Week 3

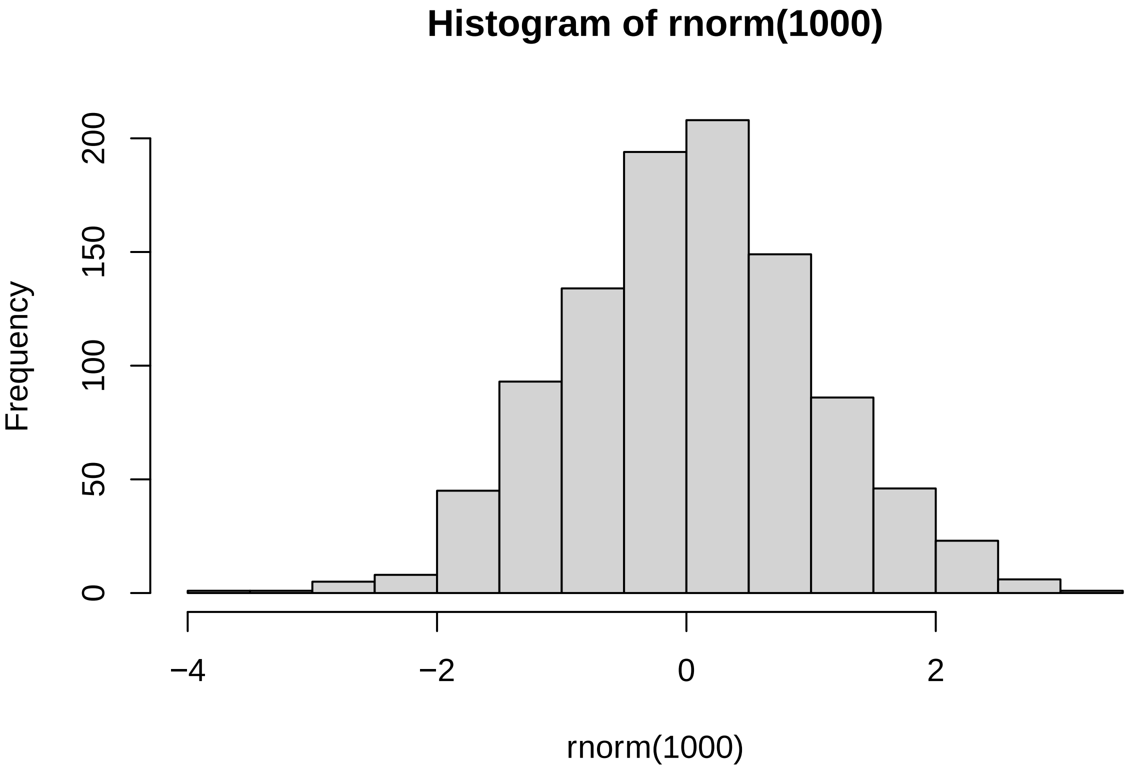
Noah Jones

3/6/2021

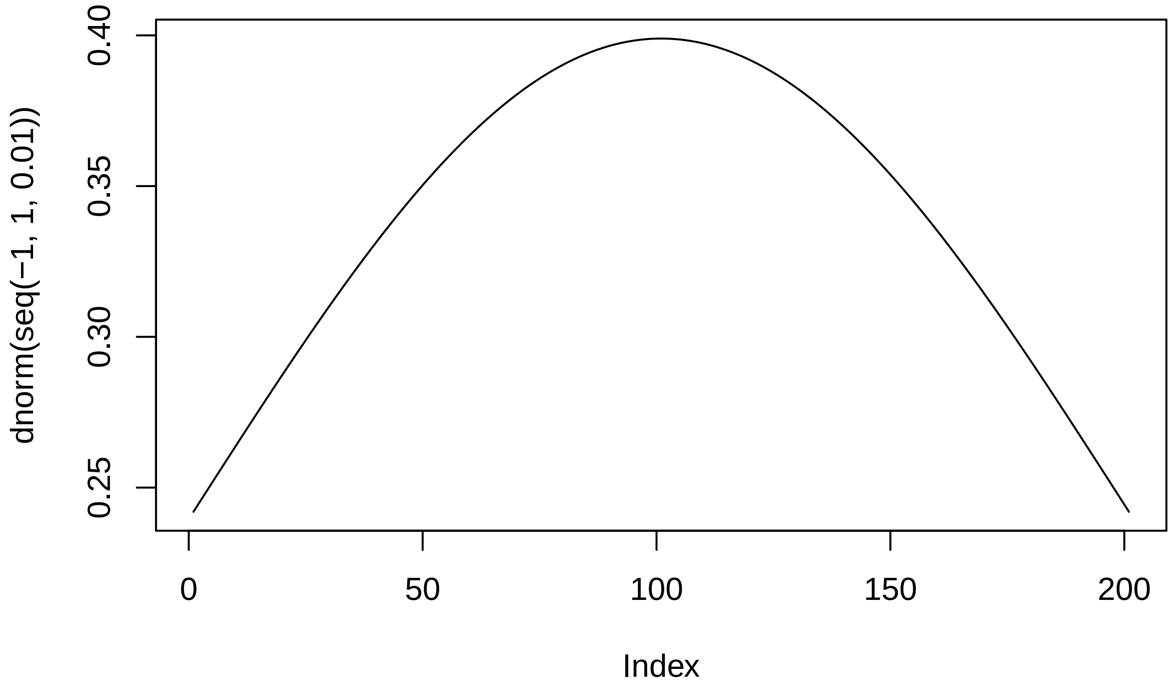
Resources

Today we are going to try to figure out how to fit to a distribution. This might be a useful resource.

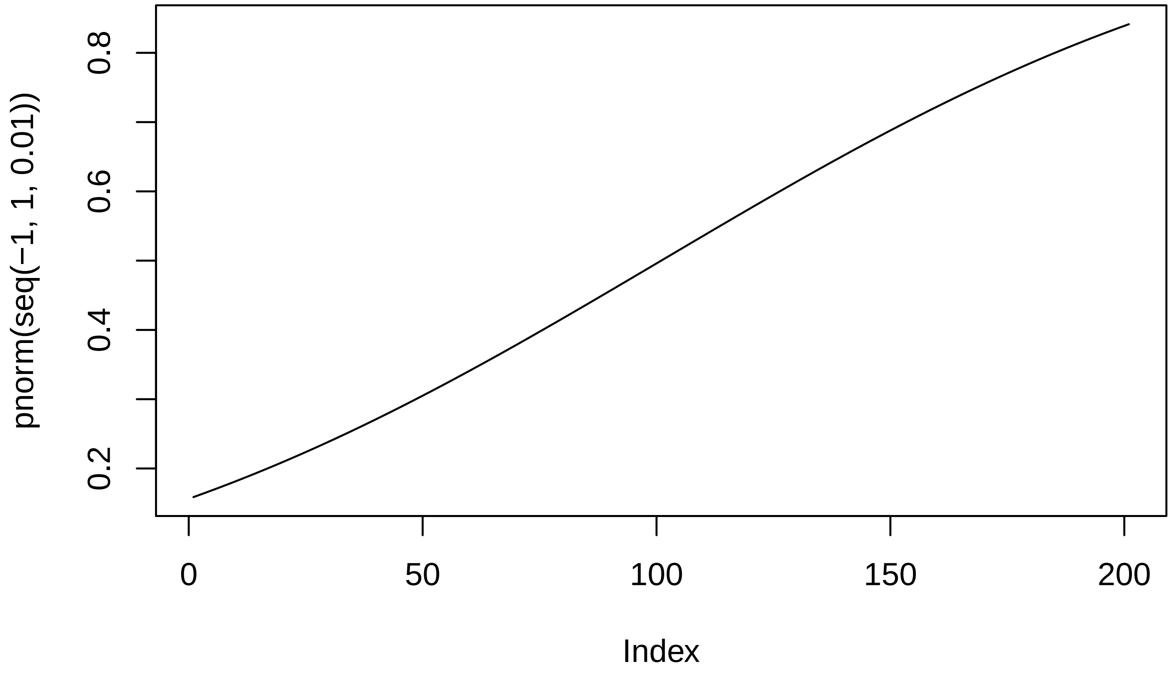
```
hist(rnorm(1000), breaks = 20)
```



```
plot(dnorm(seq(-1, 1, 0.01)), type = 'l')
```

