```
Quit[];
```

Central Limit

χ^2 distribution

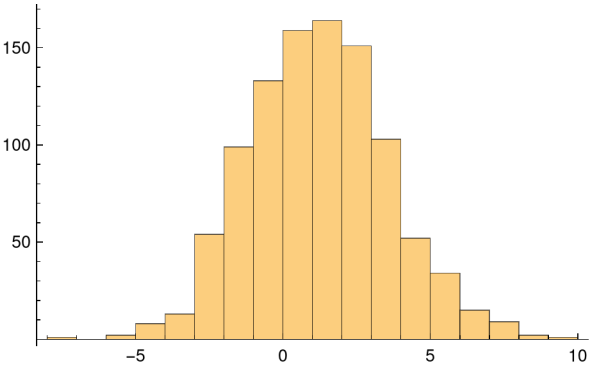
Capture/Recapture

Hypothesis Tests: z-Test

z-Test: H0 is true

In[161]:=
```
μ0 = 1.2;
Ndata = 1000;
μdata = μ0;
σdata = 2.3;
data = RandomVariate[NormalDistribution[μdata, σdata], {Ndata}];
Histogram[data]
```

Out[166]=



In[167]:=
```
(* This is what we would do *)
z = (Mean[data] - μ0) / (σdata / Sqrt[Ndata])
pvalue1 = 1 - CDF[NormalDistribution[], z] (* Right-sided*)
pvalue2 = CDF[NormalDistribution[], z] (* Left-sided*)
pvalue3 = 2 CDF[NormalDistribution[], -Abs[z]] (* Two-sided*)
```

Out[167]= 0.360679

Out[168]= 0.35917

Out[169]= 0.64083

Out[170]= 0.718339

In[171]:=
```
(* This is the Mathematica implementaion of the z-test *)
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
ZTest[data, σdata^2, μ0, "TestDataTable"]
```

Out[171]=

	Statistic	P-Value
Z	0.360679	0.35917

Out[172]=

	Statistic	P-Value
Z	0.360679	0.64083

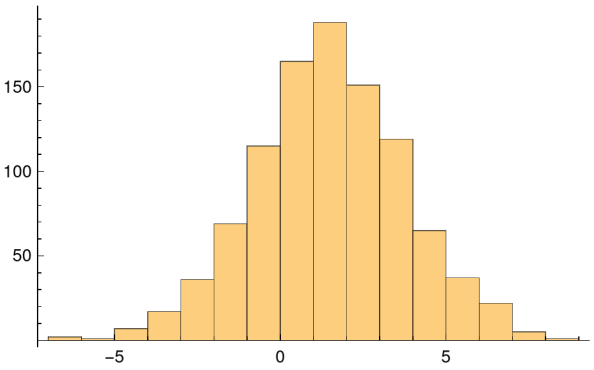
Out[173]=

	Statistic	P-Value
Z	0.360679	0.718339

z-Test: HA=H1 is true

In[192]:=
```
μ0 = 1.2;
Ndata = 1000;
μdata = μ0 + 0.35;
σdata = 2.3;
data = RandomVariate[NormalDistribution[μdata, σdata], {Ndata}];
Histogram[data]
```

Out[197]=



In[198]:=
```
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
ZTest[data, σdata^2, μ0, "TestDataTable"]
```

Out[198]=

	Statistic	P-Value
Z	3.75063	0.0000881952

Out[199]=

	Statistic	P-Value
Z	3.75063	0.999912

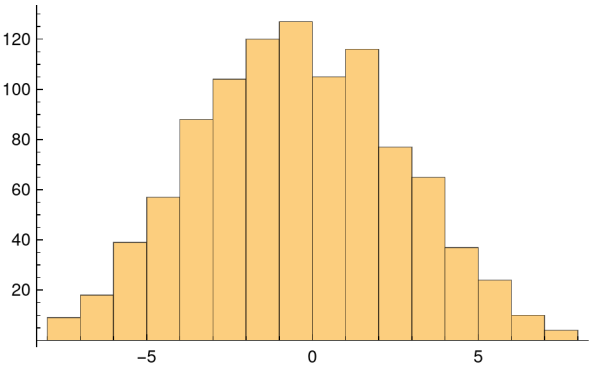
Out[200]=

	Statistic	P-Value
Z	3.75063	0.00017639

z-Test: the data is not normally distributed

In[225]:=
```
μ0 = 1.2;
Ndata = 1000;
μdata = μ0 + 0.35;
σdata = 2.3;
data = Join[RandomVariate [NormalDistribution [μdata, σdata], {Ndata / 2}],
    RandomVariate [NormalDistribution [μdata - 4, σdata], {Ndata / 2}]];
Histogram[
 data]
```

