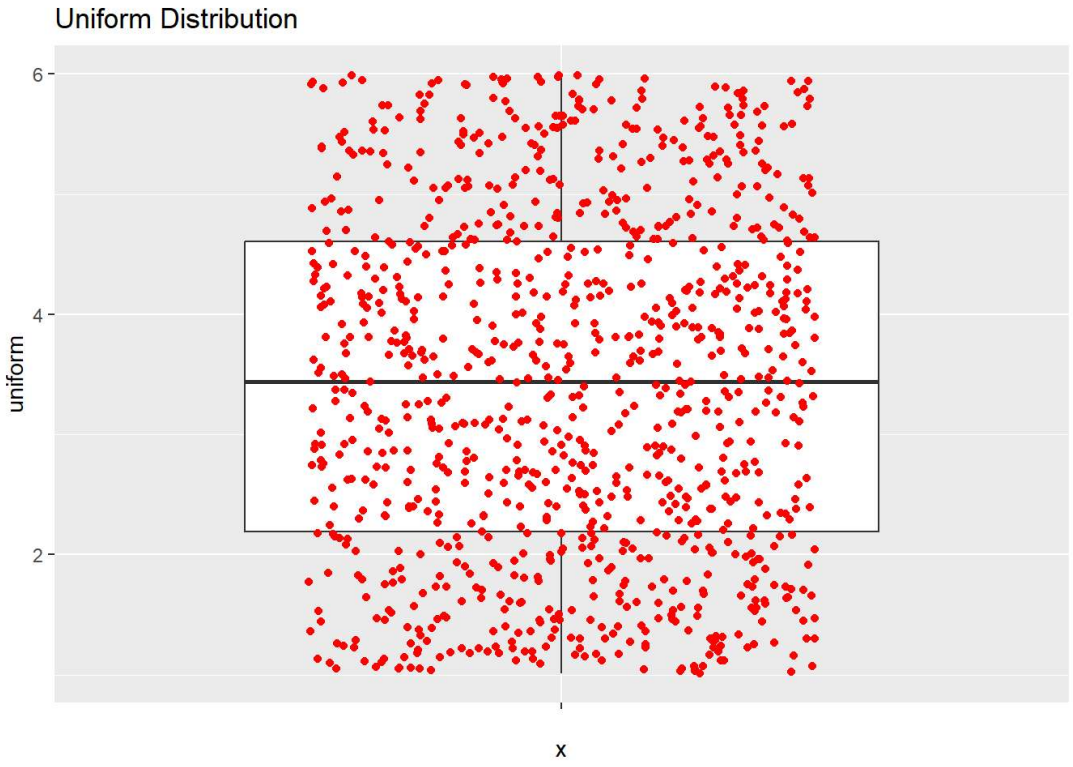
The normal distribution has outliers on either side of the IQR, and the Weibull has outliers on one side of the IQR. These both differ from the normal dustribution, which does not feature any outliers and the IQR is evenly distributed. The IQR for the weibull and normal distributions are contracted compared to the uniform distribution.

Adding data points to the plot

We can plot the density of points on top of the box, so we can see the pattern of points along the y axis

The points are “jittered” along the x axis with width =0.3 to wide the plot of points

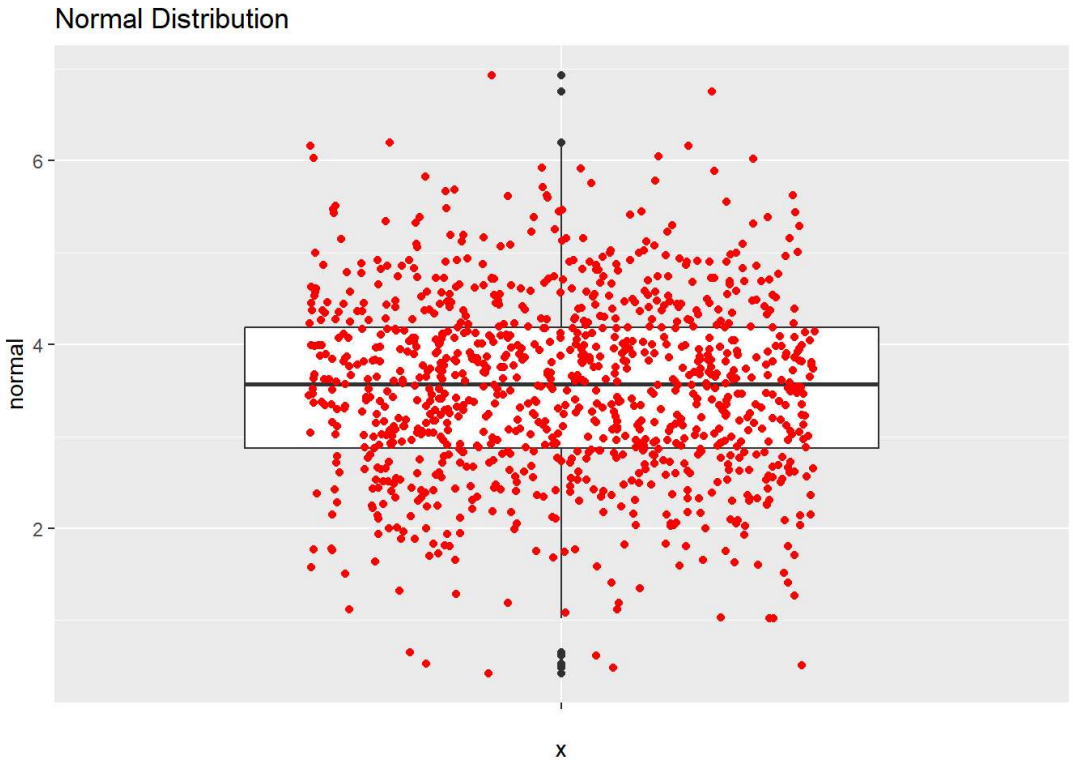
```
ggplot(distrib_df, aes(x="",y=uniform))+geom_boxplot()+ggtitle('Uniform Distribution')+geom_rug(sides='1')+geom_jitter(wi
dth=0.3, height=0,col="red")
```



