Muon

Import and Clean Data

The first step is to import the data. Mathematica only wants a .dat file not a .data file. On my mac I just deleted the extra ‘a’ from the file name and everything worked. The first choice imports the unsifted data and the second imports the sifted. The second part of the Import function imports all of the first column and not any of the unnecessary time stamps in the .data file.

Now, Histogram and delete the first bin (both the height and location) and delete the added bin location (bin location over 20,000) to keep the set rectangular and Mathematica happy and able to perform the next step. Why delete the first bin? Then, transpose the histogram list and set equal a variable. The trick is the NonLInearModelFit command requires a guess. I did not remove the first bin from the histogram in the money plot.

In[116]:=
```
(*Clear["Global`*"]*)
(*This works with the unsifted file pretty quick*)
(*sift1 = DeleteCases[
   ReadList["/Users/christophernewey/Documents/gen/winter/unsifted.dat", Number],
   x_/;x>39999];*)
(*sift1= Import["/Users/christophernewey/Desktop/sift.dat", {"Data",All,1}];*)
(*sift  = N[sift1 * 10^-3];*)
bins = {20*10^-3};
(*Curly brackets=bin width, no curly brackets=number of bins,
other option="Sturges", You may have to delete more bins accordingly*)
badbins  = 8;
binlocs = HistogramList[sift, bins][[1, badbins ;; -2]];
Length[binlocs];
heightlocs = HistogramList[sift, bins][[2, badbins ;; -1]];
Length[heightlocs];
great = Transpose[{binlocs, heightlocs}];
Length[great];(*Length[great] = number of bins check*)
```

Perform NLM Fit

In[124]:=
```
Clear[tau, c, a]
f[t_] := c + a e^(-t/tau)
fit = NonlinearModelFit[great, f[t], { {c, 1}, {a, 100}, {tau, 2}},
  t, MaxIterations → Infinity, AccuracyGoal → Infinity]
  (* ,WorkingPrecision→50 use this to get total resid=0. *)
fit["BestFitParameters"]
MatrixForm[fit["CovarianceMatrix"]]
fit["ParameterTable"]
tau = tau /. fit["BestFitParameters"]

Show[Histogram[sift, bins, PlotRange → All,
  ChartBaseStyle → EdgeForm[{Opacity[.1, Black], Thickness[.00001]}]],
 Plot[fit["BestFit"], {t, 0, 20}, PlotRange → Full], ImageSize → Large]
```

Out[126]= FittedModel[$0.682886 + 117.035\, e^{-0.492175\, t}$]

Out[127]= {c → 0.682886, a → 117.035, tau → 2.0318}

Out[128]//MatrixForm=

$$\begin{pmatrix} 0.0218584 & 0.00789386 & -0.00164846 \\ 0.00789386 & 0.707394 & -0.0126818 \\ -0.00164846 & -0.0126818 & 0.000483807 \end{pmatrix}$$

General: Exp[−1499.13] is too small to represent as a normalized machine number; precision may be lost.

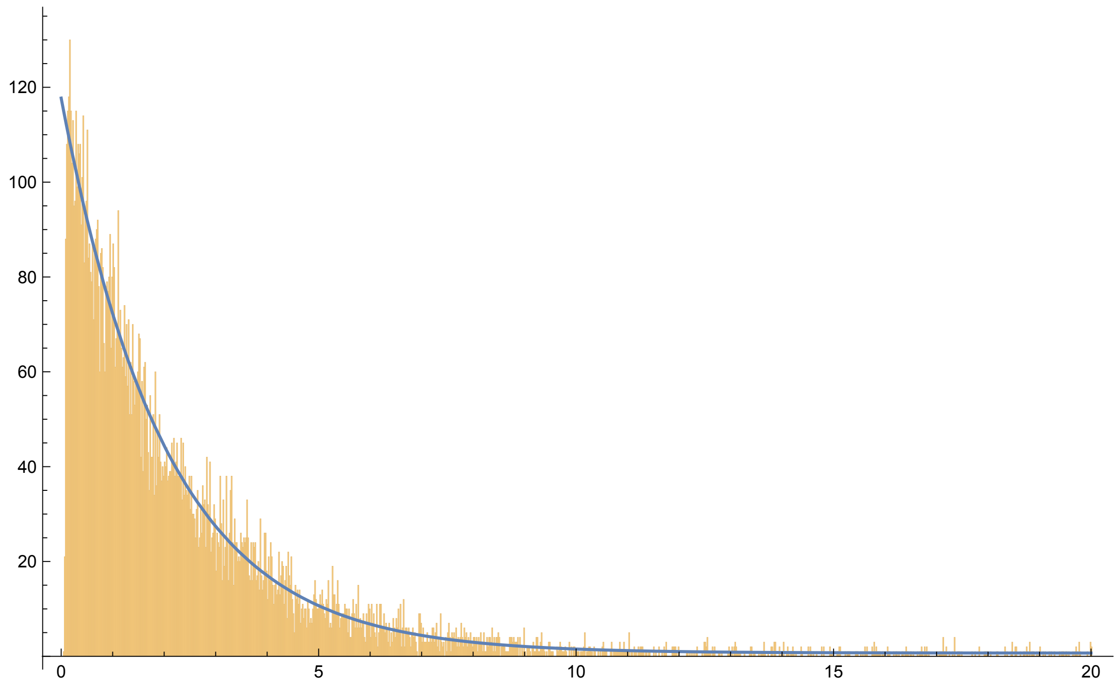
General: Exp[−1123.48] is too small to represent as a normalized machine number; precision may be lost.

Out[129]=

	Estimate	Standard Error	t-Statistic	P-Value
c	0.682886	0.147846	4.61891	4.36646×10^{-6}
a	117.035	0.841067	139.151	0.
tau	2.0318	0.0219956	92.3728	0.

