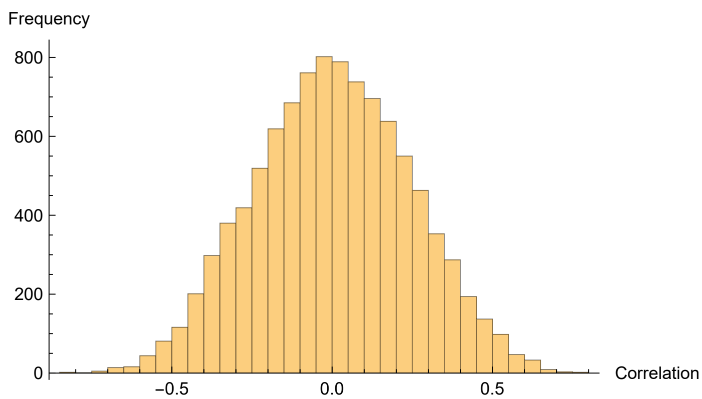
P-Value Hacking

Suppose we generate possible correlation coefficients in this way.
We have it between variables.
The variables follow a standard normal distribution.
There are 18 pairs of variables.
The correlation calculation is done 10000 times.
Notice that the correlation coefficient can easily exceed 0.5.

In[ ]:=
```
Histogram[Table[a = RandomVariate[NormalDistribution[], 18];
   b = RandomVariate[NormalDistribution[], 18]; Correlation[a, b], {10000}],
  AxesLabel → {"Correlation", "Frequency"}]
```

Out[ ]=



Compare this to a similarly fat tailed random variable.
We get the normal distribution for the distribution.
Generate the p-value for 20 variables.

```
dist1 = NormalDistribution[0.3, 1];
ta = Table[ran = RandomVariate[dist1, 20];
    SurvivalFunction[StudentTDistribution[20], Mean[ran] / (StandardDeviation[ran] / Sqrt[20])], {10^4}];
(* Not very sure about this. *)
```

We can determine the mean of the table.

In[ ]:=
```
Mean[ta] (*Most options are below the p-value 0.05*)
```

Out[ ]=
```
0.175591
```

In[ ]:=
```
? Quantile
```
Out[ ]=

Symbol

Quantile[*data*, *p*] gives the estimate of the p^{th} quantile $\hat{q}_p$ of *data*.

Quantile[*data*, {p_1, p_2, …}] gives a list of quantiles p_1, p_2, ….

Quantile[*data*, *p*, {{*a*, *b*}, {*c*, *d*}}] uses the quantile definition specified by parameters *a*, *b*, *c*, *d*.

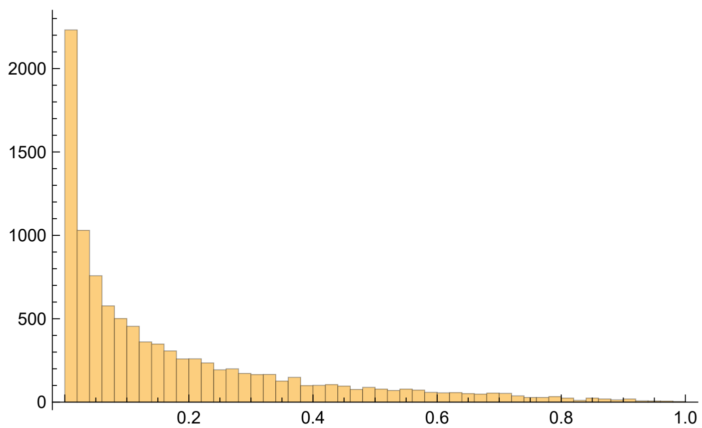
Quantile[*dist*, *p*] gives a quantile of the distribution *dist*.

We can see the various quantiles for various p-values.

In[ ]:=
```
Quantile[ta, 0.36590] // N
```
Out[ ]=
```
0.0499641
```

In[ ]:=
```
Quantile[ta, 0.025] // N
```
Out[ ]=
```
0.000794859
```

In[ ]:=
```
Histogram[ta, 60]
```
Out[ ]=



In[ ]:=
```
CDF[EmpiricalDistribution[ta], 0.01]
```
Out[ ]=
```
0.1457
```