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

AHELP for CIAO 3.4

run_paramestint

Context: sherpa

Jump to: Description Example CHANGES IN CIAO 3.2 CHANGES IN CIAO 3.1 Bugs See Also

Synopsis

Module functions to display statistics as a function of parameter value, and to retrieve the value and statistic arrays

Syntax

```
Struct_Type run_intunc(String_Type)
Struct_Type run_intproj(String_Type)

Argument:

(1) A string representing an individual thawed model parameter
```

Description

These functions initiate the INTERVAL−UNCERTAINTY and INTERVAL−PROJECTION parameter estimation methods respectively. The chosen method is run using the most recently fit datasets, which are automatically determined and hence are not function arguments. When done, each returns a structure, which are the same as those returned by get_intunc and get_intproj.

These functions can be used to retrieve information similar to that provided by the XSPEC command steppar.

Example

Fit a dataset; get information about chi−square as a function of power−law amplitude p.ampl

```
sherpa> () = load_dataset(1,"example.pha")
sherpa> () = set_subtract
sherpa> () = set_source_expr(1,"POW[p]")
sherpa> set_verbose(0)
sherpa> () = run_fit
sherpa> list_intproj()
Parameter   Current         Default                       Description
-----------------------------------------------------------------------
```

```
fast              1               1    Switch to LM/simplex: 0(n)/1(y)
expfac            3               3          Expansion factor for grid
arange            1               1              Auto−range: 0(n)/1(y)
min               0               0                      Minimum value
max               0               0                      Maximum value
log               0               0             Log−spacing: 0(n)/1(y)
nloop            20              20              Number of grid points
sigma             1               1                    Number of sigma
sherpa> sherpa.intproj.sigma = 3
sherpa> sherpa.intproj.nloop = 100
sherpa> intproj = run_intproj("p.ampl")
[...plot displayed...]
sherpa> print(intproj)
x0               =  Float_Type[100]
y                =  Float_Type[100]
name             =  p.ampl
bfit             =  0.000191983
config           =  sherpa_VisParEst_State
sherpa> printarr(intproj.x0,3)
9.19654e−05
9.39859e−05
9.60065e−05
sherpa> printarr(intproj.y,3)
270.566
267.213
263.936
```

The second−to−last call displays the first three values of the p.ampl grid, while the last call displays the best−fit statistic given those p.ampl values.

CHANGES IN CIAO 3.2

The run_intunc() and run_intproj() commands no longer fail with an error message when called. This means that you can use

```
  retval = run_intproj(parameter_name);
```

rather than having to use sherpa_eval() to call INTERVAL−PROJECTION (or INTERVAL−UNCERTAINTY) and then get_regproj() to access the results.

CHANGES IN CIAO 3.1

The structures returned by these functions contain additional fields: name, bfit, and config. These fields contain information on the name of the parameter, its best−fit value, and the values used by the "interval" command to calculate the x0 and y values.

Bugs

Functions require that FIT has been called

These functions will only run after the dataset has been fitted; i.e run_fit() called in the same session. This is unlike the Sherpa versions of these commands, which have been updated in CIAO 3.2 to not require the inital fit.

See the Sherpa bug pages online for an up−to−date listing of known bugs.

See Also

sherpa

berrors, bsyserrors, compute_errors, compute_statistic, covariance, errors, ftest, get_paramest, get_paramestint, get_paramestlim, get_paramestreg, goodness, interval−projection, interval−uncertainty, list_paramest, mlr, projection, region−projection, region−uncertainty, restore_paramest, run_paramest, run_paramestlim, run_paramestreg, set_errors, set_syserrors, staterrors, syserrors, uncertainty

---

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