Out[230]=



In[231]:=
```
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
ZTest[data, σdata^2, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
ZTest[data, σdata^2, μ0, "TestDataTable"]
```

••• ZTest : At least one of the p-values in {0.0283455 }, resulting from a test for normality , is below 0.05` . The tests in {Z} require that the data is normally distributed .

Out[231]=

	Statistic	P-Value
Z	-22.4381	1.

••• ZTest : At least one of the p-values in {0.0283455 }, resulting from a test for normality , is below 0.05` . The tests in {Z} require that the data is normally distributed .

Out[232]=

	Statistic	P-Value
Z	-22.4381	8.36374×10^{-112}

••• ZTest : At least one of the p-values in {0.0283455 }, resulting from a test for normality , is below 0.05` . The tests in {Z} require that the data is normally distributed .

Out[233]=

	Statistic	P-Value
Z	-22.4381	1.67275×10^{-111}

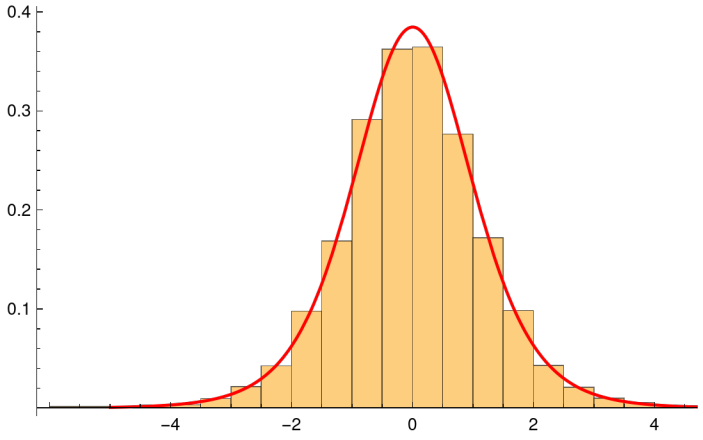
Hypothesis Tests: one-sample t-Test

t-statistic is Student's t

In[289]:=
```
(* A function that generates a value for the t-statistic with ndof dofs*)
Generatet[ndof_] := Module[{data, μ0, σ0},
    μ0 = Random[];
    σ0 = Random[];
    data = RandomVariate[NormalDistribution[μ0, σ0], {ndof + 1}];
    (Mean[data] - μ0) / (StandardDeviation[data] / Sqrt[ndof + 1])];
```

In[302]:=
```
(* The variable t is distributed as a student's t with Ndof *)
Ndof = 7;
ti = Table[Generatet[Ndof], {10^4}];
Show[Histogram[ti, Automatic, "PDF"], Plot[PDF[StudentTDistribution[Ndof], x],
  {x, -5, 5}, PlotRange → {All, {0, 0.55}}, PlotStyle → Red]]
```

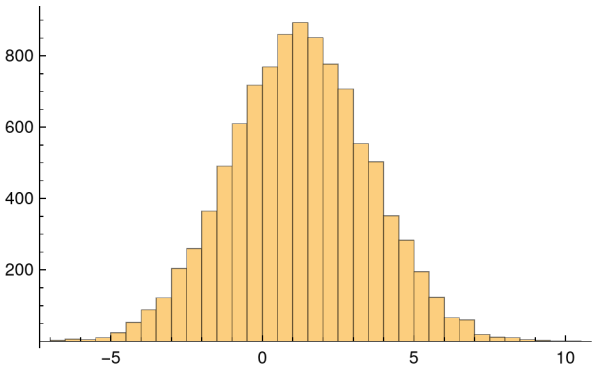
Out[304]=



t-Test: H0 is true

In[234]:=
```
μ0 = 1.2;
Ndata = 10 000;
μdata = μ0;
σdata = 2.3;
data = RandomVariate [NormalDistribution [μdata, σdata], {Ndata}];
Histogram [data]
```