Out[130]= 2.0318

Out[131]=

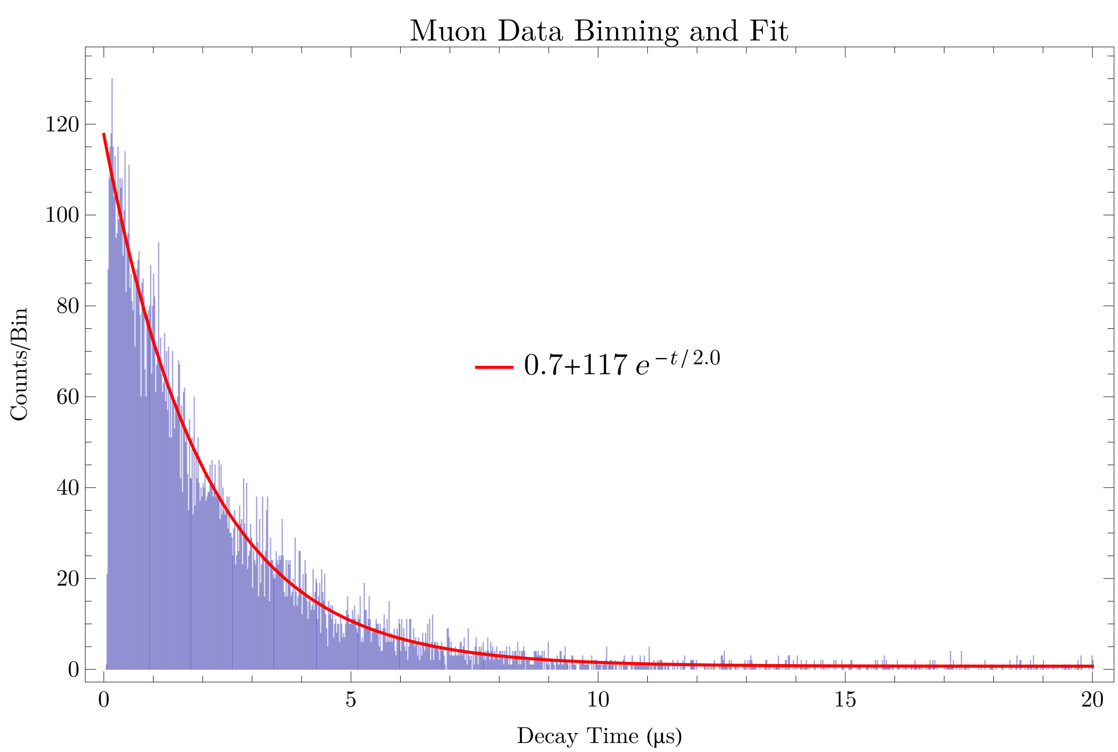


Nicer Plot

And here we have a nice looking version with a whole lot of options. The nice thing about coding this in is that if you need to change one thing it is very easy. Changing the font to a computer modern looks better if your computer has it.

In[132]:=
```
as = IntegerPart[a /. fit["BestFitParameters"]];
cs = NumberForm[c /. fit["BestFitParameters"], {2, 1}];
taus = NumberForm[tau, {2, 1}];
label = StringTemplate["``+``e^{-t/``}"][cs, as, taus];
Show[
 Histogram[sift, bins,
  PlotRange → Full,
  ChartStyle → {Opacity[.3, Blue]},
  ChartBaseStyle →
   EdgeForm[{Opacity[.1, Black],
     Thickness[.001]}]],
 Plot[fit["BestFit"], {t, 0, 20},
  PlotRange → Full,
  PlotStyle → Red,
  PlotLegends →
   Placed[{Style[label, 16, FontFamily → "Latin Modern Math"]}, {.5, .5}]],

 Frame → {{True, True}, {True, True}},
 FrameLabel → {"Decay Time (μs)", "Counts/Bin"},
 LabelStyle → {12, GrayLevel[0], FontFamily → "Latin Modern Math"},
 PlotLabel → Style["Muon Data Binning and Fit", {.1, .9}, GrayLevel[0],
   FontSize → 16, FontFamily → "Latin Modern Math"], ImageSize → Large]
```

Out[136]=



Least Squares and Weighted Least Squared Fits

Perform Fit

In[137]:=
```
(*  http://mathworld.wolfram.com/LeastSquaresFittingExponential.html  *)
ExponentialFit[x0_List, y0_List] := Module[
        {n = Length[x], i, sxlny, slny, slnx, sx2, sx, b,
         y = ReplaceAll[y0, 0 → 1]},
        sxlny = Sum[x[[i]] Log[y[[i]]], {i, n}];
        slny = Sum[Log[y[[i]]], {i, n}];
        sx2 = Sum[x[[i]]^2, {i, n}];
        sx = Apply[Plus, x];

   {Exp[(slny sx2 - sx sxlny) / (n sx2 - sx^2)], (n sxlny - sx slny) / (n sx2 - sx^2)}]
ExponentialWeightedFit[x0_List, y0_List] := Module[
        {n = Length[x], i, sxlny, sx2y, sxy, slny, sx2, sx, sy, sylny,
           sxylny, a, b,
         y = ReplaceAll[y0, 0 → 1]},
        sxlny = Sum[x[[i]] Log[y[[i]]], {i, n}];
        sx2y = Sum[x[[i]]^2 × y[[i]], {i, n}];
        sxy = x.y;
        slny = Sum[Log[y[[i]]], {i, n}];
        sx2 = Sum[x[[i]]^2, {i, n}];
        sx = Apply[Plus, x];
        sy = Apply[Plus, y];
        sylny = Sum[y[[i]] Log[y[[i]]], {i, n}];
        sxylny = Sum[x[[i]] × y[[i]] Log[y[[i]]], {i, n}];
        {a, b} = {sx2y slny - sxy sxylny, -sxy slny + sy sxylny} / (sy sx2y - sxy^2);
        {Exp[a], b}]
```

In[139]:=
```
x = great[[All, 1]];
y = great[[All, 2]];
cc = NumberForm[Mean[great[[Length[great] - 15 ;; Length[great], 2]]], {2, 1}];
ccc = N[Mean[great[[Length[great] - 15 ;; Length[great], 2]]]];
(*last 15 bins to calculate average for background count*)
a = N[ExponentialFit[x, y]]
b = N[ExponentialWeightedFit[x, y]]
lsqfit[t_] = ccc + a[[1]] * Exp[t * a[[2]]];
wlsqfit[t_] = ccc + b[[1]] * Exp[t * b[[2]]];
-1/N[a[[2]]]
-1/N[b[[2]]]
```

