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

AHELP for CIAO 3.4

covariance

Context: sherpa

Jump to: Description Examples CHANGES IN CIAO 3.2 Bugs See Also

Synopsis

Computes covariance matrices, and provides an estimate of confidence intervals for selected thawed parameters.

Syntax

```
sherpa> COVARIANCE [<dataset_range> | ALLSETS] [ <arg_1> , ... ]

where <dataset range> = #, or more generally #:#,#:#,..., such that #
specifies a dataset number, and #:# represents an inclusive range of
datasets; one may specify multiple inclusive ranges by separating them
with commas. The default is to estimate limits using data from all
appropriate datasets.
```

Description

The command−line arguments <arg_n> may be:

COVARIANCE Command Arguments

Argument	Description
<sherpa_modelname>.{<paramname> | <#>}	A specified model component parameter (e.g., GAUSS.pos).
<modelname>.{<paramname> | <#>}	A specified model component parameter (e.g., g.pos).

The user may configure COVARIANCE via the Sherpa state object structure cov. The current values of the fields of this structure may be displayed using the command print(sherpa.cov), or using the more verbose Sherpa/S−Lang module function list_cov().

The structure field is:

cov Structure Field

Field	Description
sigma	Specifies the number of sigma (i.e., the change in statistic).

Field values may be set using directly, e.g.,

```
sherpa> sherpa.cov.sigma = 2.6
```

NOTE: strict checking of value inputs is not done, i.e., the user can errantly change arrays to scalars, etc. To restore the default settings of the structure at any time, use the Sherpa/S−Lang module function restore_cov().

The confidence interval estimates are computed quickly, as described below, but are generally more accurate than those found using the command UNCERTAINTY; see also PROJECTION.

Because COVARIANCE estimates confidence intervals for each parameter independently, the relationship between sigma and the change in statistic value delta_S can be particularly simple: sigma = the square root of delta_S for statistics sampled from the chi−square distribution and for the Cash statistic, and is approximately equal to the square root of (2 * delta_S) for fits based on the general log−likelihood.

Confidence Intervals for the covariance command

Confidence	sigma	delta_chi−square	delta_log(L)
68.3%	1.0	1.00	0.50
90.0%	1.6	2.71	1.36
95.5%	2.0	4.00	2.00
99.0%	2.6	6.63	3.32
99.7%	3.0	9.00	4.50

There are a number of computations associated with the COVARIANCE command, which are described in detail in the Sherpa manual.

Output files include the information and covariance matrices, along with the eigenvectors and eigenvalues of the covariance matrix. These are recorded in three temporary ASCII files in the $ASCDS_WORK_PATH directory: ascfit.inf_matrix.<number>, ascfit.cov_matrix.<number>, and ascfit.eig_vector.<number>, where <number> refers to the process ID (pid) number for the Sherpa run. These files may be saved by copying them from the $ASCDS_WORK_PATH directory during the Sherpa session. The files are deleted from the working directory when the Sherpa session is finished. The default setting for this variable may be determined as follows:

```
unix% echo $ASCDS_WORK_PATH
```

Caveats

An estimated confidence interval is accurate if and only if:

- the chi−square or log(L) surface in parameter space is approximately shaped like a multi−dimensional paraboloid, and
- the best−fit point is sufficiently far from parameter space boundaries.

One may determine if these conditions hold by plotting the fit statistic as a function of each parameter's values (the curve should approximate a parabola) and by examining contour plots of the fit statistics made by varying the values of two parameters at a time (the contours should be elliptical, and parameter space boundaries should be no closer than approximately 3−sigma from the best−fit point).

Note that these conditions are the same as those which dictate whether the use of PROJECTION will yield accurate errors. While PROJECTION is more general (e.g. allowing the user to examine the parameter space away from the best−fit point), it is in the strictest sense no more accurate than COVARIANCE for determining confidence intervals.

If either of the conditions given above does not hold, then the output from COVARIANCE may be meaningless except to give an idea of the scale of the confidence intervals. To accurately determine the confidence intervals, one would have to reparameterize the model, or use Monte Carlo simulations or Bayesian methods.

Example 1

List the current and default values of the cov structure, and restore the default values:

```
sherpa> sherpa.cov.sigma = 5
sherpa> list_cov()
Parameter   Current         Default                        Description
----------------------------------------------------------------------
sigma             5               1                    Number of sigma
sherpa> restore_cov()
sherpa> list_cov()
Parameter   Current         Default                        Description
----------------------------------------------------------------------
sigma             1               1                    Number of sigma
```

Example 2

Determine the covariance matrix and errors for all thawed parameters:

```
sherpa> DATA example1a.dat
sherpa> PARAMPROMPT OFF
Model parameter prompting is off
sherpa> SOURCE = GAUSS1D[l2]
sherpa> FIT
...
sherpa> set_verbose(2)
sherpa> COVARIANCE
Information Matrix (Second Derivatives of Fit Statistic):

      p.c0            p.c1            p.c2            p.c3
        0.447924         1.16116         4.64118         24.2449
         1.16116         4.64109         24.2449         146.113
         4.64118         24.2449         146.113           954.8
         24.2449         146.113           954.8         6560.89

Eigenvectors (Principal Axes) of the Covariance Matrix:

      p.c0            p.c1            p.c2            p.c3
        0.646594        0.747372        0.152771      0.00368201
       -0.734474         0.55588        0.388661       0.0221011
        0.205453       -0.361973        0.897769        0.144137
      -0.0159319       0.0375378       -0.140052        0.989304

Eigenvalues of the Covariance Matrix:

         159.066         3.55303        0.118346     0.000149179

Covariance Matrix (Inverse of Information Matrix):

      p.c0            p.c1            p.c2            p.c3
```

```
        68.4903        −74.0584         20.1862        −1.54147
       −74.0584         86.9244        −24.6767         1.92902
        20.1862        −24.6767         7.27528       −0.583802
       −1.54147         1.92902       −0.583802       0.0478489

Covariance Matrix Determinant (Product of Eigenvalues): 0.00997785


Computed for covariance.sigma = 1
        −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
        Parameter Name      Best−Fit Lower Bound     Upper Bound
        −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
             p.c0         −0.303712  −8.27589        +8.27589
             p.c1          0.611953  −9.32332        +9.32332
             p.c2          0.790141  −2.69727        +2.69727
             p.c3         0.0184866  −0.218744       +0.218744
```

CHANGES IN CIAO 3.2

Prior to CIAO 3.2 the COVARIANCE command could not be used until the dataset had been fit. This was done to ensure that the parameter values were at their best−fit location, but caused problems when fitting multiple datasets or loading previously−saved analysis sessions. This restriction has now been removed. Please note that the results of COVARIANCE will not be valid unless the parameters are at their best−fit values.

Bugs

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

See Also

sherpa

berrors, bsyserrors, compute_errors, compute_statistic, 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_paramestint, run_paramestlim, run_paramestreg, set_errors, set_syserrors, staterrors, syserrors, uncertainty

---

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