Experiment 30a: Interaction of *γ*-rays with matter

In-lab preliminary analysis

0.66 MeV Full Energy Peak

```
(* We import the Cs-137 spectrum. *)
```

In[1]:=
```
<< CurveFit`
```

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

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

```
(* We assign Posson uncertainties to the spectrum,
taking the observed count for each bin as our
 maximum likelihood estimator of the expected count *)
```

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

```
(* We select the 0.66 MeV peak *)
```

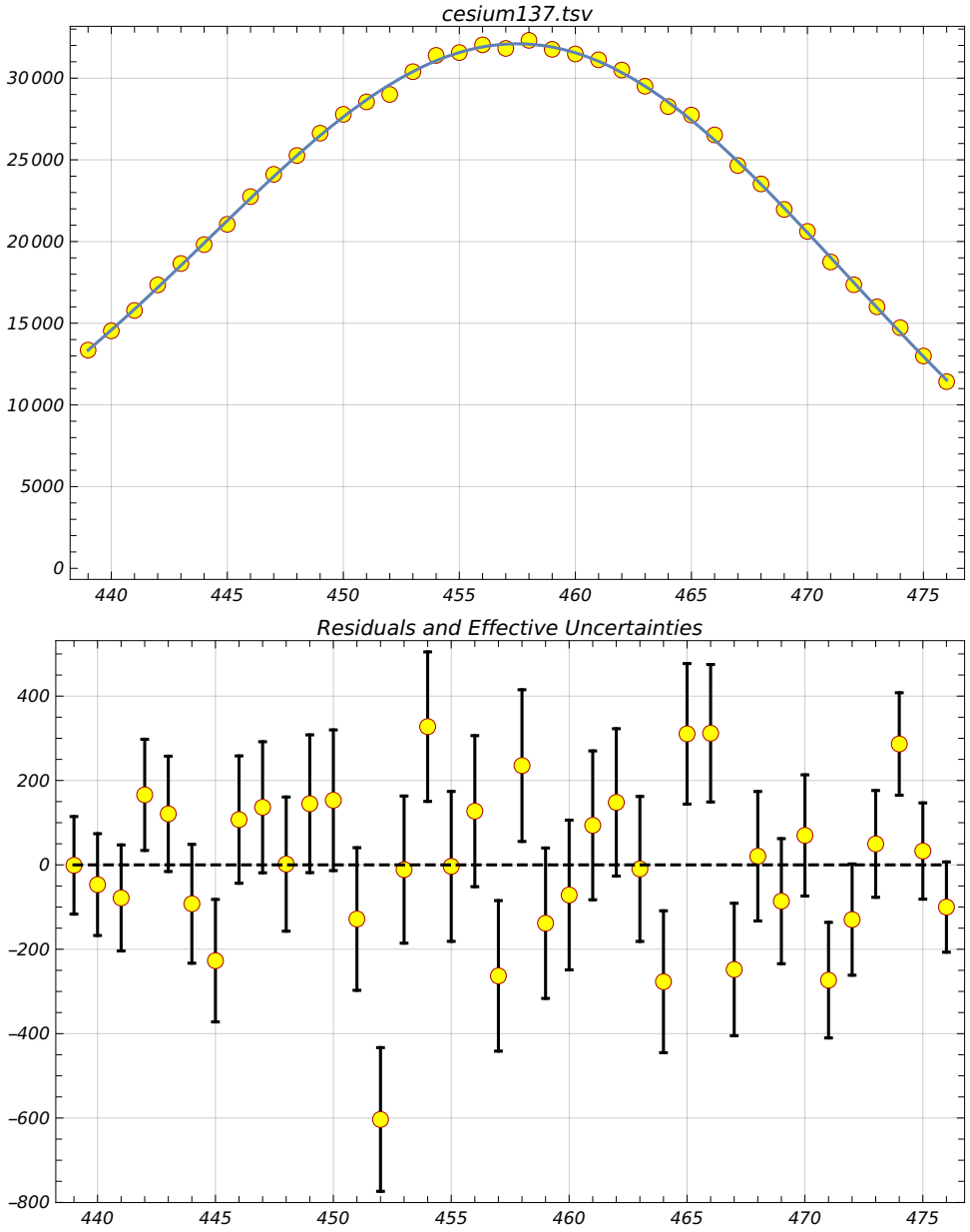
In[•]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
    Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

```
(* We now attempt a Gaussian+Linear fit of the 0.66 MeV peak *)
```

In[•]:=
```
GaussianLFit[]
```

In[•]:=
```
LinearDifferencePlot[]
```

Out[•]=



$y(x) =$

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

$c + d\ (x - \mu)$

$c =$	$d =$		
−854.481	−81.196		
$\sigma_c =$	$\sigma_d =$		
692.989	7.92824		
$y_{max} =$	$\mu =$	$sigma =$	
32936.4	457.95	13.7094	
$\sigma_{y_{max}} =$	$\sigma_\mu =$	$\sigma_{sigma} =$	$\chi^2/(n-5) =$
665.693	0.0799749	0.21572	1.69678

A reasonably good Gaussian+Linear fit is obtained $\left(\tilde{\chi}^2 = 1.69\right)$, indicating that the distribution limiting

the energy resolution described by Poisson statistics approximates a Gaussian as described in the lab manual.

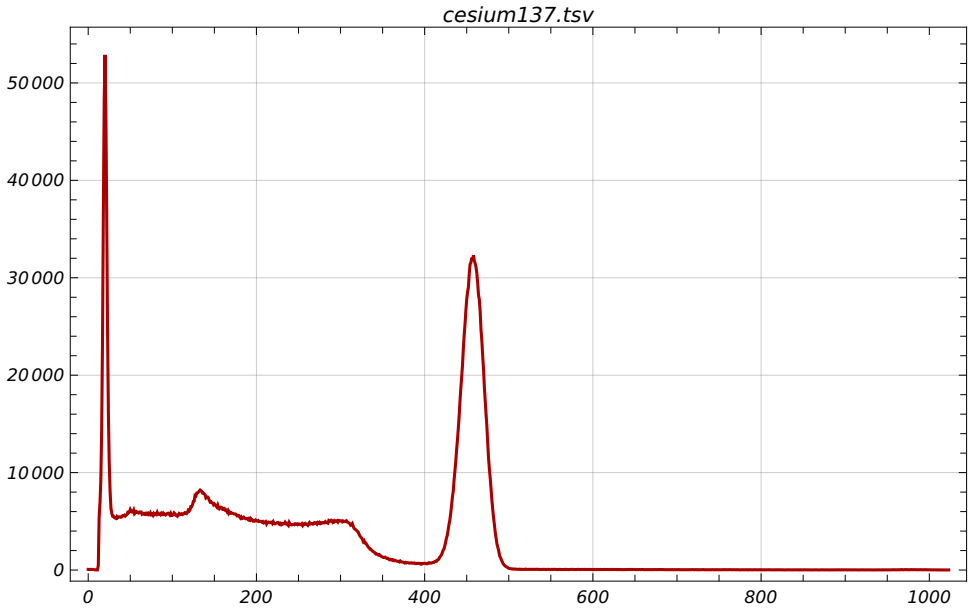
Rate-related Gain Shift

```
(* We control for rate-related gain shift by observing the spectra for Cs-
 137 with the source near and far from the photomultiplier. First,
we show the spectrum when the source is far from the photomultiplier. *)
```

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

In[•]:=
```
LinearDataPlot[]
```

Out[•]=

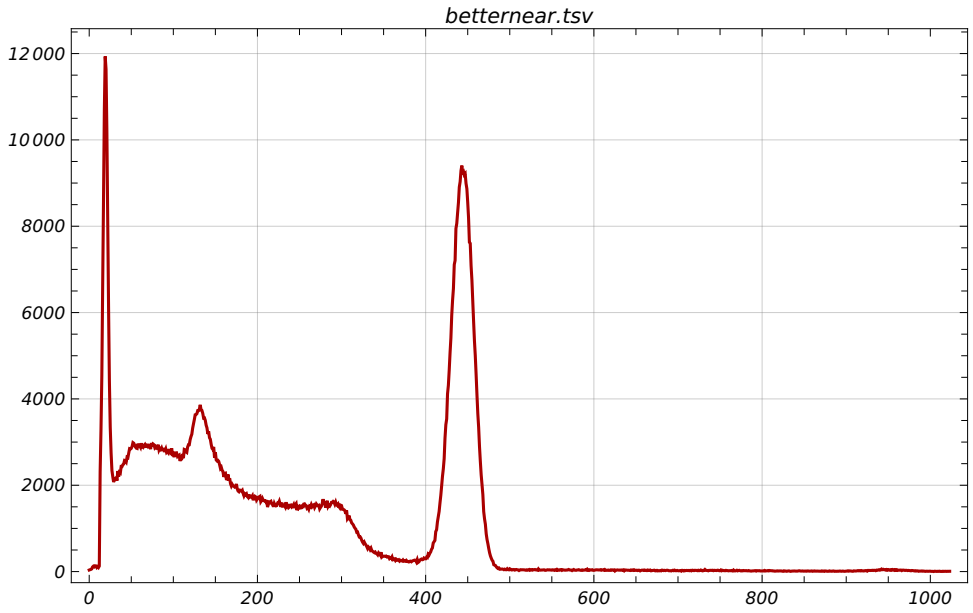



```
(* We now show the spectrum when the source is near the photomultiplier. *)
```