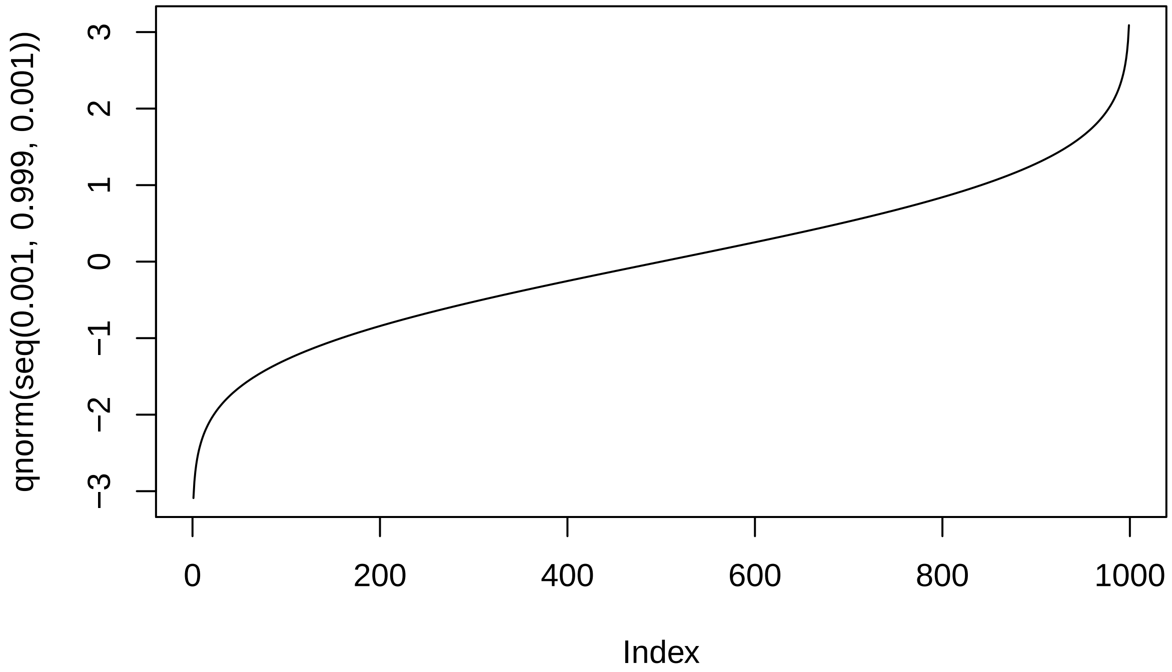


```
plot(pnorm(seq(-1, 1, 0.01)), type = 'l')
```

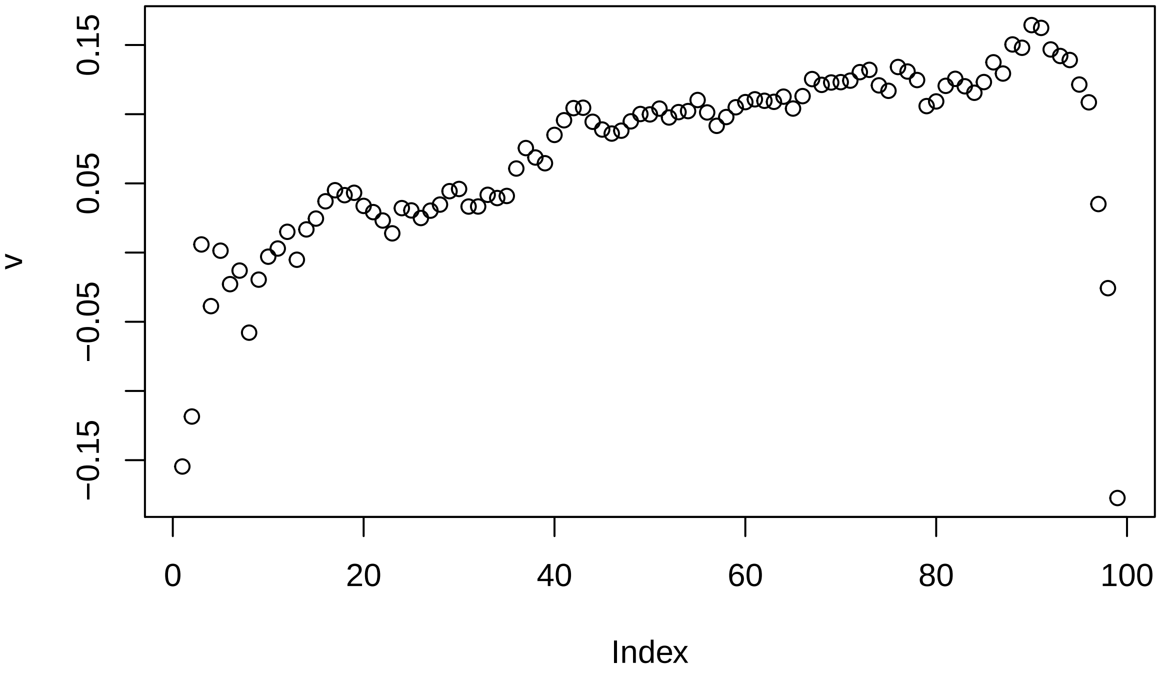


It looks like the probability density function is what we are probably looking for. To generate a linear model, we have to describe the two data sets such that they vary together. Perhaps the quantiles co-vary, or their ranks co-vary. Something like that. I think quantiles might be the most appropriate.

```
plot(qnorm(seq(.001, .999, .001)), type = "l")
```



```
v <- quantile(rnorm(1000), seq(.01, .99, .01)) -
  qnorm(seq(.01, .99, .01))
plot(v)
```

