Experiment 30B - Interaction of *γ*-rays with Matter

```
In[•]:= << CurveFit`
```

```
CurveFit for Mathematica v7.x thru v11.x, Version 1.96, 4/4/2018
Caltech Sophomore Physics Labs, Pasadena, CA
```

```
In[•]:= SetDirectory[StringJoin[NotebookDirectory[], "30b"]];
```

Lead X-ray Fluorescence

2-point energy calibration using Cs-137 spectrum

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
      {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
  If[ name =!= $Canceled,
   LoadFile[name, SkipLines → {"Data:", "Counts"}]]
 ]
```

```
In[•]:= MakePoisson[Xerrors -> True]
```

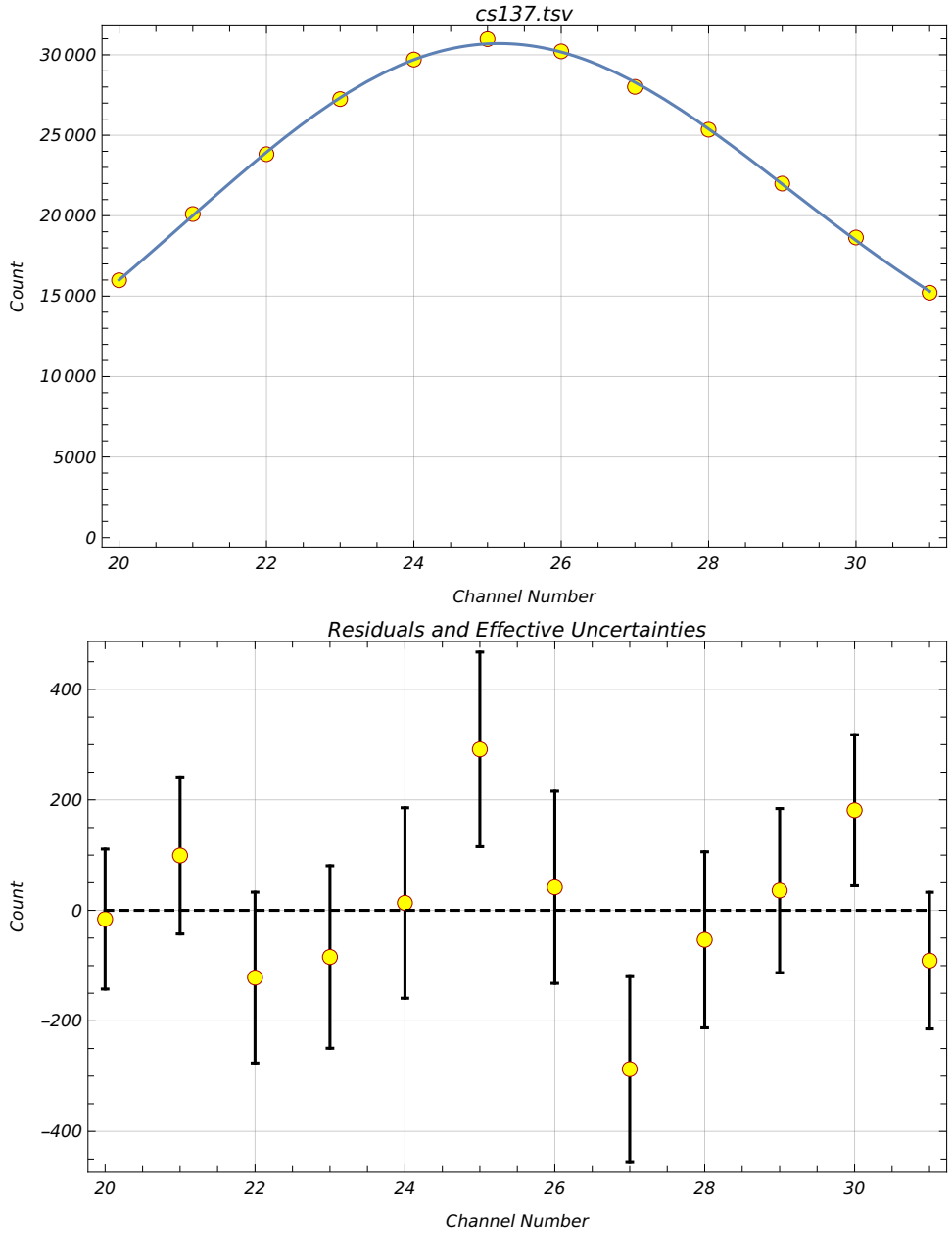
We find the channel locations of the 32 keV and 661.6 keV peaks and use the two data points to estimate correspondence between channel number and energy, assuming a linear correspondence over the entire spectrum, using the line equation $E = \frac{661.6 - 32}{\mu_{661.6} - \mu_{32}}(x - \mu_{661.6}) + 661.6\,\text{keV}$.

```
In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
      Label -> "Set the X values for the range you wish to keep." ]},
  Print[x];

  XRangeKeep[Sequence @@ x]
 ]
```

```
In[•]:= GaussianLFit[]
```

In[]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[]=



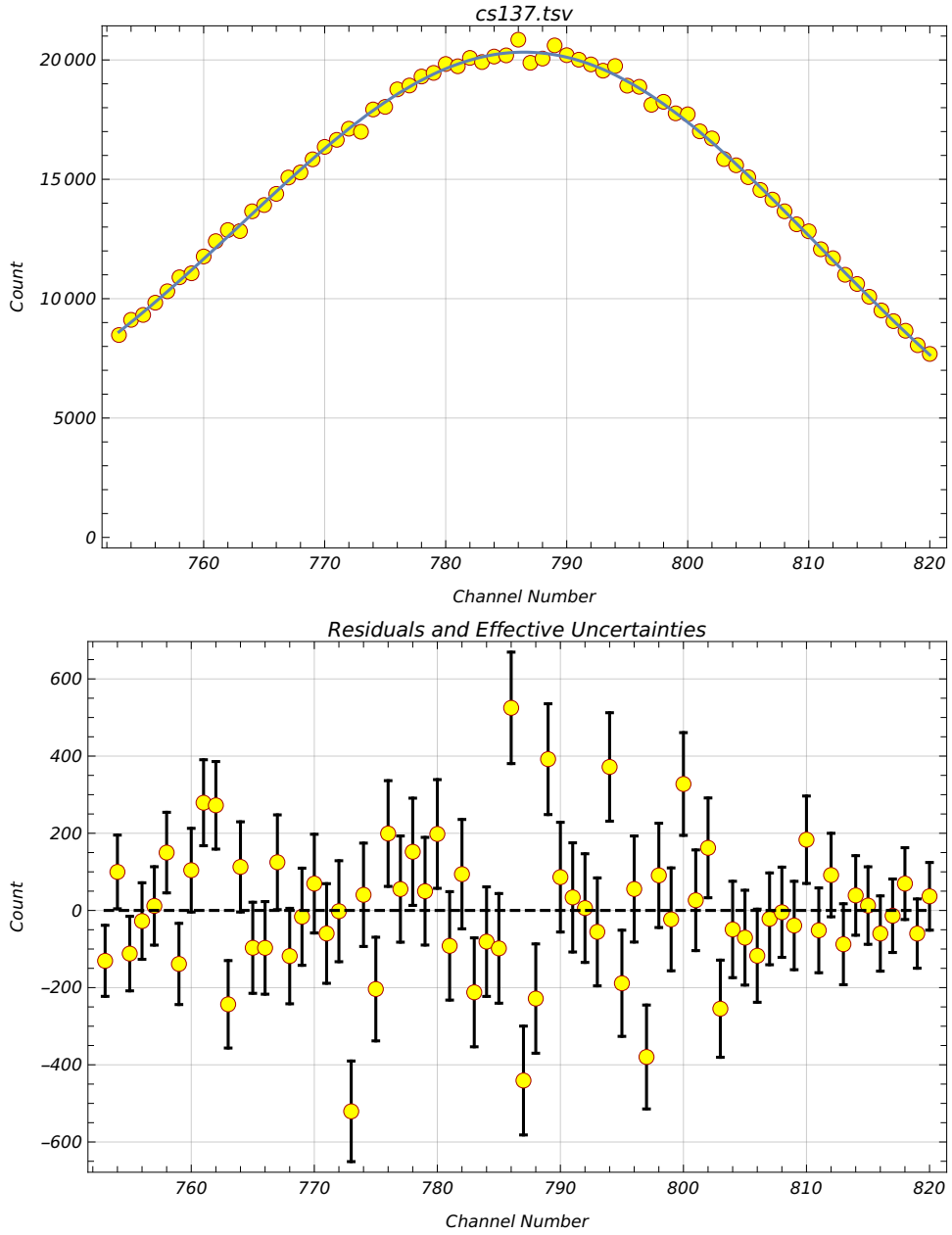
$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\ sigma^2}\right) + c + d\ (x-\mu)$$

c= 4888.64	d= 253.284		
σ_c= 1106.35	σ_d= 54.8234		
y_{max}= 25 793.6	μ= 24.9906	sigma= 4.11748	
$\sigma_{y_{max}}$= 1218.95	σ_μ= 0.0578475	σ_{sigma}= 0.15007	$\chi^2/(n-5)$= 1.37254

```
In[•]:= x32 = {mean, sigmean}; (* channel number for 32 keV peak *)

In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
        Label -> "Set the X values for the range you wish to keep." ]},
      Print[x];

      XRangeKeep[Sequence @@ x]
     ]