Out[143]= {33.4992, -0.225488}

Out[144]= {86.4139, -0.31693}

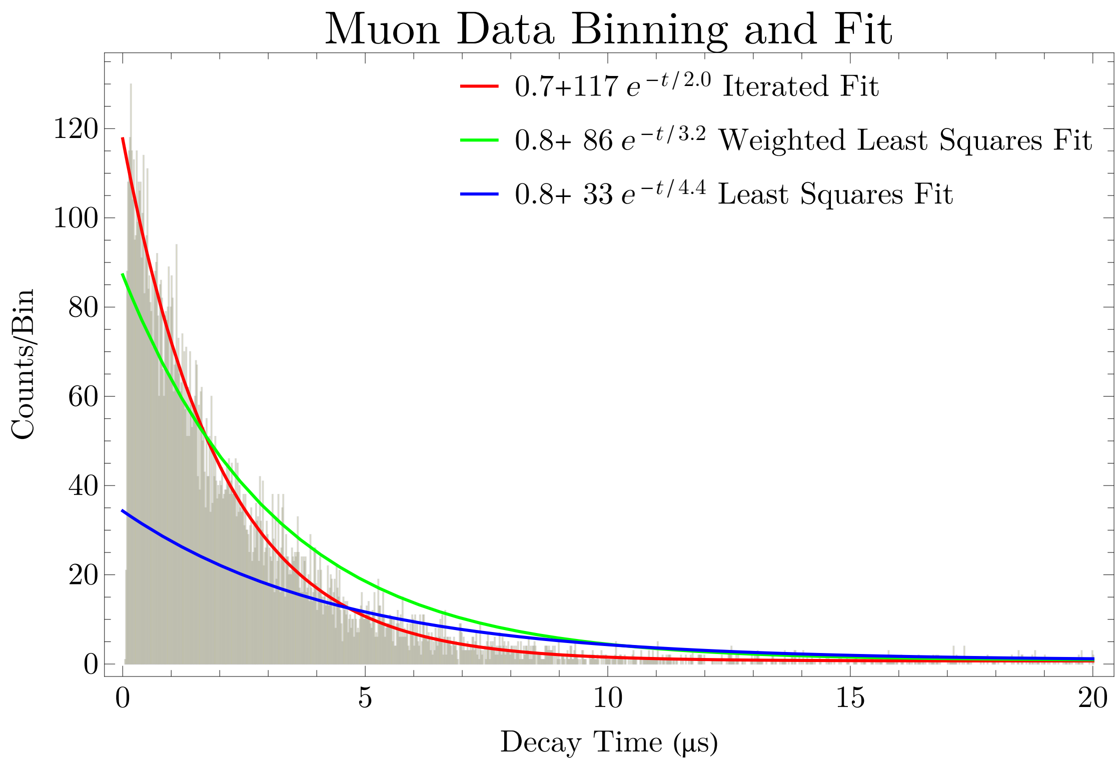
Out[147]= 4.43483

Out[148]= 3.15527

Plot

In[149]:=
```
as = IntegerPart[a /. fit["BestFitParameters"]];
cs = NumberForm[c /. fit["BestFitParameters"], {2, 1}];
taus = NumberForm[tau, {2, 1}];
aa = IntegerPart[a[[1]]];
ae = NumberForm[-1/N[a[[2]]], {2, 1}];
bb = IntegerPart[b[[1]]];
be = NumberForm[-1/N[b[[2]]], {2, 1}];
label = StringTemplate["``+``e^(-t/``) Iterated Fit"][cs, as, taus];
label1 = StringTemplate["``+ ``e^(-t/``) Weighted Least Squares Fit"][cc, bb, be];
label2 = StringTemplate["``+ ``e^(-t/``) Least Squares Fit"][cc, aa, ae];

Show[
 Histogram[sift, bins, PlotRange → Full, ChartStyle → {Opacity[.1, Yellow]},
  ChartBaseStyle → EdgeForm[{Opacity[.3, Gray], Thickness[.001]}]],

 Plot[{fit["BestFit"], wlsqfit[t], lsqfit[t]},
  {t, 0, 20}, PlotRange → Full, PlotStyle → {Red, Green, Blue},
  PlotLegends → Placed[{label, label1, label2}, {Right, Top}],
  LabelStyle → {16, GrayLevel[0], FontFamily → "Latin Modern Math"}],
 Frame → True,
 FrameLabel → {"Decay Time (μs)", "Counts/Bin"},
 LabelStyle → {16, GrayLevel[0], FontFamily → "Latin Modern Math"},
 PlotLabel → Style["Muon Data Binning and Fit", {.1, .9}, GrayLevel[0],
   FontSize → 24, FontFamily → "Latin Modern Math"], ImageSize → Large]
```

Out[159]=


Muon Data Binning and Fit
0.7+117 $e^{-t/2.0}$ Iterated Fit
0.8+ 86 $e^{-t/3.2}$ Weighted Least Squares Fit
0.8+ 33 $e^{-t/4.4}$ Least Squares Fit
Counts/Bin
0
20
40
60
80
100
120
0
5
10
15
20
Decay Time (μs)

Residual Plot and Error Plots

In[160]:=
```
time = great[[All, 1]];
height = great[[All, 2]];

nlmfitresiduals =
  Table[{time[[i]], height[[i]] - fit[time[[i]]]}, {i, 1, Length[time]}];
weightedfitresiduals = Table[{time[[i]], height[[i]] - wlsqfit[time[[i]]]},
   {i, 1, Length[time]}];
unweightedfitresiduals = Table[{time[[i]], height[[i]] - lsqfit[time[[i]]]},
   {i, 1, Length[time]}];

Total[nlmfitresiduals[[All, 2]]]
Total[weightedfitresiduals[[All, 2]]]
Total[unweightedfitresiduals[[All, 2]]]

rsenlm = Sqrt[Total[nlmfitresiduals[[All, 2]]^2] / (Length[height] - 2)];
rsewlsq = Sqrt[Total[weightedfitresiduals[[All, 2]]^2] / (Length[height] - 2)];
rselsq = Sqrt[Total[unweightedfitresiduals[[All, 2]]^2] / (Length[height] - 2)];

lbres1 = StringTemplate["Nlm RSE = ``"][NumberForm[rsenlm, {3, 2}]];
lbres2 = StringTemplate["Weighted RSE = ``"][NumberForm[rsewlsq, {3, 2}]];
```

$$\text{lbres3 = StringTemplate["Unweighted RSE = ``\textbackslash nRSE = } \sqrt{\frac{\sum_{i=1}^{n}(y_i - f(x_i))^2}{(n-2)}}\text{"][}$$