Out[239]=



In[248]:=
```
(* This is what we would do *)
t = (Mean[data] - μ0) / (StandardDeviation [data] / Sqrt[Ndata])
pvalue1 = 1 - CDF[StudentTDistribution [Ndata - 1], t] (* Right-sided*)
pvalue2 = CDF[StudentTDistribution [Ndata - 1], t] (* Left-sided*)
pvalue3 = 2 CDF[StudentTDistribution [Ndata - 1], -Abs[t]] (* Two-sided*)
```

Out[248]= 1.67755

Out[249]= 0.0467332

Out[250]= 0.953267

Out[251]= 0.0934664

In[252]:=
```
(* This is the Mathematica implementaion of the z-test *)
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
TTest[data, μ0, "TestDataTable"]
```

Out[252]=

	Statistic	P-Value
T	1.67755	0.0467332

Out[253]=

	Statistic	P-Value
T	1.67755	0.953267

Out[254]=

	Statistic	P-Value
T	1.67755	0.0934664

t-Test: HA=H1 is true

In[318]:=
```
μ0 = 1.2;
Ndata = 10 000;
μdata = μ0 + 0.1;
σdata = 2.3;
data = RandomVariate [NormalDistribution [μdata, σdata], {Ndata}];
Histogram[data]
```

Out[323]=

In[328]:=
```
(* This is the Mathematica implementaion of the z-test *)
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
TTest[data, μ0, "TestDataTable"]
```

Out[328]=

	Statistic	P-Value
T	4.14334	0.0000172555

Out[329]=

	Statistic	P-Value
T	4.14334	0.999983

Out[330]=

	Statistic	P-Value
T	4.14334	0.000034511

t-Test: the data is not normally distributed

In[358]:=
```
μ0 = 1.2;
Ndata = 10 000;
μdata = μ0 + 0.1;
σdata = 2.3;
data = Join[RandomVariate [NormalDistribution [μdata, σdata], {Ndata / 2}],
    RandomVariate [NormalDistribution [μdata - 3, σdata], {Ndata / 2}]];
Histogram[
 data]
```

Out[363]=

In[364]:=
```
(* This is the Mathematica implementaion of the z-test *)
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
TTest[data, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
TTest[data, μ0, "TestDataTable"]
```

••• TTest : At least one of the p-values in {0.000413775 }, resulting from a test for normality , is below 0.05` . The tests in {T} require that the data is normally distributed .

Out[364]=

	Statistic	P-Value
T	−50.961	1.

••• TTest : At least one of the p-values in {0.000413775 }, resulting from a test for normality , is below 0.05` . The tests in {T} require that the data is normally distributed .

••• General : 4.06176644590730448412463221358759971`23.241328587689207*^-504 is too small to represent as a normalized machine number ; precision may be lost .

Out[365]=

	Statistic	P-Value
T	−50.961	0.

••• TTest : At least one of the p-values in {0.000413775 }, resulting from a test for normality , is below 0.05` . The tests in {T} require that the data is normally distributed .

••• General : 4.06176644590730448412463221358759971`23.241328587689207*^-504 is too small to represent as a normalized machine number ; precision may be lost .

Out[366]=

	Statistic	P-Value
T	−50.961	0.