Question/Action

Repeat this for your favorite distribution

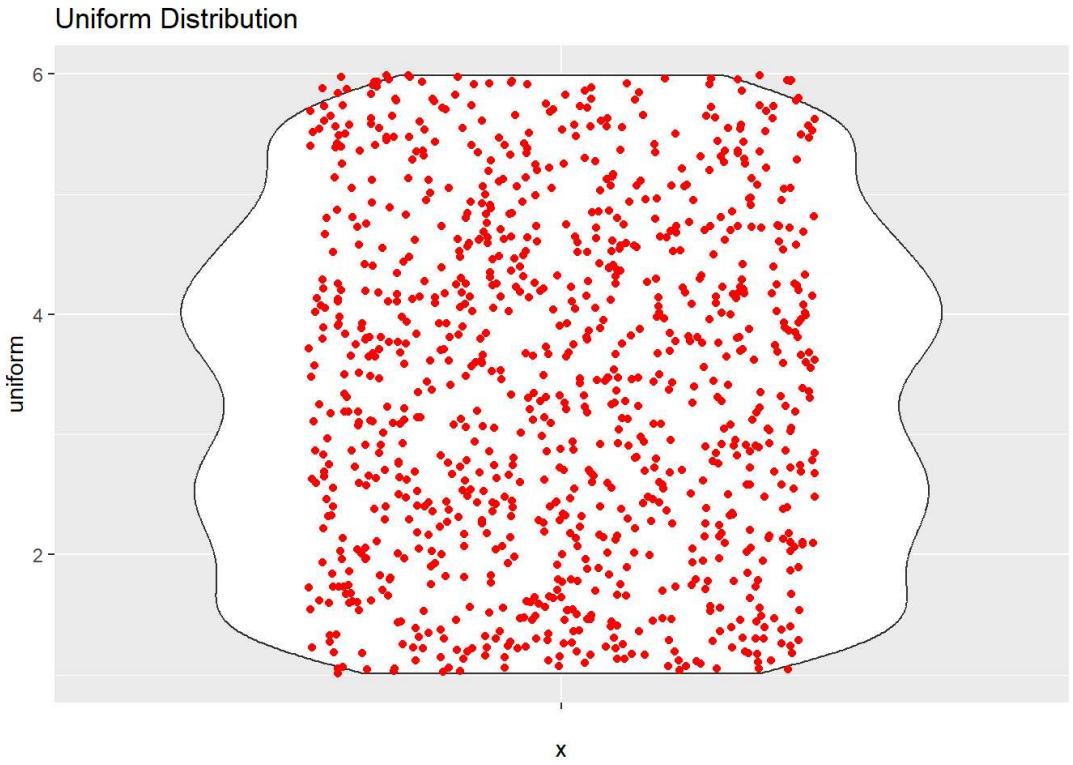
```
ggplot(distrib_df, aes(x="",y=normal))+geom_boxplot()+ggtitle('Normal Distribution')+geom_rug(sides='1')+geom_jitter(widt
h=0.3, height=0,col="red")
```



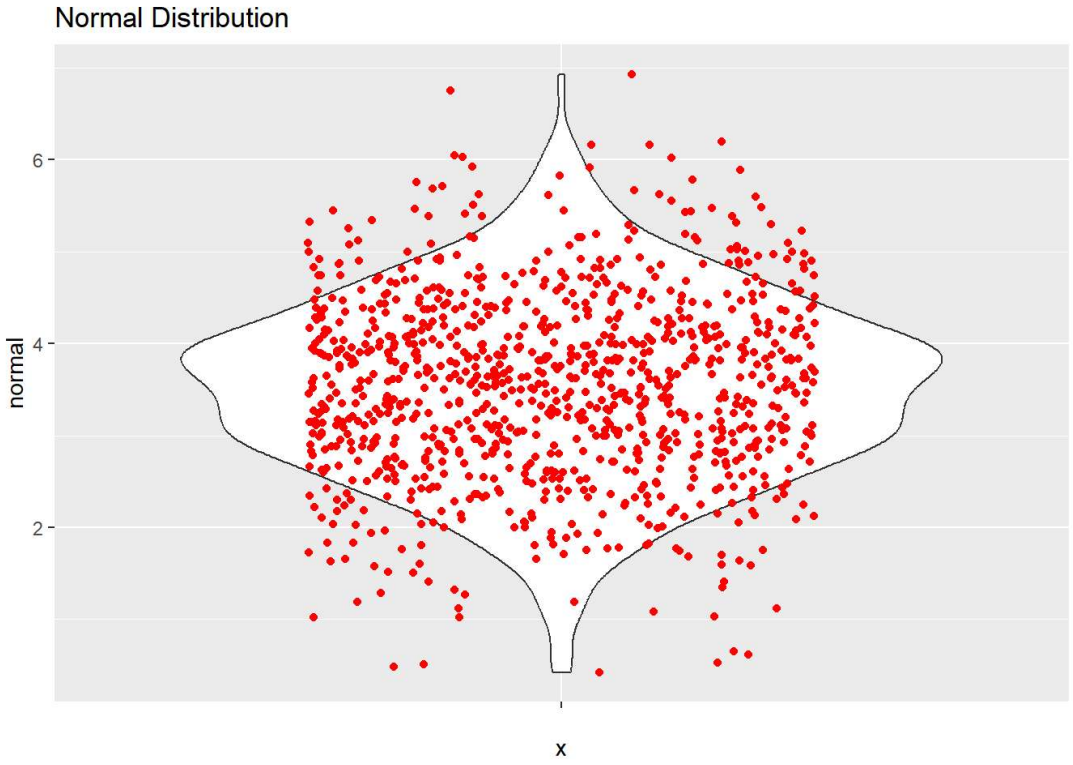
Violin plot

In this plot the width of the outline indicates the number of points present

```
ggplot(distrib_df, aes(x="",y=uniform))+geom_violin()+ggtitle('Uniform Distribution')+geom_rug(sides='1')+geom_jitter(width=0.3, height=0,col="red")
```




```
ggplot(distrib_df, aes(x="",y=normal))+geom_violin()+ggtitle('Normal Distribution')+geom_rug(sides='1')+geom_jitter(width=0.3, height=0,col="red")
```