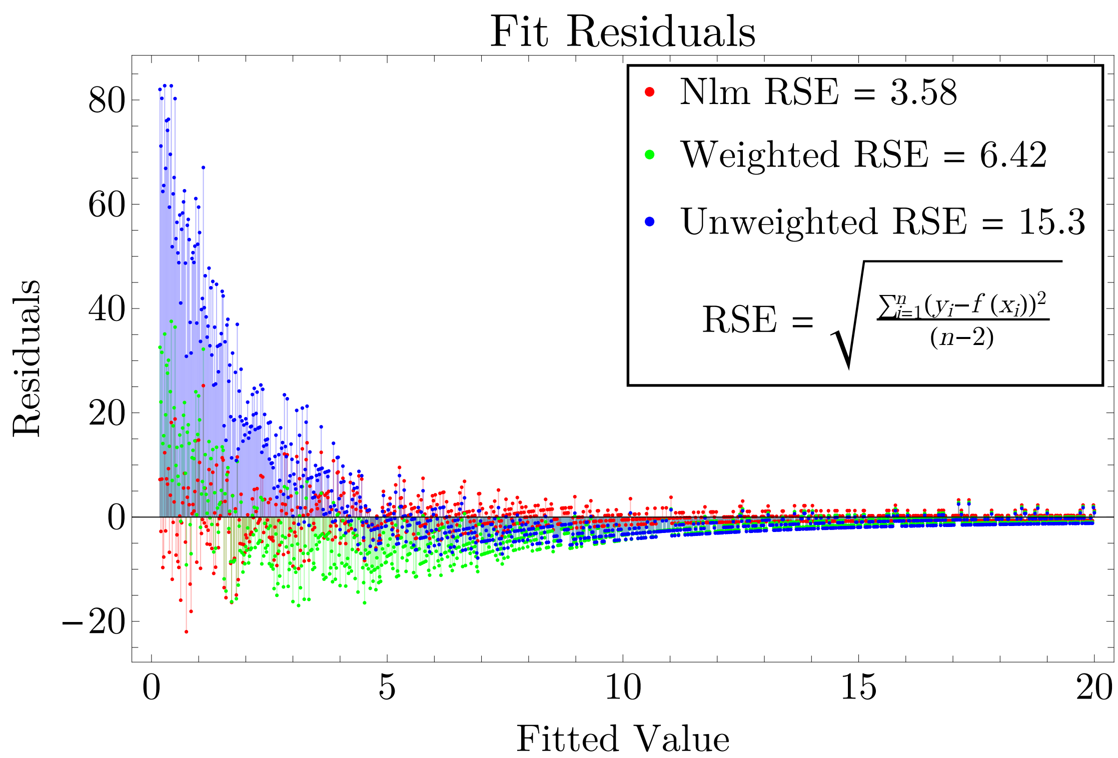
```
   NumberForm[rselsq, {3, 1}]];

ListPlot[
 {nlmfitresiduals, weightedfitresiduals, unweightedfitresiduals}, PlotRange → All,
 Filling → Axis,
 PlotStyle → {Red, Green, Blue},
 Frame → True,
 FrameLabel → {"Fitted Value", "Residuals"},
 PlotLabel → "Fit Residuals",
 PlotLegends → Placed[LineLegend[{lbres1, lbres2, lbres3},
    LegendFunction → (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
        FrameStyle → Directive[Black]] &)], {Right, Top}],
 LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
 ImageSize → Large]
```

Out[165]= 6.25278×10^{-13}

Out[166]= -2087.86

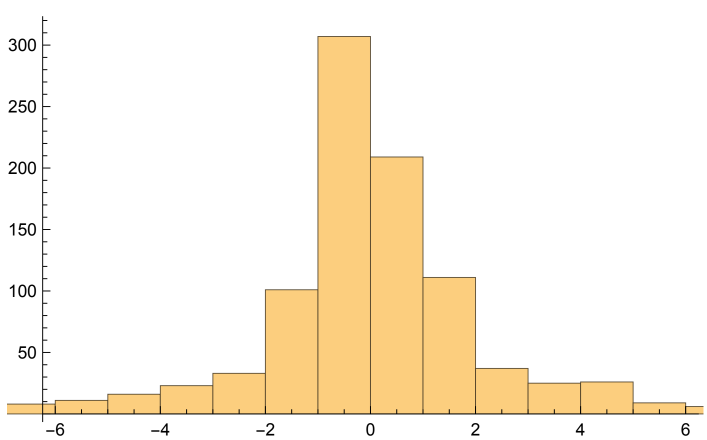
Out[167]= 3738.73

Out[174]=

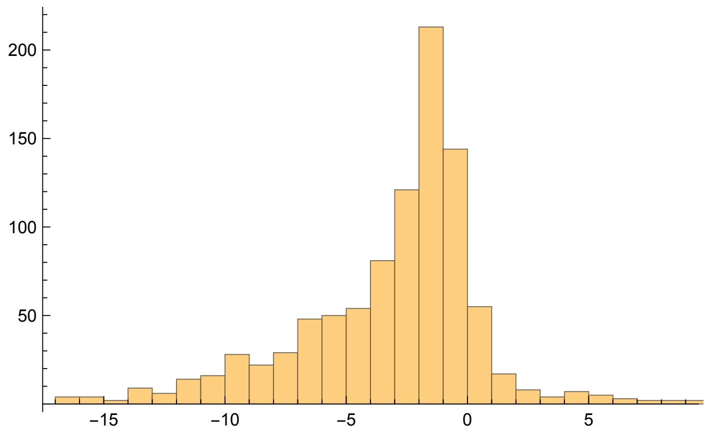


In[175]:=
```
nr = nlmfitresiduals[[ ;; , 2]];
Histogram[nr, {1}]
wr = weightedfitresiduals[[ ;; , 2]];
Histogram[weightedfitresiduals[[ ;; , 2]], {1}]
ur = unweightedfitresiduals[[ ;; , 2]];
Histogram[unweightedfitresiduals[[ ;; , 2]], {1}]
FindDistribution[nr]
FindDistribution[wr]
FindDistribution[ur]
```