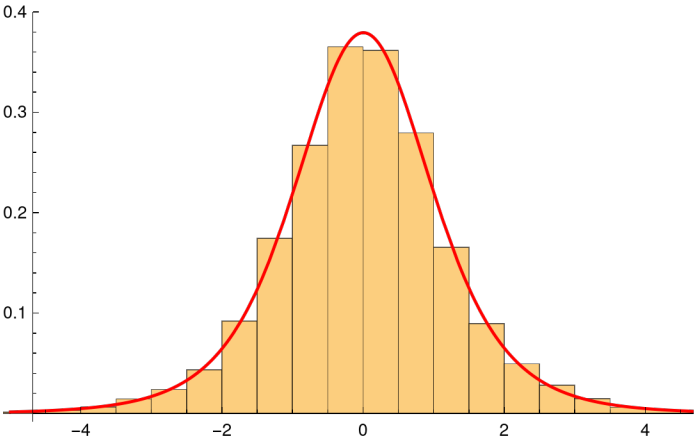
Hypothesis Tests: two-sample t-Test

two-sample t-statistic is Student's t

In[383]:=
```
(* A function that generates a value for the t-statistic with ndof dofs*)
Generatet2[ndof_] := Module[{ndata1, ndata2, data1, data2, μ1, μ2, σ, PooledVariance},
    μ1 = Random[]; μ2 = Random[];
    σ = RandomReal[{0, 2}];
    ndata1 = RandomInteger[{2, ndof}];
    ndata2 = ndof - ndata1 + 2;
    data1 = RandomVariate[NormalDistribution[μ1, σ], {ndata1}];
    data2 = RandomVariate[NormalDistribution[μ2, σ], {ndata2}];
    PooledVariance =
     (1 / ndata1 + 1 / ndata2) / ndof ((ndata1 - 1) Variance[data1] + (ndata2 - 1) Variance[data2]);
    (Mean[data1] - Mean[data2] - (μ1 - μ2)) / Sqrt[PooledVariance]];
```

In[393]:=
```
(* The variable t is distributed as a student's t with Ndof *)
Ndof = 5;
ti = Table[Generatet2[Ndof], {10^4}];
Show[Histogram[ti, Automatic, "PDF"], Plot[PDF[StudentTDistribution[Ndof], x],
  {x, -5, 5}, PlotRange → {All, {0, 0.55}}, PlotStyle → Red]]
```

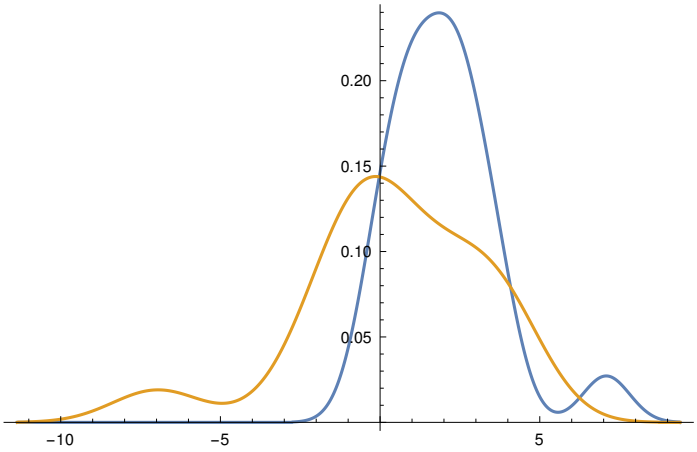
Out[395]=



two-sample t-Test: H0 is true

In[418]:=
```
μ0 = 1.2;
Ndata1 = 20;
Ndata2 = 15;
μdata1 = 1.5;
μdata2 = 0.3;
σdata = 2.3;
data1 = RandomVariate [NormalDistribution [μdata1, σdata], {Ndata1}];
data2 = RandomVariate [NormalDistribution [μdata2, σdata], {Ndata2}];
SmoothHistogram [{data1, data2}]
```

Out[426]=



In[449]:=
```
(* This is the Mathematica implementaion of the t-test *)
TTest[{data1, data2}, μ0, "TestDataTable ", AlternativeHypothesis → "Greater"]
TTest[{data1, data2}, μ0, "TestDataTable ", AlternativeHypothesis → "Less"]
TTest[{data1, data2}, μ0, "TestDataTable "]
```

