Introduction to Fitting ASCII Data with Errors: Single−Component Source Models

Sherpa Threads (CIAO 3.4)

Table of Contents

Introduction to Fitting ASCII Data with Errors: Single−Component Source Models

Sherpa Threads

Overview

Last Update: 1 Dec 2006 − reviewed for CIAO 3.4: no changes

Synopsis:

This thread provides a detailed introduction to *Sherpa*. 1−D data from an ASCII datafile are empirically fit with polynomials, then also fit with a given polynomial function. In addition, a second 1−D ASCII dataset is input and fit with a polynomial.

Proceed to the HTML or hardcopy (PDF: A4 | letter) version of the thread.

Getting Started

Please follow the "Sherpa Threads: Getting Started" thread.

Reading ASCII Data & Errors Into Sherpa

In this thread, we wish to fit 1−D data from the following ASCII dataset:

```
sherpa> $more data1.dat
0.5    1.6454   0.04114
1.5    1.7236   0.04114
2.5    1.9472   0.04114
3.5    2.2348   0.04114
4.5    2.6187   0.04114
5.5    2.8642   0.04114
6.5    3.1263   0.04114
7.5    3.2073   0.04114
8.5    3.2852   0.04114
9.5    3.3092   0.04114
10.5   3.4496   0.04114
```

This dataset is input into *Sherpa* using the READ command:

```
sherpa> READ DATA data1.dat 1 2
sherpa> SHOW DATA
Y Column: Counts
  Dimensions: 1
  Total Size: 11 bins (or pixels)
  Axis: 0; Name: Bin
  Length: 11 bins (or pixels)
```

```
File Name: data1.dat
SubSection (if any):
File Type: ASCII
[0.500000] =     1.6454
[1.500000] =     1.7236
[2.500000] =     1.9472
[3.500000] =     2.2348
[4.500000] =     2.6187
[5.500000] =     2.8642
[6.500000] =     3.1263
[7.500000] =     3.2073
[8.500000] =     3.2852
[9.500000] =     3.3092
[10.500000] =     3.4496
```

The third column of the dataset, which contains the errors, is input with the `READ ERRORS` command:

```
sherpa> READ ERRORS data1.dat 1 3
```

Plotting Data

Now the dataset may be plotted:

```
sherpa> LPLOT DATA
```

The CIAO software package includes a plotting tool called *ChIPS* (Chandra Imaging and Plotting System). *ChIPS* plotting commands are available for use within *Sherpa* and may be useful for modifying the appearance of plots:

```
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

The *ChIPS* commands `XLABEL` and `YLABEL` add labels to the X and Y axes, respectively. Note that the command `REDRAW` must be issued to update the display. Figure 1 shows the resulting plot.

Establishing a Model Component

We wish to fit these data using a polynomial. The *Sherpa* model name for a 1−D polynomial function is `POLYNOM1D`. Note that the entire of list of all models available within *Sherpa* may be obtained by typing `ahelp models`.

The `POLYNOM1D` model component is established and it is named `model1` in this session.

```
sherpa> POLYNOM1D[model1]
model1.c0 parameter value [2.5475]
model1.c1 parameter value [0]
model1.c2 parameter value [0]
model1.c3 parameter value [0]
model1.c4 parameter value [0]
model1.c5 parameter value [0]
model1.c6 parameter value [0]
model1.c7 parameter value [0]
model1.c8 parameter value [0]
model1.offset parameter value [0]
```

Since a dataset has already been input, *Sherpa* estimates the initial parameter values (and the minimum and maximum for their ranges) for this model based on the data. If a dataset had not been previously input, the parameter values of the model would be set to the defaults. *Sherpa* then prompts the user for changes to these estimates. In this example, we accept the initial parameter value estimates by hitting `<RETURN>` at each parameter value prompt. *Sherpa* prompts for the parameter values if `PARAMPROMPT` is ON.

The `SHOW` command may be used to examine the details of the established model component:

```
sherpa> SHOW model1
polynom1d[model1]  (integrate: on)
    Param   Type      Value        Min        Max                  Units
    -----   ----      -----        ---        ---                  -----
 1     c0 thawed     2.5475    -1.6454     3.4496
 2     c1 frozen          0    -18.042     18.042
 3     c2 frozen          0    -1.8042     1.8042
 4     c3 frozen          0    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset frozen          0       -0.5       10.5
```

Note from the above output that this model component is set to `(integrate: on)` by default. INTEGRATE allows to turn on the integration of the model over the bin with ON/OFF options. However, even if `(integrate: on)`, no integration will be performed when fitting unbinned data, such as that contained in dataset `data1.dat`. That is, no integration will be performed when fitting data that is not input as `HISTOGRAM` or that is not PHA or imaging data. For clarity, integration is turned off below and the model values are taken at the center of the bin:

```
sherpa> model1 INTEGRATE OFF
sherpa> SHOW model1
polynom1d[model1]  (integrate: off)
    Param   Type      Value        Min        Max                  Units
    -----   ----      -----        ---        ---                  -----
 1     c0 thawed     2.5475    -1.6454     3.4496
 2     c1 frozen          0    -18.042     18.042
 3     c2 frozen          0    -1.8042     1.8042
 4     c3 frozen          0    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset frozen          0       -0.5       10.5
```

Defining a Source Model Expression

In order to fit the dataset with the model component that has been established, the model must be defined as the source model expression to be used for fitting:

```
sherpa> SOURCE = model1
```

The current definition of *Sherpa*'s source model expression may be examined using `SHOW_SOURCE`:

```
sherpa> SHOW SOURCE
Source 1: model1
poly1d[model1]  (integrate: off)
    Param   Type      Value        Min        Max                  Units
    -----   ----      -----        ---        ---                  -----
 1     c0 thawed     2.5475    -1.6454     3.4496
 2     c1 frozen          0    −18.042     18.042
 3     c2 frozen          0    −1.8042     1.8042
 4     c3 frozen          0    −1.6454     3.4496
 5     c4 frozen          0    −1.6454     3.4496
 6     c5 frozen          0    −1.6454     3.4496
 7     c6 frozen          0    −1.6454     3.4496
 8     c7 frozen          0    −1.6454     3.4496
 9     c8 frozen          0    −1.6454     3.4496
10 offset frozen          0       −0.5       10.5
```

This output shows that `model1` is currently defined as the source model expression.

Viewing Method & Statistic Settings

We use *Sherpa*'s default optimization method and statistics for these polynomial fits. The `SHOW` command may be used to view the current method and statistics settings:

```
sherpa> SHOW METHOD
Optimization Method: Levenberg−Marquardt

      Name       Value         Min         Max                    Description
      ----       -----         ---         ---                    -----------
 1   iters        2000           1       10000   Maximum number of iterations
 2     eps       1e−03       1e−09           1              Absolute accuracy
 3   smplx           0           0           1  Refine fit with simplex (0=no)
 4 smplxep           1       1e−04        1000   Switch−to−simplex eps factor
 5 smplxit           3           1          20  Switch−to−simplex iters factor

sherpa> SHOW STATISTIC
Statistic:           Chi−Squared Gehrels
```

Further details about the Levenberg−Marquardt optimization method are available by typing:

```
sherpa> ahelp lev−mar
```

Further details about the Chi−Squared Gehrels statistic are available by typing:

```
sherpa> ahelp chigehrels
```

Thawing Model Parameters & Fitting

To start, we wish to fit these data with a first−order polynomial. The `c1` parameter of `model1` needs to be thawed so that it will be allowed to vary during the fit:

```
sherpa> THAW model1.c1
sherpa> SHOW SOURCE
Source 1: model1
poly1d[model1]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     2.5475    −1.6454     3.4496
 2     c1 thawed          0    −18.042     18.042
 3     c2 frozen          0    −1.8042     1.8042
 4     c3 frozen          0    −1.6454     3.4496
 5     c4 frozen          0    −1.6454     3.4496
 6     c5 frozen          0    −1.6454     3.4496
 7     c6 frozen          0    −1.6454     3.4496
 8     c7 frozen          0    −1.6454     3.4496
 9     c8 frozen          0    −1.6454     3.4496
10 offset frozen          0       −0.5       10.5
```

The dataset is then fit:

```
sherpa> FIT
 LVMQT: V2.0
 LVMQT: initial statistic value = 2815.14
 LVMQT: final statistic value = 151.827 at iteration 5
        model1.c0  1.58227
        model1.c1  0.198455
```

To plot the fit:

```
sherpa> LPLOT FIT
```

The appearance of the plot may be modified as follows:

```
sherpa> C 2 SIMPLELINE
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

The *ChIPS* command `C 2 SIMPLELINE` changes the plot of the fit from a histogram to a line (which is red by default). The other *ChIPS* commands add labels. Figure 2 📷 shows the resulting plot.

Next, we wish to fit these data with a second order polynomial. The `c2` parameter of `model1` needs to be thawed so that it will be allowed to vary during the fit:

```
sherpa> THAW model1.c2

sherpa> FIT
 LVMQT: V2.0
 LVMQT: initial statistic value = 151.827
 LVMQT: final statistic value = 59.0027 at iteration 4
        model1.c0  1.30826
        model1.c1  0.347303
        model1.c2  −0.0135317
```

To plot the fit:

```
sherpa> LPLOT FIT
sherpa> C 2 SIMPLELINE
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

Figure 3 📷 shows the resulting plot.

Finally, we wish to fit these data with a third order polynomial. The `c3` parameter of `model1` therefore needs to be thawed so that it will be allowed to vary during the fit. The data is then fit again and plotted:

```
sherpa> THAW model1.c3
sherpa> FIT
 LVMQT: V2.0
 LVMQT: initial statistic value = 59.0027
 LVMQT: final statistic value = 30.8491 at iteration 5
        model1.c0  1.49843
        model1.c1  0.1447
        model1.c2  0.0322936
        model1.c3  −0.00277729

sherpa> LPLOT FIT
sherpa> C 2 SIMPLELINE
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

Figure 4 shows the resulting plot.

Plotting & Examining Fit Results

A plot of both the fit and residuals may be created as follows:

```
sherpa> LPLOT 2 FIT RESIDUALS
```

Various modifications may be made to these plots:

```
sherpa> # Change the data and fit plots in the 1st drawing
sherpa> # area to block symbols, and a line, respectively:
sherpa> D 1 C 1 NOLINE
sherpa> D 1 C 2 SIMPLELINE
sherpa>
sherpa> # Modify the Y Axis limits of the 2nd drawing area:
sherpa> D 2 LIMITS Y −2.5 2.5
sherpa>
sherpa> # Add a labels to the X and Y Axes:
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> D 1 YLABEL "F(X) = SNR"
sherpa>
sherpa> # Add a title:
sherpa> TITLE "ACIS 25000 Counts Per Chip"
sherpa>
sherpa> # Make all labels and titles the same color (default is white in ChIPS window,
sherpa> # but prints as black):
sherpa> TITLE DEFAULT
sherpa> D 1 YLABEL DEFAULT
sherpa> D 2 YLABEL DEFAULT
sherpa> D 2 XLABEL DEFAULT
sherpa>
sherpa> # Remove the X Axis label from the 1st drawing area:
sherpa> D 1 XLABEL ""
sherpa>
sherpa> # Place a separation between the two drawing areas:
sherpa> SPLIT GAP y 0.04
sherpa>
sherpa> # Remove the X Axis tick marks from the 1st drawing area:
sherpa> D 1 TICKVALS X OFF
sherpa>
sherpa> # Change the format of the tick value labels on the Y Axes:
sherpa> D 1 TICKVALS Y "%1.2f"
sherpa> D 2 TICKVALS Y "%1.2f"
sherpa>
sherpa> # Add a label that contains the fit results:
sherpa> D 1 LABEL 2.0 1.7 "F(X)=(1.4984)+(0.1447)X+(0.0323)X^2+(−0.0028)X^3"
sherpa>
sherpa> REDRAW
```

Note that comments may be entered on the *Sherpa* command line if they are preceded by a pound sign (#). Further information about each of these *ChIPS* commands is available by typing `ahelp <command name>`.