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

In[•]:=
```
LinearDataPlot[]
```

Out[•]=



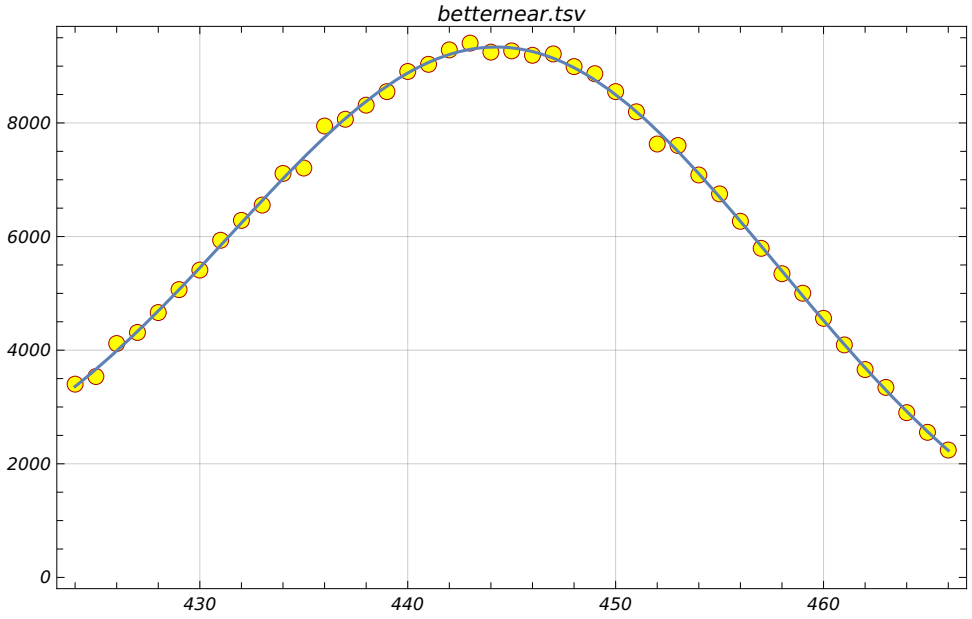

```
(* We obtain the position of the 0.66 MeV peak as before by fitting
  the 0.66 MeV peak against a Gaussian+Linear and fiding the μ. *)
```

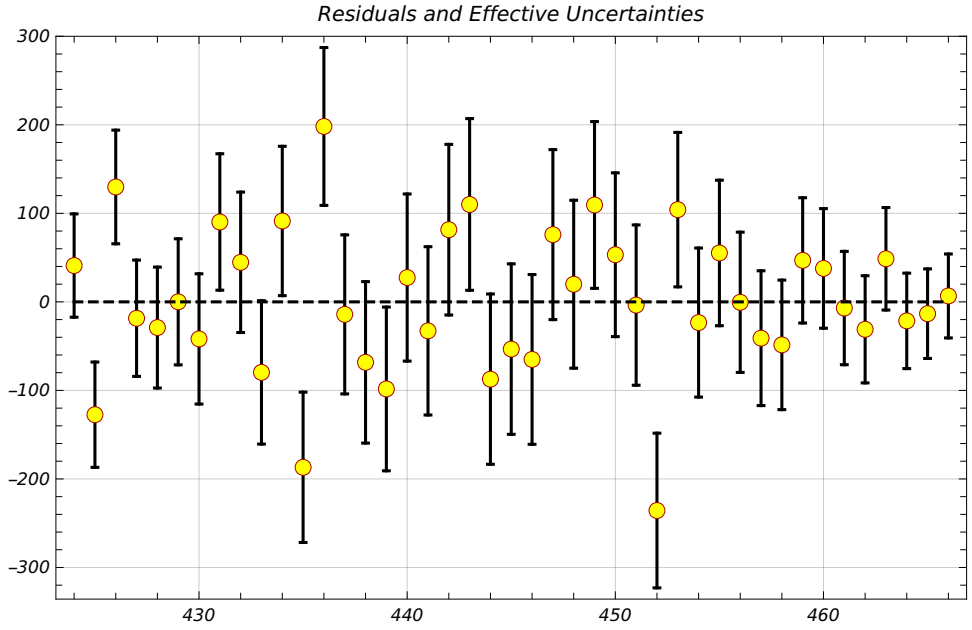
In[•]:=
```
MakePoisson[Xerrors -> True]
```

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[]**





Out[•]=

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

c= 297.559	d= −22.0464		
σ_c= 159.204	σ_d= 2.39246		
y_{max}= 9033.77	μ= 444.674	sigma= 13.1145	
$\sigma_{y_{max}}$= 163.995	σ_μ= 0.0999985	σ_{sigma}= 0.211323	$\chi^2/(n-5)$= 1.1366

We observe a shift in the position of the peak of about 13 channels, or less than 3%. We therefore have significant rate-related gain shift in our spectra and may have to carry out the experiment again while

controlling for rate at the detector. We may also have to account for the systematic error introduced by the shift (and the drift described in the next section) in the data analysis.

Long-term stability of gain

We similarly verify the long-term stability of the system's gain by comparing Co-60 spectra taken at an interval of each other and observing any drift in the position of the peaks. We accurately find the positions of the peaks using the same procedure as above (fit Gaussian+Linear to full-energy a full energy peak, and the mean μ will give a maximum likelihood estimator of the position of the peak).

```
(* We find the position of one of the Co-60 peaks for the first spectrum. *)
```

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

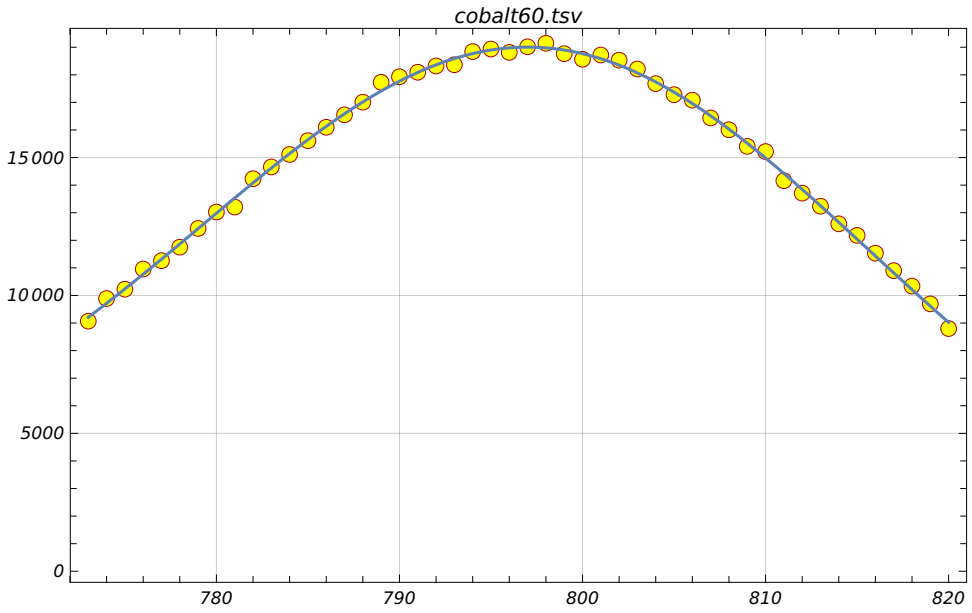
In[•]:=
```
MakePoisson[Xerrors -> True]
```

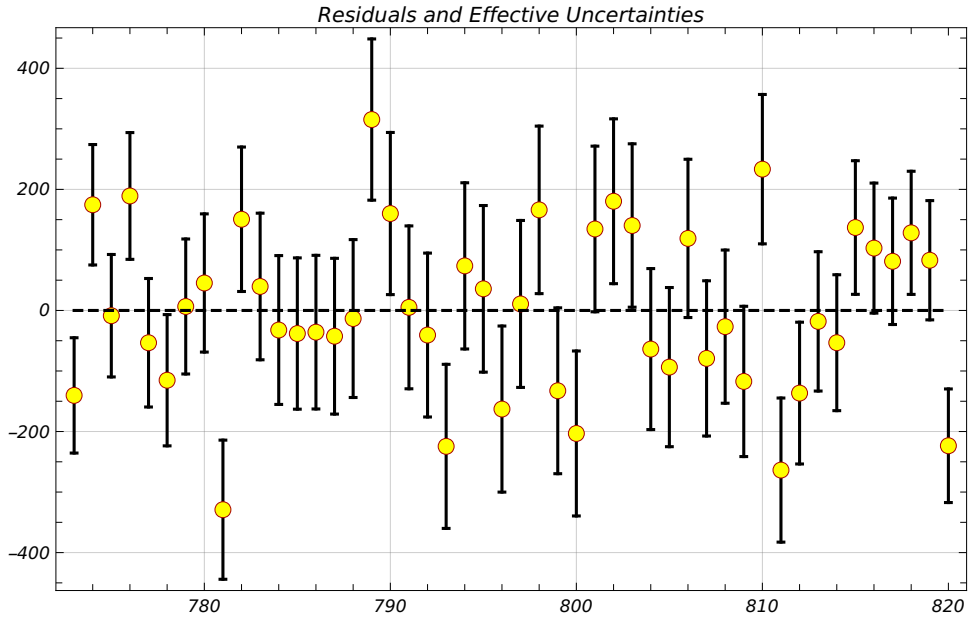
Change the plot from LinearDataPlot[ ] to something else like LogDataPlot[ ] if you wish. If the plot has a log X-axis (like LogLogDataPlot[ ]), then the Log option to SetXRange[ ] must be changed to Log → True

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[]
```





Out[•]=

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

c=	d=		
1406.1	−27.4483		
σ_c=	σ_d=		
583.759	5.38559		
y_{max}=	μ=	sigma=	
17591.	797.52	18.2437	
$\sigma_{y_{max}}$=	σ_μ=	σ_{sigma}=	$\chi^2/(n-5)$=
624.25	0.166652	0.384855	1.49795