Question/Action

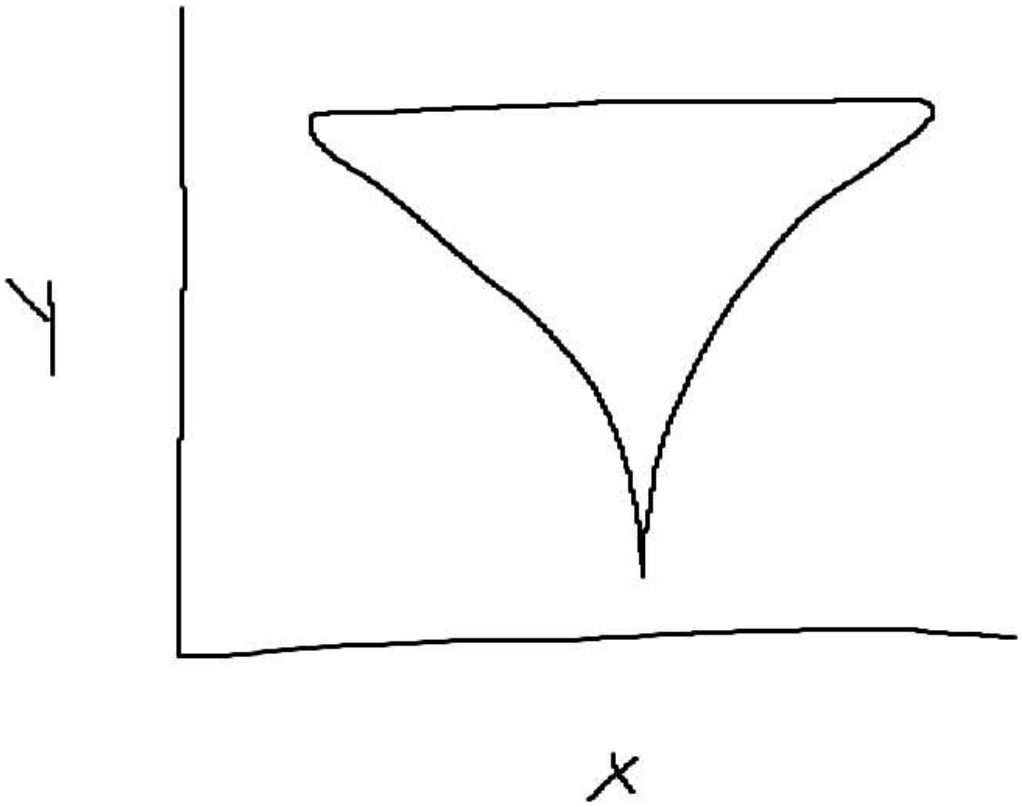
Why does this violin shape change so much from Uniform to normal?

The violin plot is a visualization of distribution density over the observation range. For a uniform distribution, the density should be, well, uniform, therefore the violin plot should be roughly square (depending on the random variation). A normal distribution has a much higher observation count density near the mean, tapering off to the ends of the distribution. The violin plot for the normal distribution ends up making a mirrored bell curve about the x-axis.

What feature of the violin plot can tell you about skew?

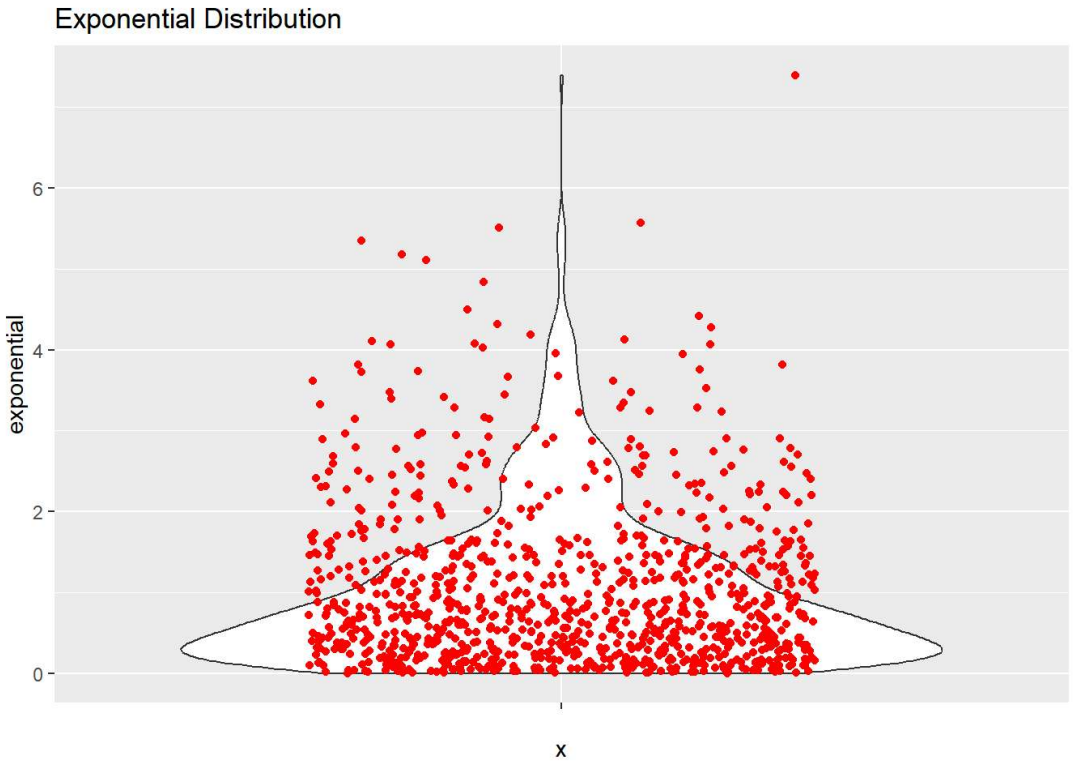
Skew can be determined by observing the width/length of the violin plot edges. A higher skew would indicate a larger tail length/width.

Predict what the violin plot of the exponential data will look like, then plot it.



Exponential Violin Plot

```
ggplot(distrib_df, aes(x="",y=exponential))+geom_violin()+ggtitle('Exponential Distribution')+geom_rug(sides='1')+geom_ji
tter(width=0.3, height=0,col="red")
```



Oops! Inverted… My mistake was that there should be more densely populated values at the low end of the distribution, because the points get exponentially less dense over the range of y.