Figure 5, may be saved as a PostScript file:

```
sherpa> PRINT POSTFILE sherpa.basic.5.ps
```

To view estimates of the confidence intervals for the thawed parameters, use the `PROJECTION` command:

```
sherpa> PROJECTION
Projection complete for parameter: model1.c0
Projection complete for parameter: model1.c1
Projection complete for parameter: model1.c2
Projection complete for parameter: model1.c3

Computed for sherpa.proj.sigma = 1
        --------------------------------------------------------
        Parameter Name      Best−Fit Lower Bound     Upper Bound
        --------------------------------------------------------
        model1.c0            1.49843  −0.0518826      +0.0518826
        model1.c1             0.1447  −0.0412216      +0.0412216
        model1.c2          0.0322936  −0.00874421     +0.00874421
        model1.c3        −0.00277729  −0.00051864     +0.00051864
```

For information on the chi−squared goodness−of−fit, use the `GOODNESS` command:

```
sherpa> GOODNESS
Goodness: computed with Chi−Squared Gehrels

DataSet 1: 11 data points -- 7 degrees of freedom.
 Statistic value       = 30.8491
 Probability [Q−value] = 6.62868e-05
 Reduced statistic     = 4.40701
```

Linking Model Parameters

Instead of empirically fitting a polynomial to the data as before, we now wish to fit a first order polynomial such that the offset constant parameter is the following product:

```
  offset = 4.3979 * c1
```

where `c1` is the first order coefficient and `4.3979 = log(25000)`.

First, another `POLYNOM1D` model component is established and is named `model2`:

```
sherpa> PARAMPROMPT OFF
Model parameter prompting is off
sherpa> POLYNOM1D[model2]
```

Since a dataset has been previously input, *Sherpa* estimates the initial parameter values (and the minimum and maximum for their ranges) for this model based on the data. The command `PARAMPROMPT OFF` cancels prompting for changes to these model parameter value estimates.

The `SHOW` command may again be used to examine the details of the established model component:

```
sherpa> SHOW model2
polynom1d[model2]  (integrate: on)
    Param   Type      Value         Min         Max                 Units
    -----   ----      -----         ---         ---                 -----
 1     c0 thawed     2.5475    −1.6454     3.4496
 2     c1 frozen          0    −18.042     18.042
 3     c2 frozen          0    −1.8042     1.8042
 4     c3 frozen          0    −1.6454     3.4496
 5     c4 frozen          0    −1.6454     3.4496
 6     c5 frozen          0    −1.6454     3.4496
 7     c6 frozen          0    −1.6454     3.4496
 8     c7 frozen          0    −1.6454     3.4496
 9     c8 frozen          0    −1.6454     3.4496
10 offset frozen          0       −0.5       10.5
```

As was the case with `model1`, this model component is set to `(integrate: on)` by default. However, no integration will be performed when fitting unbinned data, such as that contained in dataset `data1.dat`. For clarity, integration is also turned off for this model:

```
sherpa> model2 INTEGRATE OFF
```

Next, we change the source model expression to be the model component that we have just established (`model2`):

```
sherpa> SOURCE = model2
```

For a first order polynomial fit, we thaw the `c1` parameter of `model2`:

```
sherpa> THAW model2.c1
```

To set the offset constant parameter to the desired product given above, the `offset` parameter of `model2` is linked to the value of the `c1` parameter:

```
sherpa> model2.offset => (model2.c1)*(4.3979)
```

The source model expression for fitting is:

```
sherpa> SHOW SOURCE
Source 1: model2
poly1d[model2]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     2.5475    −1.6454     3.4496
 2     c1 thawed          0    −18.042     18.042
 3     c2 frozen          0    −1.8042     1.8042
 4     c3 frozen          0    −1.6454     3.4496
 5     c4 frozen          0    −1.6454     3.4496
 6     c5 frozen          0    −1.6454     3.4496
 7     c6 frozen          0    −1.6454     3.4496
 8     c7 frozen          0    −1.6454     3.4496
 9     c8 frozen          0    −1.6454     3.4496
10 offset   link          0     expression: (model2.c1 * 4.3979)
```

The dataset is then fit:

```
sherpa> FIT
 LVMQT: V2.0
 LVMQT: initial statistic value = 2815.14
 LVMQT: final statistic value = 151.827 at iteration 5
        model2.c0  1.75548
        model2.c1  0.198455
```

To plot the fit:

```
sherpa> LPLOT FIT
```

The appearance of the plot may be modified as follows:

```
sherpa> C 2 SIMPLELINE
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

Figure 6 📷 shows the resulting plot.

To compare the fit obtained using `model2` to the fit obtained using `model1`, the command `SHOW MODELS` is issued:

```
sherpa> SHOW MODELS
−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
```

```
Defined source/background model components:
-------------------------------------------

poly1d[model1]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     1.4984    -1.6454     3.4496
 2     c1 thawed     0.1447    -18.042     18.042
 3     c2 thawed 3.2294e-02    -1.8042     1.8042
 4     c3 thawed -2.777e-03    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset frozen          0       -0.5       10.5

poly1d[model2]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     1.7555    -1.6454     3.4496
 2     c1 thawed     0.1985    -18.042     18.042
 3     c2 frozen          0    -1.8042     1.8042
 4     c3 frozen          0    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset   link     0.8728     expression: (model2.c1 * 4.3979)
```

One may also wish to obtain the chi−squared goodness−of−fit for this fit and compare it with that of the previous fit (shown in the Plotting & Examining Fit Results section):

```
sherpa> GOODNESS
Goodness: computed with Chi-Squared Gehrels

DataSet 1: 11 data points -- 9 degrees of freedom.
 Statistic value       = 151.827
 Probability [Q-value] = 3.68791e-28
 Reduced statistic     = 16.8697
```

Independently Fitting a Second Dataset

Finally, we wish to fit a different dataset, again using a first order polynomial.

This dataset and its errors are input into *Sherpa* using the `READ` command:

```
sherpa> READ DATA 2 data2.dat 1 2
sherpa> READ ERRORS 2 data2.dat 1 3
```

Note that by issuing these commands as `READ 2`, the first dataset is ***not*** overwritten. Instead, the new data are input as dataset 2.

Dataset 2 may now be plotted:

```
sherpa> LPLOT DATA 2
sherpa> XLABEL "X = Off-Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

Yet another `POLYNOM1D` model component is established for use, this time named `model3`:

```
sherpa> POLYNOM1D[model3]
```

Since parameter prompting has previously been turned off, the user is not prompted for the initial model parameter values.