In[•]:= GaussianLFit[]
```

In[ ]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[ ]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\ sigma^2}\right) + c + d\ (x-\mu)$$

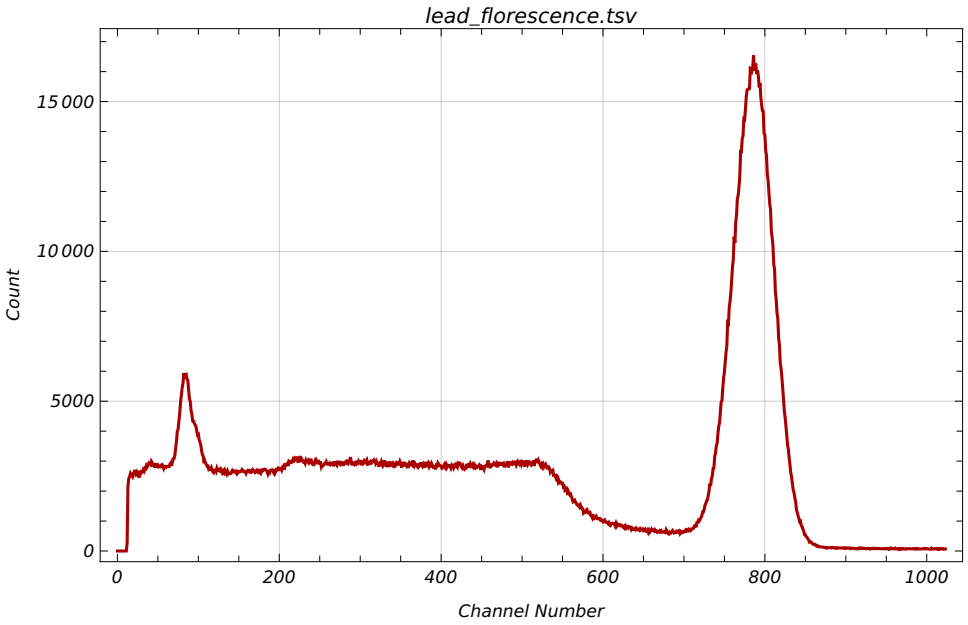
c= 769.582	d= −22.1922		
σ_c= 361.886	σ_d= 2.53914		
y_{max}= 19548.8	μ= 787.114	sigma= 23.937	
$\sigma_{y_{max}}$= 365.004	σ_μ= 0.134073	σ_{sigma}= 0.357908	$\chi^2/(n-5)$= 2.04713

```
In[•]:= x661 = {mean, sigmean}; (* channel number for 661.6 keV peak *)

        (* We write a 2-
         point calibration function for our spectrum with error estimate: *)

In[•]:= cal1[channel_] := ((661.6 - 32)/(x661[[1]] - x32[[1]])) (channel[[1]] - x661[[1]]) + 661.6

        {1, Sqrt[(x661[[2]]^2 + x32[[2]]^2)/(x661[[1]] - x32[[1]])^2 + (x661[[2]]^2 + channel[[2]]^2)/(channel[[1]] - x661[[1]])^2]};
```

Estimating energy of lead X-ray fluorescence peak

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
            {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
         If[ name =!= $Canceled,
          LoadFile[name, SkipLines → {"Data:", "Counts"}]]
        ]

In[•]:= MakePoisson[Xerrors -> True]

In[•]:= LinearDataPlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=

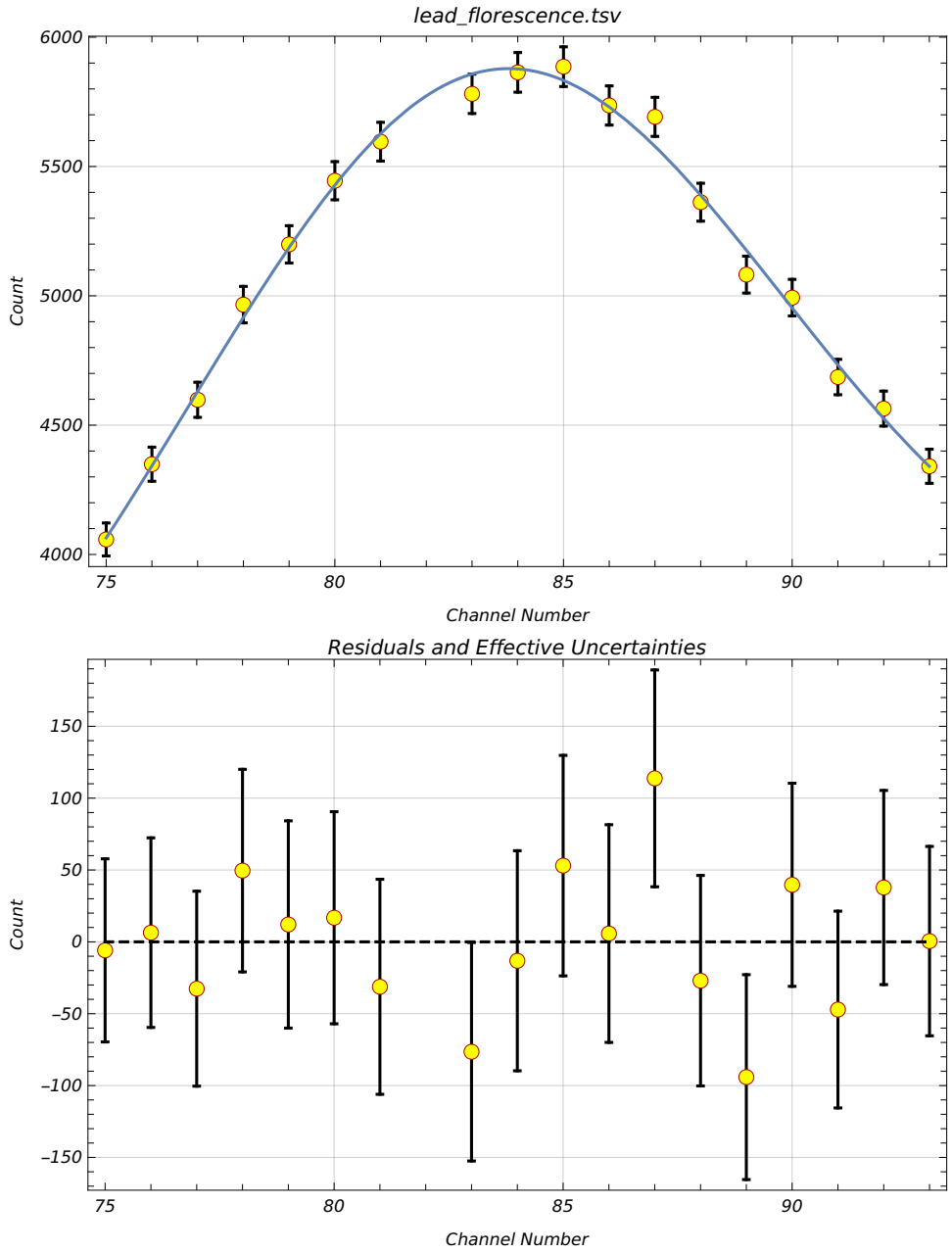



```
In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
            Label -> "Set the X values for the range you wish to keep." ]},
         Print[x];

         XRangeKeep[Sequence @@ x]
        ]
```

In[•]:= GaussianLFit[]

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\ sigma^2}\right) + c + d\ (x-\mu)$$

$c=$	$d=$		
3136.95	32.6793		
$\sigma_c=$	$\sigma_d=$		
330.007	13.4154		
$y_{max}=$	$\mu=$	sigma=	
2733.31	83.3008	6.46351	
$\sigma_{y_{max}}=$	$\sigma_\mu=$	$\sigma_{sigma}=$	$\chi^2/(n-5)=$
509.11	0.29331	0.88919	0.598377

```
In[•]:= flChannel = {mean, sigmean};

In[•]:= cal1[flChannel]

Out[•]= {80.1709, 0.0398179}
```

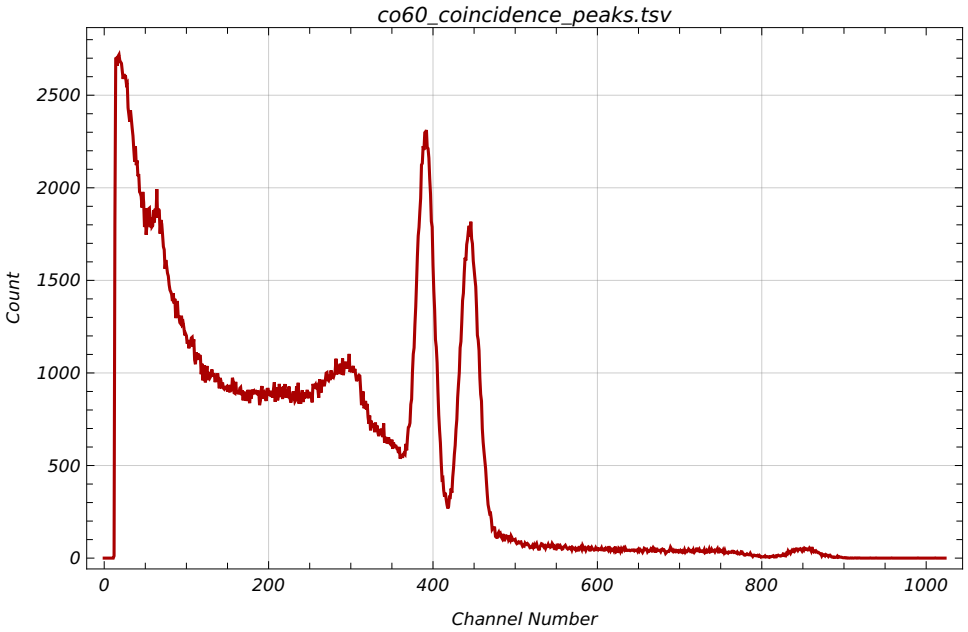
From the above, we can estimate the energy of the lead X-ray fluorescence peak to be 80.17 ± 0.04 keV.

Thorium-232 Pair Production

We again assume a linear dependence of energy on channel count and construct a new calibration function for the detector settings used for the Cobalt-60 and Thorium 232 dataset.