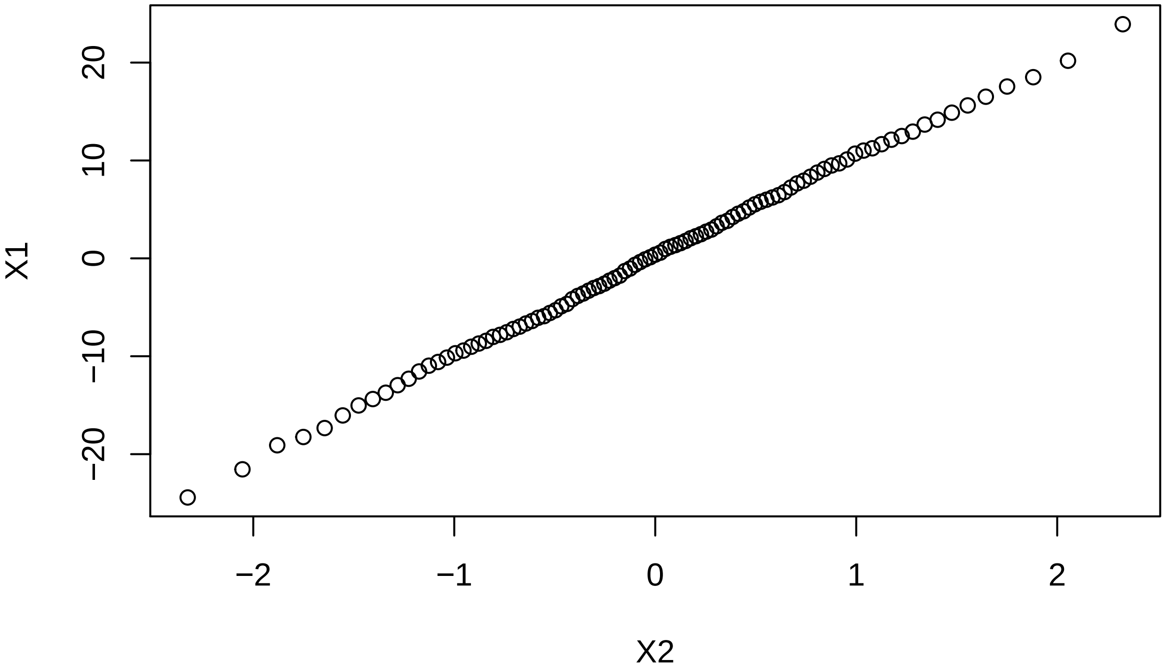


Quantile Plots

We demonstrate that by plotting the quantiles of a dataset against the quantiles of a distribution, we can infer whether the data set matches the distribution.

```
cbind(quantile(rnorm(1000)*10, seq(.01, .99, .01)), qnorm(seq(.01, .99, .01))) %>%
  data.frame() -> df1
```

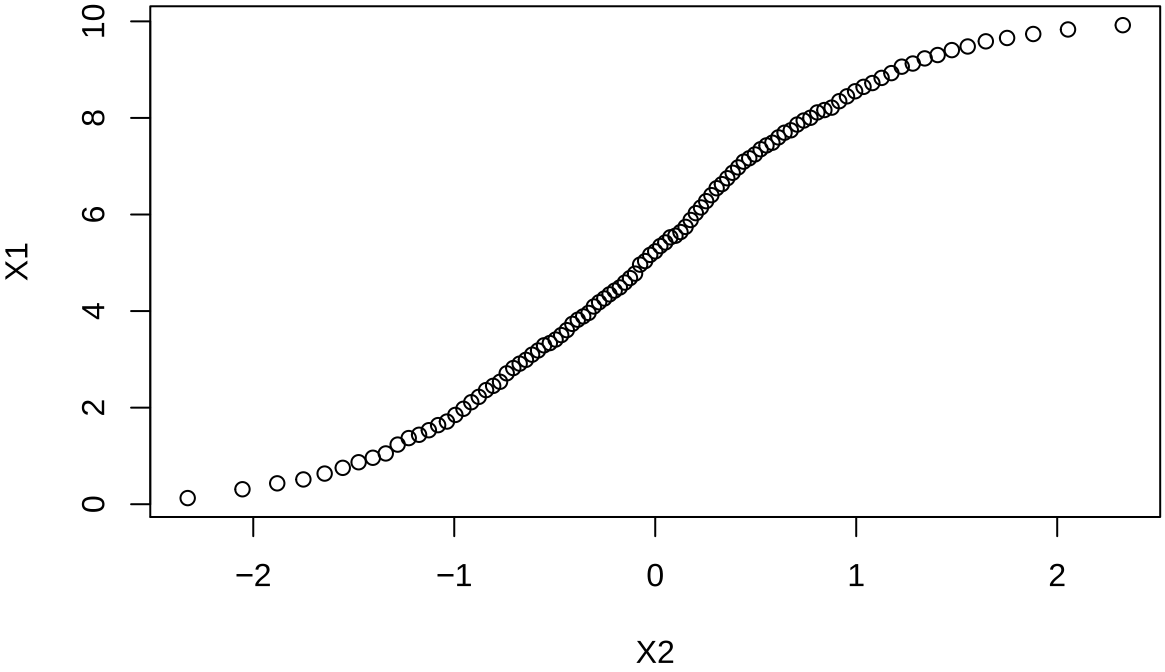
```
plot(X1 ~ X2, df1)
```



```
lm(X1 ~ X2, df1) %>% summary()
```

```
##
## Call:
## lm(formula = X1 ~ X2, data = df1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.97432 -0.12099  0.04114  0.21178  0.44429
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.12802    0.02913   4.395 2.83e-05 ***
## X2          10.24164    0.03033 337.654  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2898 on 97 degrees of freedom
## Multiple R-squared:  0.9991, Adjusted R-squared:  0.9991
## F-statistic: 1.14e+05 on 1 and 97 DF,  p-value: < 2.2e-16
```

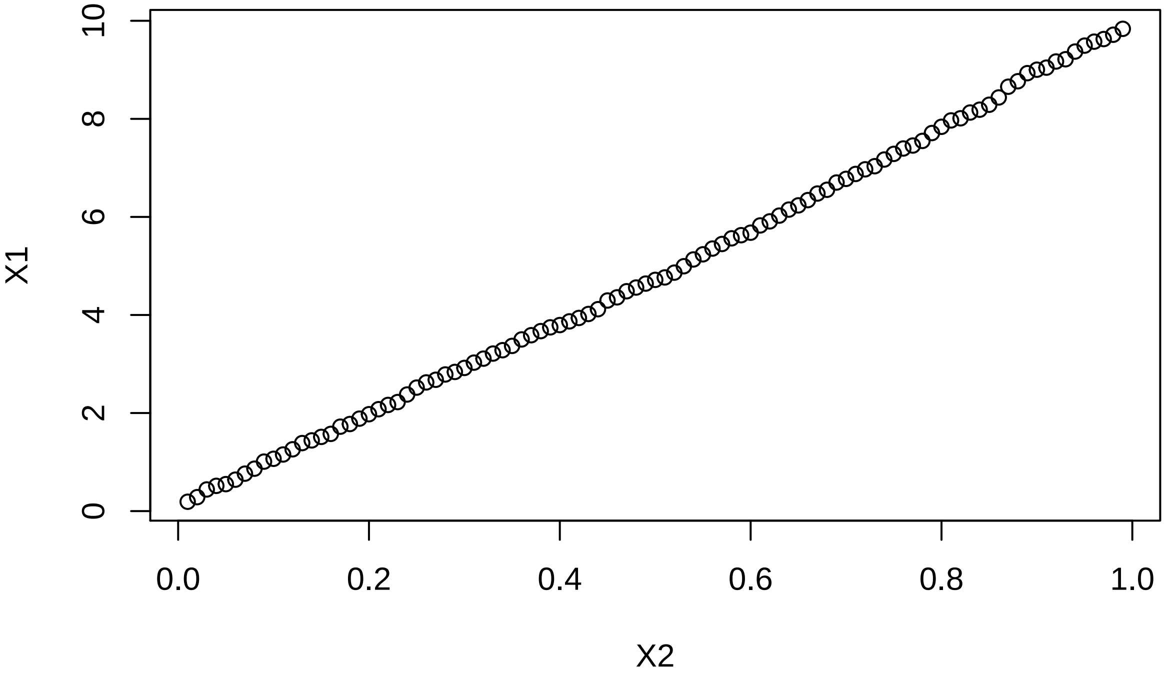
```
cbind(quantile(runif(1000)*10, seq(.01, .99, .01)), qnorm(seq(.01, .99, .01))) %>%
  data.frame() -> df2
plot(X1 ~ X2, df2)
```



```
lm(X1 ~ X2, df2) %>% summary()
```

```
##
## Call:
## lm(formula = X1 ~ X2, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.05534 -0.35839 -0.00371  0.42824  1.63342
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.23500    0.05036  103.94   <2e-16 ***
## X2           2.89769    0.05245   55.25   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5011 on 97 degrees of freedom
## Multiple R-squared:  0.9692, Adjusted R-squared:  0.9689
## F-statistic:  3053 on 1 and 97 DF,  p-value: < 2.2e-16
```