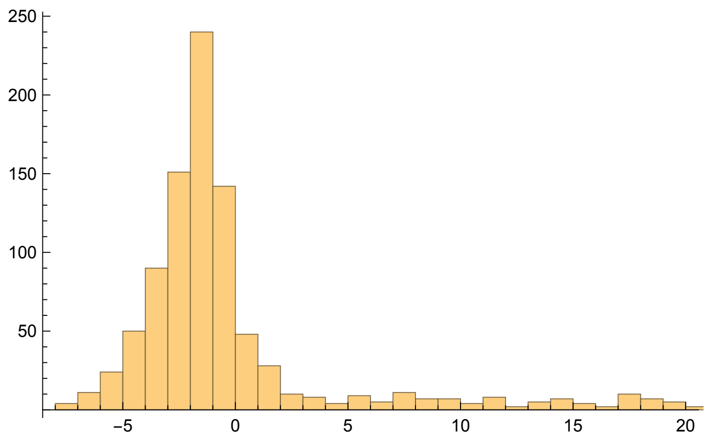
Out[176]=



Out[178]=



Out[180]=



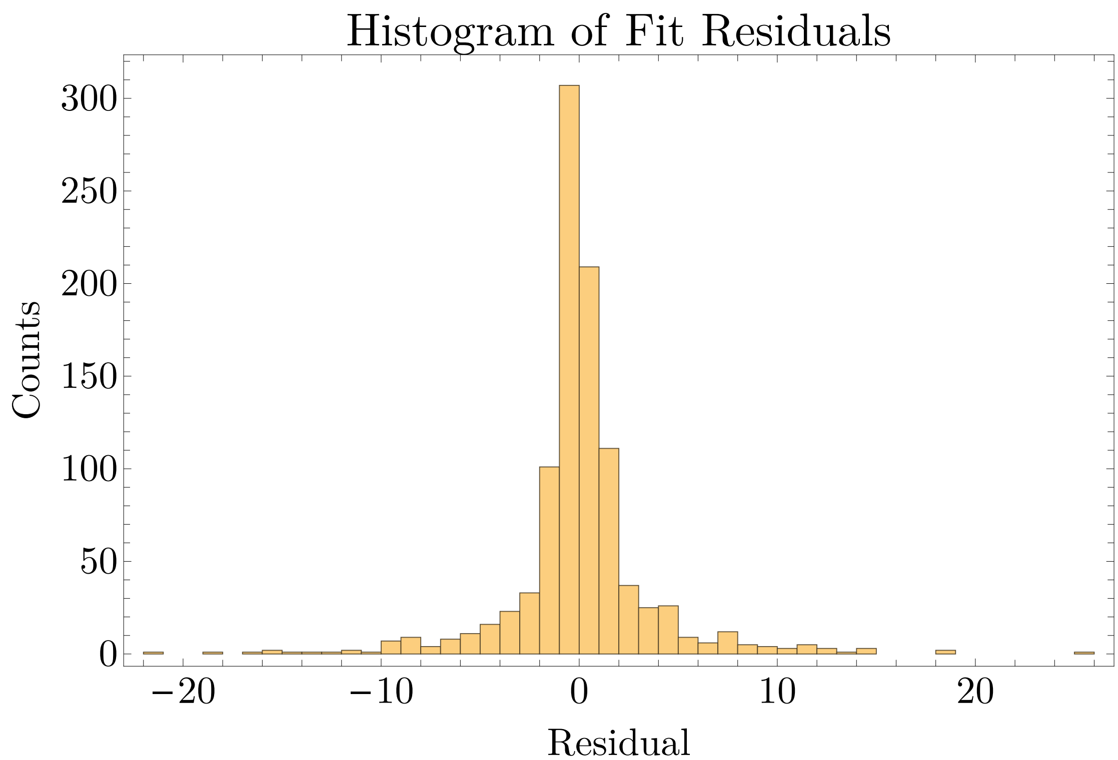
Out[181]=
```
MixtureDistribution[{0.663599, 0.336401},
  {NormalDistribution[-0.125576, 1.0116], NormalDistribution[1.09245, 6.98646]}]
```

Out[182]=
```
StudentTDistribution[-1.90759, 2.12967, 1.4905]
```

Out[183]=
```
MixtureDistribution[{0.830551, 0.169449},
  {LaplaceDistribution[-1.87905, 1.27691], GammaDistribution[1.32685, 23.211]}]
```

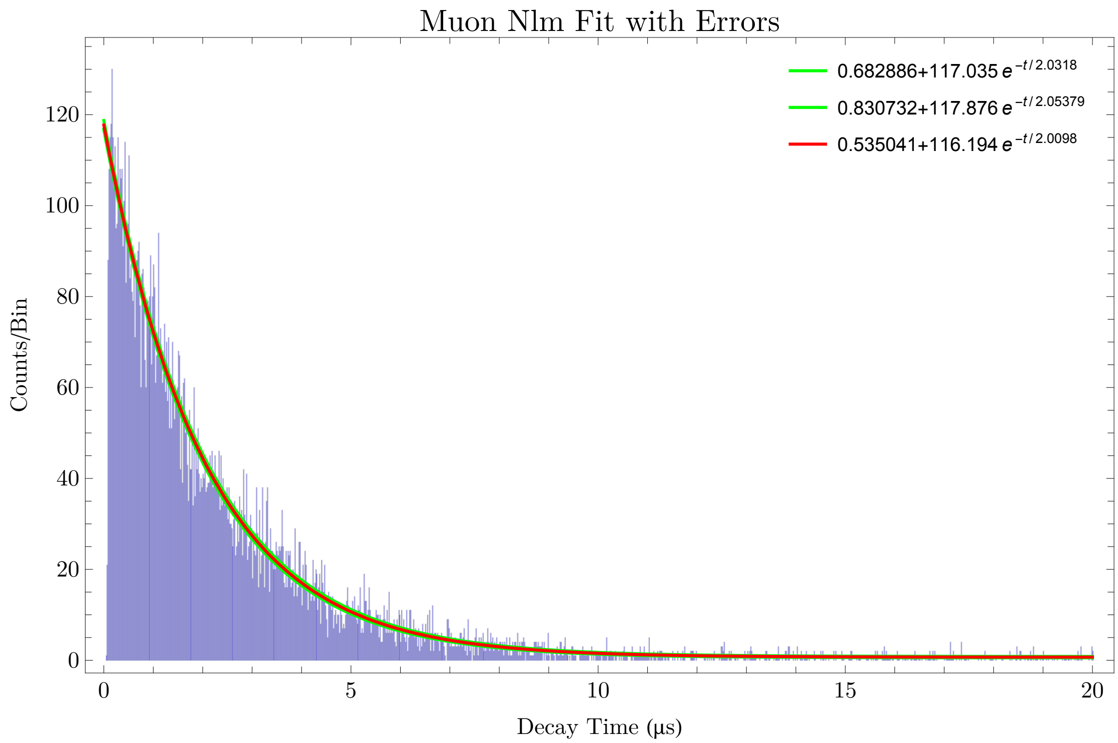
In[184]:=
```
Show[Histogram[nlmfitresiduals[[ ;; , 2]], {1}], PlotRange → All,
  Filling → Axis,
  Axes → False,
  PlotStyle → {Red, Green, Blue},
  Frame → True,
  FrameLabel → {"Residual", "Counts"},
  PlotLabel → "Histogram of Fit Residuals",
  PlotLegends → Placed[LineLegend[{lbres1, lbres2, lbres3},
     LegendFunction → (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
         FrameStyle → Directive[Black]] &)], {Right, Top}],
  LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
  ImageSize → Large]
```

