URL: http://cxc.harvard.edu/ciao3.4/plot_rprof.html
Last modified: December 2006

AHELP for CIAO 3.4

plot_rprof

Context: sherpa

Synopsis

Display a radial profile (circular annuli) of 2D data, fit, and residuals.

Syntax

```
plot_rprofr( [dnum,] min, max, step )
plot_rprofr( [dnum,] modelname, min, max, step )
plot_rprofr( [dnum,] xval, yval, min, max, step )
plot_rprofr( [dnum,] xname, yname, min, max, step )

plot_rprofd( [dnum,] min, max, step )
plot_rprofd( [dnum,] modelname, min, max, step )
plot_rprofd( [dnum,] xval, yval, min, max, step )
plot_rprofd( [dnum,] xname, yname, min, max, step )

plot_rprof() is a synonym for plot_rprofr()
```

Description

The 'image fit' command shows the data, best−fit model, and residuals for your 2D dataset. Sometimes this can be hard to interpret − in particular when dealing with low−count data as is often the case with Chandra images − and a different view of the fit may be useful. The plot_rprofr() and plot_rprofd() functions provide such a view, as they display two plots; the first is a radial profile of the data and best−fit model, while the second is a radial profile of the residual image. The residual plot created by plot_rprofr() has the y axis in units of counts, whilst the residual plot from plot_rprofd() uses counts / error.

These plots are similar to the output of 'lplot fit resid' − for plot_rprofr() − and 'lplot fit delchi' − for plot_rprofd() − for 1D data.

The functions provide a usage message if called with either no, or an incorrect set of, arguments. The plot_eprofr() and plot_eprofd() functions produce the same plots but using elliptical, rather than circular, annuli: see "ahelp plot_eprof" for more information.

How are the annuli chosen?

The min, max, and step arguments determine the the minimum and maximum radii of the annuli, and the width of each annnulus repectively. The units of these arguments match the current setting of Sherpa's coordinate command. If no other arguments − other than the dataset number (dnum) − are given, then the

center of the annuli is taken from the xpos and ypos parameters of a source component for the dataset. If no such component exists, or, more likely, there is more than one component in the source expression, then you have to specify the center of the annuli. This can be done in one of three ways:

- The name of the component to use (the modelname parameter). This model should have xpos and ypos parameters.
- The actual coordinates of the center (the xval and yval parameters) in the same coordinate system as min, max, and step.
- The names of model parameters (the xname and yname arguments) which contain the center position.

The following examples assume that a source model has been fit to a 2D dataset. See the "NOTES" section below for information on how to load the function into Sherpa, and the "CHANGING THE PLOTS" section for information on how to customise the appearance of the plots.

Example 1

```
sherpa> plot_rprofr(0,200,20)
```

This will plot up a radial profile of the data and model (top plot) and a radial profile of the residual image (bottom plot) for dataset 1. Assuming the values in sherpa.fitplot and sherpa.resplot have not been changed then the data is plotted using squares, the best−fit as the red line, and the residuals as filled circles.

The three arguments in the call give the minimum and maximum radii − here 0 and 200 respectively − and the width of each annulus (20). The units are taken to match the current coordinate settings for the dataset (see 'ahelp coord'). The source expression is assumed to contain only one component with "xpos" and "ypos" parameters; these are used to define the center of the profile.

A radial bin will only be displayed if there are any pixels that contribute to that bin. For instance, if the central region around the source were excluded from the fit out to a distance of 30, then the plots created by plot_rprof(0,200,20) would start at the second bin (i.e. for radii of 20−40).

Example 2

```
sherpa> plot_rprofd(0,200,20)
```

This creates the same plot as in the previous example except that the residual plot has units of "sigma" − i.e. counts divided by the error estimate − rather than counts.

Example 3

```
sherpa> plot_rprofr(2,0,200,20)
```

This creates the radial profiles for dataset 2.

Example 4

```
sherpa> plot_rprof("core",0,200,20)
```

Here we explicitly list which model should be used to define the center of the profile; we assume in this example that "core" is a model component such as BETA2D (i.e. that it has xpos and ypos parameters).

This is only necessary when one of the following holds:

- The source expression contains more than one component with "xpos" and "ypos" parameters, so you have to specify which component to use.
- There is no model component with both "xpos" and "ypos" parameters so you have to name which parameters define the center of the profile.
- You wish to use a center which does not match any of the model components.

CHANGES

Version 1.27

Version 1.27 of sherpa_utils.sl was released at the same time as CIAO 3.1. The major changes to the previously−released version (1.19) are:

- The plot_rprofr() and plot_rprofd() functions have been added to plot the residuals in units of counts or sigma (analogous to the "resid" and "delchi" options of the lplot command). The plot_rprof() command is kept for backwards compatability, and is the same as plot_rprofr().
- The plot_eprofr() and plot_eprofd() functions have been added to calculate the profiles within elliptical annuli.
- The residuals are no longer noramlized by the area of each annulus.
- Error bars are now displayed (where the errors are estimated using the approximations given in Gehrels, N., 1986, Ap.J., 303, 336).
- You can give just the model name, rather than the names of the "xpos" and "ypos" parameters, to distinguish which model to use to calculate the center of the profile.
- The residual plots are now smaller than the main plot; previously they were the same size.
- The residual plots now contain a line at "y=0".