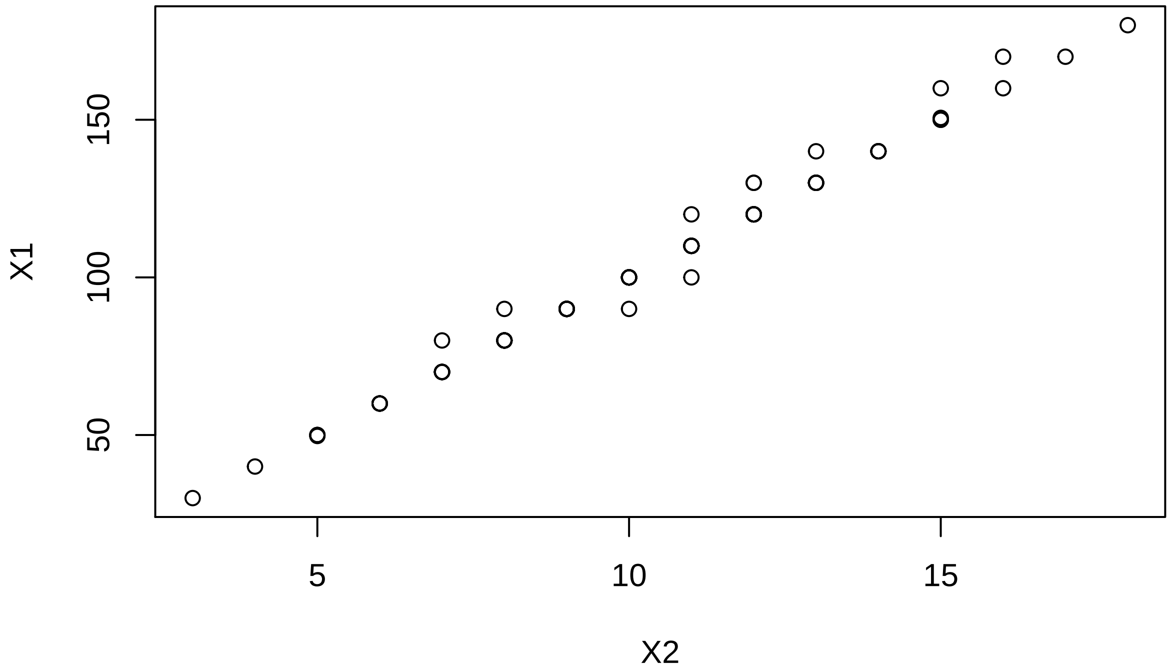
```
cbind(quantile(runif(1000)*10, seq(.01, .99, .01)),
      qunif(seq(.01, .99, .01))
      ) %>%
  data.frame() -> df3
plot(X1 ~ X2, df3)
```



```
lm(X1 ~ X2, df3) %>% summary()
```

```
##
## Call:
## lm(formula = X1 ~ X2, data = df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.20998 -0.09744 -0.00349  0.08840  0.23962
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.01213    0.02269  -0.534    0.594
## X2           9.78015    0.03940 248.203   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.112 on 97 degrees of freedom
## Multiple R-squared:  0.9984, Adjusted R-squared:  0.9984
## F-statistic: 6.16e+04 on 1 and 97 DF,  p-value: < 2.2e-16
```

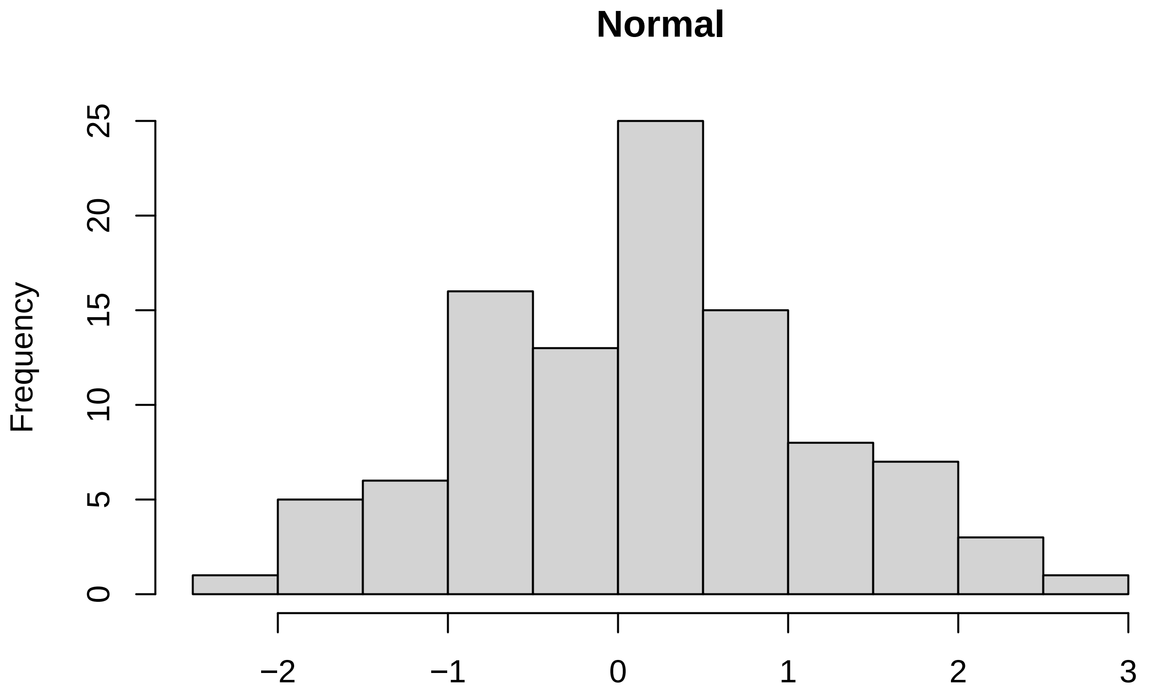
```
cbind(quantile(rpois(1000, 10)*10, seq(.01, .99, .01)),
      qpois(seq(.01, .99, .01), 10)
      ) %>%
  data.frame() -> df3
plot(X1 ~ X2, df3)
```



```
lm(X1 ~ X2, df3) %>% summary()
```