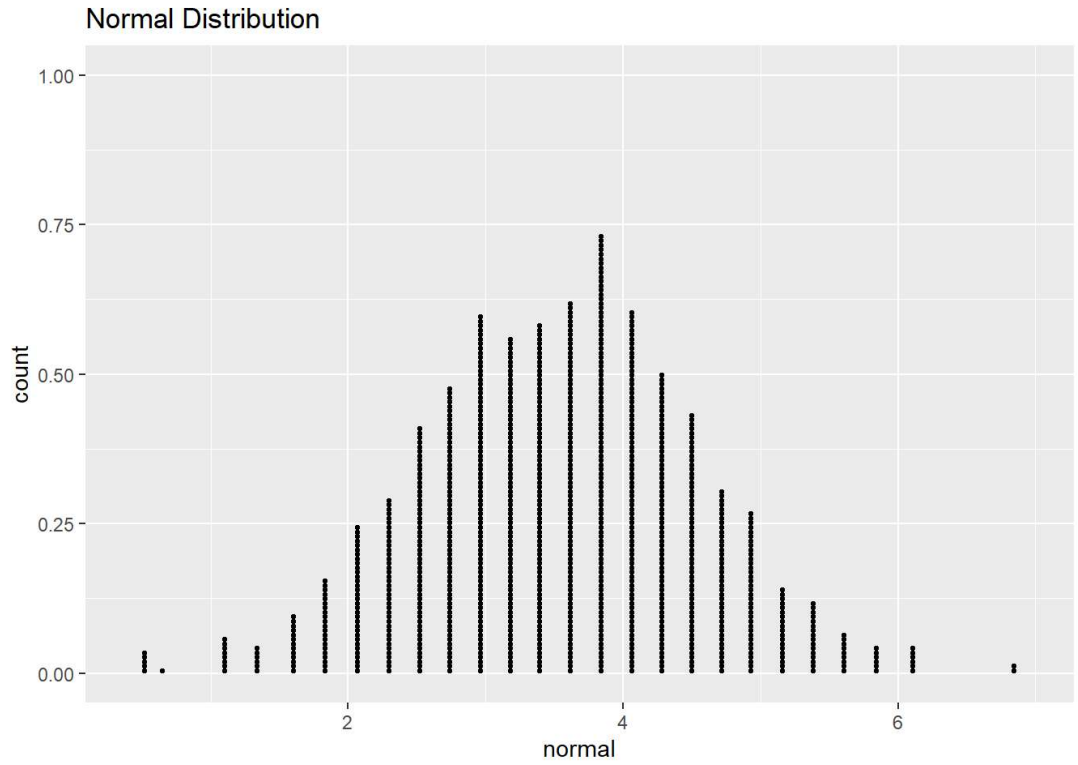
DotPlot

Shows a histogram as stacked dots

Really just another histogram

```
ggplot(distrib_df, aes(x=normal))+geom_dotplot(dotsize=0.15)+ggtitle('Normal Distribution')
```

```
## Bin width defaults to 1/30 of the range of the data. Pick better value with
## `binwidth`.
```

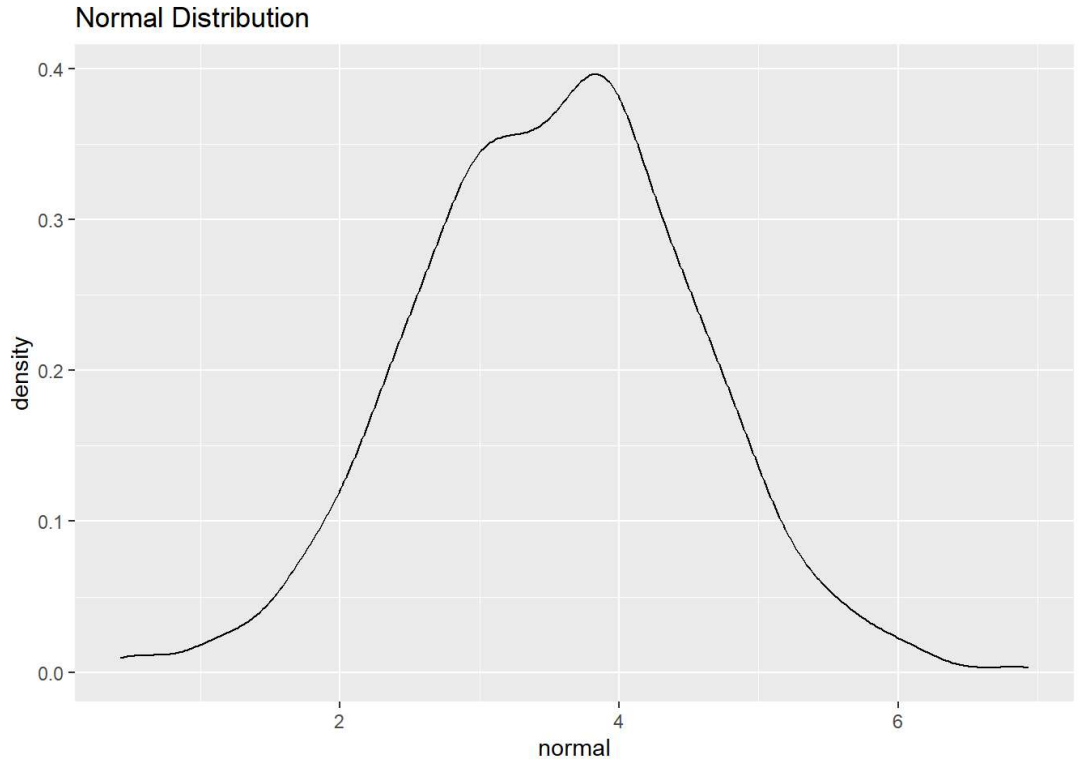


Density estimate

This is the estimated density of points, shown as a curve on a histogram

Essentially a curve fitted to the histogram

```
ggplot(distrib_df, aes(x=normal))+geom_density()+ggtitle('Normal Distribution')
```



Histogram with a density estimate

Histogram with a fitted curve

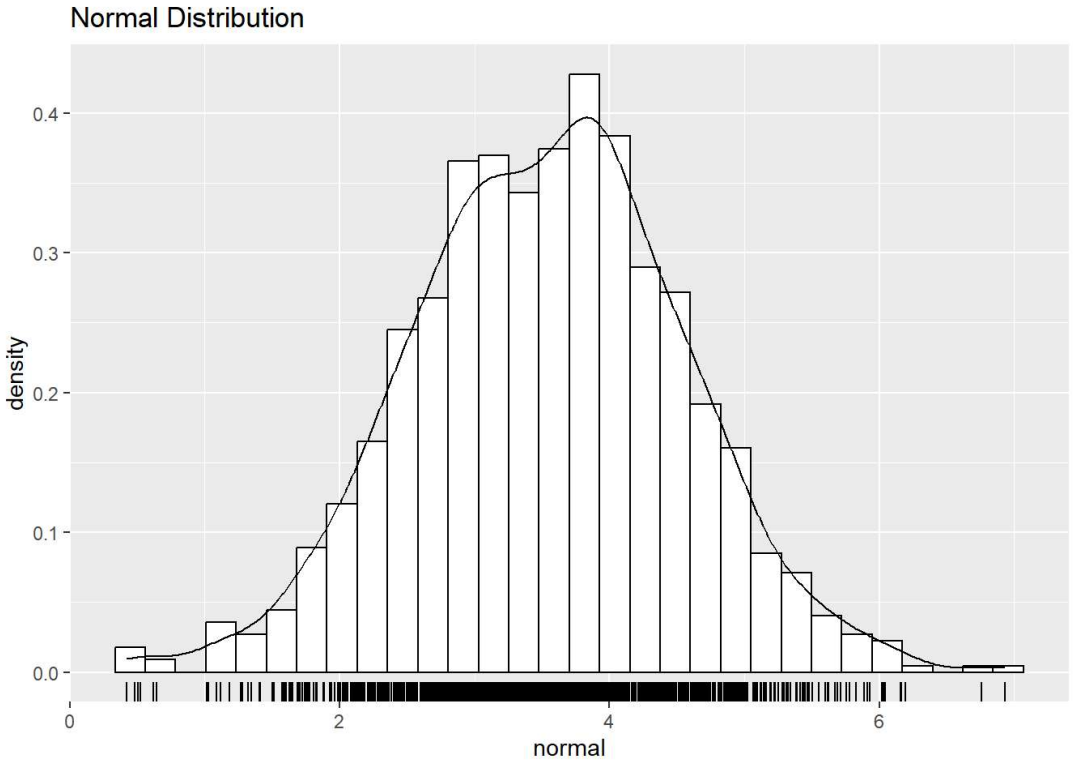
The “rug” plot only the bottom shows the density of values at each point along the axis