Next, we set the source model expression to be `model3` for dataset 2:

```
sherpa> SOURCE 2 = model3
sherpa> model3 INTEGRATE OFF
```

For a first order polynomial fit, we thaw the `c1` parameter of `model3`:

```
sherpa> THAW model3.c1
```

Dataset 2 is then fit using the `SOURCE 2` model expression:

```
sherpa> FIT 2
 LVMQT: V2.0
 LVMQT: initial statistic value = 2534.99
 LVMQT: final statistic value = 38.7541 at iteration 4
        model3.c0  2.30466
        model3.c1  0.177985
```

And this fit is plotted:

```
sherpa> LPLOT FIT 2
sherpa> C 2 SIMPLELINE
sherpa> XLABEL "X = Off−Axis (arcmin)"
sherpa> YLABEL "F(X) = SNR"
sherpa> REDRAW
```

Figure 7 shows the resulting plot.

For information on the chi−square goodness−of−fit, use the `GOODNESS` command:

```
sherpa> GOODNESS 2
Goodness: computed with Chi−Squared Gehrels

DataSet 2: 11 data points -- 9 degrees of freedom.
 Statistic value       = 38.7541
 Probability [Q−value] = 1.27581e−05
 Reduced statistic     = 4.30601
```

Note that fitting this second dataset with the `model3` polynomial did not affect the previous fit of the first dataset with the `model2` polynomial.

Checking Sherpa Session Status

The final overall status of this *Sherpa* session may be viewed as follows:

```
sherpa> SHOW

Optimization Method: Levenberg−Marquardt
Statistic:           Chi−Squared Gehrels

-----------------
Input data files:
-----------------

Data 1: data1.dat ascii 1 2.
Total Size: 11 bins (or pixels)
```

```
Dimensions: 1
Total counts (or values): 29.411500

Current errors for dataset 1:
READ ERRORS data1.dat 1 3
Data 2: data2.dat ascii 1 2.
Total Size: 11 bins (or pixels)
Dimensions: 1
Total counts (or values): 36.119300

Current errors for dataset 2:
READ ERRORS 2 data2.dat 1 3

------------------------------
Defined analysis model stacks:
------------------------------

source 1 = model2
source 2 = model3

-------------------------------------------
Defined source/background model components:
-------------------------------------------

poly1d[model1]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     1.4984    -1.6454     3.4496
 2     c1 thawed     0.1447    -18.042     18.042
 3     c2 thawed 3.2294e-02    -1.8042     1.8042
 4     c3 thawed -2.777e-03    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset frozen          0       -0.5       10.5

poly1d[model2]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     1.7555    -1.6454     3.4496
 2     c1 thawed     0.1985    -18.042     18.042
 3     c2 frozen          0    -1.8042     1.8042
 4     c3 frozen          0    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset   link     0.8728     expression: (model2.c1 * 4.3979)

poly1d[model3]  (integrate: off)
    Param   Type      Value        Min        Max                 Units
    -----   ----      -----        ---        ---                 -----
 1     c0 thawed     2.3047    -1.6454     3.4496
 2     c1 thawed      0.178    -18.042     18.042
 3     c2 frozen          0    -1.8042     1.8042
 4     c3 frozen          0    -1.6454     3.4496
 5     c4 frozen          0    -1.6454     3.4496
 6     c5 frozen          0    -1.6454     3.4496
 7     c6 frozen          0    -1.6454     3.4496
 8     c7 frozen          0    -1.6454     3.4496
 9     c8 frozen          0    -1.6454     3.4496
10 offset frozen          0       -0.5       10.5
```

Exiting Sherpa

To exit the current *Sherpa* session:

```
sherpa> BYE
Goodbye.
```

History

14 Jan 2005	reviewed for CIAO 3.2: no changes
21 Dec 2005	reviewed for CIAO 3.3: no changes
01 Dec 2006	reviewed for CIAO 3.4: no changes