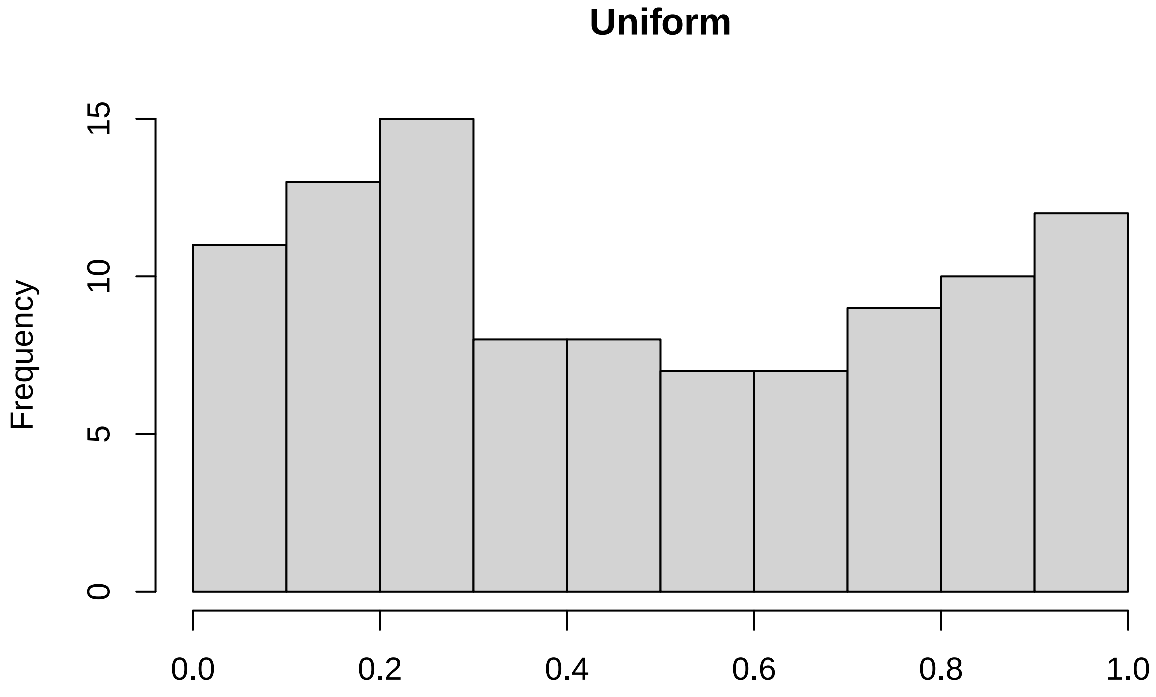
```
##
## Call:
## lm(formula = X1 ~ X2, data = df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.7629  -0.9123  -0.4643  -0.1656   9.8344
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.8797     1.0757  -0.818    0.415
## X2           10.1493     0.1032  98.333   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.119 on 97 degrees of freedom
## Multiple R-squared:  0.9901, Adjusted R-squared:   0.99
## F-statistic:  9669 on 1 and 97 DF,  p-value: < 2.2e-16
```

```
rnorm(100) %>% hist(breaks = 10, main = "Normal")
```



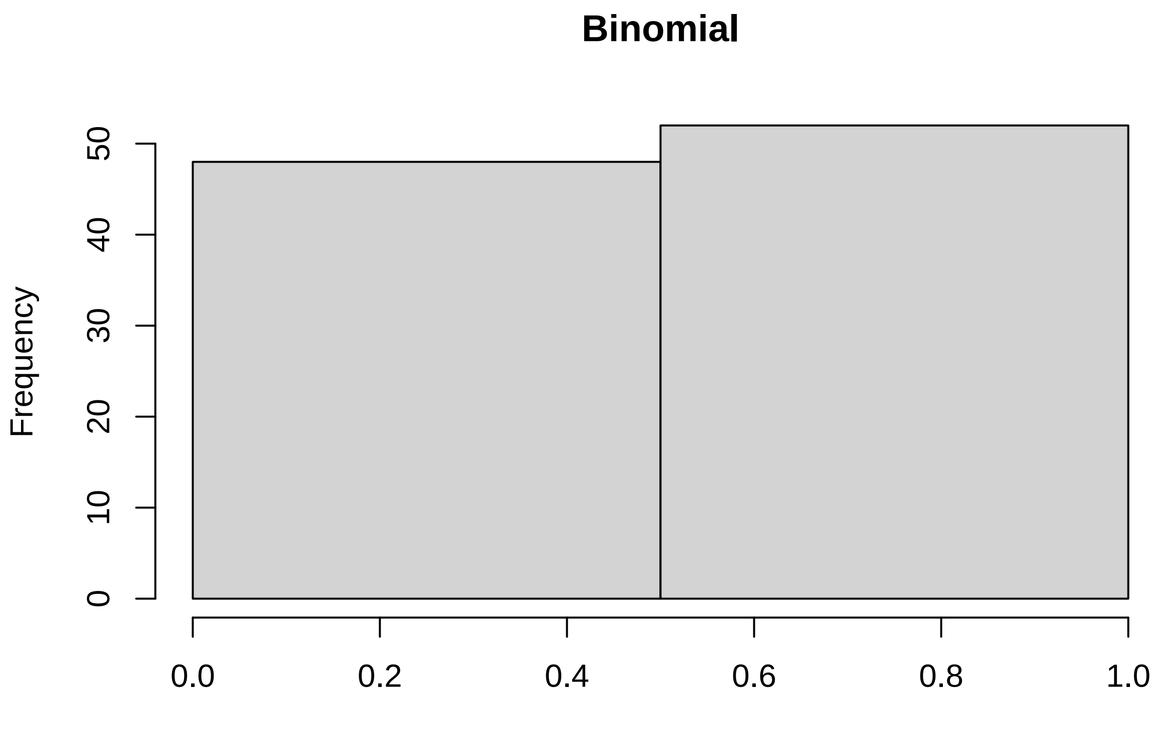
.

```
runif(100) %>% hist(breaks = 10, main = "Uniform")
```