This is sometimes called a marginal density or marginal distribution plot

```
ggplot(distrib_df, aes(x=normal))+geom_histogram(aes(y = ..density..),
                 colour = 1, fill = "white")+ggtitle('Normal Distribution')+geom_density()+geom_rug()
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

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



Biplots or scatter plots

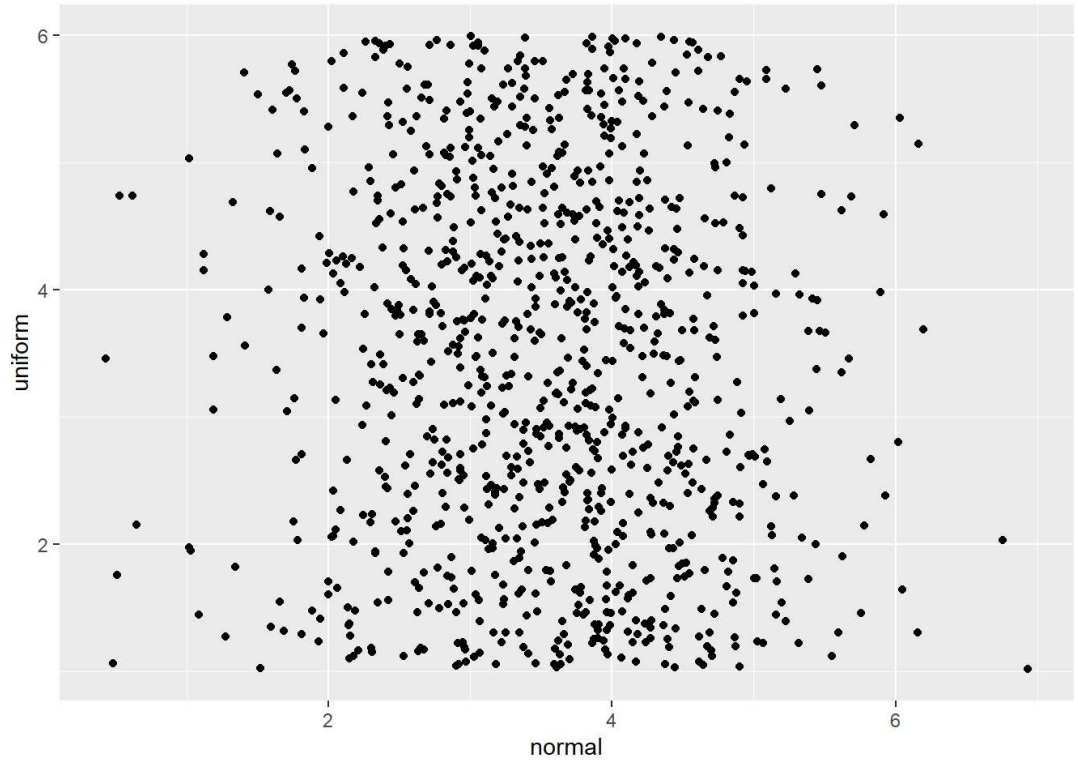
Allow us to see the relationships between two variables

Our random values are not good for this, but let’s try anyway, just to see how the plot works

geom_point() plots x and y

There is no relationship between the x and y axes used here, the result is just a featureless blob- this is evidence of a lack of a relationship between these two random variables

```
ggplot(distrib_df, aes(y=uniform, x=normal))+geom_point()
```



Data on human height and

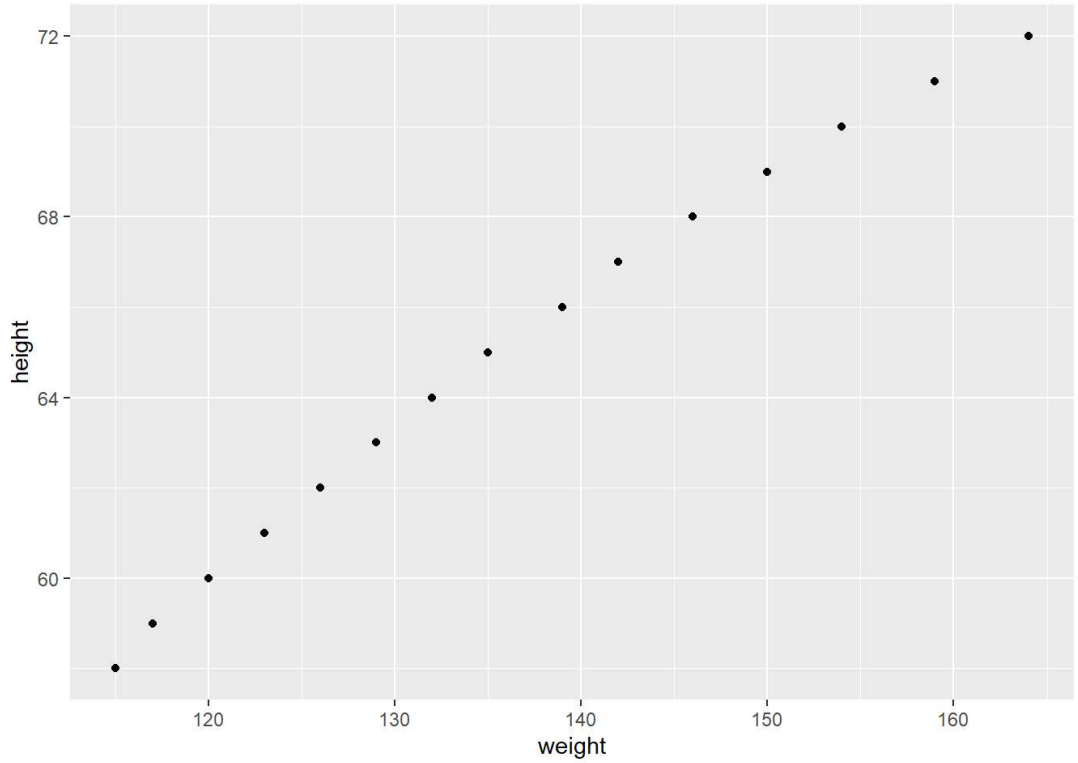
weight

We would expect some structure or relationship between height and weight (why?)

Typically, assuming a non-obese data set, taller people have a larger volume, therefore weigh more because the density of the human body is relatively consistent.