```
(* We now do the same for the second Co-60 spectrum. *)
```

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

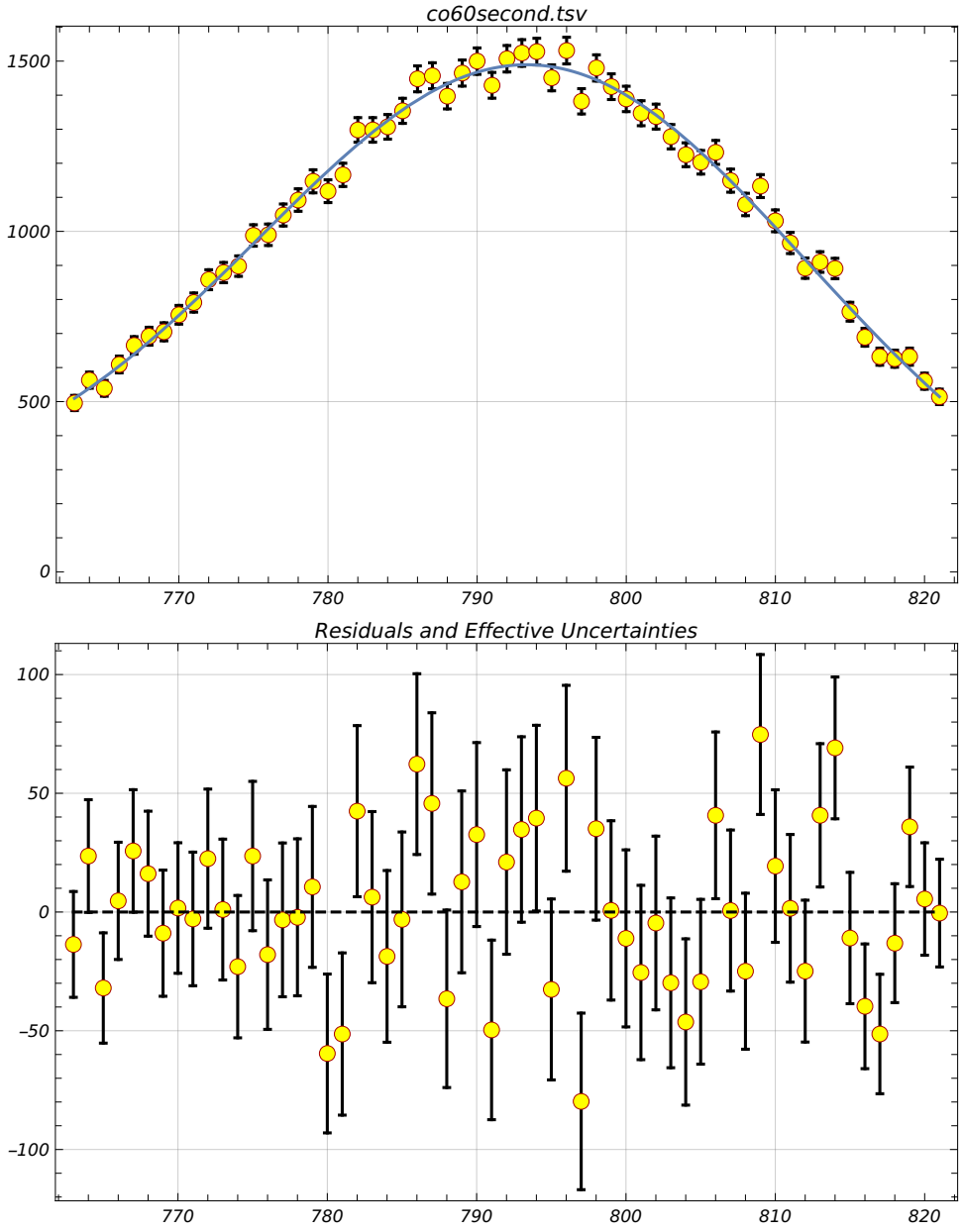
```
In[•]:= MakePoisson[Xerrors -> True]
```

Change the plot from LinearDataPlot[ ] to something else like LogDataPlot[ ] if you wish. If the plot has a log X-axis (like LogLogDataPlot[ ]), then the Log option to SetXRange[ ] must be changed to Log → 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[]**



Out[•]=

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

$c =$ 107.41	$d =$ −2.04662		
$\sigma_c =$ 62.5355	$\sigma_d =$ 0.721534		
$y_{max} =$ 1381.52	$\mu =$ 793.809	sigma = 18.3831	
$\sigma_{y_{max}} =$ 67.5434	$\sigma_\mu =$ 0.318507	$\sigma_{sigma} =$ 0.801073	$\chi^2/(n-5) =$ 1.10908

We observe a shift in the peak position of about 4 channels or 0.5%, such that small but observable drift occurs. The response of the system is relatively stable with respect to time, but we will have to consider

in what way the observed drift affects our estimate for the uncertainties in our peak positions (for instance, what type of error this contributes to).

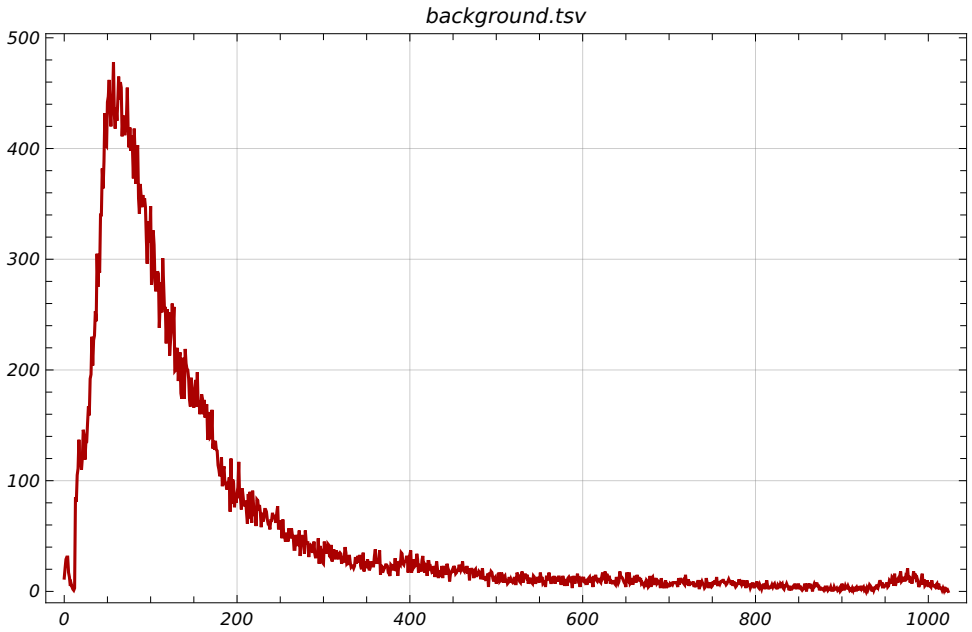
Background spectrum

A spectrum of the background noise is taken over several minutes, to be used as a baseline.

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

```
In[•]:= LinearDataPlot[]
```

Out[•]=



We observe that the count is small enough that the background will not contribute to observable features of the spectra.

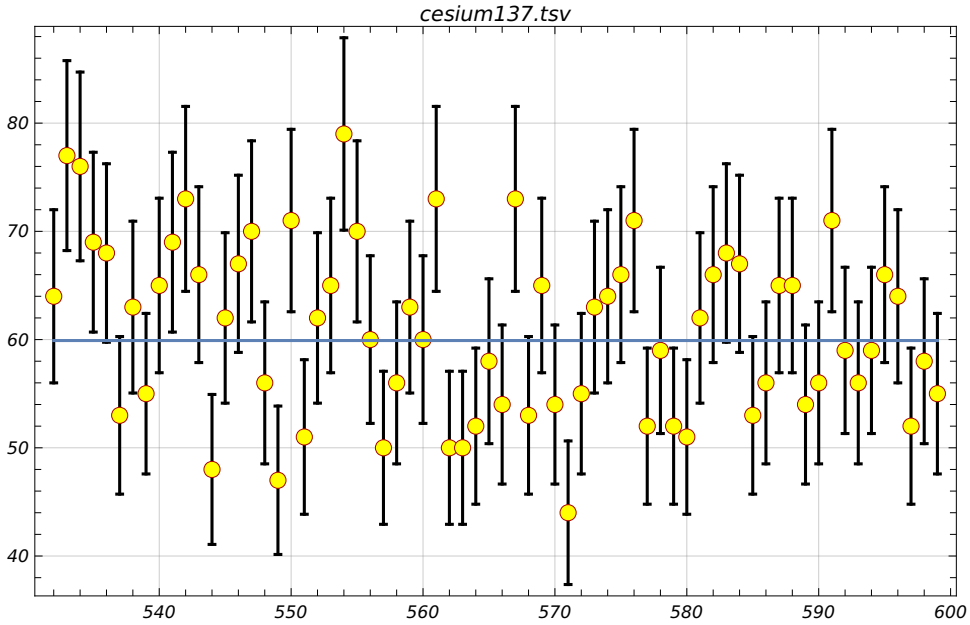
By a rough calibration using the peaks above, we observed a peak with increased counts at ~0.1 MeV. This may correspond to the NaI K-edge as described in the photoelectric absorption section of the lab manual.

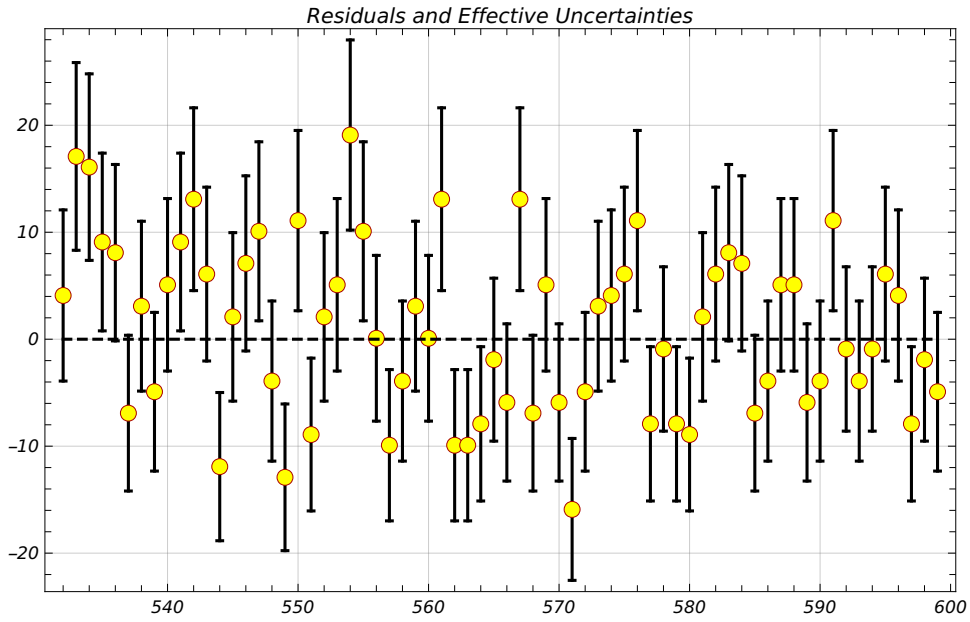
Data Analysis

```
(* We consider a small, flat portion of the C-
 137 spectrum and fit to a constant to observe the channel-to-channel scatter. *)
```