NOTES

This script is not an official part of the CIAO release but is made available as "contributed" software via the CIAO scripts page. Please see the installation instructions page for help on installing the package.

Loading the functions

These functions are not an official part of the CIAO release but are made available as part of the sherpa_plotfns.sl script from the CIAO scripts page. Since the functions print a usage message if called with no parameters, the easiest way to see if it has been loaded into Sherpa is to try:

```
sherpa> plot_rprofr()
```

If the message

```
Parse Error/Undefined Name: plot_rprofr();
```

is seen then you need to load the sherpa_plotfns.sl package into Sherpa. See the "NOTES" section of the sherpa_plotfns ahelp file − "ahelp −b ADESC −t NOTES sherpa_plotfns" − for more information.

CHANGING THE PLOTS

The plots produced by the plot_rprof() function can be configured using the sherpa.fitplot and sherpa.resplot configuration variables (see 'ahelp sherpa.fitplot' and 'ahelp sherpa.resplot' for more information). This means that you can use the set_log(), set_lin(), and related functions ('ahelp set_log', 'ahelp set_lin') to control the scales (linear or log) of the plots.

The main items that are ignored from the sherpa.fitplot and sherpa.resplot variables are:

- The y−axis of the residual plot is always linear (sherpa.resplot.y_log is ignored)
- No error bars are drawn (the "error_bars" and "errs" variables are ignored).
- The prefunc and postfunc hooks are ignored.

The table below lists those fields that are used (the leading 'sherpa.' has been dropped from all the variable names):

Variable	Meaning
fitplot.symbolstyle	symbol used for the data
fitplot.symbolcolor	symbol color for the data
fitplot.symbolsize	symbol color for the data
fitplot.curvestyle	curve style for the data
fitplot.curvecolor	curve style for the data
fitplot.fit_symbolstyle	symbol used for the model
fitplot.fit_symbolcolor	symbol color for the model
fitplot.fit_symbolsize	symbol color for the model
fitplot.fit_curvestyle	curve style for the model
fitplot.fit_curvecolor	curve style for the model
resplot.symbolstyle	symbol used for the residuals
resplot.symbolcolor	symbol color for the residuals
resplot.symbolsize	symbol color for the residuals
resplot.curvestyle	curve style for the residuals
resplot.curvecolor	curve style for the residuals
fitplot.title_size	The size of the title
fitplot.ylabel_size	The size of the y−axis label for the data/fit plot
fitplot.tickvals_size	The size of the axis numbers for the data/fit plot
resplot.xlabel_size	The size of the x−axis label for the residual plot
resplot.ylabel_size	The size of the y−axis label for the residual plot
resplot.tickvals_size	The size of the axis numbers for the residual plot
fitplot.x_log	Should the x−axis be logarithmic for the data/fit plot
fitplot.y_log	Should the y−axis be logarithmic for the data/fit plot
resplot.x_log	Should the x−axis be logarithmic for the residual plot

CALCULATING THE RADIAL PROFILE

The intent of the functions (both the circlular and elliptical annuli versions) is to provide a quick comparison of the model to the data. The radial profiles are calculated by looping through each radial bin and finding all those pixels which lie within the limits of the bin: for simplicity, only those pixels whose centers lie within the

radial limits of the bin are used. The error on each bin is then calculated using the 1−sigma upper−limit approximation from Gehrels, N., 1986 (Ap.J., 303, 336), namely:

```
error = ( 1 + sqrt(N + 0.75) )
```

where N is the sum of the pixel values within the bin. This assumes that the pixel values are Poisson distributed.

The same routine is used to calculate the profiles of the model and residual data (without calculating any errors). When displaying the profiles, the data and model curves are normalised by the number of pixel in each bin, while the residuals are plotted with no normalisation (prior to version 1.27 they were also normalised by the bin area).

See Also

chandra

guide

sherpa

bye, calc_kcorr, dataspace, dcounts, dollarsign, echo, eflux, eqwidth, erase, flux, get, get_dcounts_sum, get_dir, get_eflux, get_eqwidth, get_filename, get_flux2d, get_flux_str, get_lfactorial, get_mcounts_sum, get_pflux, get_source_components, get_verbose, groupbycounts, guess, is, journal, list, list_par, mcounts, numbersign, paramest, plot_eprof, prompt, reset, run, set, set_analysis, set_axes, set_coord, set_dataspace, set_dir, set_verbose, setplot, sherpa−module, sherpa_plotfns, sherpa_utils, show, simspec, use, version

---

URL: http://cxc.harvard.edu/ciao3.4/plot_rprof.html
Last modified: December 2006