A Typical Sherpa Session
(The boiled-down version.)

The user:

• reads in data (and sets filters, etc.);
• builds model expressions;
• chooses a statistic;
• fits the model expressions to the data, one at a time;
• compares the results of the fits in order to select a best-fit model;
  and
• estimates the errors for the best-fit model parameters.
Choosing a Statistic

(So many choices, so little guidance.)

A key feature of Sherpa is its large array of statistics appropriate for analyzing Poisson-distributed (i.e. counts) data.

- Statistics based on $\chi^2$:
  - CHI GEHRELS
  - CHI DVAR
  - CHI MVAR
  - CHI PARENT
  - CHI PRIMINI

- Statistics based on the Poisson likelihood $L$:
  - CASH
  - BAYES

If the data are not Poisson-distributed (e.g. fluxes), then alternatives include:

- least-squares fitting: setting all variances to one; or

- providing errors in an input file.
**χ²-Based Statistics**

The $\chi^2$ statistic is

$$\chi^2 \equiv \sum_i \frac{(D_i - M_i)^2}{\sigma_i^2},$$

where

- $D_i$ represents the observed datum in bin $i$;
- $M_i$ represents the predicted model counts in bin $i$; and
- $\sigma_i^2$ represents the variance of the sampling distribution for $D_i$.

<table>
<thead>
<tr>
<th>$\chi^2$ Statistic</th>
<th>$\sigma_i^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEHRELS</td>
<td>$[1 + \sqrt{D_i + 0.75}]^2$</td>
</tr>
<tr>
<td>DVAR</td>
<td>$D_i$</td>
</tr>
<tr>
<td>MVAR</td>
<td>$M_i$</td>
</tr>
<tr>
<td>PARENT</td>
<td>$\frac{\sum_{i=1}^{N} D_i}{N}$</td>
</tr>
<tr>
<td>PRIMINI</td>
<td>$M_i$ from previous best-fit</td>
</tr>
</tbody>
</table>
Likelihood-Based Statistics

The **CASH** statistic is

\[
C \equiv 2 \sum_i [M_i - D_i \log M_i] \propto -2 \log \mathcal{L},
\]

where

- \( D_i \) represents the observed datum in bin \( i \);
- \( M_i \) represents the predicted model counts in bin \( i \); and
- \( \mathcal{L} = \prod_i \frac{M_i^{P_i}}{D_i!} \exp(-M_i) \).
Statistics: Caveats

(Potholes on the road to publication.)

Things to remember when using $\chi^2$:

- $\chi^2$ is an approximation of $\log \mathcal{L}$ in the Gaussian (high-counts) limit. So...

- All estimations of variance (except GEHRELS) assume a Gaussian sampling distribution, not Poisson. Hence the number of counts in each bin should be $\geq 5$.

- CHI GEHRELS works with low-count data, but does not generally follow the $\chi^2$ distribution: best fits are often “too good.”

- And $\chi^2$ is a biased estimator.

Things to remember when using CASH or BAYES:

- In the limit of high counts, $\Delta C \sim \Delta \chi^2$.

- Likelihood estimators are unbiased. But...

- Background subtraction is not allowed.

- There is no “goodness-of-fit” measure.

- And negative model amplitudes are not allowed.
Optimization in Sherpa

Optimization is the action of minimizing $\chi^2$ or $-\log\mathcal{L}$ by varying the thawed parameters of the model. The user may choose between several optimization methods in Sherpa, including ones which:

- Find the local minimum.
  - POWELL
  - SIMPLEX
  - LEVENBERG–MARQUARDT

These algorithms are not computationally expensive, but they are also not appropriate for finding the global minimum of a complex statistical surface when starting from a random point.

- Attempt to find the global minimum.
  - GRID and GRID–POWELL
  - MONTE and MONTE–POWELL
  - SIMULATED ANNEALING

These are computationally intensive algorithms which are useful for searching complex statistical surfaces, starting from a random point.
Main SHERPA Components

- Data Input/Output.
- Visualization through ChIPS and ds9
- Model library and model language.
- Statistics and Error Analysis.
- Optimization Methods.
MODELS

• Three main type of models:
  
  Source
  Background
  Instrument

• Model library consists of several models (plus XSPEC v.11) which can be used to define a source or background model.

• There are three type of instrument models:
  
  RSP
  PsfFromTCD
  PsfFromFile

• Instrument models are convolved with Source and Background models before the model predicted data is compared with the observed data.

• Instrument and Background models are NOT required. Source models have to be defined for fitting.
Data Entry

• During a *Sherpa* session, the user may read in files containing:
  – source data
  – background data
  – errors on the source and/or background data
  – filters
  – statistical weights

These data may be integer, float, or double precision and may be of arbitrary dimensionality.

• Currently supported file types include:
  – ASCII
  – FITS binary table
  – FITS image
  – PHA types I & II
  – IRAF imh
  – ROSAT qpoe

For all types except ASCII, data entry is accomplished through the Data Model interface, so that, *e.g.*, one can use the Data Model filtering syntax within *Sherpa* data entry commands.
Instrument Models

- Instrument models describe the mapping from photon space (where source and background models are evaluated) to counts space (where fit statistics are computed) for a particular detector.

⇒ *The instrument model class, by hiding detector-dependent details, allows Sherpa to be a mission-independent fitting application.*

- Currently *Sherpa* offers three instrument model types:
  - **RSP**, in which the evaluated one-dimensional model is multiplied by an ancillary response (*i.e.* effective area) and then folded through a response matrix;
  - **PSFFromTCD**, in which the evaluated one- or two-dimensional model is convolved with an analytic kernel (*e.g.* Gaussian) defined in *CIAO’s TCD* library;
  - and **PSFFromFile**, in which the evaluated two-dimensional model is convolved with a numeric kernel.
Building Model Expressions

• In *Sherpa*, one can build model expressions that represent the
  – source
  – background
  – instrument

for each dataset.

⇒ Note, however, background and instrument model expressions are *not* required to carry out fits.

• Currently, nearly 40 one- and two-dimensional *Sherpa* models and 90 one-dimensional *XSPEC v. 11* models are available for building model expressions.
Building Model Expressions

- Model parameters can be *linked* to other parameters.
- Example:
  - A particular atomic line is observed by detectors with different resolutions. One can model this line with two Gaussian functions whose centroids (but not amplitudes or widths) are linked:
    
    ```
    sherpa> source 1 = gauss1d[g1]
    sherpa> source 2 = gauss1d[g2]
    sherpa> g1.pos => g2.pos
    ```
  - Model parameters can also be linked to other models.
  - Example:
    - One can model emission from an accretion disk using a blackbody function whose temperature is a function of radius:
      
      ```
      sherpa> Temperature = POLY
      sherpa> BB.kT => Temperature
      ```
Building Model Expressions

• A model can be *nested* within another model.

• Example:
  – Transform an input dataspace to log-space, and evaluate a blackbody in that space:

        sherpa> logenergy = shlog
        sherpa> source = bb{logenergy}

• Different models can be defined along each axis of a multi-dimensional dataset.

• Example:
  – Model two-dimensional data that have spectral information along one axis and spatial information (*e.g.* radius) along the other:

        sherpa> data image.fits
        sherpa> lorentz[ Spatial ]
        sherpa> pow[ Spec  ]
        sherpa> source = Spatial{x1} * Spec{x2}