```
(* We pull the Co-60 and Th-
 232 full energy peak channel counts for our calibration. *)

In[•]:= With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
  If[ name =!= $Canceled,
   LoadFile[name, SkipLines → {"Data:", "Counts"}]]
 ]

In[•]:= MakePoisson[Xerrors -> True]

In[•]:= LinearDataPlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=

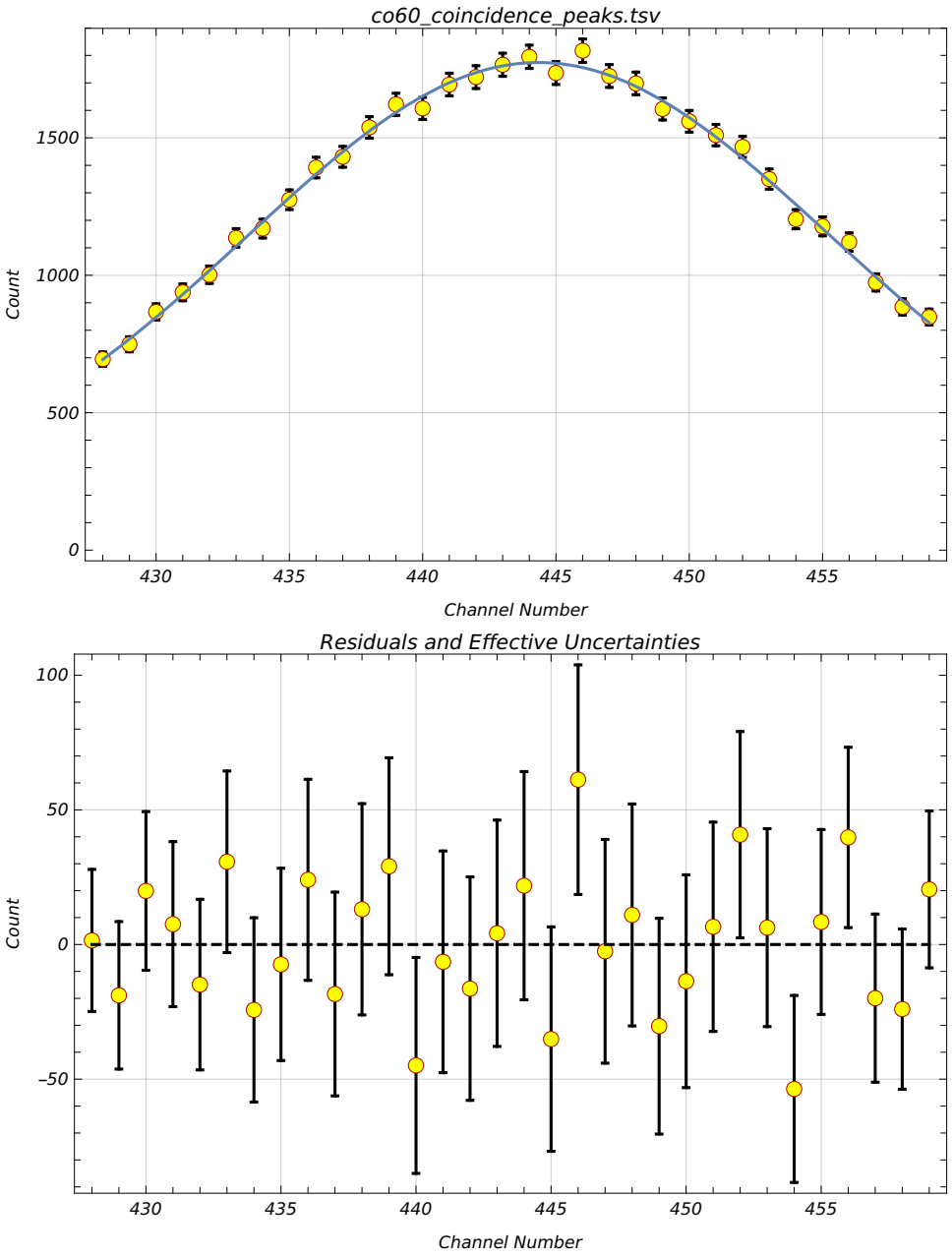


```
In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
          Label -> "Set the X values for the range you wish to keep." ]},
       Print[x];

       XRangeKeep[Sequence @@ x]
      ]
```

```
In[•]:= GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



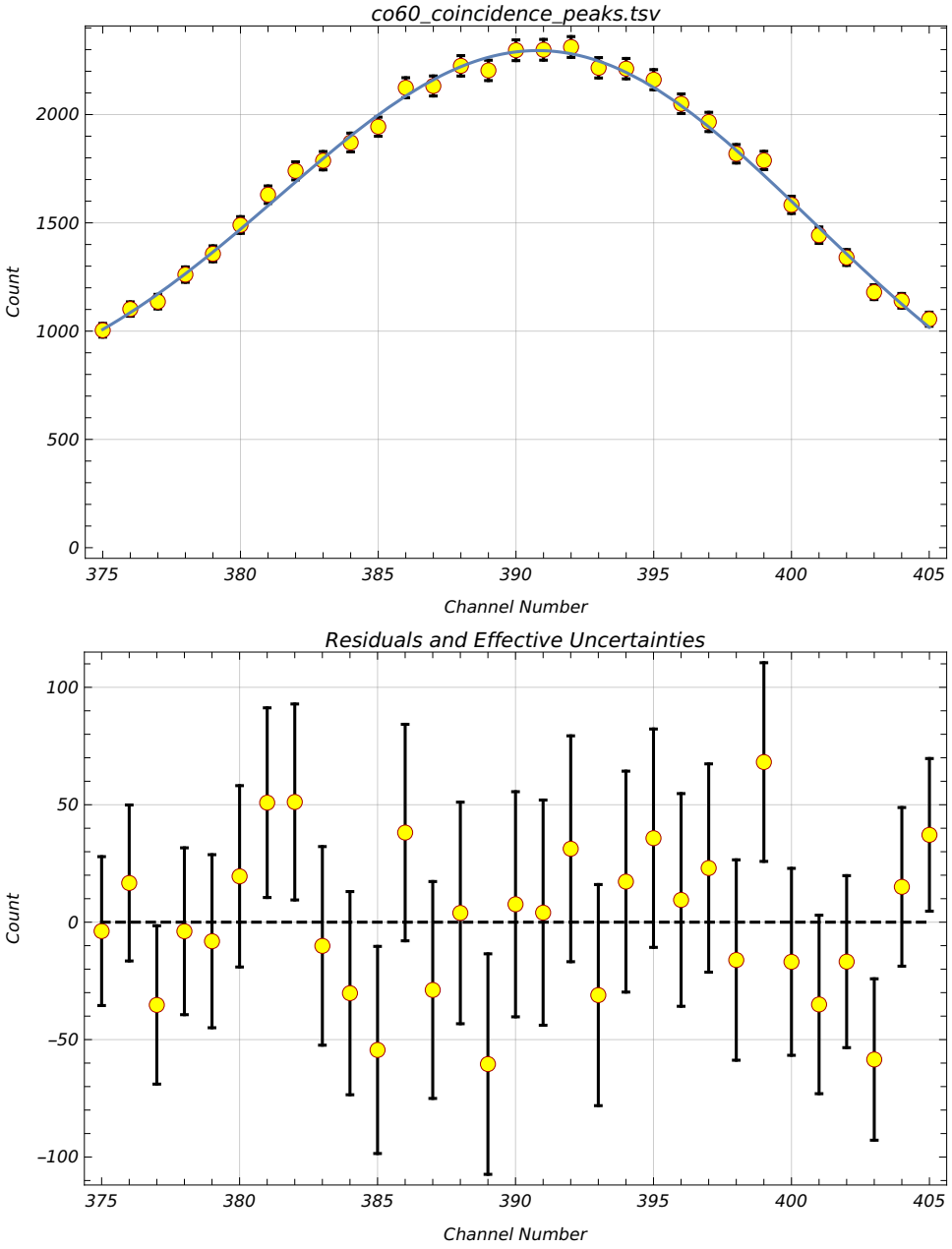
$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\ sigma^2}\right) + c + d\ (x-\mu)$$

$c=$	$d=$		
177.187	0.280333		
$\sigma_c=$	$\sigma_d=$		
131.504	2.28792		
$y_{max}=$	$\mu=$	sigma=	
1596.78	444.328	10.9082	
$\sigma_{y_{max}}=$	$\sigma_\mu=$	$\sigma_{sigma}=$	$\chi^2/(n-5)=$
160.773	0.299899	0.85892	0.582479

```
In[•]:= co133 = {1.332, mean, sigmean};

In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
        Label -> "Set the X values for the range you wish to keep." ]},
      Print[x];

      XRangeKeep[Sequence @@ x]
     ]

In[•]:= GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=

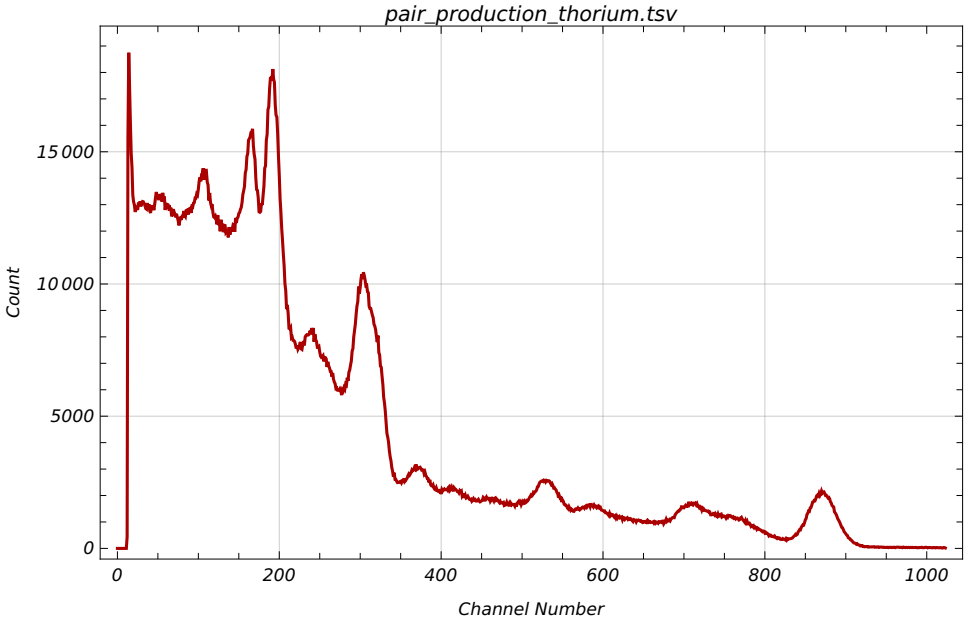


$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\,sigma^2}\right) + c + d\ (x-\mu)$$