Out[184]=



```
In[185]:= a1 = a /. fit["BestFitParameters"];
c1 = c /. fit["BestFitParameters"];
a2 = (a /. fit["BestFitParameters"]) + fit["ParameterErrors"][[2]];
c2 = (c /. fit["BestFitParameters"]) + fit["ParameterErrors"][[1]];
tau2 = tau + fit["ParameterErrors"][[3]];
a3 = (a /. fit["BestFitParameters"]) - fit["ParameterErrors"][[2]];
c3 = (c /. fit["BestFitParameters"]) - fit["ParameterErrors"][[1]];
tau3 = tau - fit["ParameterErrors"][[3]];

taus = NumberForm[tau, {2, 1}];
label = StringTemplate["``+``e^-t/``"][c1, a1, tau];
label2 = StringTemplate["``+``e^-t/``"][c2, a2, tau2];
label3 = StringTemplate["``+``e^-t/``"][c3, a3, tau3];
Show[
 Histogram[sift, bins,
  PlotRange → Full,
  ChartStyle → {Opacity[.3, Blue]},
  ChartBaseStyle →
   EdgeForm[{Opacity[.1, Black],
     Thickness[.001]}]],
 Plot[{c2 + a2 * Exp[-t / tau2], c3 + a3 * Exp[-t / tau3], fit["BestFit"]}, {t, 0, 20},
  PlotRange → Full,
  PlotStyle → {Green, Green, Red},
  PlotLegends → Placed[LineLegend[{label, label2, label3},
     LegendFunction → (Framed[#, FrameMargins → 0, Background → White,
         FrameStyle → Directive[White]] &)], {Right, Top}], Frame → All],
 Frame → {{True, True}, {True, True}},
 FrameLabel → {"Decay Time (μs)", "Counts/Bin"},
 LabelStyle → {12, GrayLevel[0], FontFamily → "Latin Modern Math"},
 PlotLabel → Style["Muon Nlm Fit with Errors", {.1, .9}, GrayLevel[0],
   FontSize → 16, FontFamily → "Latin Modern Math"], ImageSize → Large]
```

Out[197]=



Step 13

Code

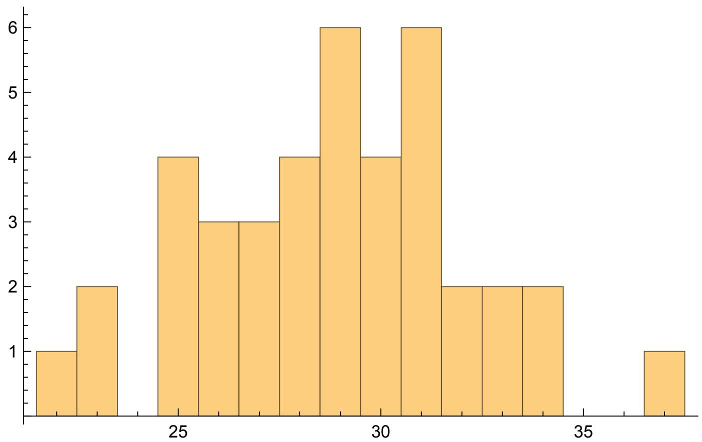
Step 13: fresh data sample of 2000 good decay events. For each successive group of 50 events, count how many have a decay time less than 1 lifetime. Histogram the number of “successes.” This gives you 40 experiments to do.

In[198]:=
```
initelement = 1;
setsize = 50;
numbersets = 40;
bincount = {1};
p = .63;
q = .37;
theoryvar = setsize * p * q;
theorymean = p * setsize;
groupall =
  Partition[sift[[initelement ;; initelement + (setsize * numbersets)]], setsize];
b = Table[Length[Select[groupall[[i]], # < tau &]], {i, numbersets}];
Histogram[b, bincount];
check = HistogramList[b, bincount, "Probability"][[2, 1]];
a = HistogramList[b, bincount];
start = a[[1, 1]] - .5;
end = a[[1, -1]] + .5;
max = Max[a[[2]]];
Print["The Theoretical mean is ", theorymean,
 "\nThe theoretical variance is ", theoryvar]
Print["The mean is ", N[Mean[b]], "\nThe variance is ", N[Variance[b]]]
ℋ = DistributionFitTest[b, Automatic, "HypothesisTestData"]

distlabel = StringTemplate["μ = ``\nσ² = ``"][N[Mean[b]], N[Variance[b]]];
Histogram[b, bincount]
Show[Histogram[b, bincount, "Probability"], Plot[PDF[ℋ["FittedDistribution"], x],
  {x, start, end}, PlotStyle → Thick, PlotRange → All]]
p3 = Plot[CDF[ℋ["FittedDistribution"], x], {x, start, end}, PlotStyle → Red];
p4 = Histogram[b, bins, "CDF"];
Show[p4, p3]
```

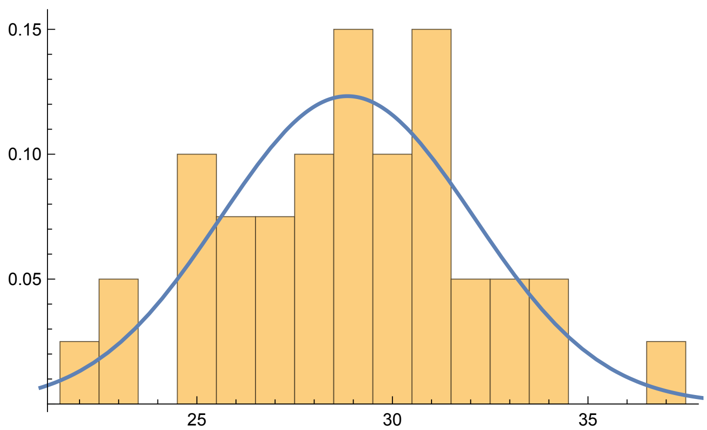
```
The Theoretical mean is 31.5
The theoretical variance is 11.655

The mean is 28.85
The variance is 10.7462
```

Out[216]= HypothesisTestData[Type: DistributionFitTest p−Value: 0.6]