```
data(women)

ggplot(women,aes(x=weight, y=height))+geom_point()
```



Turns out in this limited data set, the relationship is very strong

Question/Action

Do an exploratory data analysis on the univariate values height and weight in the data set women

Weight Analysis

```
mean(women$weight)
```

```
## [1] 136.7333
```

```
median(women$weight)
```

```
## [1] 135
```

```
var(women$weight)
```

```
## [1] 240.2095
```

```
sd(women$weight)
```

```
## [1] 15.49869
```

```
skewness(women$weight)
```

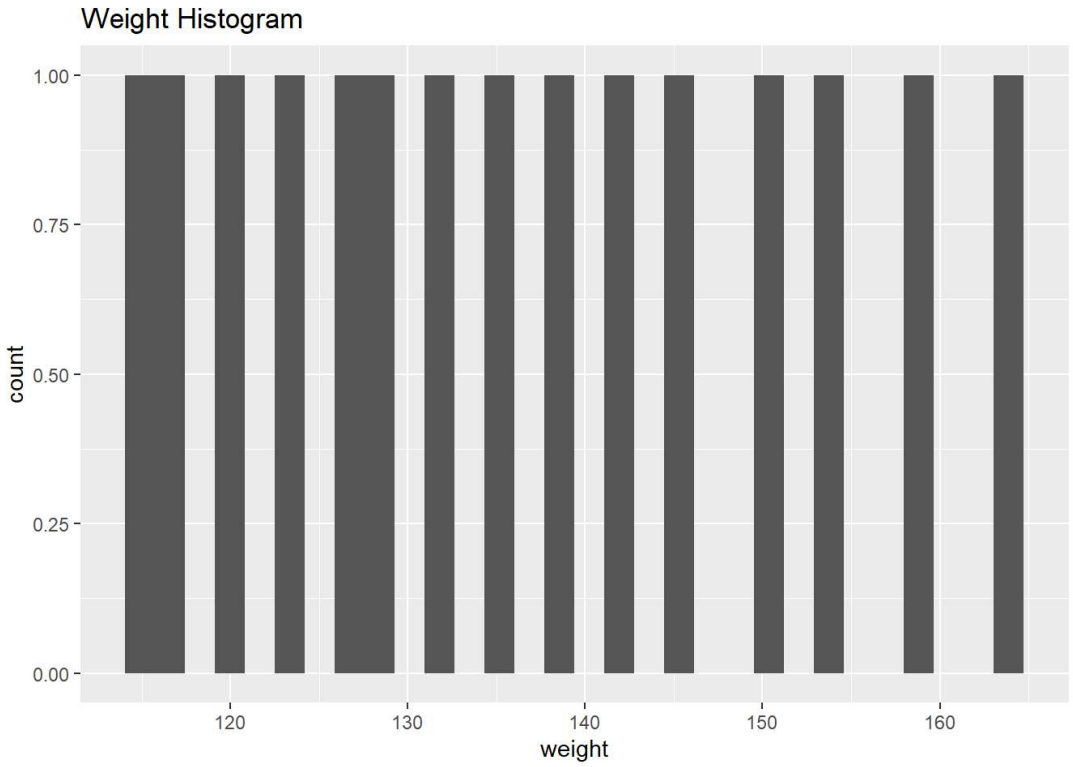
```
## [1] 0.2524665
```

```
kurtosis(women$weight)
```

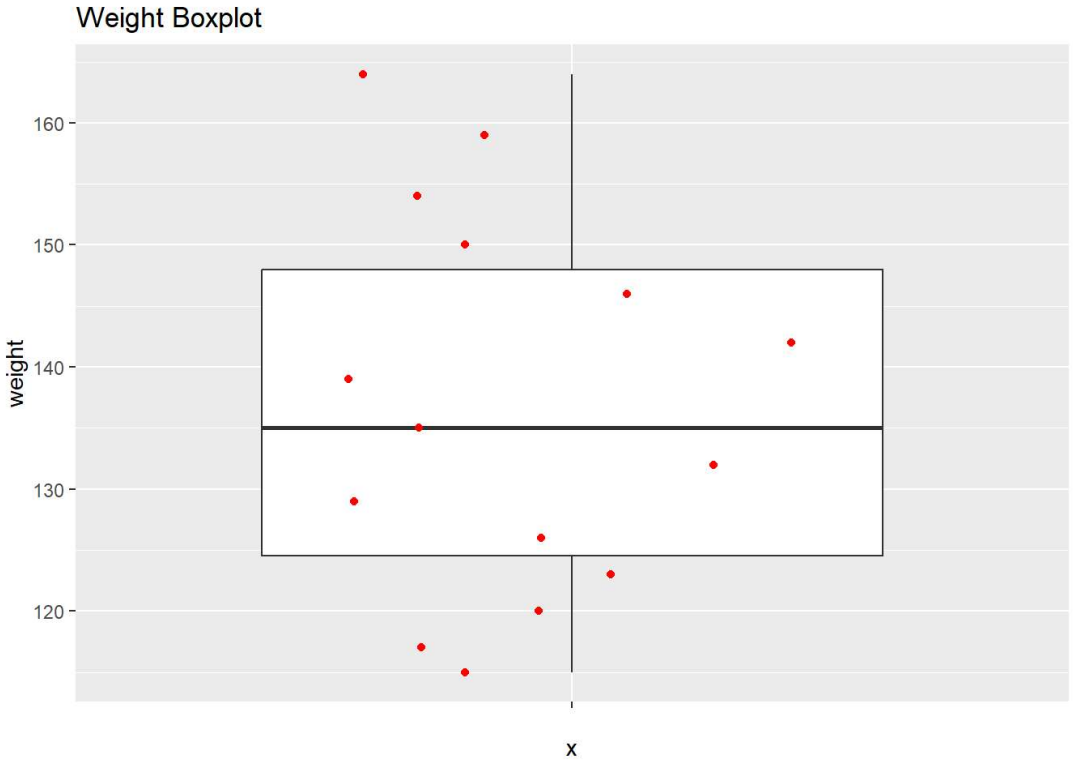
```
## [1] 1.900217
```