$c =$	$d =$		
451.238	−6.10821		
$\sigma_c =$	$\sigma_d =$		
110.086	2.5918		
$y_{max} =$	$\mu =$	sigma =	
1843.23	391.091	9.64444	
$\sigma_{y_{max}} =$	$\sigma_\mu =$	$\sigma_{sigma} =$	$\chi^2/(n-5) =$
126.119	0.223392	0.571286	0.765

```
In[•]:= co117 = {1.173, mean, sigmean};

In[•]:= With[ {name = SystemDialogInput["FileOpen",
          {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
       If[ name =!= $Canceled,
        LoadFile[name, SkipLines → {"Data:", "Counts"}]]
      ]

In[•]:= MakePoisson[Xerrors -> True]

In[•]:= LinearDataPlot[FrameLabel → {"Channel Number", "Count"}]
```

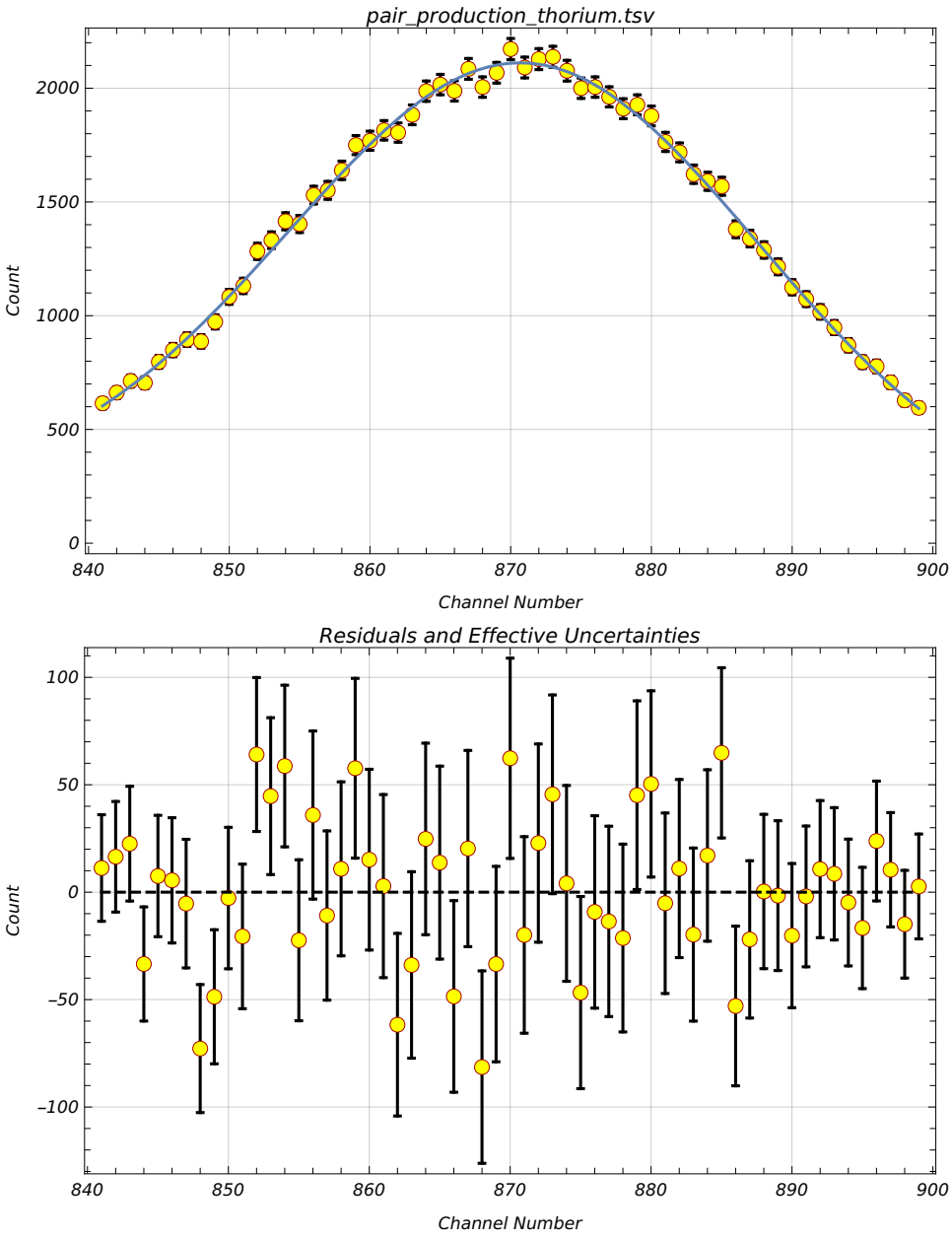
Out[•]=




```
In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
          Label -> "Set the X values for the range you wish to keep."]},
       Print[x];

       XRangeKeep[Sequence @@ x]
      ]

In[•]:= GaussianLFit[]
```

In[•]:= **LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]**

Out[•]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\ sigma^2}\right) + c + d\ (x-\mu)$$

c=	d=		
168.233	−1.58703		
σ_c=	σ_d=		
47.725	0.602624		
y_{max}=	μ=	sigma=	
1942.85	870.904	16.6606	
$\sigma_{y_{max}}$=	σ_μ=	σ_{sigma}=	$\chi^2/(n-5)$=
46.7638	0.209758	0.419356	0.843657

```
In[•]:= th261 = {2.61, mean, sigmean};
```

Constructing the calibration function

```
In[•]:= Export["calData.dat", Partition[Join[co117, co133, th261], 3]];
```

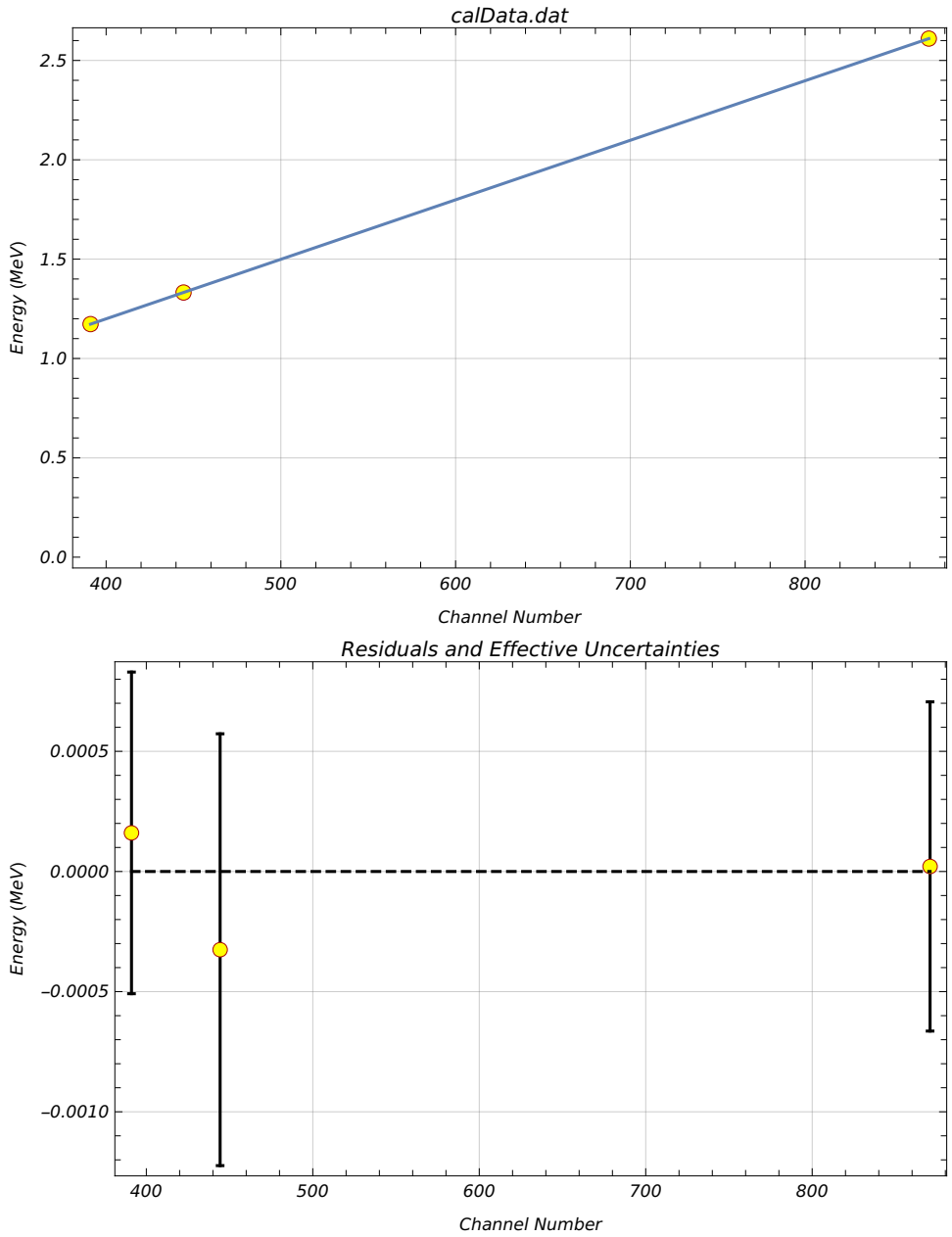
```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
        {DataFileName, {"data files" -> {"*.dat", "*.mca"}, "all files" -> {"*"}}}]},
     If[ name =!= $Canceled,
      LoadFile[name]]
    ]
```

```
In[•]:= SwitchXXandYY[]
```

```
In[•]:= LinearFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Energy (MeV)"}]
```

Out[•]=



$y(x) = a + b\,x$

$a =$	$b =$	
0.00121159	0.00299579	
$\sigma_a =$	$\sigma_b =$	$\chi^2/(n-2) =$
0.00117895	1.88276×10^{-6}	0.189787

Since we will be using this calibration function to measure energy spacings, we do not have to concern ourselves with the offset a.