URL: http://cxc.harvard.edu/sherpa/threads/basic/ Last modified: 1 Dec 2006

Image 1: Sherpa LPLOT command after manipulation by ChIPS

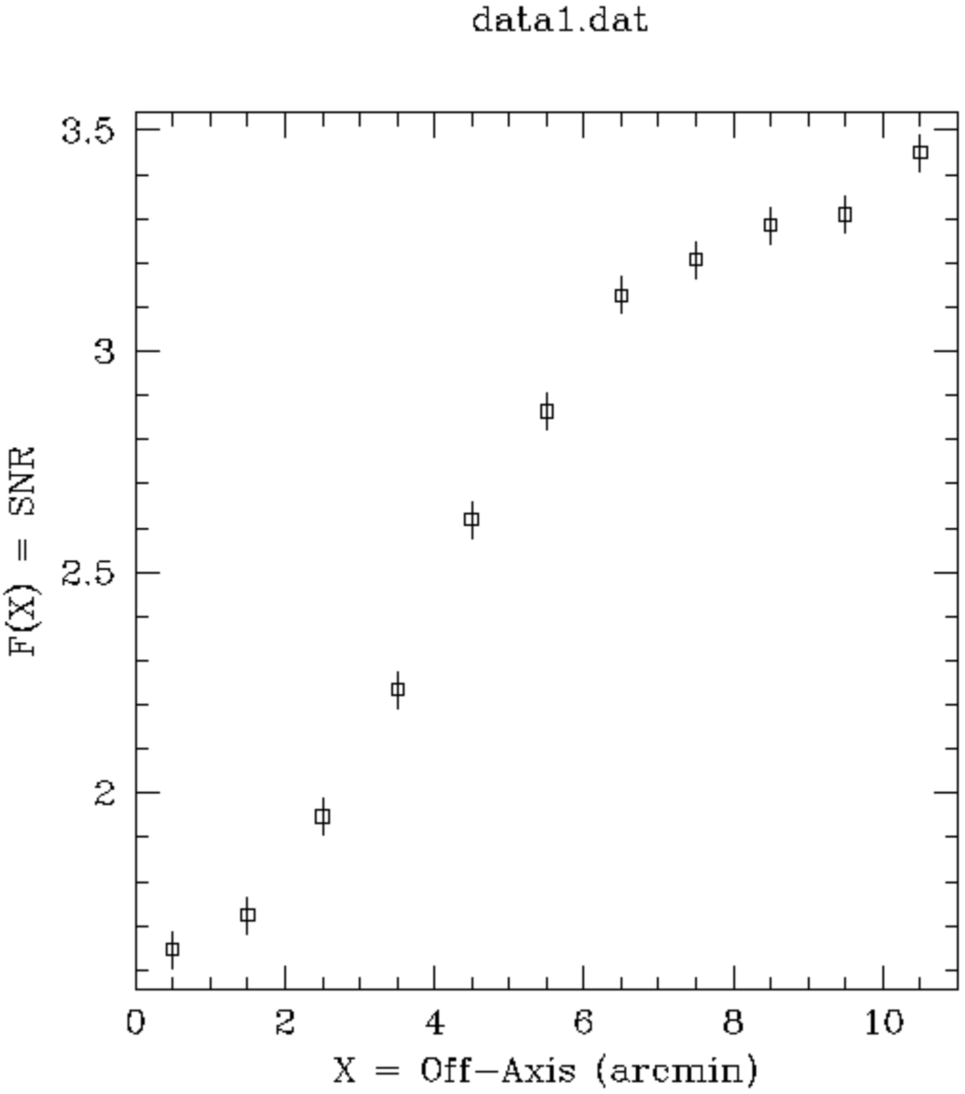