```
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[•]:= ConstantFit[]
```

In[•]:=
```
LinearDifferencePlot[]
```





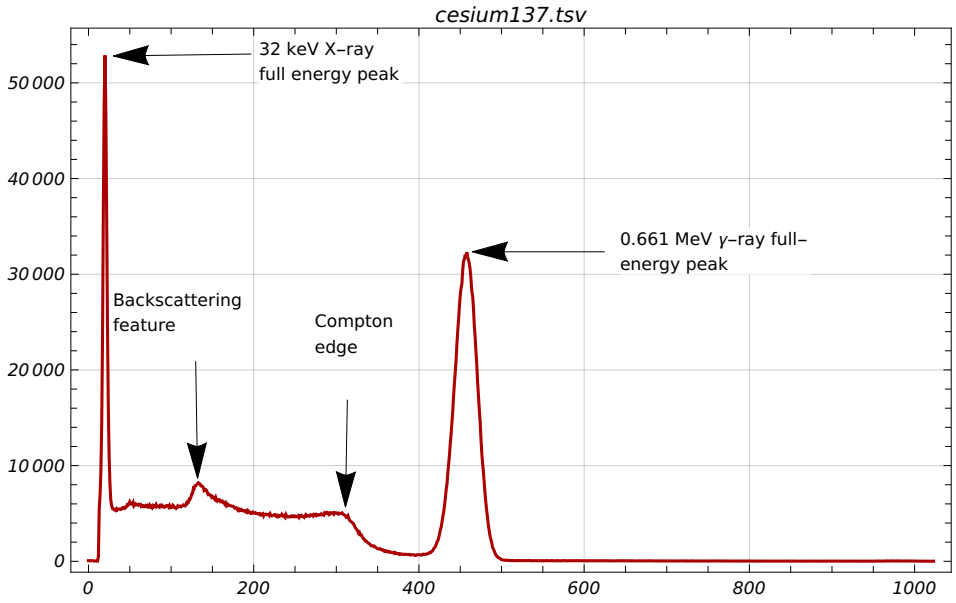
Out[•]=
```
y(x) = a
a=
59.9112
σa=                χ²/(n-1)=
0.938641           1.07515
```

As expected, the linear background very closely fits a constant, with a reduced $\tilde{\chi}^2$very close to 1. The residual plot also allows us to observe that the channel-to-channel scatter should not be significant over high-count features and will not pose an issue when interpreting the spectra of short half-life isotopes, despite being significant with respect to the background count (we will observe a noise feature due to the K-edge of NaI for longer half-life isotopes such as Na-22).

Cs-137 spectrum

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

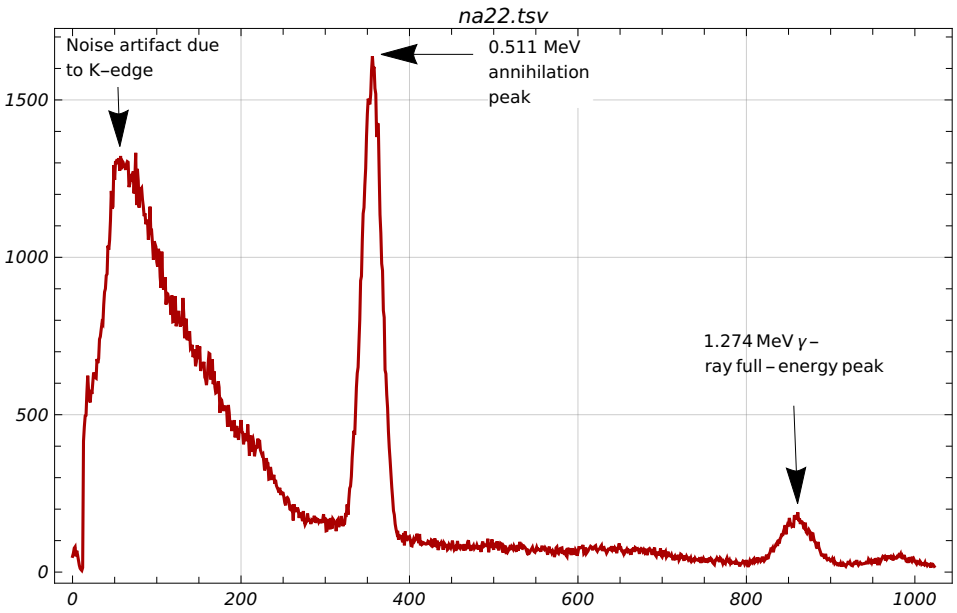
```
In[•]:= LinearDataPlot[]
```



Na-22 spectrum

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

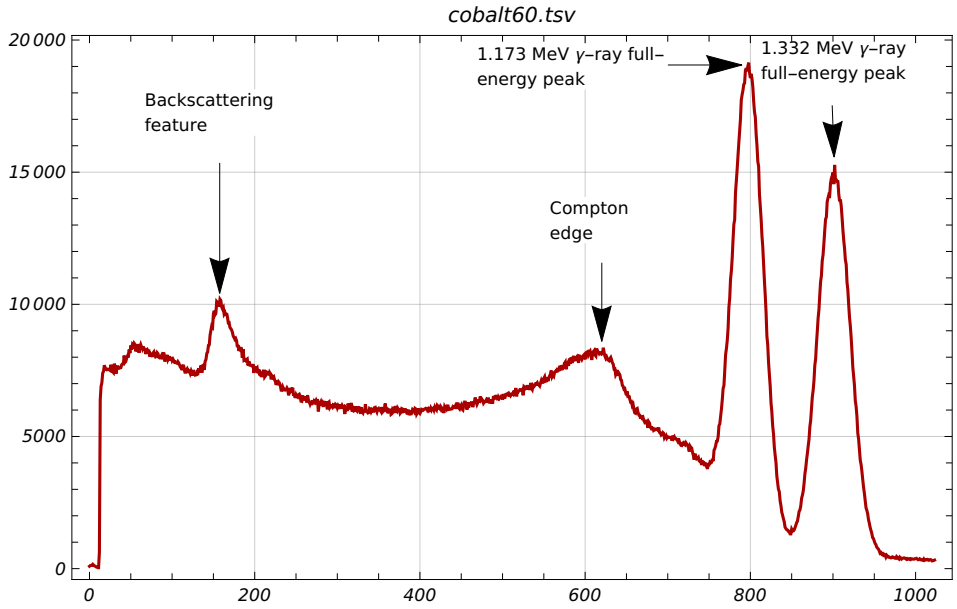
```
In[•]:= LinearDataPlot[]
```



Co-60 spectrum

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

In[•]:=
```
LinearDataPlot[]
```

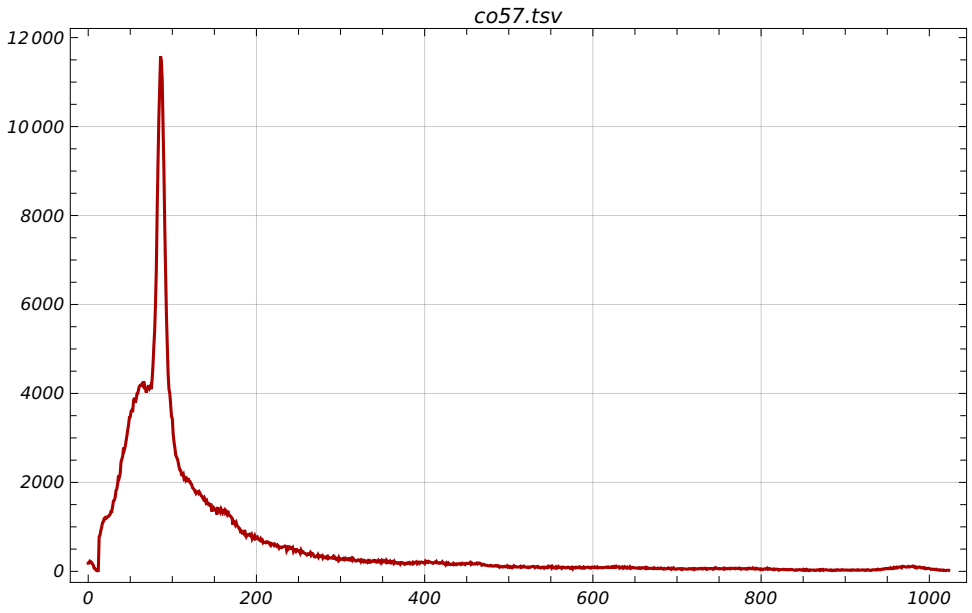


Co-57 spectrum

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

In[•]:=
```
LinearDataPlot[]
```

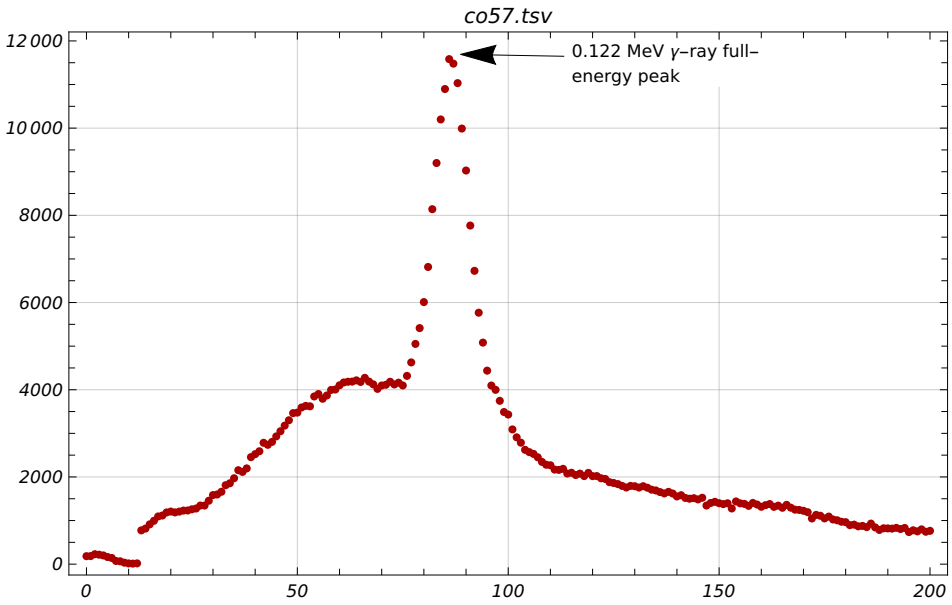
Out[•]=