In[•]:=
```
scaling = {b, sigb};
```

```
In[•]:= cal2[spacing_] :=
          scaling[[1]] spacing[[1]] {1, Sqrt[(scaling[[2]]/scaling[[1]])^2 + (spacing[[2]]/spacing[[1]])^2]};

In[•]:= spacing[escPeak_] := {th261[[2]] - escPeak[[1]], Sqrt[th261[[3]]^2 + escPeak[[2]]^2]};
```

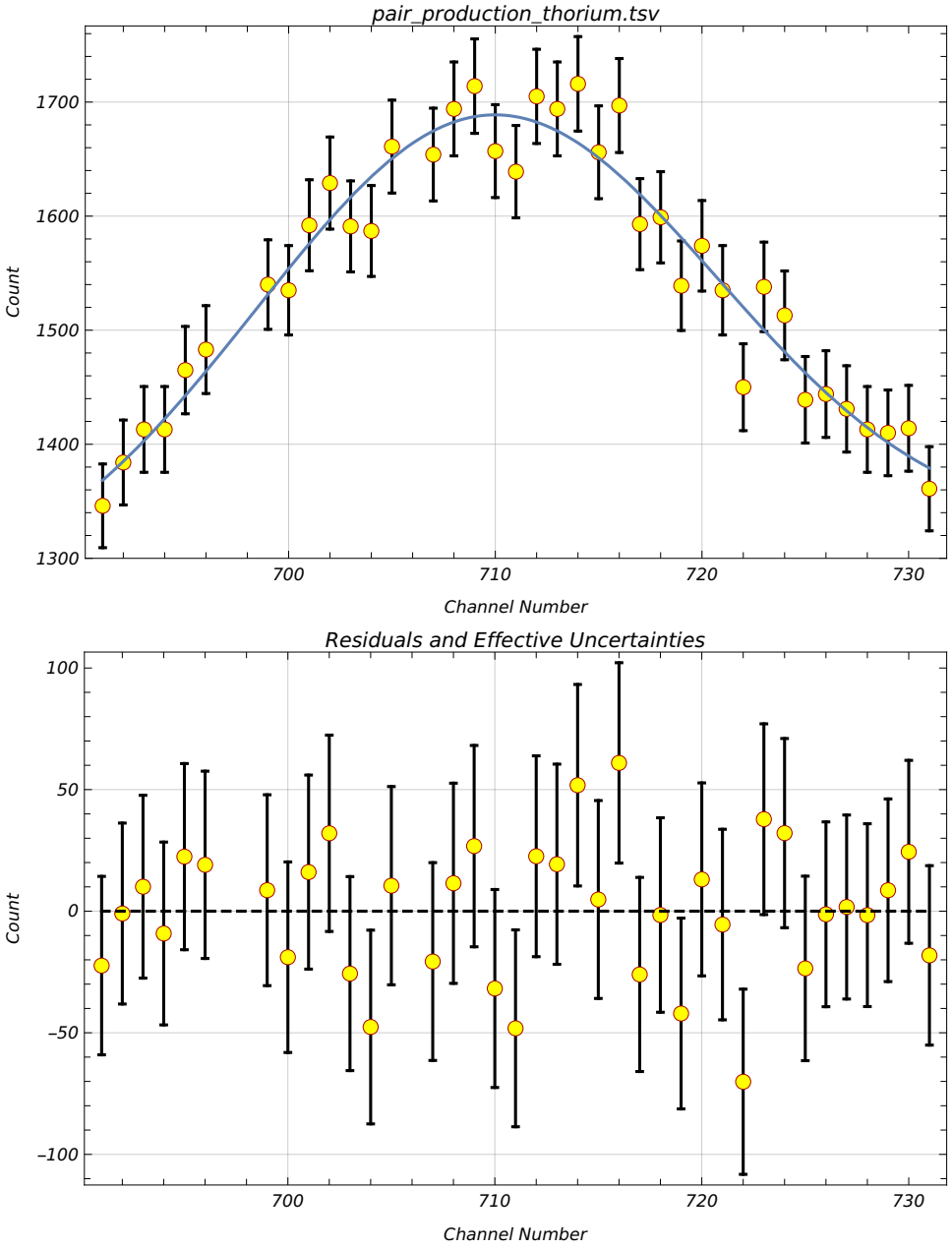
Spacing between full-energy peak and escape peaks

One-escape peak

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
            {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
          If[ name =!= $Canceled,
           LoadFile[name, SkipLines → {"Data:", "Counts"}]]
         ]

In[•]:= MakePoisson[Xerrors -> True]

In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
            Label -> "Set the X values for the range you wish to keep." ]},
          Print[x];

          XRangeKeep[Sequence @@ x]
         ]


In[•]:= GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\,sigma^2}\right) + c + d\,(x-\mu)$$

c=	d=		
1291.43	1.14832		
σ_c=	σ_d=		
53.5445	1.6736		
y_{max}=	μ=	sigma=	
397.158	709.626	11.1475	
$\sigma_{y_{max}}$=	σ_μ=	σ_{sigma}=	$\chi^2/(n-5)$=
98.4558	0.897566	2.48909	0.574586

```
In[•]:= oneEsc = spacing[{mean, sigmean}];

In[•]:= cal2[oneEsc]
Out[•]= {0.482874, 0.00279134}
```

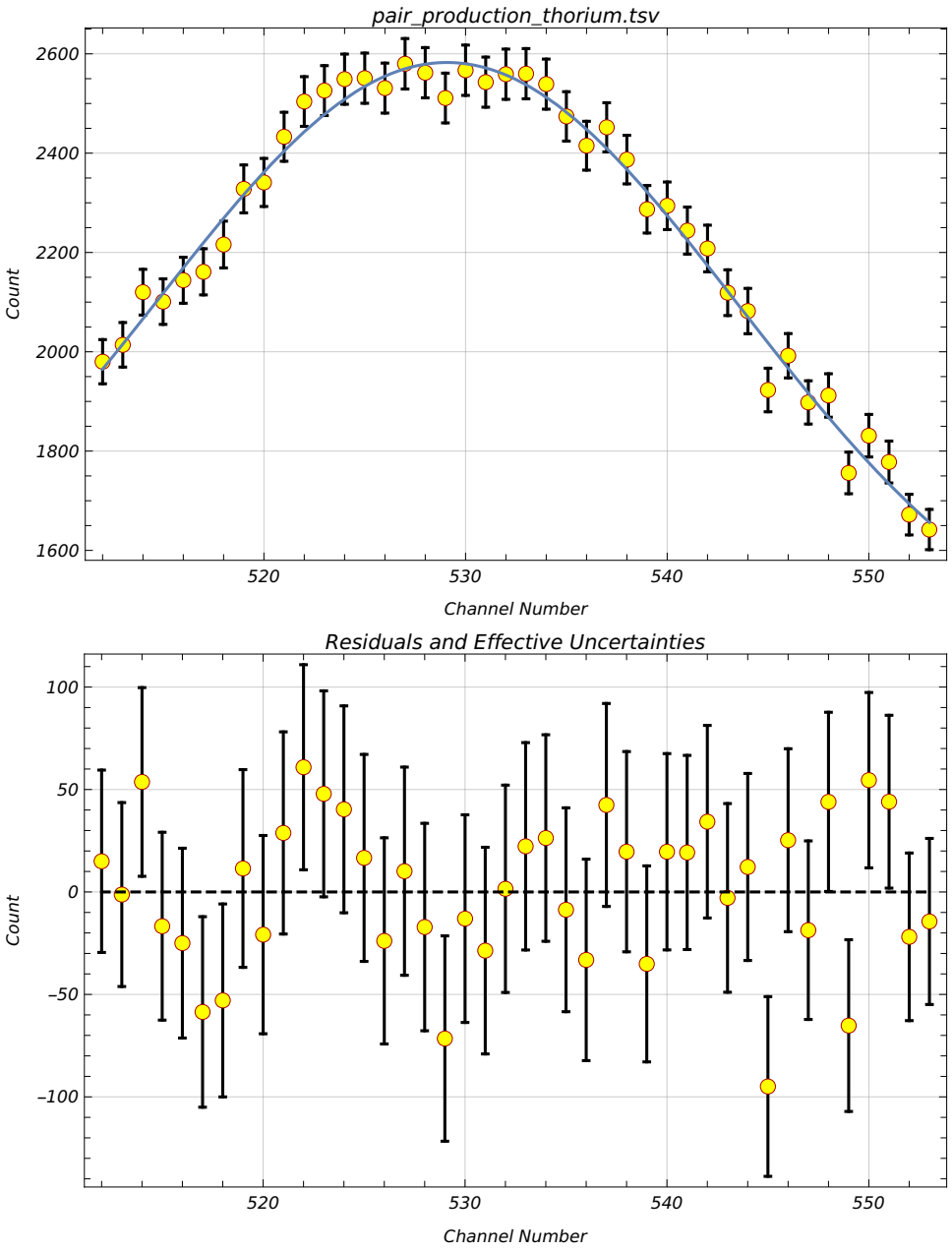
Therefore, using the one-escape peak, we estimate an electron rest mass energy $m_e = 0.4829 \pm 0.0028$ MeV.

Two-escape peak

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name, SkipLines → {"Data:", "Counts"}]]
]

In[•]:= MakePoisson[Xerrors -> True]

In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
     Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]


In[•]:= GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\,sigma^2}\right) + c + d\ (x-\mu)$$

$c =$	$d =$		
1352.63	−0.342994		
$\sigma_c =$	$\sigma_d =$		
187.125	3.16697		
$y_{max} =$	$\mu =$	sigma =	
1229.91	529.133	14.4087	
$\sigma_{y_{max}} =$	$\sigma_\mu =$	$\sigma_{sigma} =$	$\chi^2/(n-5) =$
217.48	0.867373	1.74202	0.71222

```
In[•]:= twoEsc = spacing[{mean, sigmean}];

In[•]:= cal2[twoEsc] / 2

Out[•]= {0.511797, 0.00138158}
```

Therefore, using the one-escape peak, we estimate an electron rest mass energy $m_e = 0.5118 \pm 0.0014$ MeV.

Comments

We observe that we obtain a much better estimate of m_e using the two-escape peak than using the one-escape peak (in fact we are within error of the NIST value whereas the one-escape peak estimate is $>5\sigma$ off).