Image 2: 1st order polynomial fit to the data

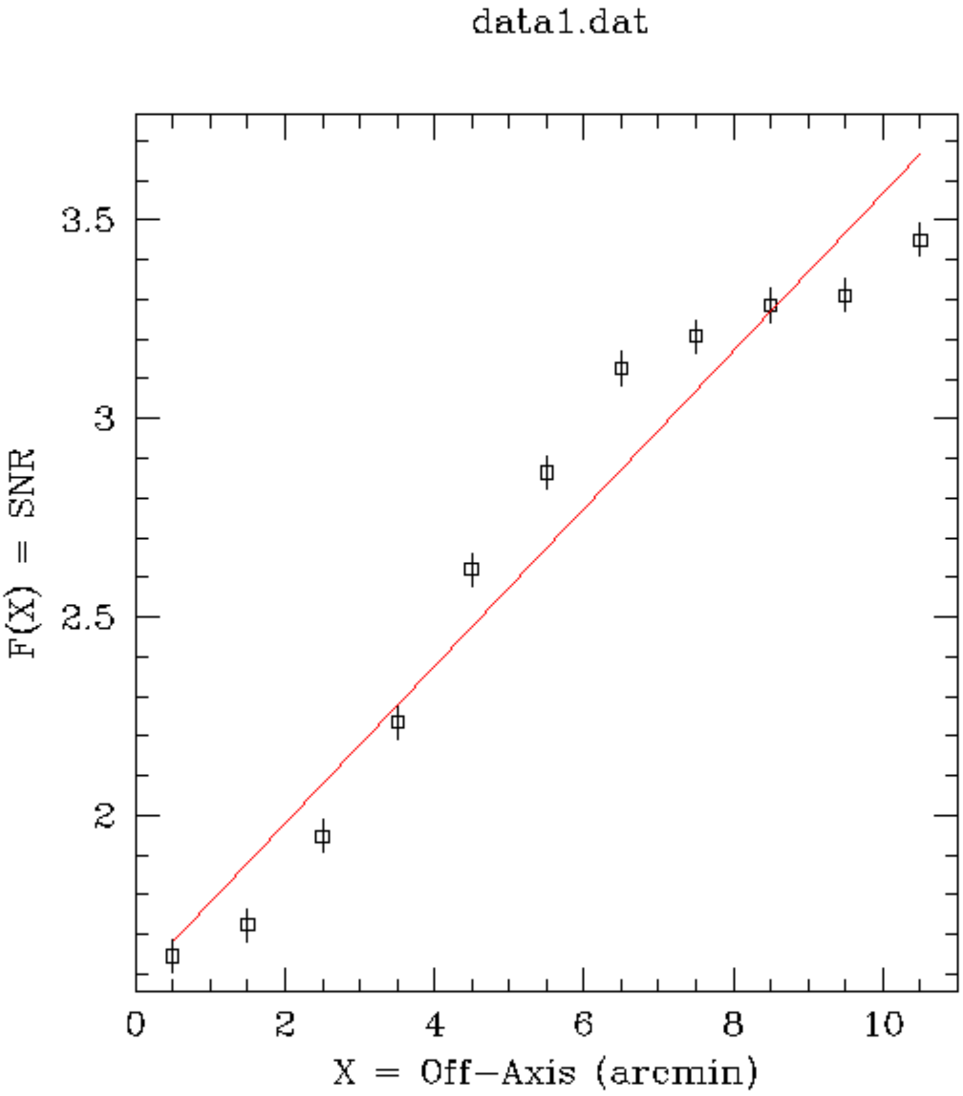

Image 3: 2nd order polynomial fit to the data

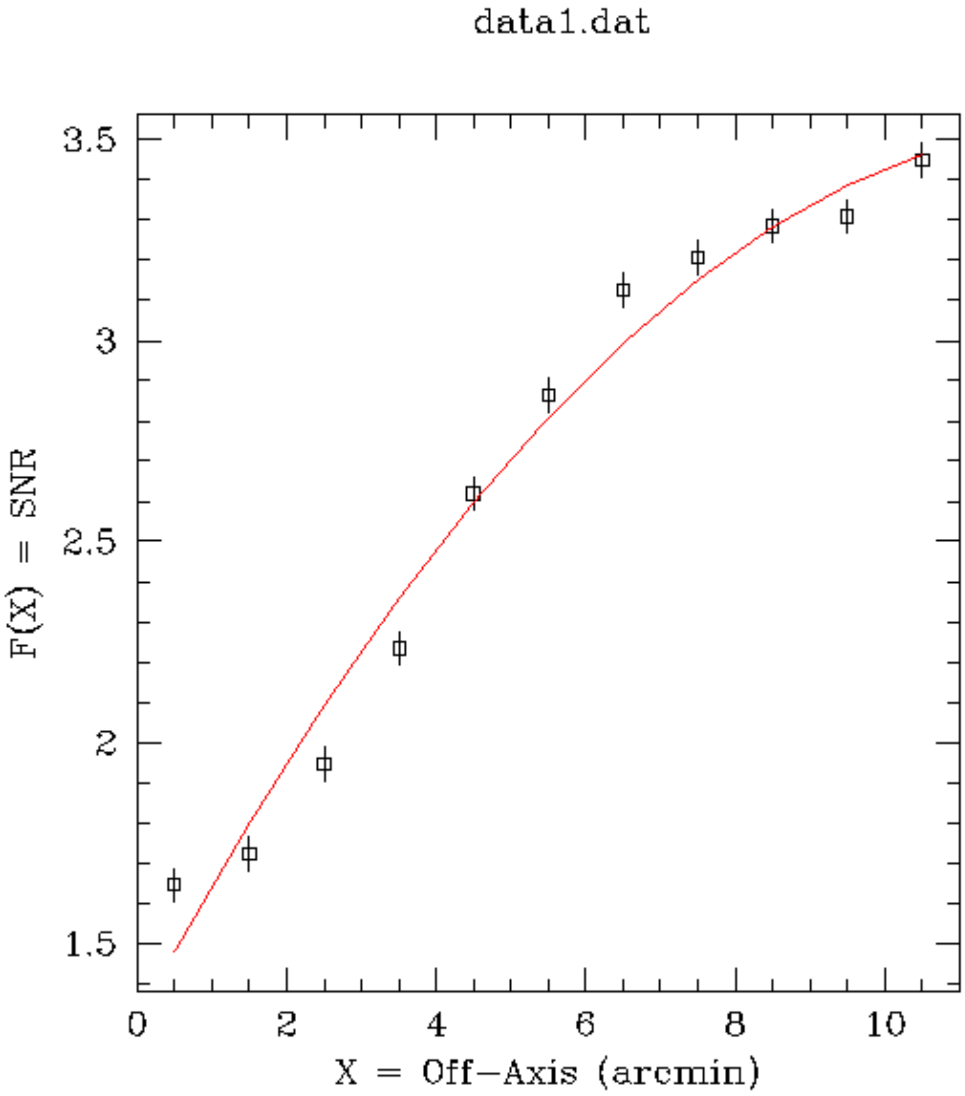