```
ggplot(women, aes(x=weight))+geom_histogram()+ggtitle('Weight Histogram')
```

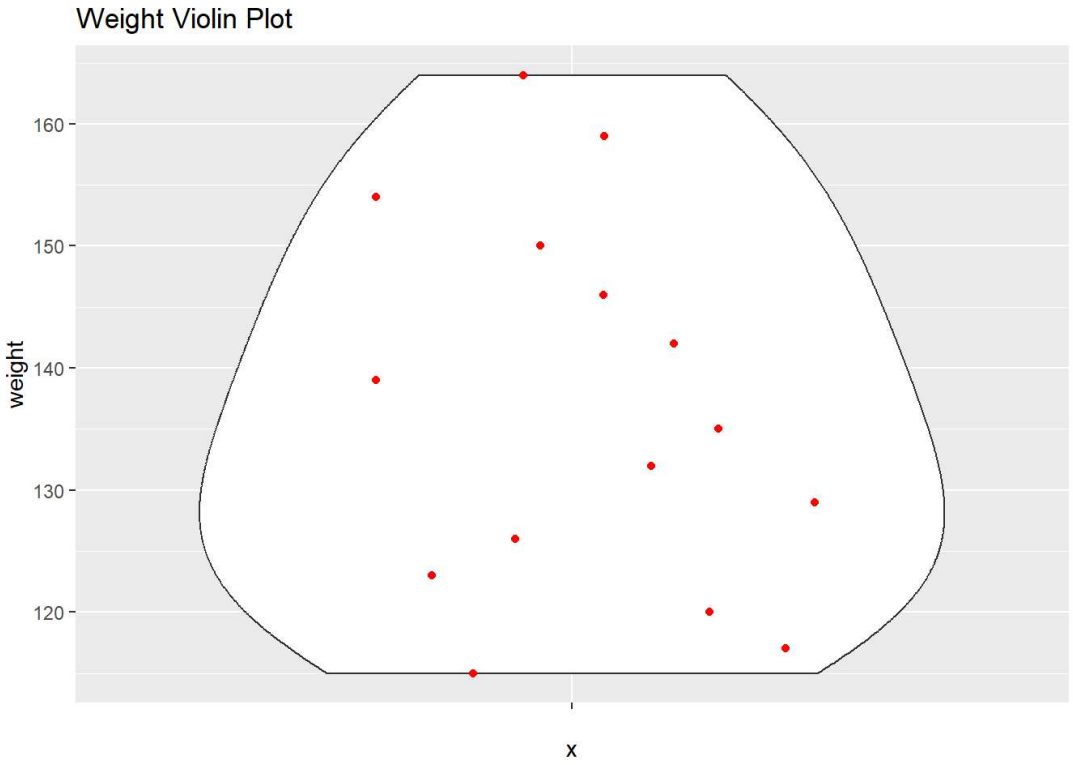
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(women, aes(x="",y=weight))+geom_boxplot()+ggtitle('Weight Boxplot')+geom_rug(sides='1')+geom_jitter(width=0.3, hei
ght=0,col="red")
```




```
ggplot(women, aes(x="",y=weight))+geom_violin()+ggtitle('Weight Violin Plot')+geom_rug(sides='1')+geom_jitter(width=0.3,
height=0,col="red")
```



Height Analysis

```
mean(women$height)
```

```
## [1] 65
```

```
median(women$height)
```

```
## [1] 65
```

```
var(women$height)
```

```
## [1] 20
```

```
sd(women$height)
```

```
## [1] 4.472136
```

```
skewness(women$height)
```

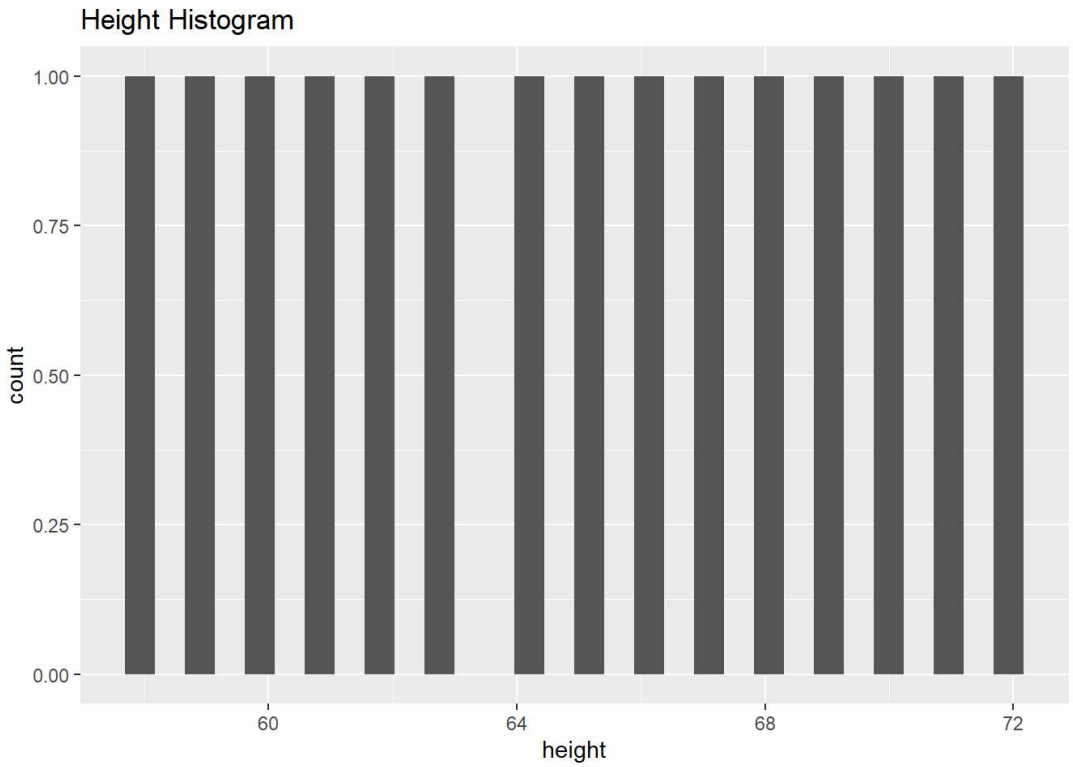
```
## [1] 0
```

```
kurtosis(women$height)
```