Out[449]=

	Statistic	P-Value
T	0.504273	0.309621

Out[450]=

	Statistic	P-Value
T	0.504273	0.690379

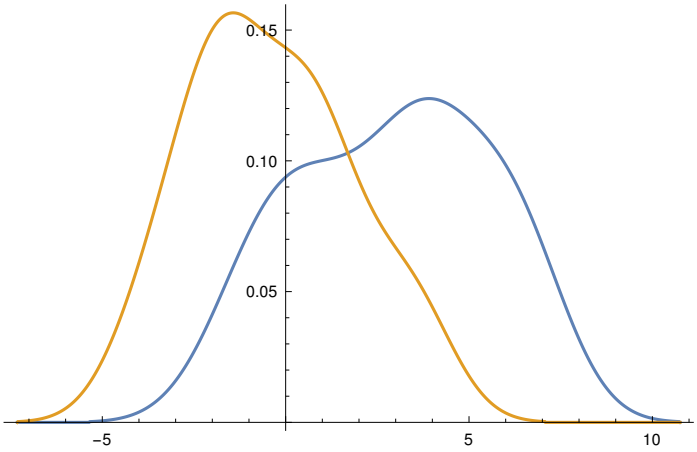
Out[451]=

	Statistic	P-Value
T	0.504273	0.619243

two-sample t-Test: HA=H1 is true

In[548]:=
```
μ0 = 1.2;
Ndata1 = 20;
Ndata2 = 15;
μdata1 = 2.9;
μdata2 = -0.7;
σdata = 2.3;
data1 = RandomVariate [NormalDistribution [μdata1, σdata], {Ndata1}];
data2 = RandomVariate [NormalDistribution [μdata2, σdata], {Ndata2}];
SmoothHistogram [{data1, data2}]
```

Out[556]=



In[557]:=
```
(* This is the Mathematica implementaion of the t-test *)
TTest[{data1, data2}, μ0, "TestDataTable", AlternativeHypothesis → "Greater"]
TTest[{data1, data2}, μ0, "TestDataTable", AlternativeHypothesis → "Less"]
TTest[{data1, data2}, μ0, "TestDataTable"]
```

Out[557]=

	Statistic	P-Value
T	2.54491	0.00789534

Out[558]=

	Statistic	P-Value
T	2.54491	0.992105

Out[559]=

	Statistic	P-Value
T	2.54491	0.0157907

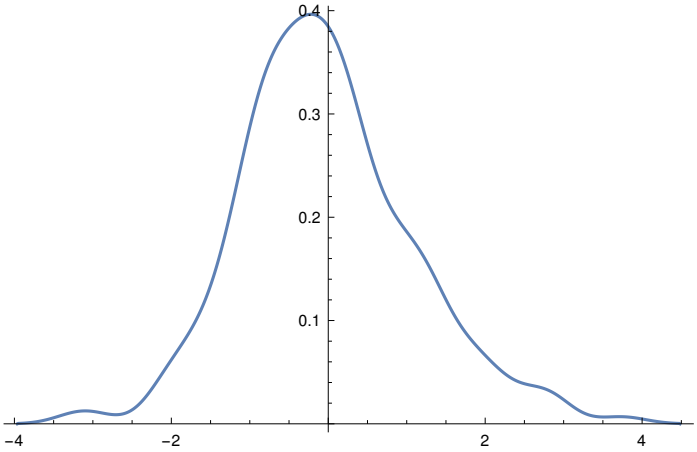
Hypothesis Tests: Kolmogorov-Smirnov

One - sample

In[593]:=
```
Ndata = 200;
data = RandomVariate [StudentTDistribution [13], {Ndata}];
```

In[595]:=
```
SmoothHistogram [data]
```

Out[595]=



In[596]:=
```
KolmogorovSmirnovTest [data , StudentTDistribution [13], "TestDataTable "]
KolmogorovSmirnovTest [data , StudentTDistribution [4], "TestDataTable "]
KolmogorovSmirnovTest [data , NormalDistribution [], "TestDataTable "]
```