```
(* We select the leftmost section of the
 spectrum to better discern any interesting features. *)
```

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[•]:=
```
LinearDataPlot[]
```



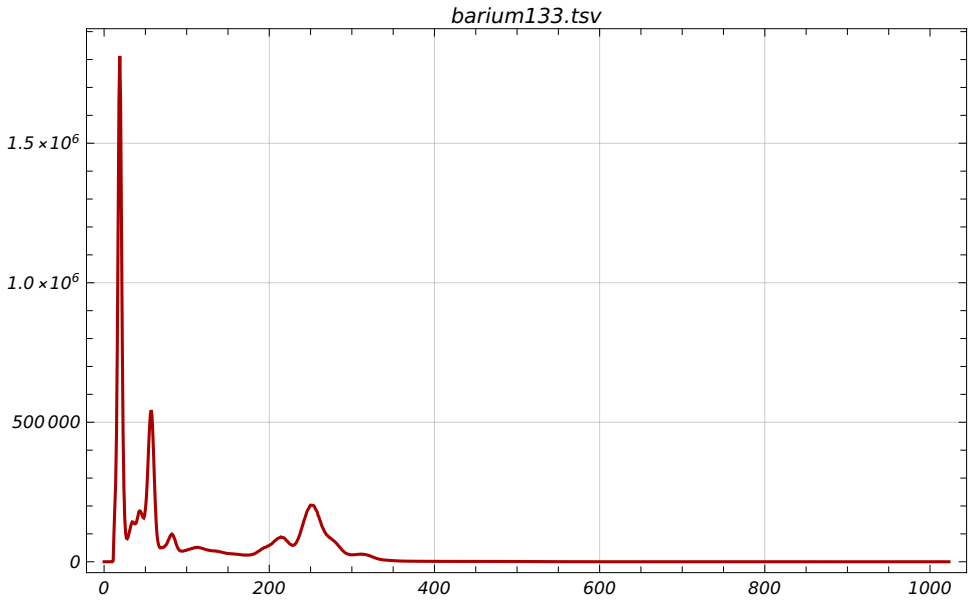
The 0.136 γ-ray full energy peak has a low count (as expected) and cannot be clearly discerned, and therefore will be left out of the calibration set. The 0.0144 MeV γ-ray full energy peak is too close to the LLD to be reliably discerned, and will also be left out of the calibration set.

Ba-133 spectrum

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

In[•]:=
```
LinearDataPlot[]
```

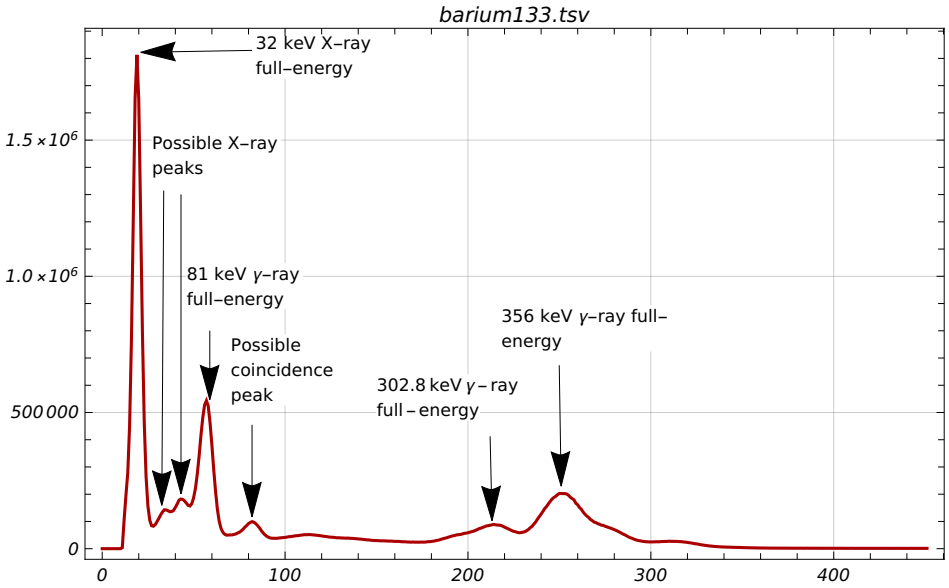
Out[•]=




```
(* We again select the leftmost section of the
 spectrum to better discern any interesting features. *)
```

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[•]:=
```
LinearDataPlot[]
```



Other features we would expect to see from the decay cycle of Ba-133 are not clearly discernible and therefore not labelled inMinimizing Chi^2 the above figure. We only consider the full-energy peaks for the calibration set, because those are reliably located on the spectrum above. We may have to exclude the 302.8 keV peak if we find that it is too close to the larger 356 keV peak and that they have superposed significantly near the 302.8 keV peak, skewing the Gaussian+Linear fit.

Constructing calibration dataset

In[42]:=
```
(* Define loop to prompt for selection of peak from data in file,
then returns fit and prompt for appending
 {energy, mean, sigmean} to calibration dataset calData *)

SetDirectory[NotebookDirectory[]];
fileList =
  {"na22.tsv", "barium133.tsv", "cesium137.tsv", "co57.tsv", "cobalt60.tsv"};
calData = {{"Energy (MeV)", "mean", "sigmean"}};
ClearAll[energy]

For[
 i = 1,
 i ≤ Length[fileList],
 (
  Block[
   {Print},
   LoadFile[fileList[[i]], SkipLines → {"Data:", "Counts"}];
   MakePoisson[Xerrors -> True];
   With[{x = SetXRange[ LinearDataPlot[], Log -> False,
        Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
  ];
  GaussianLFit[];
  DialogInput[
   Grid[
    {
     {"Energy (MeV):", InputField[Dynamic[energy], Number]},
     {
      CancelButton[DialogReturn[]],
      DefaultButton[
       DialogReturn[(AppendTo[calData, {energy, mean, sigmean}];)]],
```

```
      Button["Next", DialogReturn[(AppendTo[calData, {energy, mean, sigmean}];
         i++;)]]
     }
    },
    Spacings → {1, Automatic}, Alignment → Left]
  ];
 )
]

calData = Table[If[calData[[i]][[1]] == 0, Null, calData[[i]]],
    {i, Length[calData]}] /. Null → Sequence[];
```

In[235]:=
```
table = Table[
   {NumberForm[calData[[i]][[1]], {4, 2}], NumberForm[calData[[i]][[2]], {4, 2}],
    NumberForm[calData[[i]][[3]], {4, 2}]}, {i, 2, Length[calData]}];
PrependTo[table, {"Energy (MeV)", "μ", "σμ"}];
Grid[table, Alignment → Left, Spacings → {2, 1}, Frame → All,
 ItemStyle → "Text", Background → {{Gray, None}, {LightGray, None}}]
```

Out[237]=

Energy (MeV)	μ	σ_μ
1.27	859.20	0.48
0.66	458.20	0.14
0.12	86.41	0.09
1.17	797.10	0.26
1.33	901.40	0.26

The table above is our calibration dataset, whereby the μ's represent the channel-positions of full-energy peaks and σ_μ's represent the uncertainty in μ. Note that we must adjust the uncertainties to account for rate-related gain shift and time drift before we can use them to construct the calibration function.

Note that we have not considered X-ray peaks for the calibration dataset because it was not possible to obtain a Gaussian+Linear fit with a $\tilde{\chi}^2$ reasonably close to 1. For the same reason, we have not considered the Ba-133 spectrum - in this case, the bad fit may be because of close and numerous peaks superposing.

Constructing the calibration function

We start by plotting the calibration data as-is, as we cannot adjust the uncertainties for rate-related gain-shift without information about the rates at the detector (which our data does not contain), and drift will be treated as a systematic uncertainty later on.

```
(* We export our calibration dataset for use with CurveFit. *)
```

In[61]:=
```
Export["calibration_data.dat", calData /. "mean" → "S mean"];
```

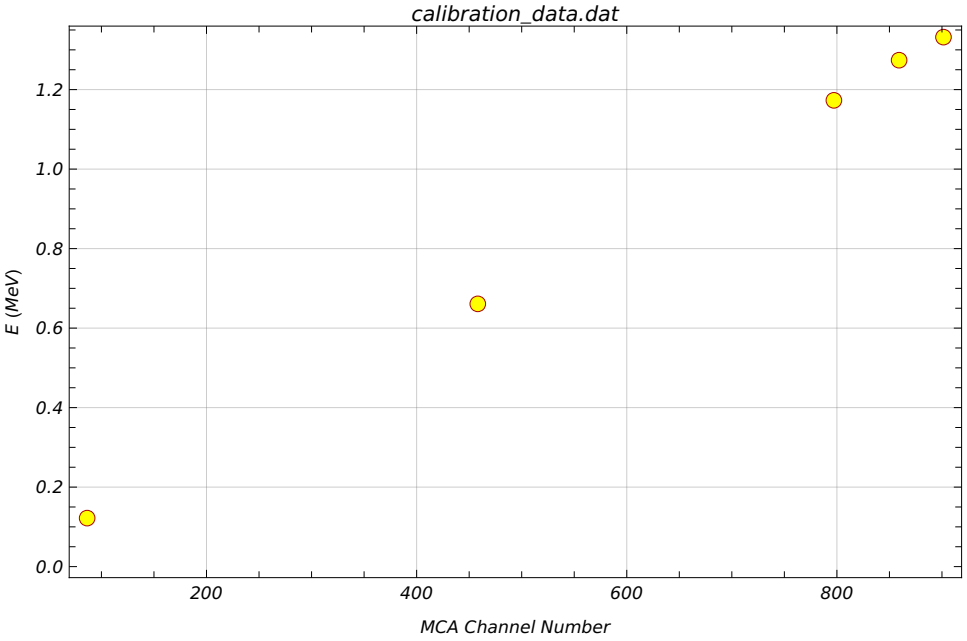
In[93]:=
```
With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"data files" -> {"*.dat", "*.mca"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name]]
]
```

In[94]:=
```
SwitchXXandYY[]
```

xx and yy have been switched (so have sx and sy).

In[84]:=
```
LinearDataPlot[FrameLabel → {"MCA Channel Number", "E (MeV)"}]
```

Out[84]=

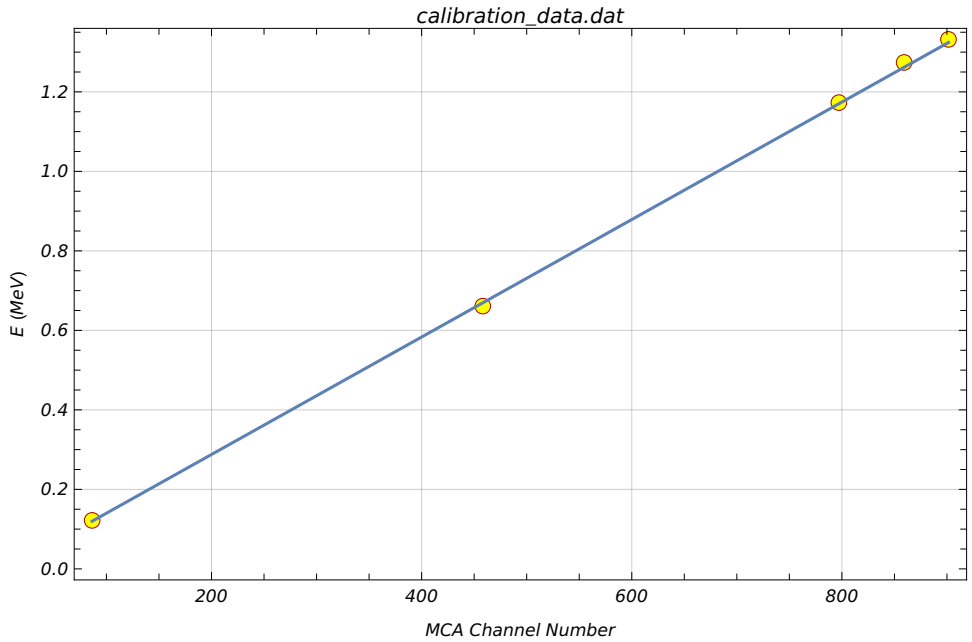


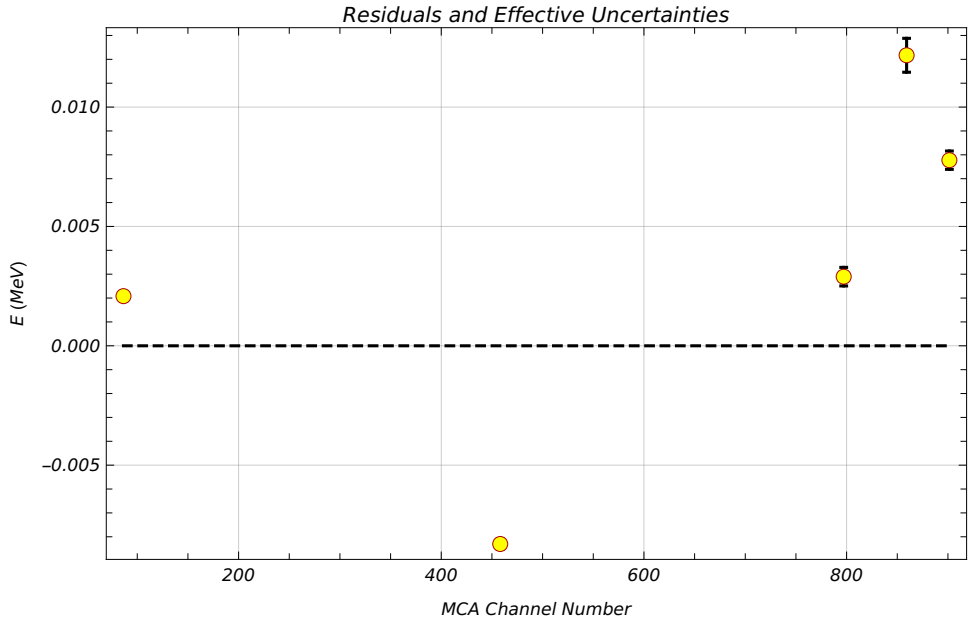