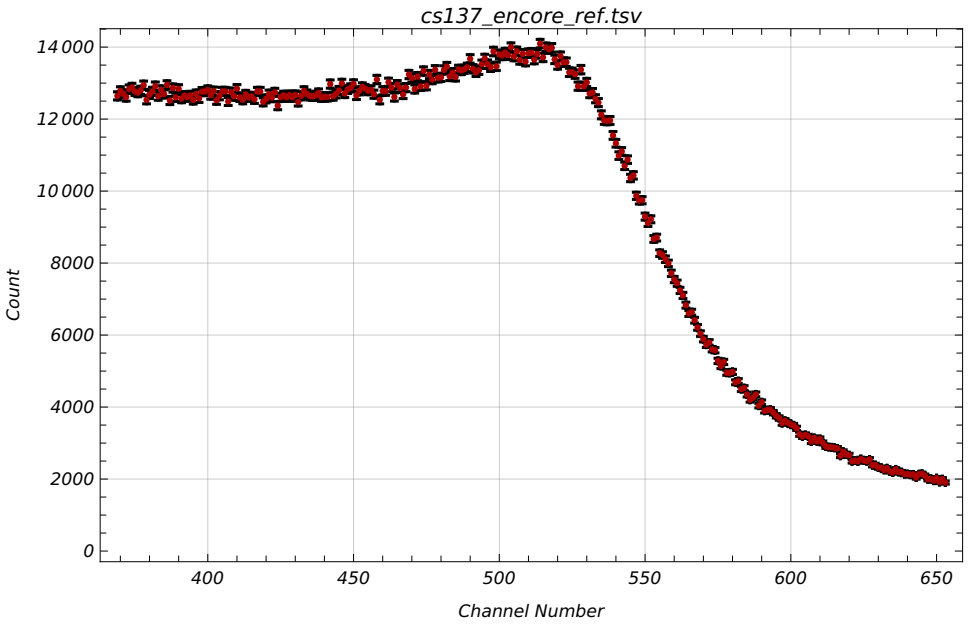
Plastic Scintillator

Cs-137 NaI Scintillator

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
      {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
  If[ name =!= $Canceled,
   LoadFile[name, SkipLines → {"Data:", "Counts"}]]
 ]

In[•]:= With[{x = SetXRange[ LinearDataPlot[], Log -> False,
      Label -> "Set the X values for the range you wish to keep." ]},
  Print[x];

  XRangeKeep[Sequence @@ x]
 ]


In[•]:= MakePoisson[Xerrors -> True]
```

In[•]:=
```
LinearDataPlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



Cs-137 Plastic Scintillator

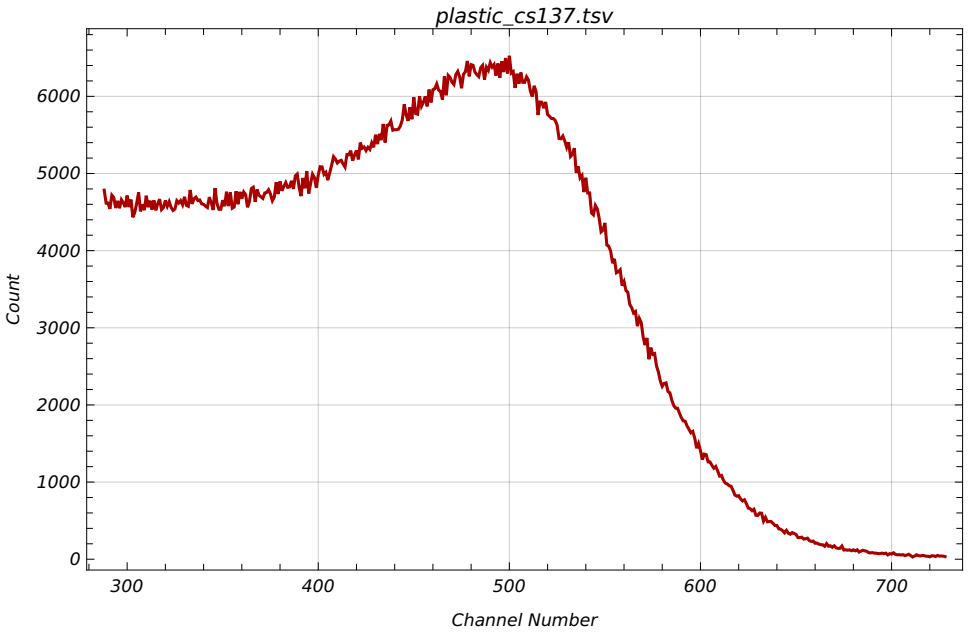
In[•]:=
```
With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
  If[ name =!= $Canceled,
   LoadFile[name, SkipLines → {"Data:", "Counts"}]]
 ]
```

In[•]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
     Label -> "Set the X values for the range you wish to keep." ]},
  Print[x];

  XRangeKeep[Sequence @@ x]
 ]
```

In[•]:=
```
MakePoisson[Xerrors -> True]
```

In[•]:=
```
LinearDataPlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



The plastic scintillator Compton edge is much more pronounced, and appears as a discernible peak. The peak also appears slightly shifted towards lower energies for the plastic scintillator.

Estimating mean number of photoelectrons generated by 661.6 keV detection - NaI detector

In[•]:=
```
With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name, SkipLines → {"Data:", "Counts"}]]
]
```

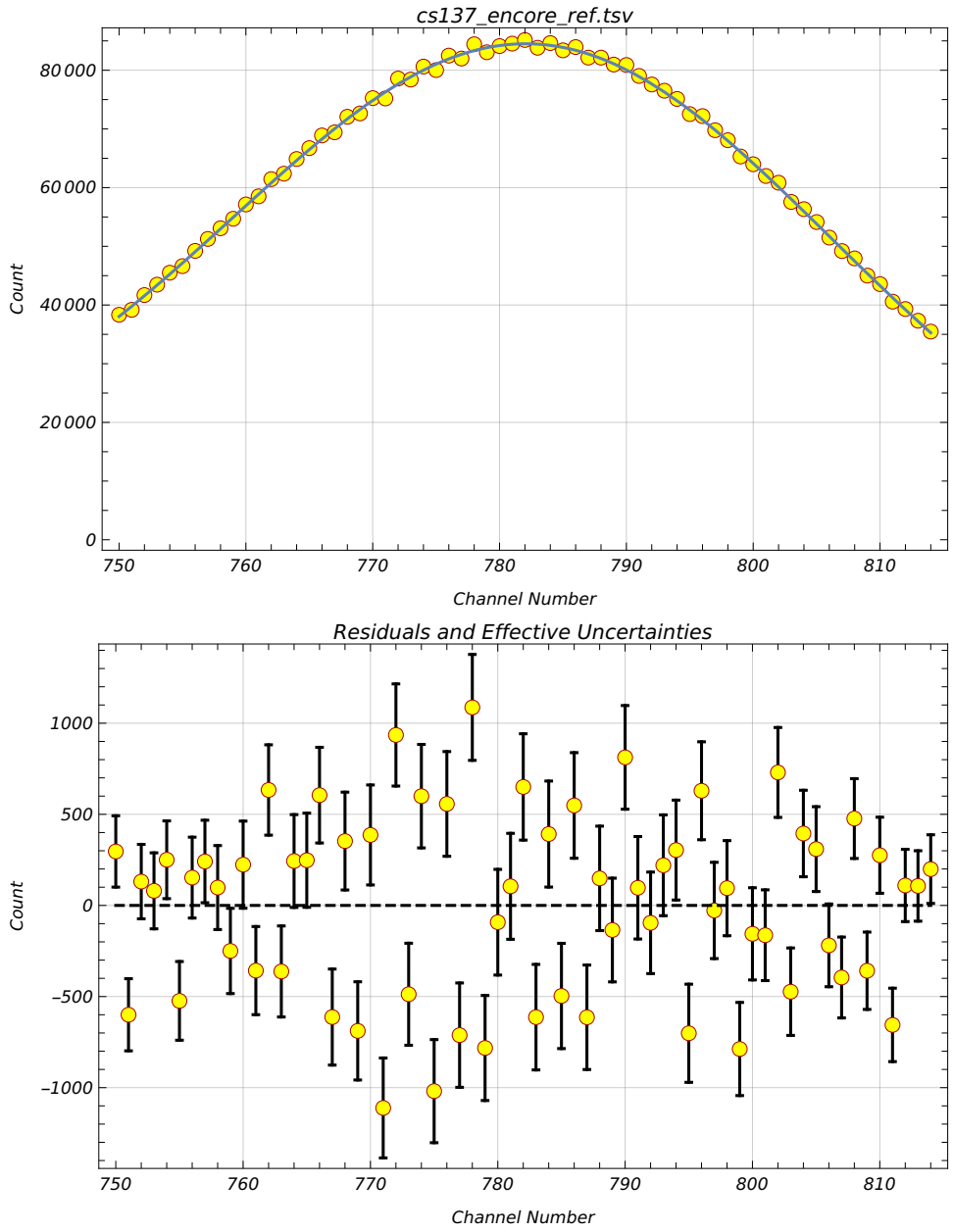
In[•]:=
```
MakePoisson[Xerrors -> True]
```

In[•]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
     Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

In[•]:=
```
GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



$$y(x) = y_{max}\ \exp\left(\frac{-(x-\mu)^2}{2\,sigma^2}\right) + c + d\ (x-\mu)$$

$c =$ 5184.42	$d =$ −77.0335		
$\sigma_c =$ 873.285	$\sigma_d =$ 5.98655		
$y_{max} =$ 79301.8	$\mu =$ 782.594	sigma = 23.5151	
$\sigma_{y_{max}} =$ 841.754	$\sigma_\mu =$ 0.0728689	$\sigma_{sigma} =$ 0.193784	$\chi^2/(n-5) =$ 3.91872

From equation (30.C.2) in the lab manual, derived from basic Poisson statistics, we can estimate the

mean number of photoelectrons generated by a 661.6 MeV detection by the variance of the full-energy peak:

```
In[•]:= photoelectrons = {sigma^2, 2 sigsigma sigma}
Out[•]= {552.958, 9.11371}
```

That is, the mean number of photoelectrons generated by a 661.6 MeV detection is 553 ± 9. We can then use the definition of μ in Appendix C (alternatively, equation (30.C.6)) to estimate E_{pe}, the energy per photoelectron. We use as in the pre-lab $\frac{\sigma(\text{bin})}{\mu(\text{bin})} = \frac{\sigma(T_e)}{T_e}$.

Note: Even if there is an offset in the calibration (so far all calibrations have had offset consistent with 0), the expression is adequate since we are at a channel number far enough from 0.