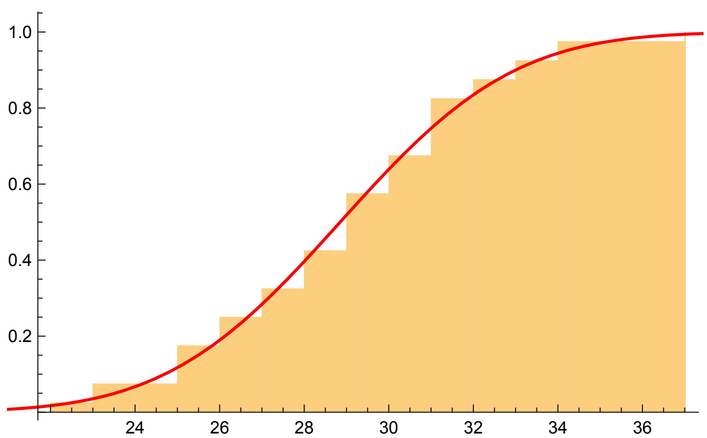
Out[218]=



Out[219]=



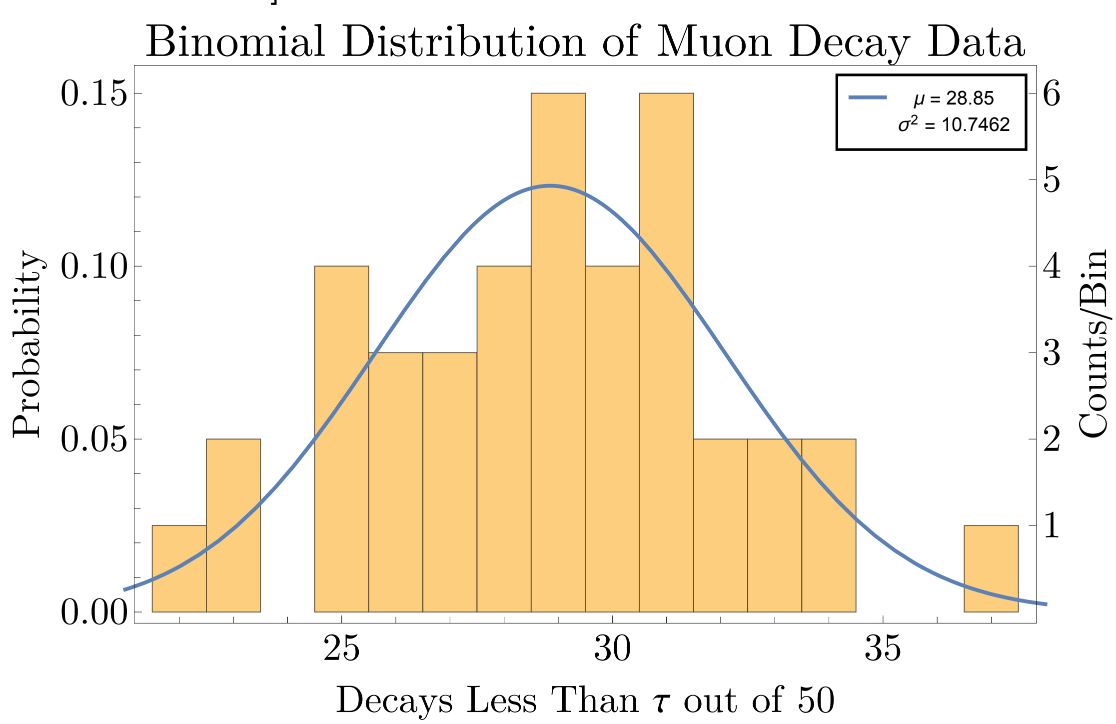
Out[222]=



Nicer Plots

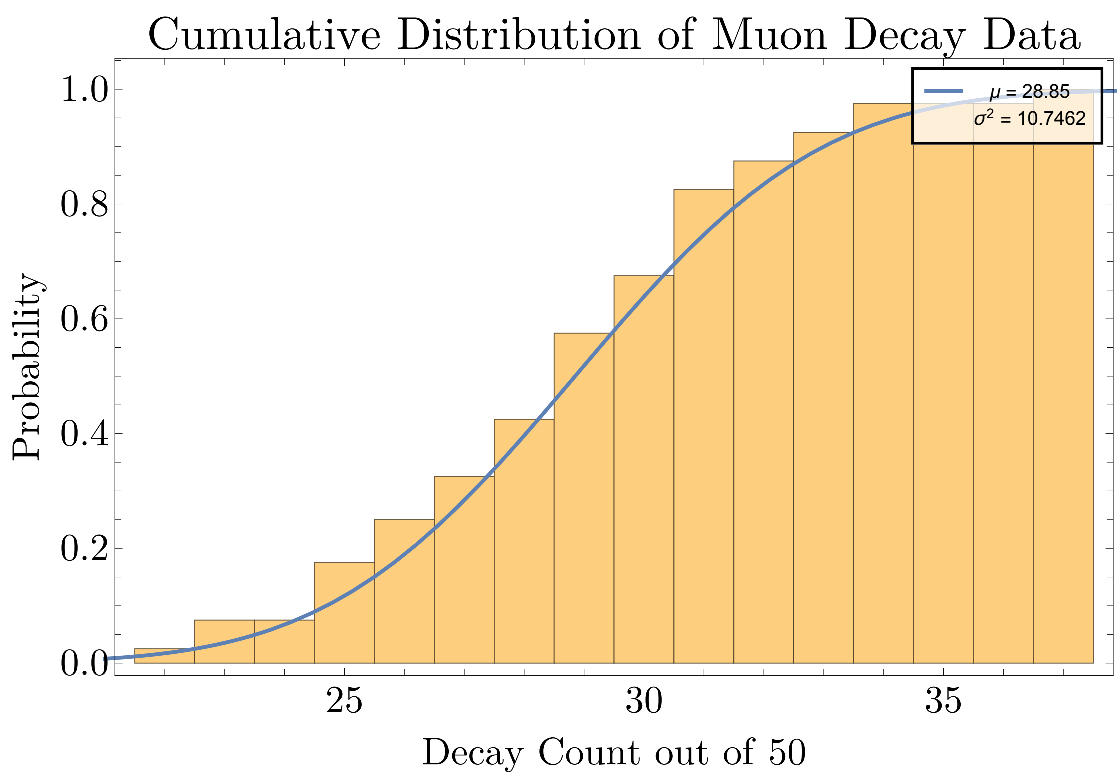
In[223]:=
```
Show[Histogram[b, bincount, "Probability"], Plot[PDF[ℋ["FittedDistribution"], x],
   {x, start, end}, PlotStyle → Thick, PlotRange → All,
   PlotLegends → Placed[LineLegend[{distlabel},
      LegendFunction → (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
          FrameStyle → Directive[Black]] &)], {Right, Top}]],
  Frame → True,
  FrameLabel → {{"Probability", "Counts/Bin"},
    {StringForm["Decays Less Than τ out of ``", setsize], None}},
  FrameTicks → {{All, Table[{i * check, i},
      {i, 1, max + Round[max / 10], Round[max / 10]}]}, {True, None}},
  (*FrameLabel→{"Decays Less Than tau out of 50","Probability Density"},*)
  PlotLabel → "Binomial Distribution of Muon Decay Data",
  LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
  ImageSize → Large]
```

Out[223]=



```
In[224]:= Show[Histogram[b, bincount, "CDF"], Plot[CDF[ℋ["FittedDistribution"], x],
    {x, start, end}, PlotStyle → Thick, PlotRange → All,
    PlotLegends → Placed[LineLegend[{distlabel},
       LegendFunction → (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
           FrameStyle → Directive[Black]] &)], {Right, Top}]],
   Frame → True,
   FrameLabel →
    {{"Probability", None}, {StringForm["Decay Count out of ``", setsize], None}},
   PlotLabel → "Cumulative Distribution of Muon Decay Data",
   LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
   ImageSize → Large]
```