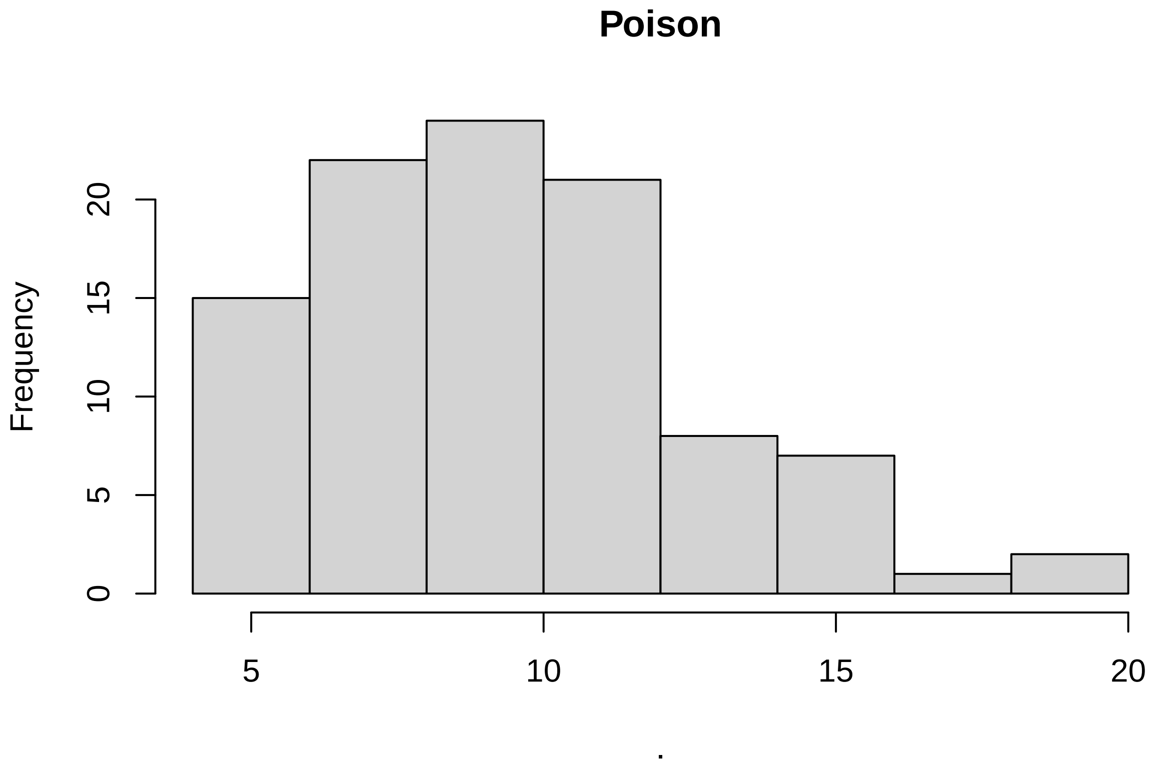


.

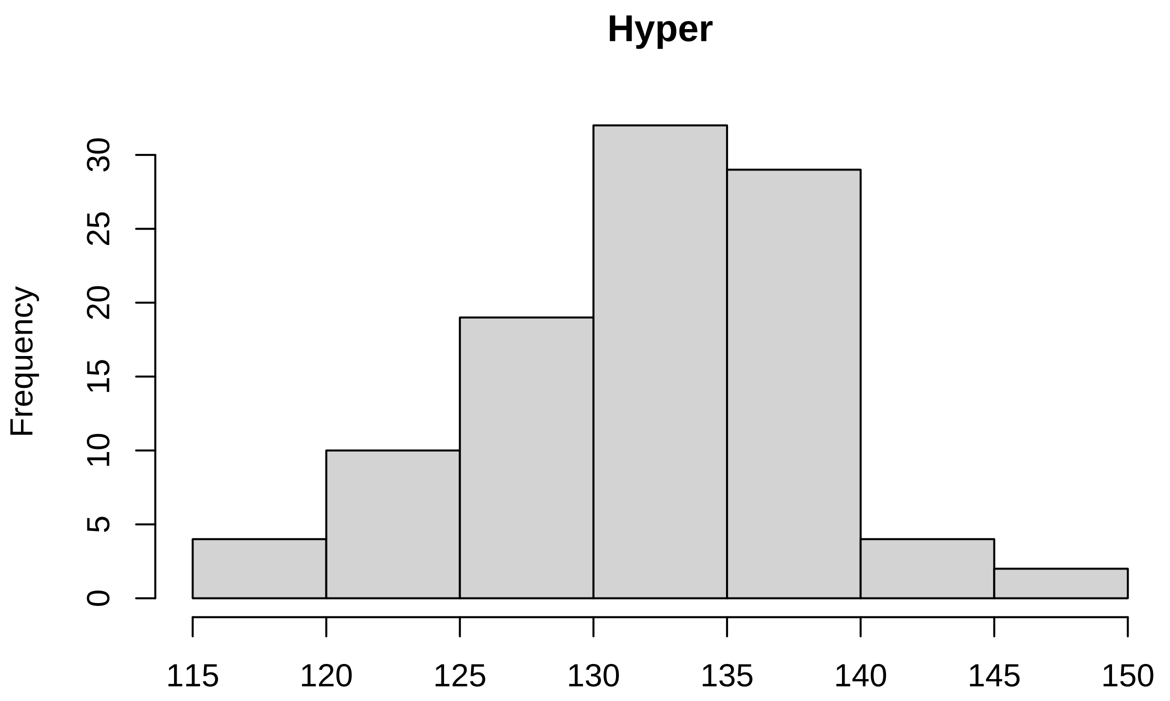
```
rbinom(100, 1, 0.5) %>% hist(breaks = 2, main = "Binomial")
```



```
rpois(100, 10) %>% hist(breaks = 10, main = "Poison")
```

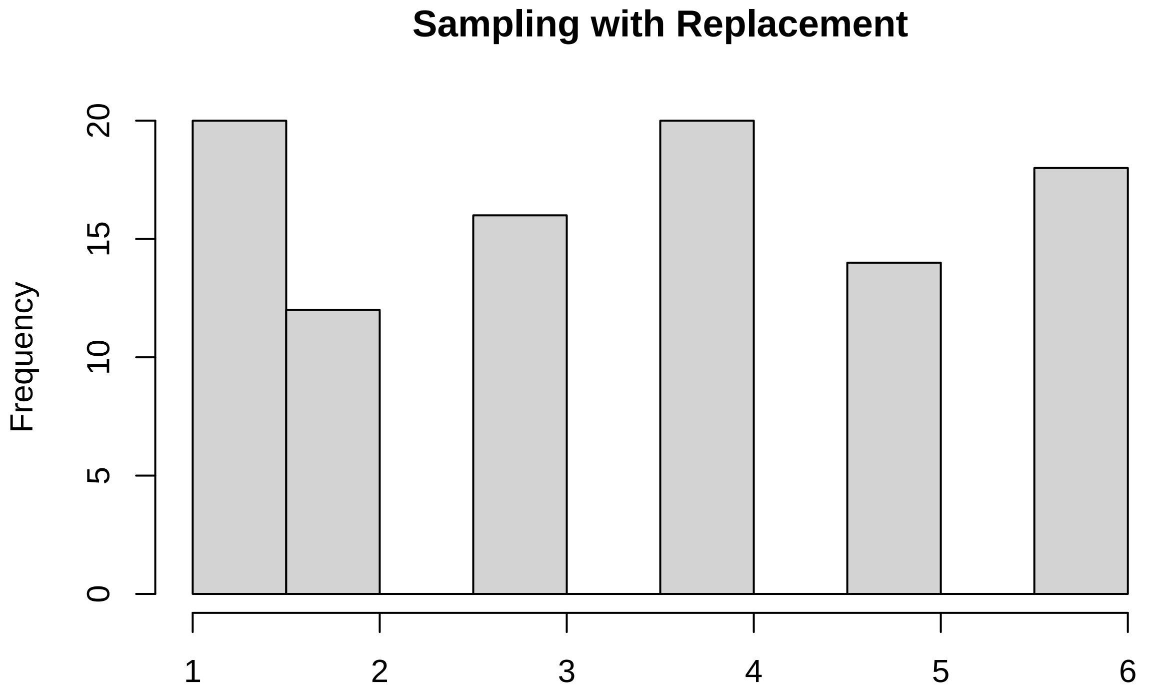


```
rhyper(100, 1000, 500, 200) %>%
  hist(breaks = 10, main = "Hyper")
```