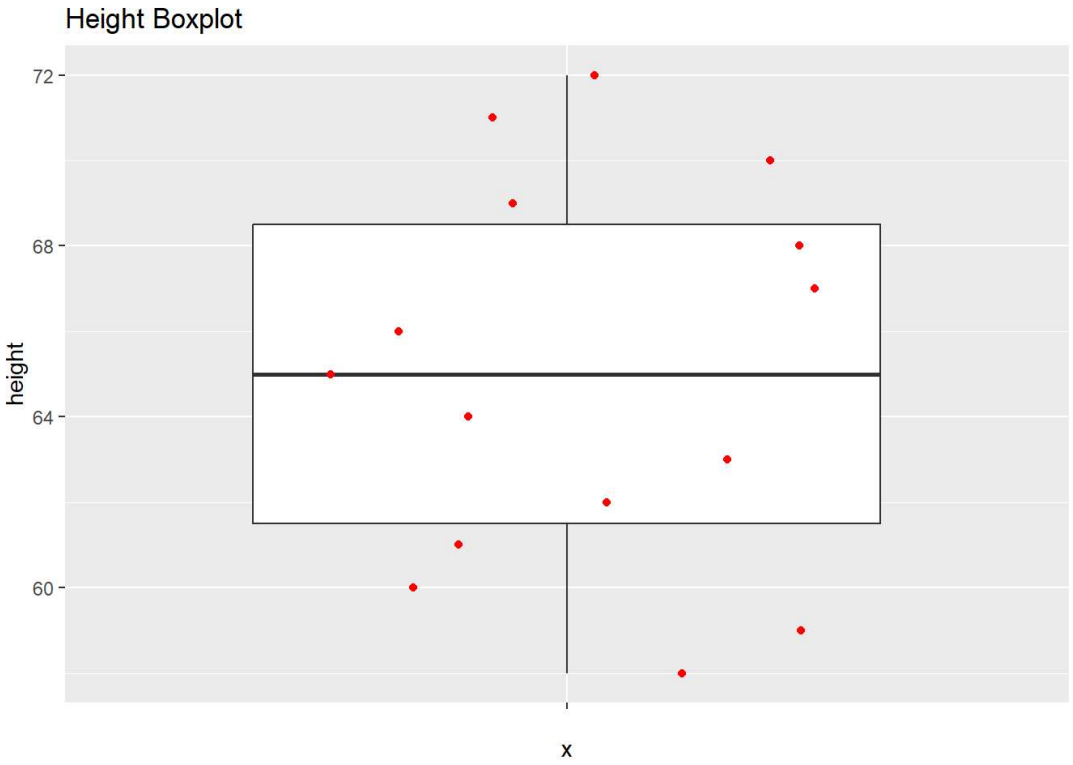
```
## [1] 1.789286
```

```
ggplot(women, aes(x=height))+geom_histogram()+ggtitle('Height Histogram')
```

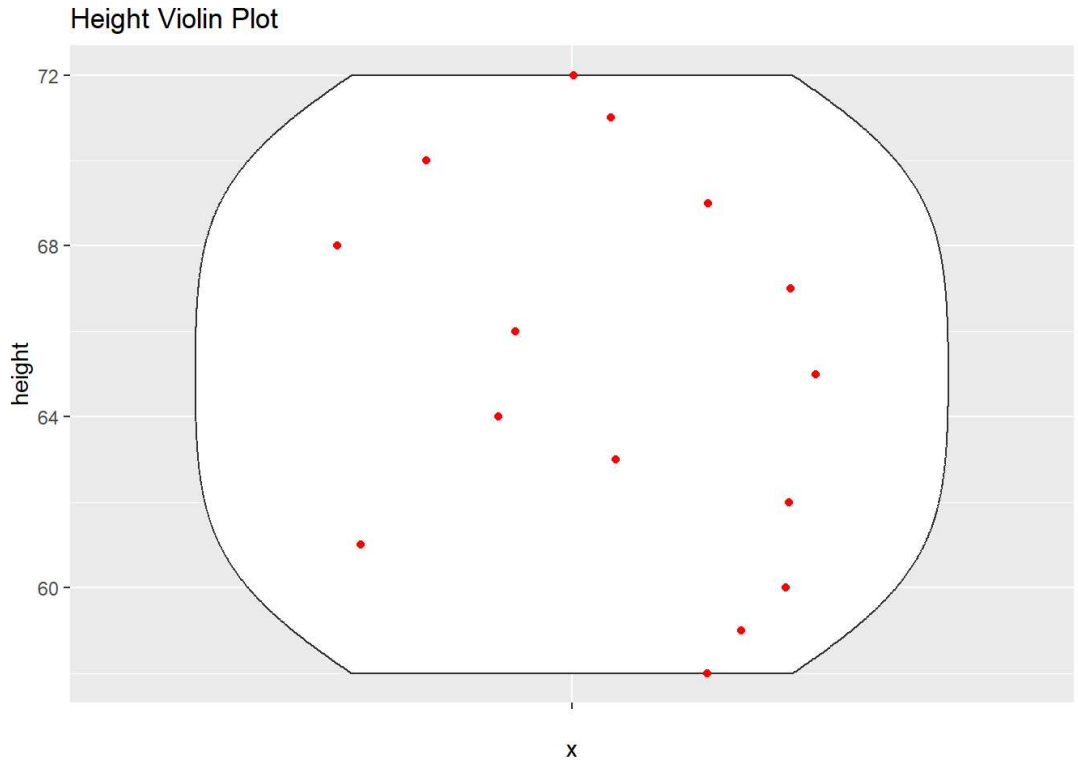
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(women, aes(x="",y=height))+geom_boxplot()+ggtitle('Height Boxplot')+geom_rug(sides='1')+geom_jitter(width=0.3, hei
ght=0,col="red")
```




```
ggplot(women, aes(x="",y=height))+geom_violin()+ggtitle('Height Violin Plot')+geom_rug(sides='1')+geom_jitter(width=0.3,
height=0,col="red")
```



Use whatever tools and plots seem useful- try to use them all in fact

This is a small data set, so the results will not look “perfect”

What is the central tendency?- find several measures

The tendency for the height data is almost perfect uniformity, based on the violin plots, skewness, and kurtosis. Weight, however, is more of a weibull distribution, using the same metrics.

Find 3 measures of spread

Skewness, kurtosis, and standard deviation are the three measures of spread. These values for each data set can be seen, below:

Weight [1] 15.49869 [1] 0.2524665 [1] 1.900217

Height [1] 4.472136 [1] 0 [1] 1.789286

What is the kurtosis? is this data skewed? provide two forms of evidence of this

Based on the weight and height numerical analysis, above, the kurtosis is 1.9 and 1.79 for weight and height, respectively. The data is not skewed based on the skewness values (0 and 0.25) and the boxplots for both data sets.

If you had to pick one of the 4 distributions we have seen to describe the height and weight data, which would you use? Would you use the same for both? Why is the distribution you picked the best choice in each case?

I would use uniform for height and Weibull for weight. Uniform is fit for height because it has a very low kurtosis and zero skew. Weight fits the Weibull distribution becase it has a slightly higher kurtosis, non-zero skew, and the violin plot is suggestive of Weibull.