Out[596]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.067956	0.30041

Out[597]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0724085	0.233623

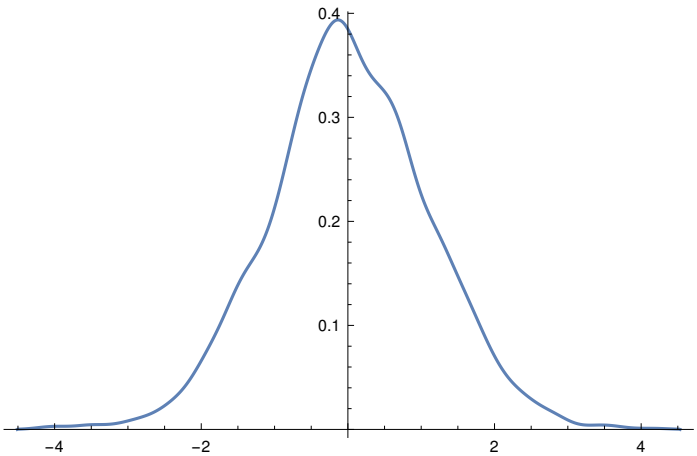
Out[598]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0658866	0.335648

In[599]:=
```
Ndata = 2000 ;
data = RandomVariate [StudentTDistribution [13], {Ndata}];
```

In[601]:=
```
SmoothHistogram [data]
```

Out[601]=



In[602]:=
```
KolmogorovSmirnovTest [data , StudentTDistribution [13], "TestDataTable "]
KolmogorovSmirnovTest [data , StudentTDistribution [4], "TestDataTable "]
KolmogorovSmirnovTest [data , NormalDistribution [], "TestDataTable "]
```

Out[602]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0175413	0.563513

Out[603]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0294135	0.0615822

Out[604]=

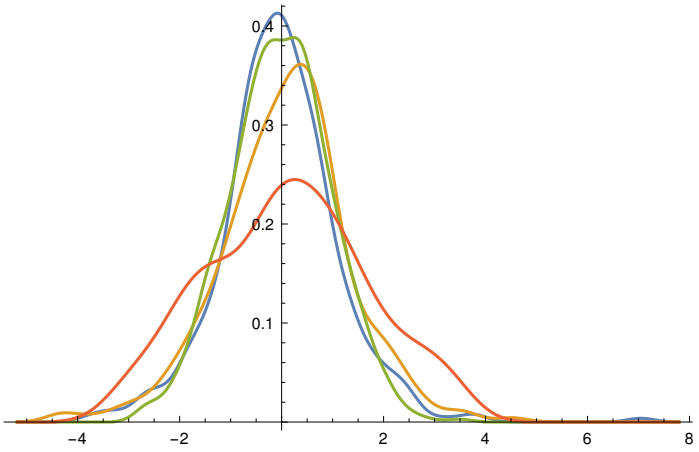
	Statistic	P-Value
Kolmogorov -Smirnov	0.0279352	0.0865304

Two - sample

In[657]:=
```
Ndata1 = 400;
Ndata2 = 300;
Ndata3 = 600;
Ndata4 = 200;
data1 = RandomVariate [StudentTDistribution [5], {Ndata1}];
data2 = RandomVariate [StudentTDistribution [5], {Ndata2}];
data3 = RandomVariate [NormalDistribution [], {Ndata3}];
data4 = RandomVariate [TriangularDistribution [{-4, 4}], {Ndata4}];
```

In[665]:=
```
SmoothHistogram [{data1 , data2 , data3 , data4}]
```

Out[665]=



In[666]:=
```
KolmogorovSmirnovTest [data1 , data2 , "TestDataTable "]
KolmogorovSmirnovTest [data1 , data3 , "TestDataTable "]
KolmogorovSmirnovTest [data2 , data3 , "TestDataTable "]
KolmogorovSmirnovTest [data1 , data4 , "TestDataTable "]
```

Out[666]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.1075	0.0380444

Out[667]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0425	0.778877

Out[668]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.075	0.210552

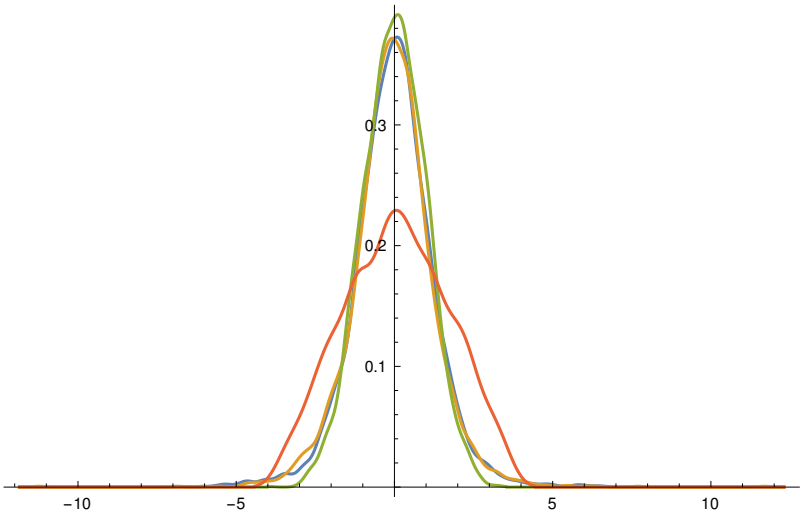
Out[669]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.165	0.00132728