```
In[•]:= Epe = 661.6 photoelectrons[[1]] / mean^2 {1, Sqrt[(photoelectrons[[2]] / photoelectrons[[1]])^2 + (2 sigmean / mean)^2]}
Out[•]= {0.597331, 0.00984567}
```

and we estimate E_{pe} = 597 ± 10 eV.

Estimating mean number of photoelectrons generated by 661.6 keV detection - Plastic detector

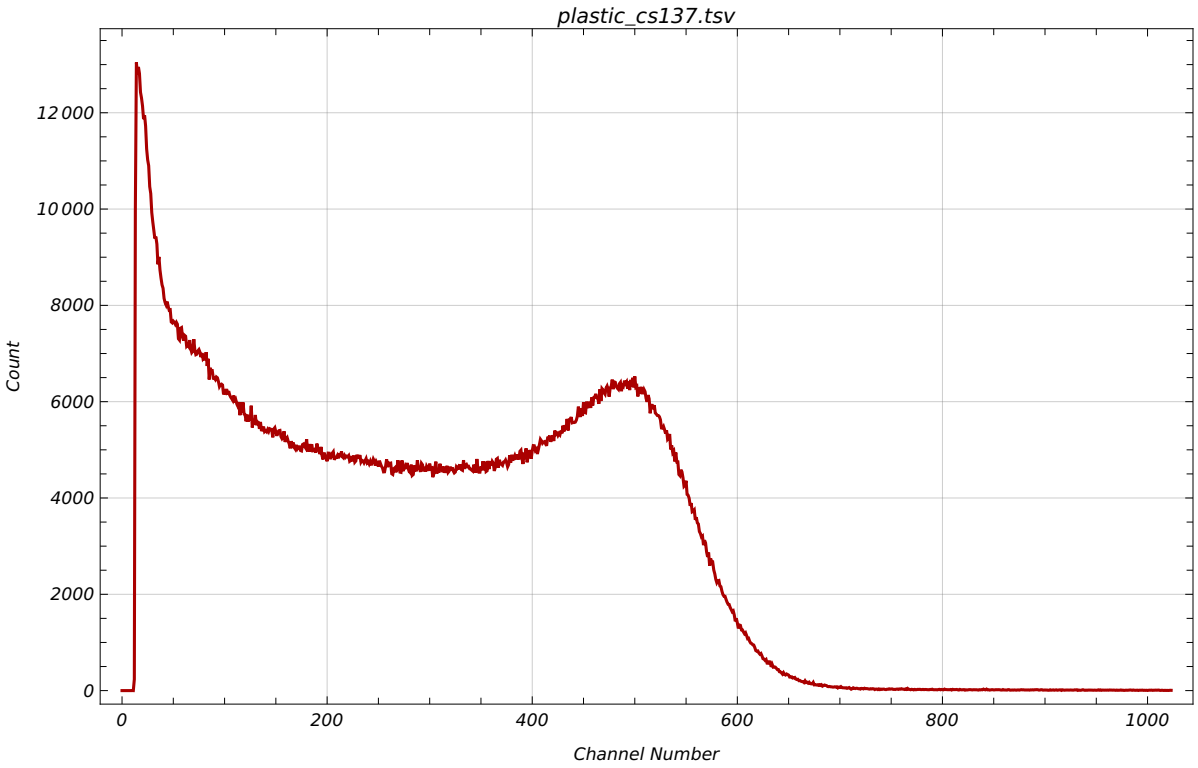
Here we cannot use the approach described above, as our spectrum for the plastic scintillator contains no full-energy peak. We first attempt to manually vary the parameter E_{pe} in the Compton_Spectra1.nb provided until we obtain a Compton edge that appears to fit our data.

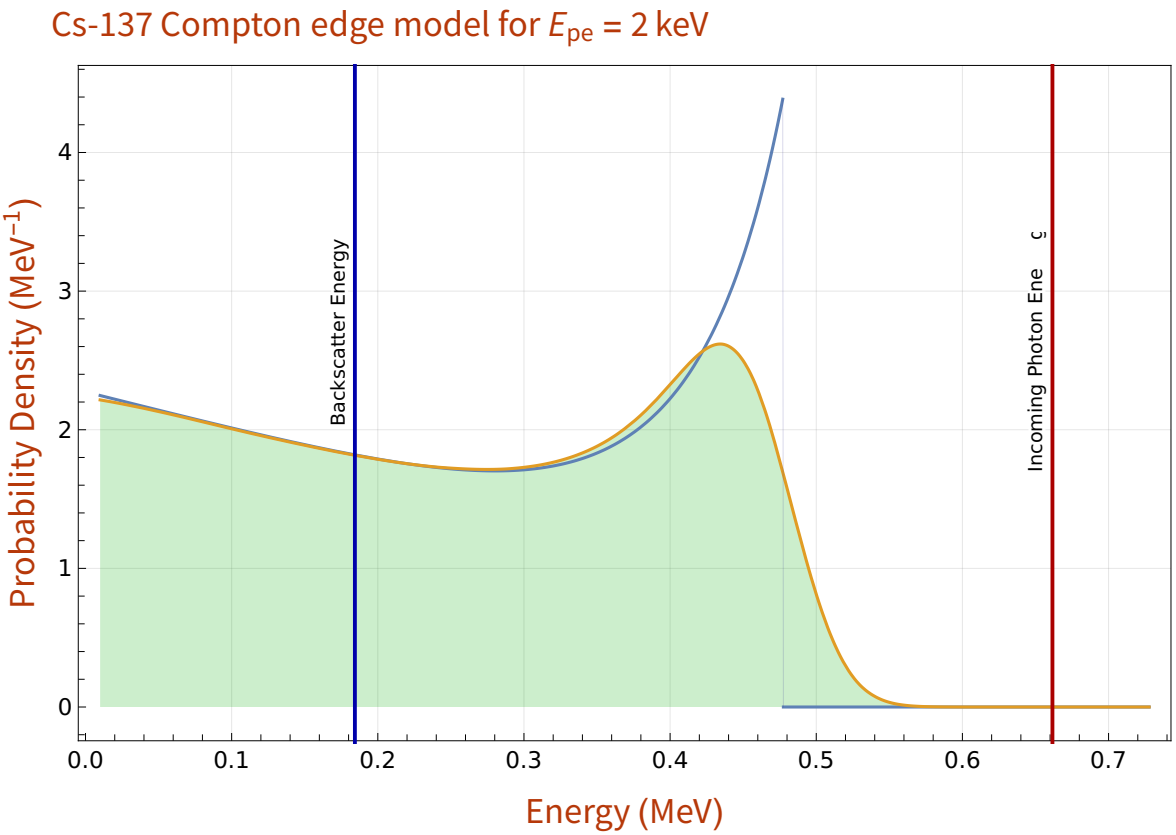
Cs-137 Compton edge for plastic scintillator