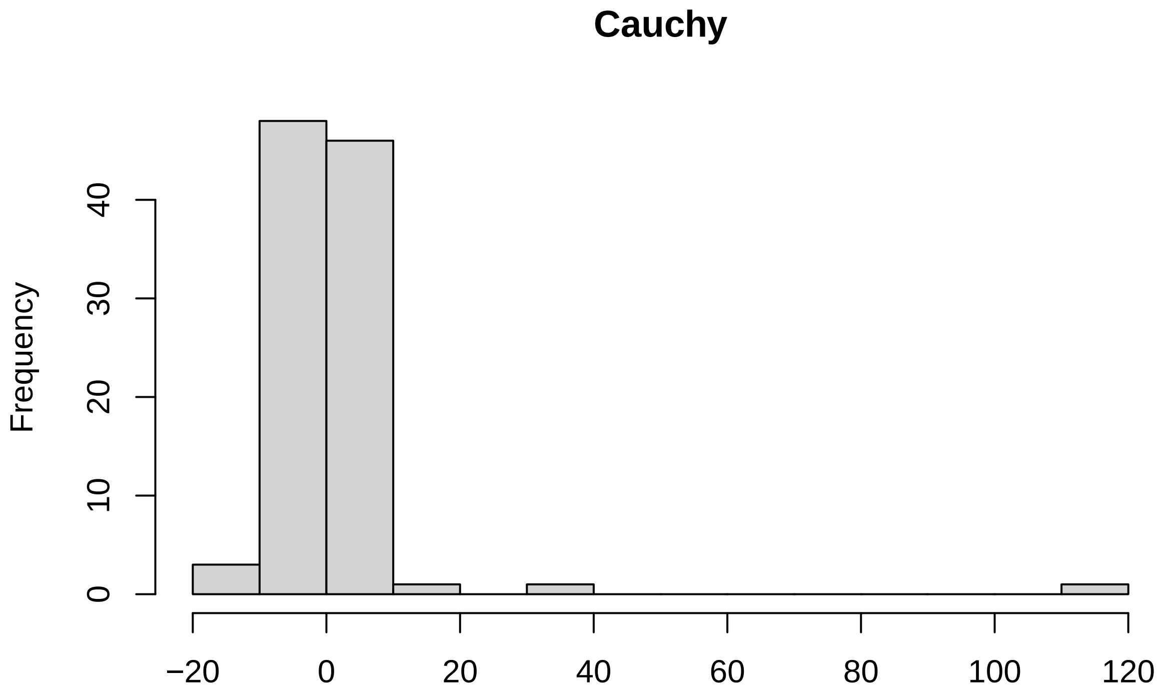
.

```
sample(1:6, 100, replace = T, prob = rep(1/6,6)) %>%
  hist(breaks = 10, main = "Sampling with Replacement")
```



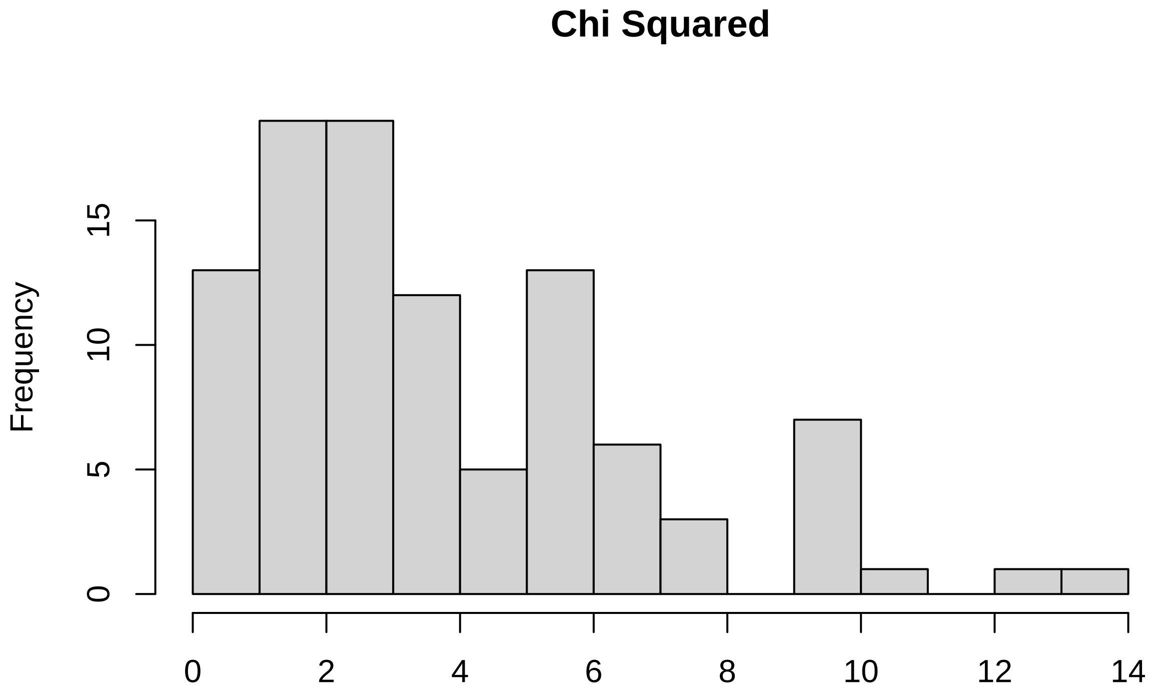
.

```
rcauchy(100) %>% hist(breaks = 10, main = "Cauchy")
```



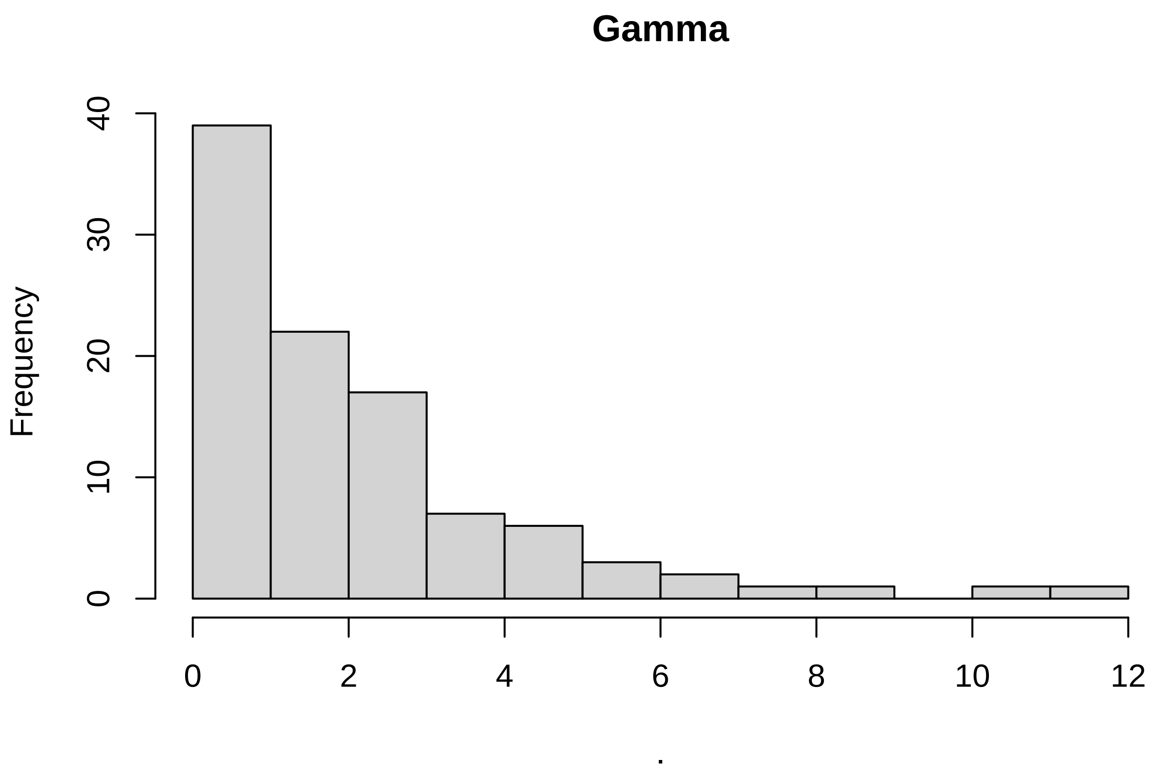
.

```
rchisq(100, 4) %>% hist(breaks = 10, main = "Chi Squared")
```