In[670]:=
```
Ndata1 = 4000;
Ndata2 = 3000;
Ndata3 = 6000;
Ndata4 = 2000;
data1 = RandomVariate [StudentTDistribution [5], {Ndata1}];
data2 = RandomVariate [StudentTDistribution [5], {Ndata2}];
data3 = RandomVariate [NormalDistribution [], {Ndata3}];
data4 = RandomVariate [TriangularDistribution [{-4, 4}], {Ndata4}];
```

In[679]:=
```
SmoothHistogram [{data1 , data2 , data3 , data4}]
```

Out[679]=



In[680]:=
```
KolmogorovSmirnovTest [data1, data2, "TestDataTable "]
KolmogorovSmirnovTest [data1, data3, "TestDataTable "]
KolmogorovSmirnovTest [data2, data3, "TestDataTable "]
KolmogorovSmirnovTest [data1, data4, "TestDataTable "]
```

Out[680]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.01725	0.687455

Out[681]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0361667	0.0037523

Out[682]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0428333	0.00129969

Out[683]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.1265	0.

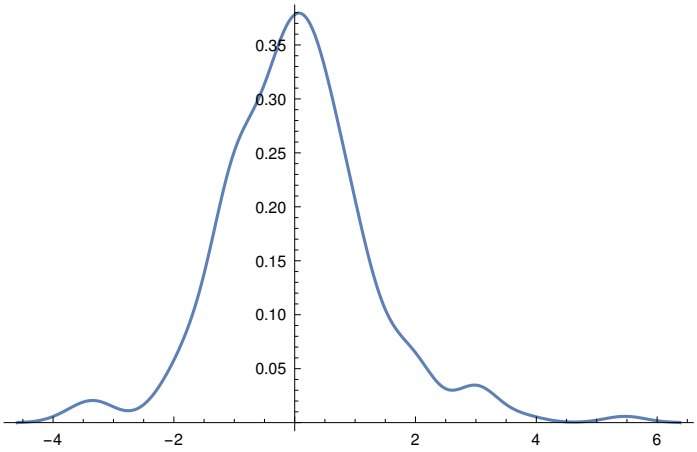
Hypothesis Tests: ShapiroWilk vs KS

One - sample H0 false

In[710]:=
```
Ndata = 200;
data = RandomVariate [StudentTDistribution [7], {Ndata}];
```

In[712]:=
```
SmoothHistogram [data]
```

Out[712]=



In[713]:=
```
KolmogorovSmirnovTest [data, NormalDistribution [], "TestDataTable "]
ShapiroWilkTest [data, "TestDataTable "]
```

Out[713]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0513125	0.648948

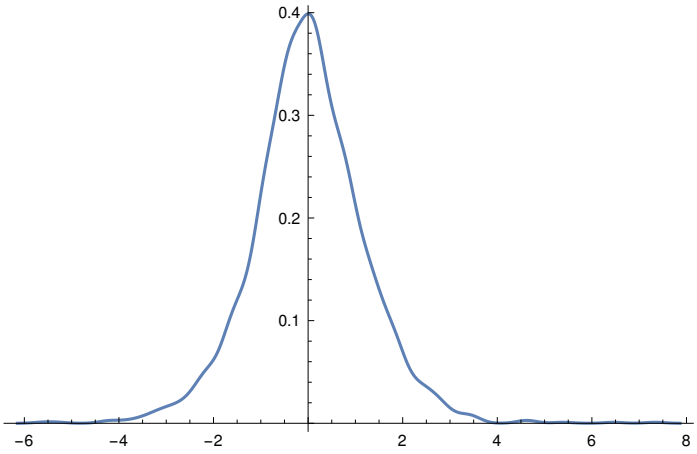
Out[714]=

	Statistic	P-Value
Shapiro -Wilk	0.965924	0.0000915426

In[715]:=
```
Ndata = 2000;
data = RandomVariate [StudentTDistribution [7], {Ndata}];
```