```
(* We attempt a linear fit. *)
```

In[85]:=
```
LinearFit[]
```

In[86]:=
```
LinearDifferencePlot[FrameLabel → {"MCA Channel Number", "E (MeV)"}]
```

Out[86]=





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

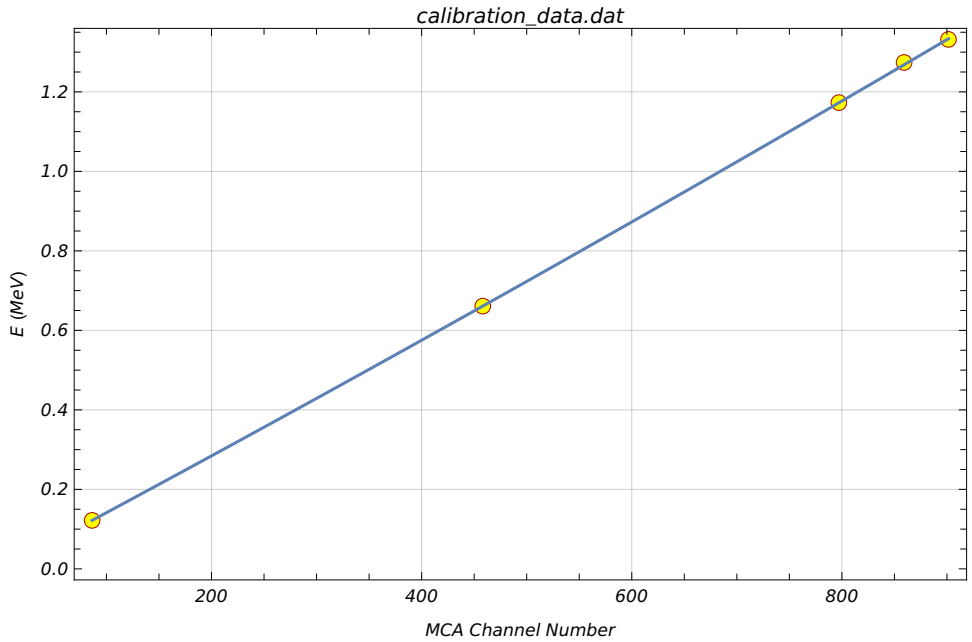
$a =$	$b =$	
-0.00775362	0.00147762	
$\sigma_a =$	$\sigma_b =$	$\chi^2/(n-2) =$
0.000151915	3.58895×10^{-7}	904.393

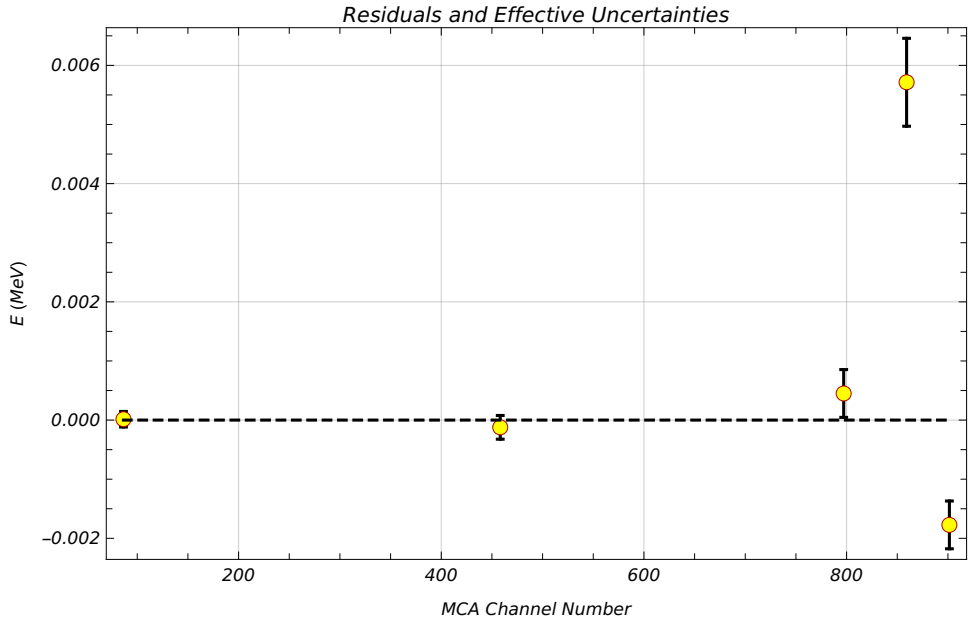
```
(* We attempt a quadratic fit. *)
```

In[87]:=
```
QuadraticFit[]
```

In[89]:=
```
LinearDifferencePlot[FrameLabel → {"MCA Channel Number", "E (MeV)"}]
```

Out[89]=





$y(x) = a + b\,x + c\,x^2$

$a =$	$b =$	$c =$	
-0.0000261005	0.00140492	8.28818×10^{-8}	
$\sigma_a =$	$\sigma_b =$	$\sigma_c =$	$\chi^2/(n-3) =$
0.000207108	1.45869×10^{-6}	1.63662×10^{-9}	39.9795

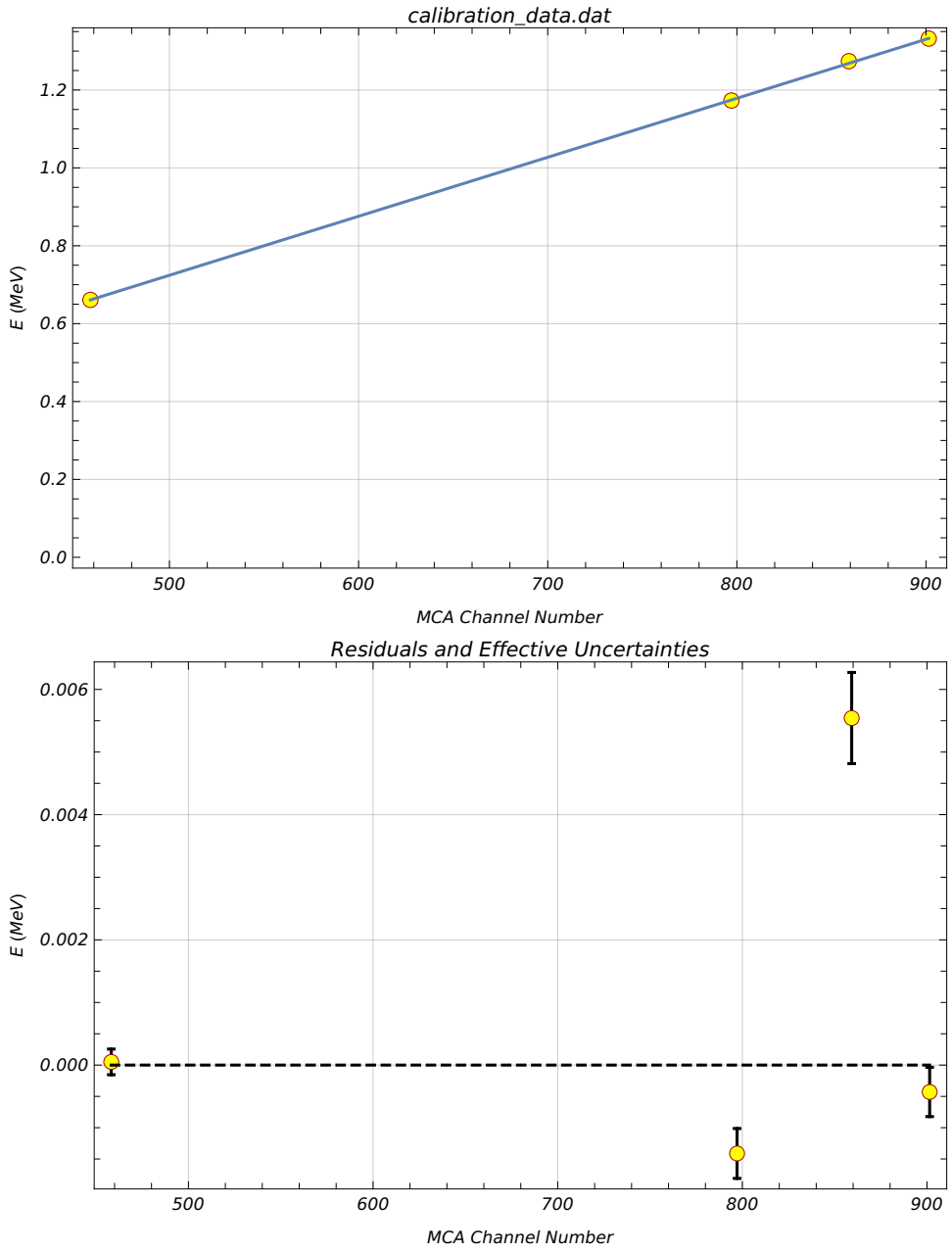
```
(* We attempt a linear fit of only the last 4 data points. *)
```

In[90]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
    Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

In[91]:=
```
LinearFit[]
```

In[92]:=
```
LinearDifferencePlot[FrameLabel → {"MCA Channel Number", "E (MeV)"}]
```

Out[92]=



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

$a =$	$b =$	
-0.0332233	0.00151497	
$\sigma_a =$	$\sigma_b =$	$\chi^2/(n-2) =$
0.000533104	8.35693×10^{-7}	35.8873

We observe high $\tilde{\chi}^2$ for the fits (of the order of 30-40) because of the unreasonably small error bars we have used. It would be difficult to systematically adjust the error bars for rate-related gain shift, as we do not know the detector's rate response function, and we do not have information regarding the rate at the detector anyway. Therefore, we simply use the $\tilde{\chi}^2$ as an order of magnitude measure by which we compare the fits we have obtained above.

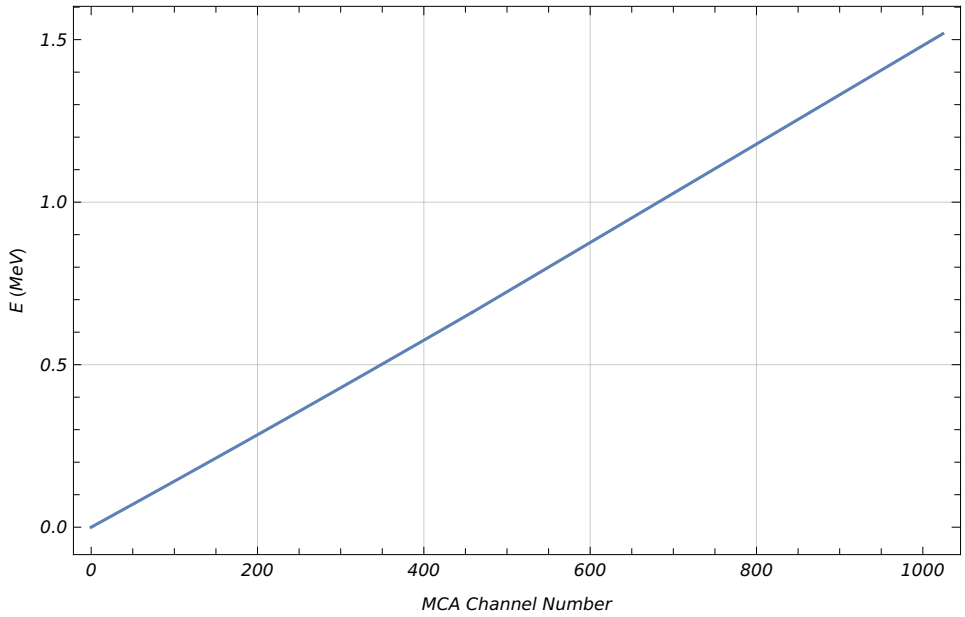
We observe that the response for the data points well above 100 keV are well modelled by the linear fit, whereas those closer to and below 100 keV seem better modelled by the quadratic fit. We tentatively define a calibration function as a piecewise of the two functions. We can see that the linear fit and the quadratic fit both have negligible residuals for the data point at channel number 458. Therefore, we chose this to be the point of transition of our piecewise (i.e. our calibration curve will correspond to the quadratic fit for channels 458 and below, and to the linear fit for channels over 458.