```
In[•]:= With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name, SkipLines → {"Data:", "Counts"}]]
]
```

In[•]:=
```
LinearDataPlot[FrameLabel → {"Channel Number", "Count"}, ImageSize → 620]
```

Out[•]=



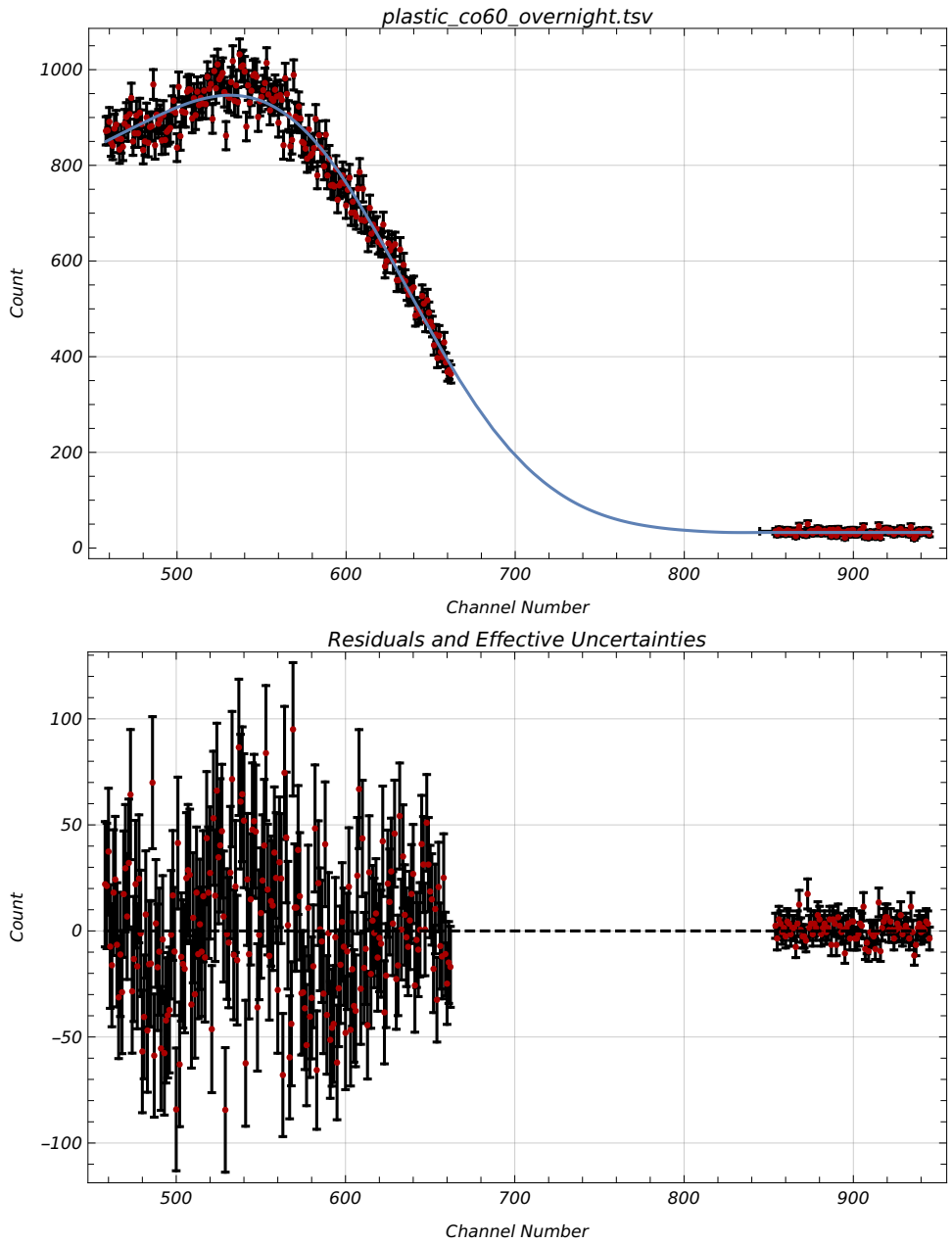


The model appears qualitatively similar to our Compton edge data (reproduced above for comparison) at $E_{pe} \approx 2$ MeV.

Fit of Cs-137 Compton edge for plastic scintillator

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```

Out[•]=



y(x) = Compton spectrum from a photon with energy k_0, scaled in X and Y and convolved with the scintillator resolution e_{pe}(energy/photoelectron).

K0 (MeV) =
0.6616

Tedge (MeV) =
0.477281

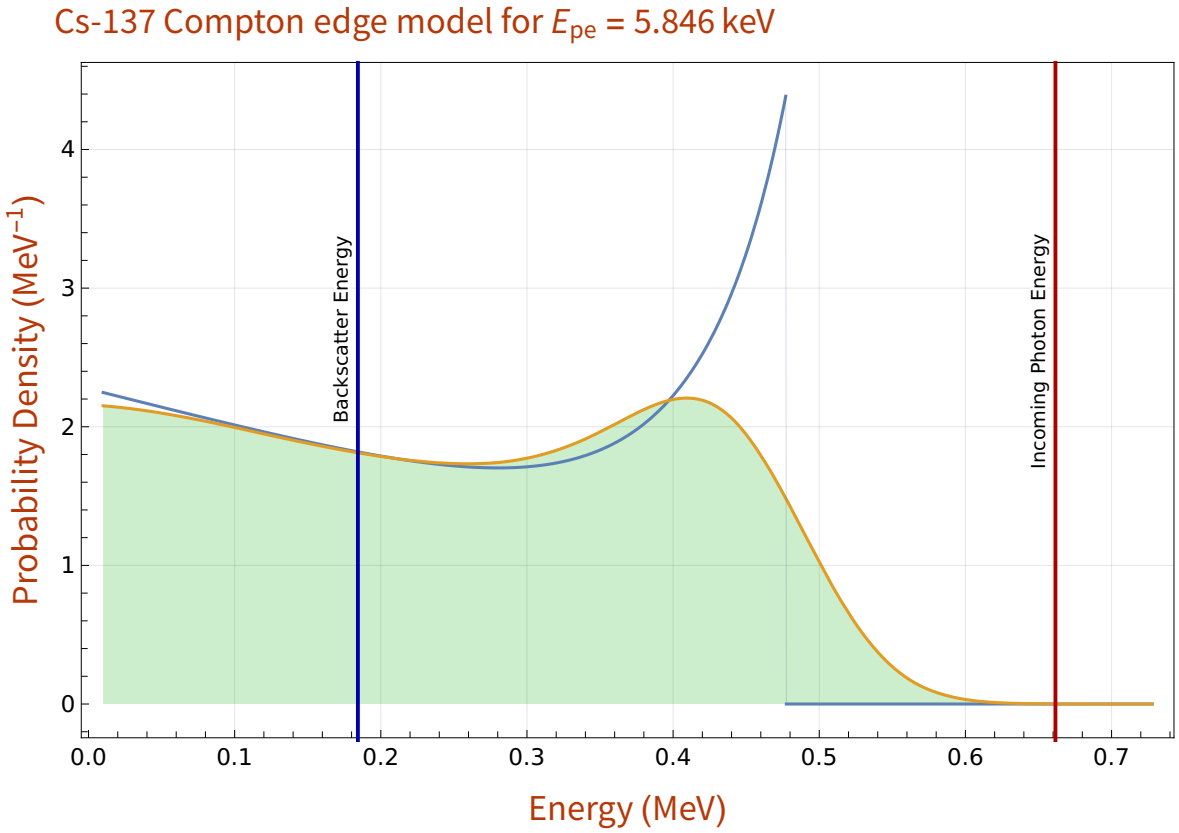
EPE (eV/photoelectron) =

sigEPE =

5846.36	147.789
Xscale (channels/MeV) = 1299.2	sigXscale = 1.0572
Yscale = (counts/channel)/(probability/MeV) 413.836	sigYscale = 1.84396
Yoffset (counts/channel) = 32.57	sigYoffset = 0.594918
Xedge (channel) = 620.083	sigXedge = 0.504581

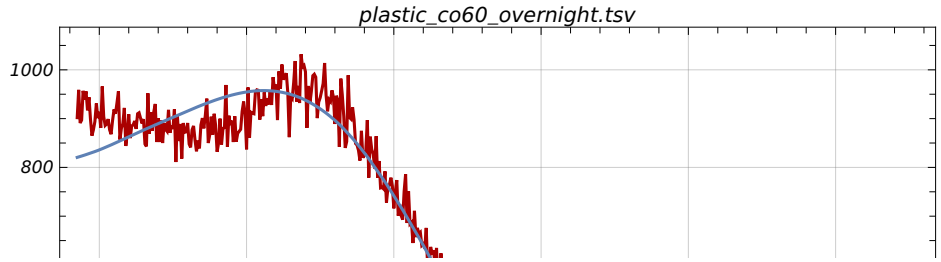
χ^2 = 1.26291

From this we obtain an estimate of $E_{pe} = 5.85 \pm 0.15$ keV.

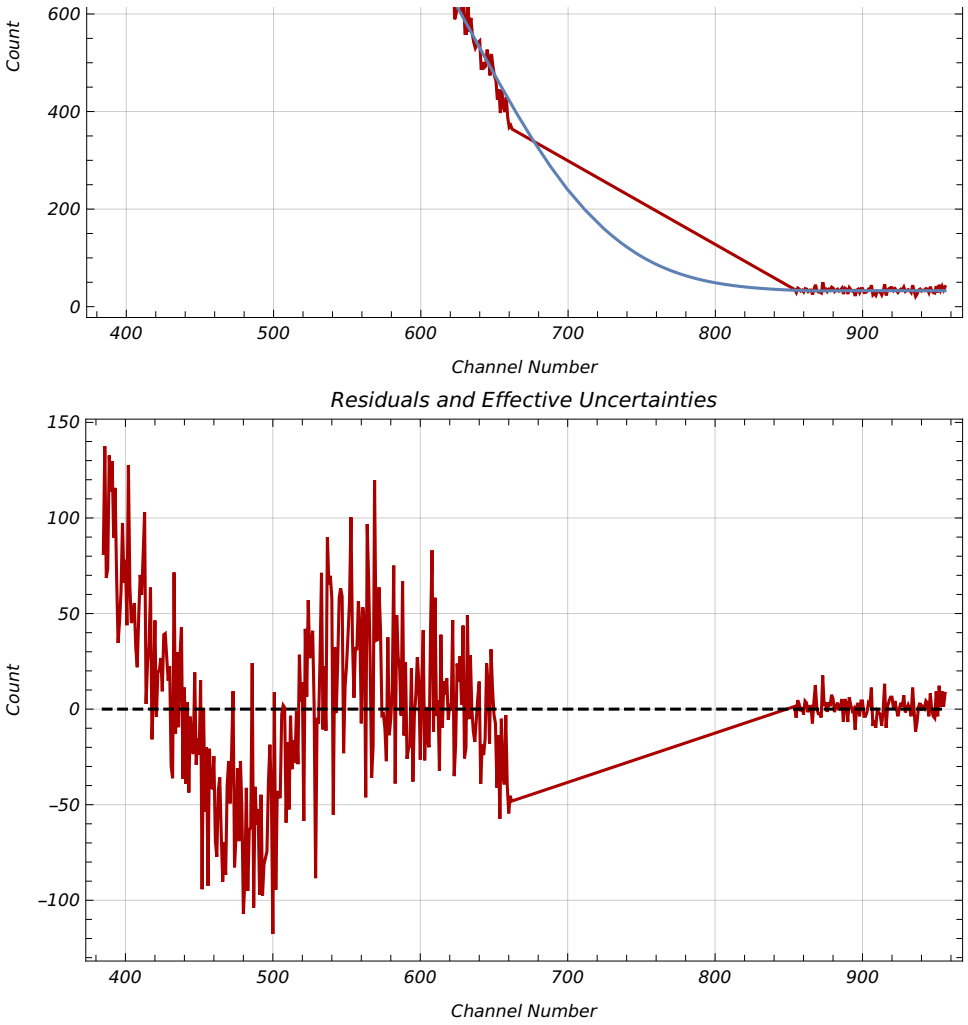


We plot the Compton edge model for E_{pe} matching the fit parameter obtained above. Interestingly, we observe that the model generated qualitatively appears to be less similar to our Compton edge data outside of a very small region around the Compton edge than our initial guess of 2 keV (which we know is typical of plastic scintillators). However, attempting to remedy this by fitting a larger section of the data significantly worsens the fit:

In[•]:=
```
LinearDifferencePlot[FrameLabel → {"Channel Number", "Count"}]
```



Out[ ]=



y(x) = Compton spectrum from a photon with energy k_0, scaled in X and Y and convolved with the scintillator resolution e_{pe}(energy/photoelectron).

K0 (MeV) = 0.6616	Tedge (MeV) = 0.477281	
EPE (eV/photoelectron) = 8327.61	sigEPE = 162.84	
Xscale (channels/MeV) = 1292.4	sigXscale = 1.15018	
Yscale = (counts/channel)/(probability/MeV) 443.216	sigYscale = 1.35316	
Yoffset (counts/channel) = 32.7521	sigYoffset = 0.56615	
Xedge (channel) = 616.837	sigXedge = 0.548961	χ^2 = 2.29648

This indicates that the Compton edge model we are fitting against does not adequately describe our measurements, at least outside a narrow region around the Compton edge. For instance, the count drops much faster than expected right before the Compton edge. This could be due to other effects that become significant at lower energies (we know such effects exist due to the observed dramatic count increase at low energies observed on the full Cs-137 plastic scintillator spectrum, although we do

not know at what energies those effects become significant or how they affect the spectrum outside of the very low-energy region).