Image 4: 3rd order polynomial fit to the data

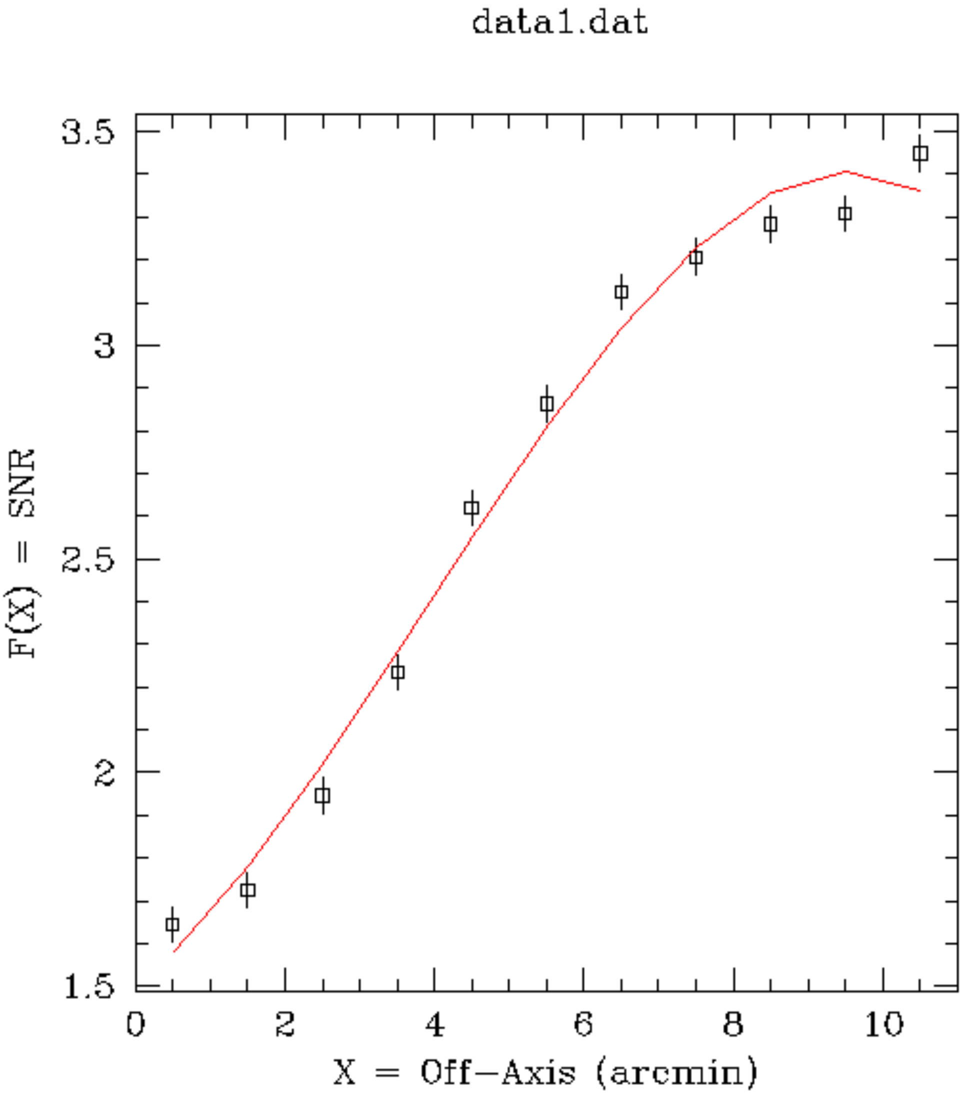

Image 5: Plotting the fit and residuals

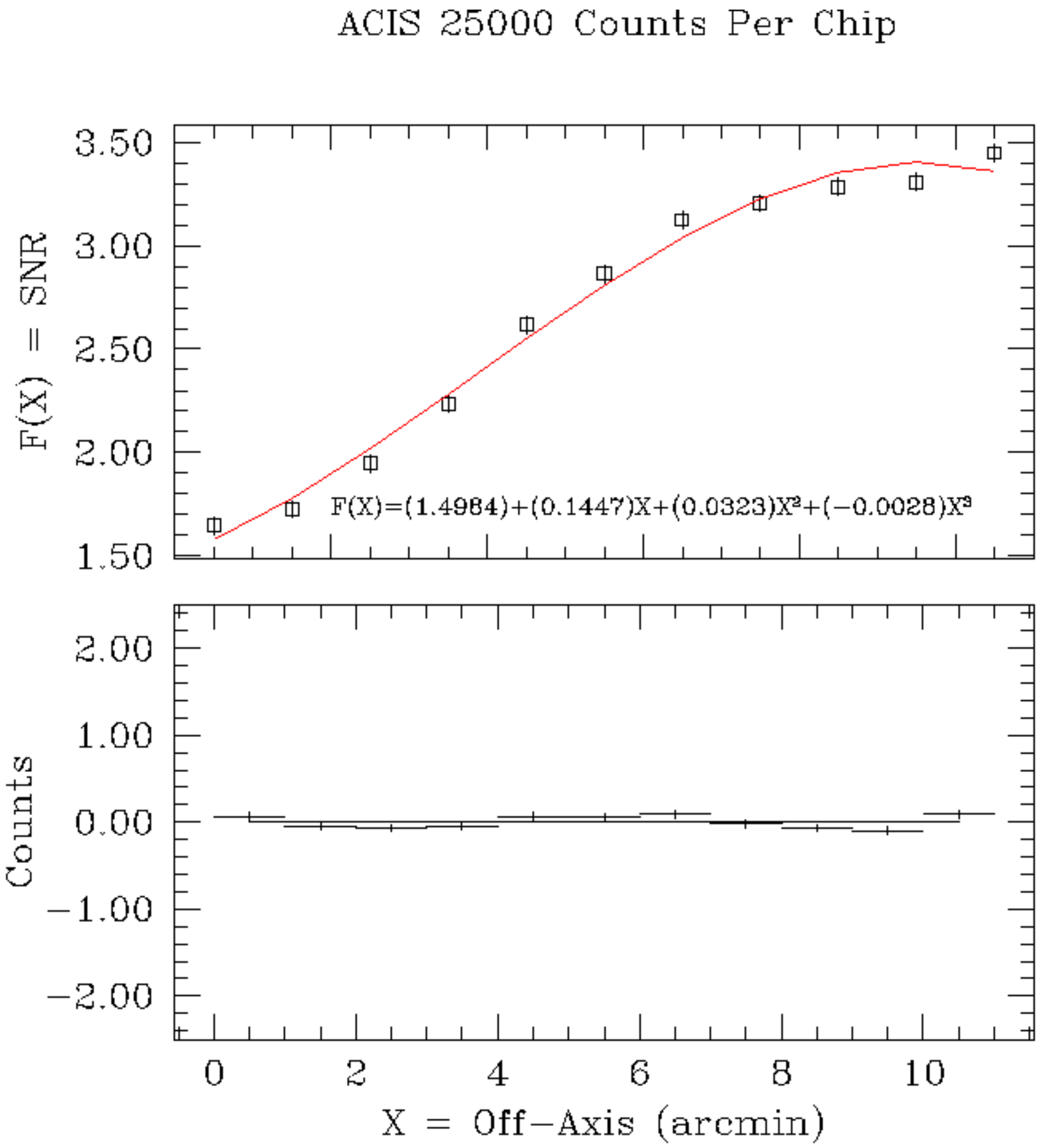