```
(* We define our calibration function. *)
```

In[101]:=
```
calPiecewise[number_] :=
 Piecewise[{{0.00140492 number + (8.28818 * 10^-8) number^2, number ≤ 458},
   {-0.0332233 + 0.00151497 number, number > 458}}]
```

In[114]:=
```
Plot[calPiecewise[x], {x, 0, 1024}, FrameLabel → {"MCA Channel Number", "E (MeV)"}]
```

Out[114]=



Note that even though this is the best we can come up with from the calibration data we have, this function is not very reliable at low energies, since we only have one data point for the seemingly quadratic region.

We observe above from the order of magnitude of the σ's that uncertainties in peak positions originating from the statistical nature of the data contribute negligibly to uncertainties in our calibration. As we will see below, we will in fact neglect those in favor of much more significant sources of uncertainty.

Identifying and finding the energies of features using the calibration

Na-22 positron annihilation peak

```
(* We first identify as before the channel location of the annihilation peak. *)
```

In[136]:=
```
With[ {name = SystemDialogInput["FileOpen",
    {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name, SkipLines → {"Data:", "Counts"}]]
]
```

In[137]:=
```
MakePoisson[Xerrors -> True]
```

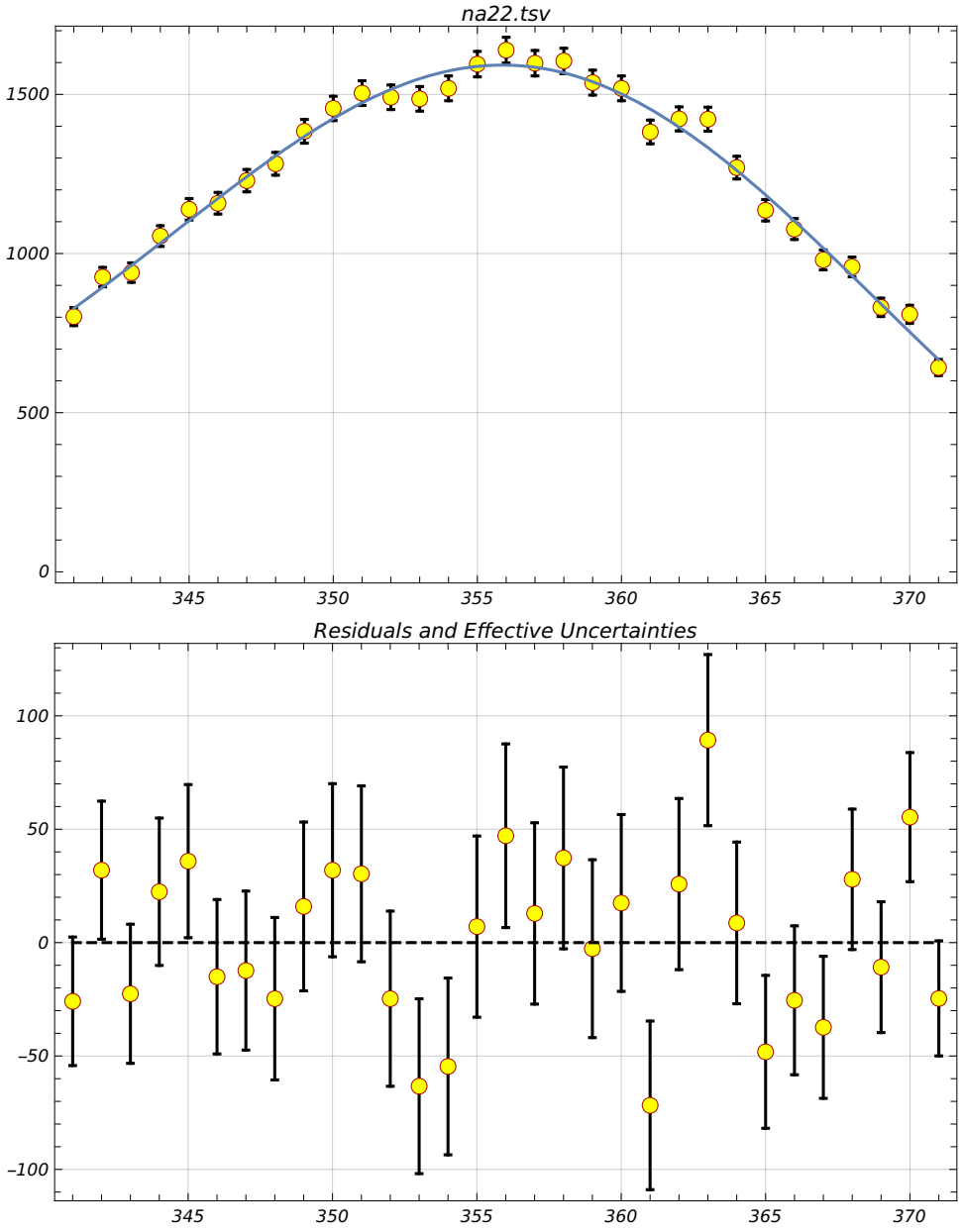
Change the plot from LinearDataPlot[ ] to something else like LogDataPlot[ ] if you wish. If the plot has a log X-axis (like LogLogDataPlot[ ]), then the Log option to SetXRange[ ] must be changed to Log → True

In[138]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
    Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

In[139]:=
```
GaussianLFit[]
```

In[140]:= **LinearDifferencePlot[]**

Out[140]=



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

$c =$	$d =$		
-10.3182	-8.9165		
$\sigma_c =$	$\sigma_d =$		
358.209	3.64034		
$y_{max} =$	$\mu =$	$sigma =$	
1598.58	356.703	12.1994	
$\sigma_{y_{max}} =$	$\sigma_\mu =$	$\sigma_{sigma} =$	$\chi^2/(n-5) =$
363.84	0.52538	1.59897	1.29293