Out[224]=

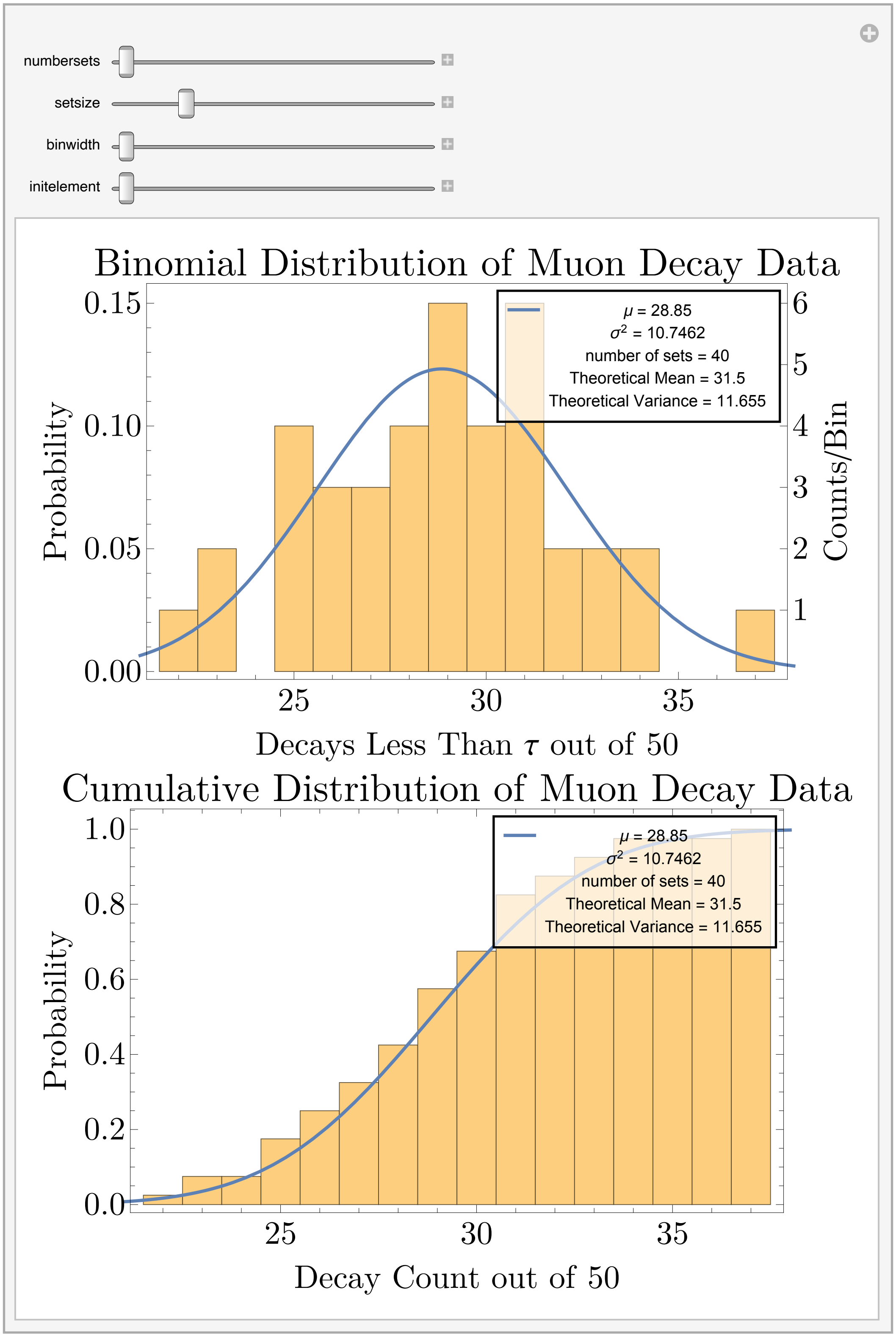


Manipulate

```
In[225]:= Manipulate[
   size = 500;
   bincount = {binwidth};
   groupall =
    Partition[sift[[initelement ;; initelement + (setsize * numbersets)]], setsize];
   p = .63;
   q = .37;
   theoryvar = setsize * p * q;
   theorymean = p * setsize;
```

```
b = Table[Length[Select[groupall[[i]], # < tau &]], {i, numbersets}];
ℋ = DistributionFitTest[b, Automatic, "HypothesisTestData"];
a = HistogramList[b, bincount];
check = HistogramList[b, bincount, "Probability"][[2, 1]];
start = a[[1, 1]] - .5;
end = a[[1, -1]] + .5;
max = Max[a[[2]]];
distlabel =
 StringTemplate["μ = ``\nσ² = ``\nnumber of sets = ``\nTheoretical Mean
     = ``\nTheoretical Variance = ``"][
  N[Mean[b]], N[Variance[b]], numbersets, theorymean, theoryvar];
Grid[{{
   Show[
    Histogram[b, bincount, "ProbabilityDensity", PlotRange → All],
    Plot[PDF[ℋ["FittedDistribution"], x],
     {x, start, end}, PlotStyle → Thick, PlotRange → All,
     PlotLegends → Placed[LineLegend[{distlabel}, LegendFunction →
         (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
            FrameStyle → Directive[Black]] &)], {Right, Top}]],
    Frame → True,
    FrameLabel → {{"Probability", "Counts/Bin"},
      {StringForm["Decays Less Than τ out of ``", setsize], None}},
    FrameTicks → {{All, Table[{i * check, i}, {i, 1, max + Round[max / 10],
         Round[max / 10]}]}, {True, None}},
    PlotLabel → "Binomial Distribution of Muon Decay Data",
    LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
    ImageSize → size]}, {
   Show[Histogram[b, bincount, "CDF"], Plot[CDF[ℋ["FittedDistribution"], x],
     {x, start, end}, PlotStyle → Thick, PlotRange → All,
     PlotLegends → Placed[LineLegend[{distlabel}, LegendFunction →
         (Framed[#, FrameMargins → 0, Background → Opacity[.7, White],
            FrameStyle → Directive[Black]] &)], {Right, Top}]],
    Frame → True,
    FrameLabel → {{"Probability", None},
      {StringForm["Decay Count out of ``", setsize], None}},
    PlotLabel → "Cumulative Distribution of Muon Decay Data",
    LabelStyle → {20, GrayLevel[0], FontFamily → "Latin Modern Math"},
    ImageSize → size]}}],
{{numbersets, 40}, 40, 243, 1}, {{setsize, 50}, 1, 240, 1},
{{binwidth, 1}, 1, 10, 1}, {{initelement, 1}, 1, 12000, 1}]
```

Out[225]=



numbersets
setsize
binwidth
initelement
Binomial Distribution of Muon Decay Data
Probability
Counts/Bin
Decays Less Than τ out of 50
μ = 28.85
σ2 = 10.7462
number of sets = 40
Theoretical Mean = 31.5
Theoretical Variance = 11.655
Cumulative Distribution of Muon Decay Data
Probability
Decay Count out of 50

Credits

If you have any questions email Chris Newey (cnewey@smu.edu).

In[226]:=