PyBLoCXS Documentation

Release 0.0.2

CHASC Astro-Statistics Collaboration

June 11, 2010

CONTENTS

Contents:

CHAPTER

ONE

HIGH-LEVEL USER INTERFACE FUNCTIONS

list_priors()

List the dictionary of currently set prior functions for the set of thawed Sherpa model parameters

get_prior(*par*)

Return the prior function set for the input Sherpa parameter *par*

> func = get_prior(abs1.nh)

set_prior(*par, prior*)

Set a prior function *prior* for the the input Sherpa parameter *par*. The function signature for *prior* is of the form lambda x.

set_sampler(*sampler*)

Set a sampler type *sampler* as the default sampler for use with pyblocxs. *sampler* should be of type str. Native samplers available include *“MH”* and *“MetropolisMH”*. The default sampler is *“MetropolisMH”*. For example:

```
set_sampler("MetropolisMH")
```

get_sampler_name()

Return the name of the currently set sampler type. For example:

```
print get_sampler_name()
"MH"
```

get_sampler()

Return the current sampler’s Python dictionary of configuration options. For example:

```
print get_sampler()
"{’priorshape’: False, ’scale’: 1, ’log’: False, ’defaultprior’: True,
’inv’: False, ’sigma_m’: False, ’priors’: (), ’originalscale’: True,
’verbose’: False}"
```

set_sampler_opt(*opt, value*)

Set a configuration option for the current sampler type. A collection of configuration options is found by calling get_sampler() and examining the Python dictionary. For example:

```
# Set all parameters to log scale
set_sampler_opt(’log’, True)
```

```
# Set only the first parameter to log scale
set_sampler_opt('log', [True, False, False])
```

get_sampler_opt(*opt*)

Get a configuration option for the current sampler type. A collection of configuration options is found by calling get_sampler() and examining the Python dictionary. For example:

```
get_sampler_opt('log')
False
```

get_draws(*id=None, otherids=(), niter=1e3*)

Run pyblocxs using current sampler and current sampler configuration options for *niter* number of iterations. The results are returned as a 3-tuple of Numpy ndarrays. The tuple specifys an array of statistic values, an array acceptance booleans, and a 2-D array of associated parameter values. The arguments *id* and *otherids* are used to access the Sherpa fit object to be used in the run by Sherpa data id. Note, before running *get_draws* a Sherpa fit must be complete and the covariance matrix should be calculated at the resultant fit minimum. For example:

```
stats, accept, params = get_draws(1, niter=1e4)
```

get_error_estimates(*x, sorted=False*)

Compute the quantiles and return the median, -1 sigma value, and +1 sigma value for the array *x*. The *sorted* argument indicates whether *x* has been sorted. For example:

```
median, low_val, hi_val = get_error_estimates(x, sorted=True)
```

CHAPTER

TWO

INDICES AND TABLES

- *Index*
- *Module Index*
- *Search Page*

INDEX