Image 6: Fitting using linked model parameters

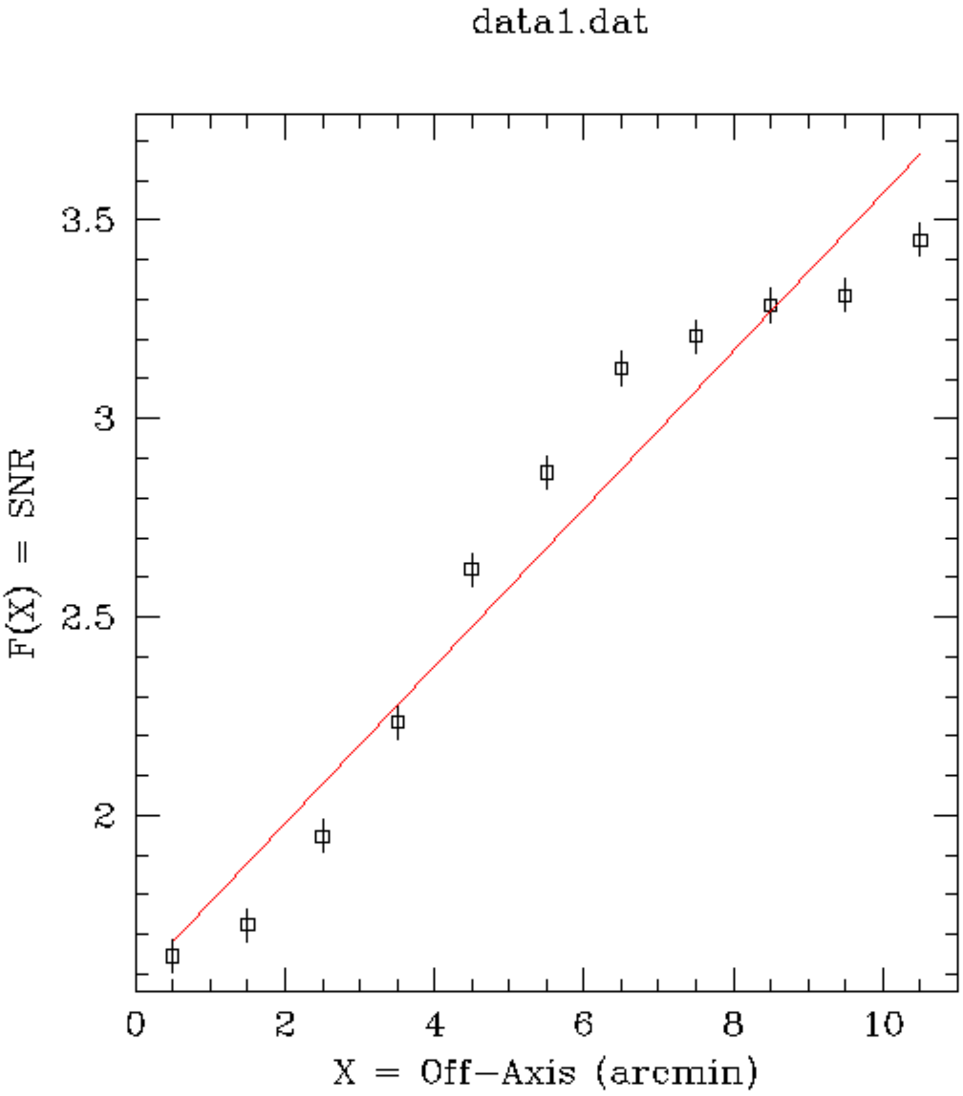

Image 7: Fitting a second dataset

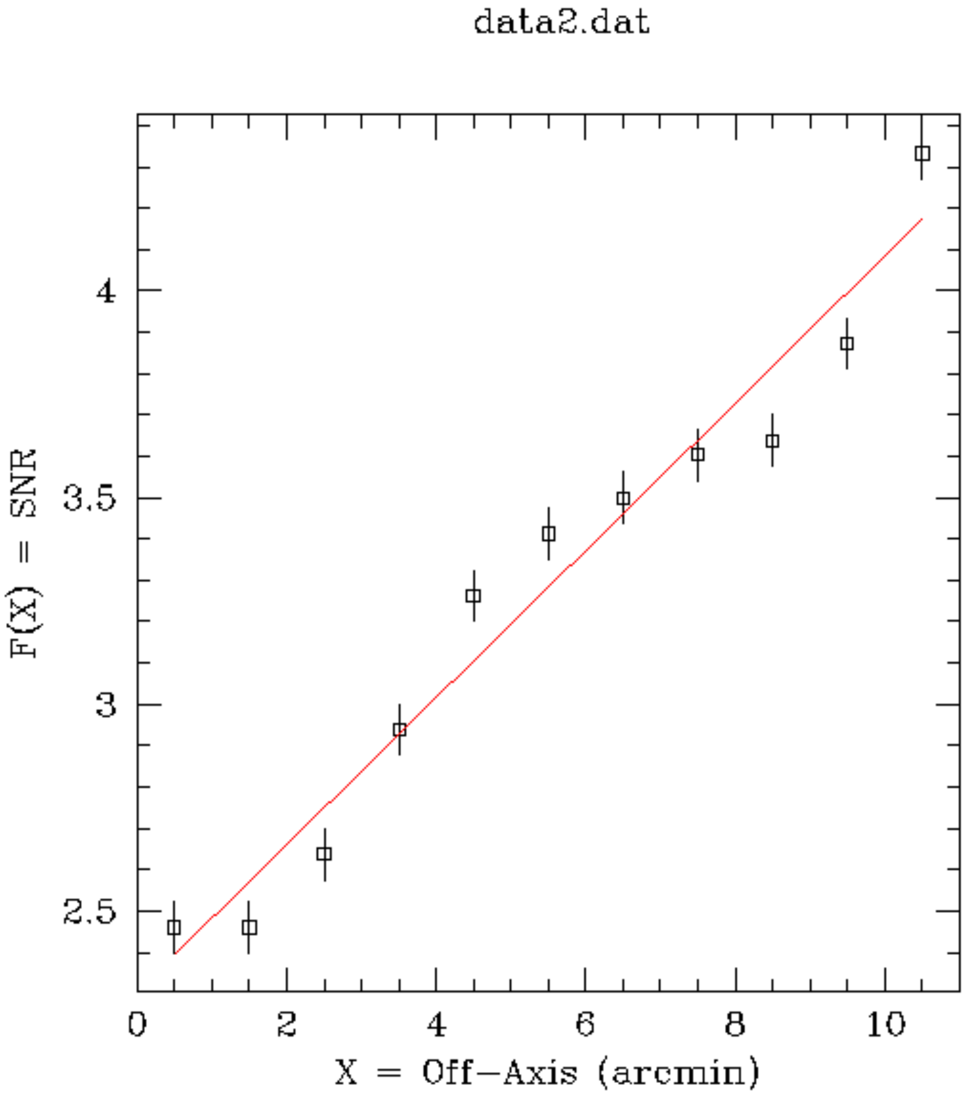

Image 7: Fitting a second dataset