



AHELP for CIAO 3.4

run_paramestreg

Context: [sherpa](#)

Jump to: [Description](#) [Example](#) [CHANGES IN CIAO 3.2](#) [CHANGES IN CIAO 3.1](#) [Bugs](#) [See Also](#)

Synopsis

Module functions to display contours of statistics as a function of parameter values, and to retrieve arrays of values and statistics

Syntax

```
Struct_Type run_regunc(Array_Type)
Struct_Type run_regproj(Array_Type)
```

Argument:

(1) An array of two strings representing two thawed model parameters.

Description

These functions initiate the REGION-UNCERTAINTY and REGION-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_regunc` and `get_regproj`.

Example

Fit a dataset; get information about chi-square as a function of power-law amplitude `p.ampl` and slope `p.gamma`

```
sherpa> () = load_ascii(1,"spec.dat")
sherpa> () = create_model("POLY","p")
sherpa> set_thawed(["p.c1","p.c2","p.c3"])
sherpa> () = set_source_expr(1,"p")
sherpa> () = run_fit
LVMQT: V2.0
LVMQT: initial statistic value = 82.2297
LVMQT: final statistic value = 62.2247 at iteration 3
      p.c0  61.4774
      p.c1  -0.380228
      p.c2  0.00993229
      p.c3  -7.01741e-05

sherpa> list_regproj()
```

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]	[0,0]	Minimum values, each axis
max	[0,0]	[0,0]	Maximum values, each axis
log	[0,0]	[0,0]	Log-spacing: 0(n)/1(y), each axis
nloop	[10,10]	[10,10]	Number of grid points, each axis
sigma	[1,2,3]	[1,2,3]	Number of sigma, each contour

```

sherpa> sherpa.regproj.sigma = [1.6,2.6]
sherpa> regproj = run_regproj(["p.c0","p.c1"])
[...plot displayed...]
sherpa> print(regproj)
x0          = Float_Type[100]
x1          = Float_Type[100]
y           = Float_Type[100]
levels      = Float_Type[2]
name        = String_Type[2]
bfit        = Double_Type[2]
config      = sherpa_VisParEst_State
sherpa> printarr(regproj.x0,3)
61.3661
61.3661
61.3661
sherpa> printarr(regproj.x1,3)
-1.79518
-1.48075
-1.16631
sherpa> printarr(regproj.y,3)
154.651
118.31
90.9931
    
```

The third-to-last and second-to-last function calls cause the first three values of the p.c0 and p.c1 grid axes to be displayed (with p.c0 repeated because a 2-D grid is being computed). The last function call displays the best-fit statistic given the p.c0 and p.c1 values.

CHANGES IN CIAO 3.2

The run_regunc() and run_regproj() commands no longer place two arrays on the stack. This means that you can use

```
retval = run_regproj(parameter_names);
```

rather than having to say something like

```
(, ,retval) = run_regproj(parameter_names);
```

CHANGES IN CIAO 3.1

The structures returned by these functions contain additional fields: levels, name, bfit, and config. These fields contain information on the statistic value for each contour, the name of the parameters, their best-fit values, and the values used by the "region" command to calculate the x0, x1, 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 initial

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](#), [fctest](#), [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_paramestint](#), [run_paramestlim](#), [set_errors](#), [set_syserrors](#), [staterrors](#), [syserrors](#), [uncertainty](#)

The Chandra X-Ray Center (CXC) is operated for NASA by the
Smithsonian Astrophysical Observatory.
60 Garden Street, Cambridge, MA 02138 USA.
Smithsonian Institution, Copyright © 1998–2006. All rights reserved.

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