```
(* We observe that the annihilation peak occurs at channel μ = 356.7,
and we can use the calibration function to find the corresponding energy. *)
```

In[145]:=
```
annihilationEnergy = NumberForm[calPiecewise[356.703], 4] (* MeV *)
```

Out[145]//NumberForm=
```
0.5117
```

We have no information regarding the count-rate and difference in rates for the two data points we took to investigate rate-dependent gain-shift, or regarding the count rates for our other spectra. We only have a single data point regarding the gain shift, making it difficult to quantify the error induced by this shift in our calibration model. The simplest choice here is to assume that the gain function is linear (we have seen that this is not quite the case, but should be a good enough approximation to give us an idea of the error induced) and assume that the shift measured is representative of the error in peak position for that peak due to rate-related gain shift (we have no way of finding out if this is correct, but it is the only data point we have and therefore the only guess we can make). Then we use the shift to estimate the fractional uncertainty in the peak position, which by our assumptions should then be the uncertainty in the gain and therefore the fractional uncertainty in the position of our other features as well (including the annihilation peak for Na-22 we are interested in).

In[238]:=
```
fractionalUncertainty = (13 / 458) // N
```

Out[238]=
```
0.0283843
```

In[239]:=
```
uncertaintyAnnihilation = fractionalUncertainty * 356.7
```

Out[239]=
```
10.1247
```

In[240]:=
```
calPiecewise[366.8] - calPiecewise[356.7]
```

Out[240]=
```
0.0147953
```

In[241]:=
```
calPiecewise[346.6] - calPiecewise[356.7]
```

Out[241]=
```
-0.0147784
```

By the above, we then have the energy of the positron annihilation peak as 0.512 ± 0.015 MeV. However, there are also errors induced by time-drift of the gain, as measured above. This time again, we only have a single data point to investigate the shift, and therefore do not know the time-scale of the variation of the gain. However, the very good Gaussian+Linear fits we have obtained over relatively long run-times suggest that the time scale is relatively long (otherwise the spectra would have been distorted and we would not have obtained clean fits), so we treat the error induced by the time-drift of gain as a systematic error, separate from the error above. We again resort to the same assumptions as above to estimate the systematic error.

In[244]:=
```
fractionalSystematic = (4 / 798) // N
```

Out[244]=
```
0.00501253
```

In[245]:=
```
systematicAnnihilation = fractionalSystematic * 797.5
```

Out[245]=
```
3.99749
```

In[247]:=
```
calPiecewise[360.7] - calPiecewise[356.7]
```
Out[247]=
```
0.00585752
```

In[248]:=
```
calPiecewise[352.8] - calPiecewise[356.7]
```
Out[248]=
```
-0.00570853
```

We then have the energy of the positron annihilation peak as 0.512 ± 0.015 ± 0.006 MeV, which should also correspond to the rest energy of the electron. While our estimates for the error bars are admittedly rough, hardly rigorous and may be unreliable, here we find that we are within error of the accepted value for m_e = 0.510999 MeV.

Compton edge for Cs-137

We expect the Compton edge to occur at T_e = 0.4773 MeV. We use our calibration function to find the channel that this energy corresponds to:

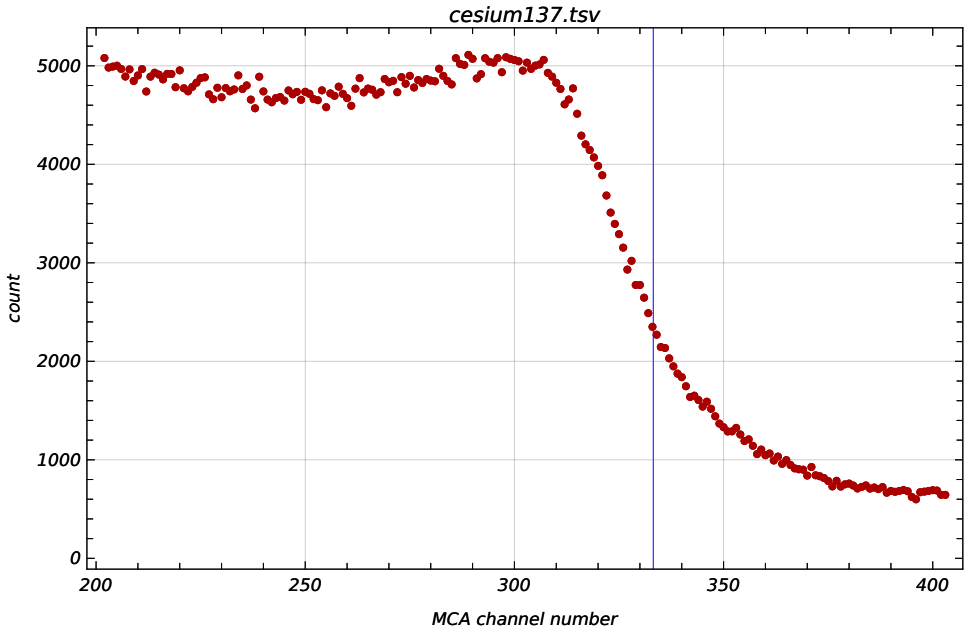
In[250]:=
```
NSolve[calPiecewise[channel] == 0.4773 && channel > 0, channel]
```
Out[250]=
```
{{channel → 333.186}}
```

In[266]:=
```
With[ {name = SystemDialogInput["FileOpen",
     {DataFileName, {"spectrum files" -> {"*.tsv"}, "all files" -> {"*"}}}]},
 If[ name =!= $Canceled,
  LoadFile[name, SkipLines → {"Data:", "Counts"}]]
]
```

In[267]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
     Label -> "Set the X values for the range you wish to keep." ]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

In[270]:=
```
Overlay[{LinearDataPlot[FrameLabel → {"MCA channel number", "count"},
   GridLines → {{333.186}, {}}, GridLinesStyle → Blue],
  LinearDataPlot[FrameLabel → {"MCA channel number", "count"}]}]
```

Out[270]=



The vertical blue line represents the channel corresponding to T_e, the expected energy at which the Compton edge occurs. We observe that a little more than halfway down from the peak of the hump (as opposed to the expected location about 1/3 of the way down shown in the lab manual). This may be due to the error in the calibration function.

Backscatter feature

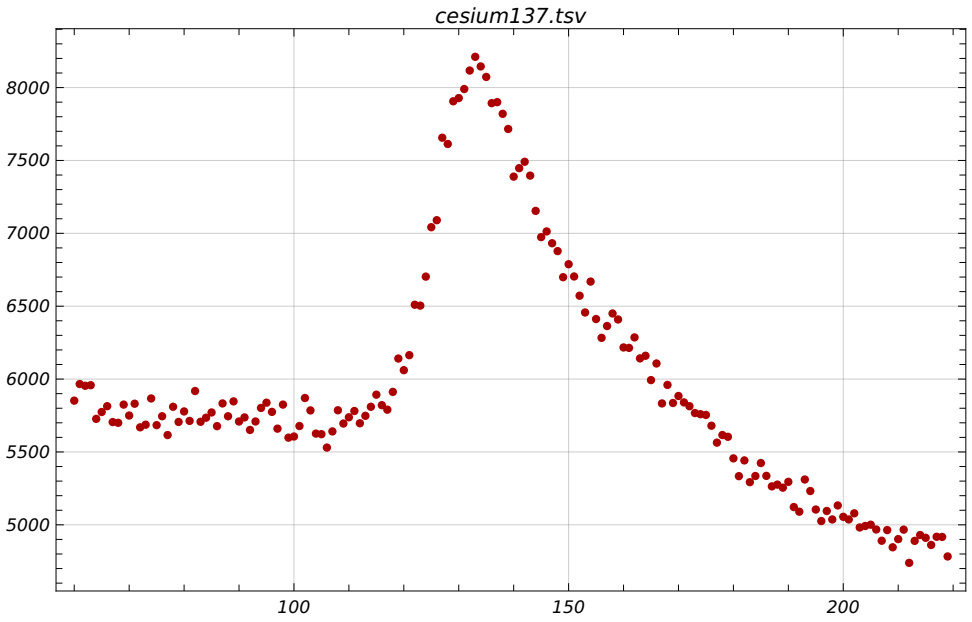
In[271]:=
```
Undo[]
```

Change the plot from LinearDataPlot[ ] to something else like LogDataPlot[ ] if you wish. If the plot has a log X-axis (like LogLogDataPlot[ ]), then the Log option to SetXRange[ ] must be changed to Log → True

In[272]:=
```
With[{x = SetXRange[ LinearDataPlot[], Log -> False,
    Label -> "Set the X values for the range you wish to keep."]},
 Print[x];

 XRangeKeep[Sequence @@ x]
]
```

In[276]:=
```
LinearDataPlot[]
```

Out[276]=




```
(* We observe that the backscatter feature is located at
 the maximum point of our dataset. By opening the data table,
we find that this is the point corresponding to the MCA channel 133. *)
```

In[278]:=
```
backscatterEnergy = calPiecewise[133] (* MeV *)
```

Out[278]= 0.18832

We recall that in the pre-lab we calculated a backscattering energy of 0.1843 MeV. Our estimate of 0.1883 MeV based on the location of the peak is therefore within ~2.2% of the expected energy of the backscattering peak, which again is consistent with the error in our calibration function. Indeed, we can find the error bars on our estimate as before:

In[280]:=
```
uncertaintyBackscatter = fractionalUncertainty * 133
```

Out[280]= 3.77511

In[284]:=
```
calPiecewise[133 + 3.775] - calPiecewise[133]
```

Out[284]= 0.00538798

In[285]:=
```
calPiecewise[133 - 3.775] - calPiecewise[133]
```

Out[285]= −0.00538562

In[282]:=
```
systematicBackscatter = fractionalSystematic * 133
```

Out[282]= 0.666667

In[286]:=
```
calPiecewise[133 + 0.6666] - calPiecewise[133]
```

Out[286]= 0.000951253

In[287]:=
```
calPiecewise[133 - 0.6666] - calPiecewise[133]
```

Out[287]=
```
-0.000951179
```

and from this we obtain an estimate with uncertainty for the backscattering energy of 0.188 ± 0.005 ± 0.001 MeV, and indeed we are within error of the expected value.