In[717]:=
```
SmoothHistogram [data]
```

Out[717]=



In[718]:=
```
KolmogorovSmirnovTest [data , NormalDistribution [], "TestDataTable "]
ShapiroWilkTest [data , "TestDataTable "]
```

Out[718]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0291319	0.0657923

Out[719]=

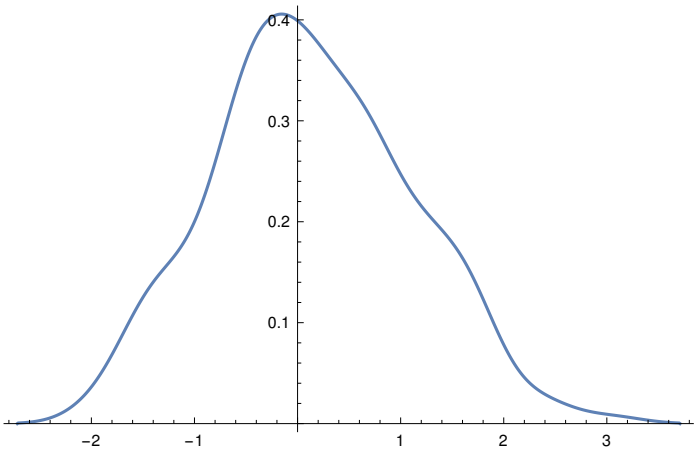
	Statistic	P-Value
Shapiro -Wilk	0.982504	6.09403×10^{-15}

One - sample H0 true

In[730]:=
```
Ndata = 200;
data = RandomVariate [NormalDistribution [], {Ndata}];
```

In[732]:=
```
SmoothHistogram [data]
```

Out[732]=



In[733]:=
```
KolmogorovSmirnovTest [data , NormalDistribution [], "TestDataTable "]
ShapiroWilkTest [data , "TestDataTable "]
```

Out[733]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0754465	0.194888

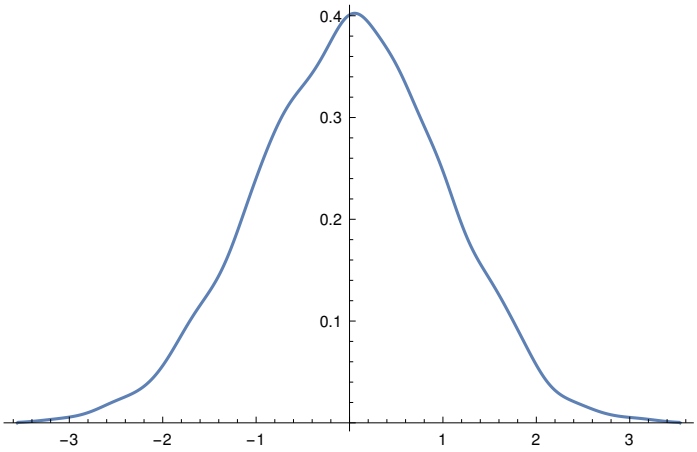
Out[734]=

	Statistic	P-Value
Shapiro -Wilk	0.992011	0.343199

In[735]:=
```
Ndata = 2000 ;
data = RandomVariate [NormalDistribution [], {Ndata}];
```

In[737]:=
```
SmoothHistogram [data]
```

Out[737]=



In[738]:=
```
KolmogorovSmirnovTest [data , NormalDistribution [], "TestDataTable "]
ShapiroWilkTest [data , "TestDataTable "]
```

Out[738]=

	Statistic	P-Value
Kolmogorov -Smirnov	0.0167374	0.62345

Out[739]=

	Statistic	P-Value
Shapiro -Wilk	0.999406	0.813273