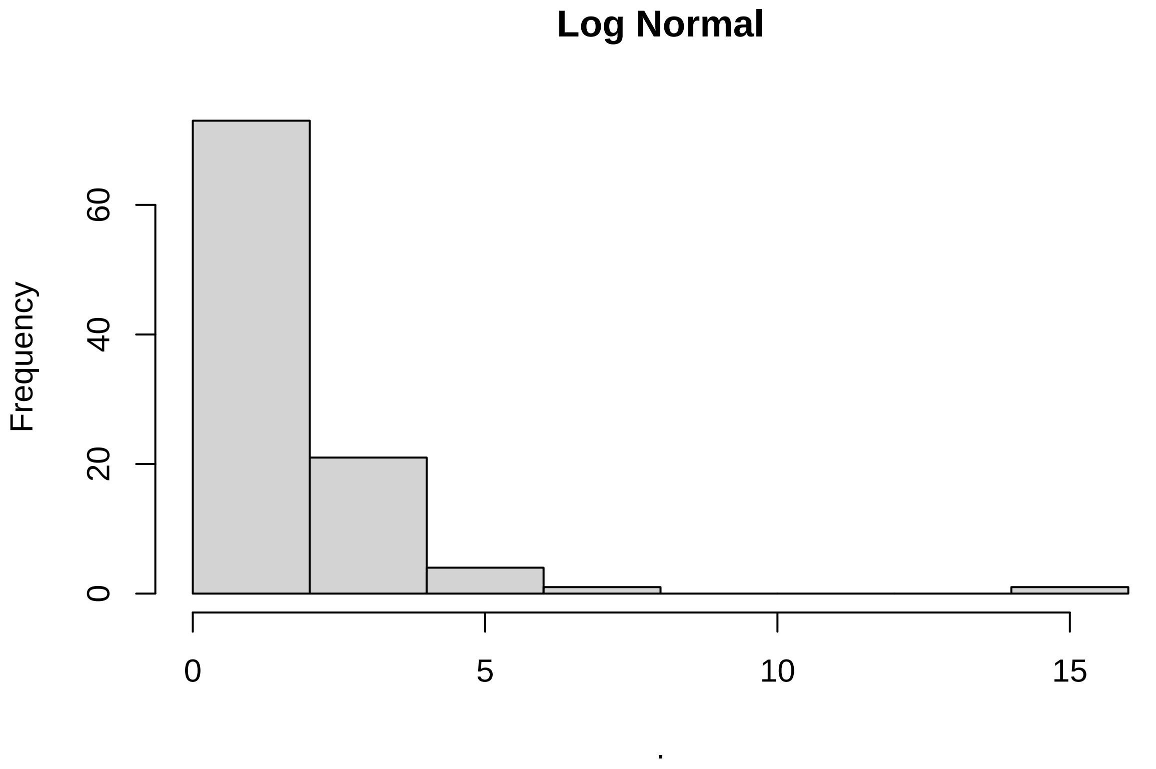


.

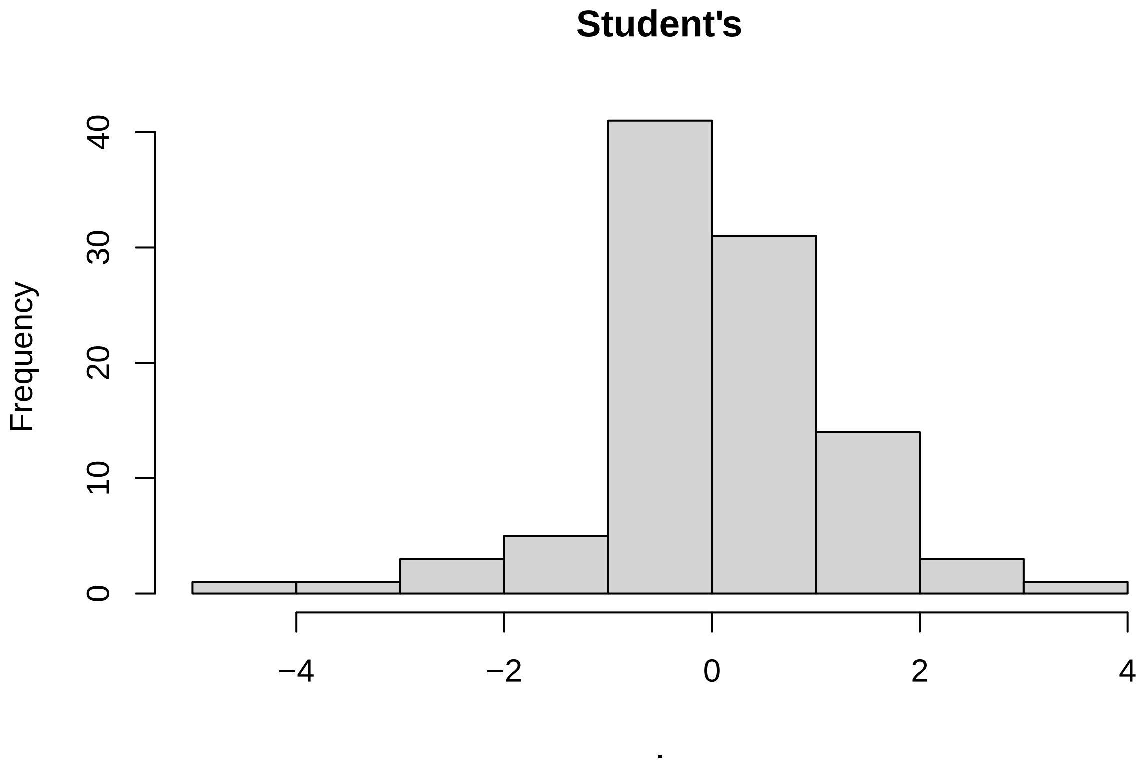
```
rgamma(100, 1, 0.5) %>% hist(breaks = 10, main = "Gamma")
```



```r
rlnorm(100, 0, 1) %>% hist(breaks = 10, main = "Log Normal")
```



```r
rt(100, 10) %>% hist(breaks = 10, main = "Student's")
```


Student's
Frequency
0
10
20
30
40
−4
−2
0
2
4
.