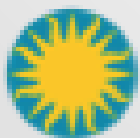




Sherpa Technical Notes

Omar Laurino

Aneta Siemiginowska



Smithsonian Astrophysical Observatory
Chandra X-Ray Center

Procedural vs Object Oriented

Sherpa 3.4

```
data 3c273.pi  
subtract  
notice energy 0.1:6.0  
analysis wave  
sherpa.dataplot.x_log = 1  
lplot data
```

Procedural vs Object Oriented

```
from sherpa.astro import ui
ui.load_arrays(1, x, y)
ui.set_source('powlaw1d.p1')
ui.fit()
```

Procedural vs Object Oriented

```
from sherpa.astro import ui
ui.load_arrays(1, x, y)
ui.set_source('powlaw1d.p1')
ui.fit()
```

```
from sherpa.models import PowLaw1D
from sherpa.data import Data1D
from sherpa.optmethods import NelderMead
from sherpa.stats import Cash
from sherpa.fit import Fit

data = Data1D(1, x, y)
model = PowLaw1D("p1")
fitter = Fit(data, model, Cash(), NelderMead())
results = fitter.fit()
print(results)
```

Integration with Astropy

Saba – Sherpa/Astropy Bridge

Google Summer of Code 2016

`saba`

साबा

`bridge`

Integration with Astropy

Saba – Sherpa/Astropy Bridge

Google Summer of Code 2016

Install Sherpa, Astropy (v1.3+), Saba, then:

```
from astropy.modeling.fitting import SherpaFitter
```

saba

साबा

bridge

Integration with Astropy

`astropy.modeling` package:

Weighted least square

No uncertainties

`astropy.modeling.fitting.SherpaFitter` class:

Select optimization algorithm

Configurable fit statistics

Estimate parameter confidence intervals, including coupled non-Gaussian errors.

MCMC sampler for exploration of posterior probability distribution

C/C++ Extensions

Sherpa's extensions depend on third-party dependencies (FFTW, WCSSubs, CIAO libraries).

C/C++ Extensions

Sherpa's extensions depend on third-party dependencies (FFTW, WCSSubs, CIAO libraries).

Sherpa's source code includes them:

- for convenience

- for lack of a native Python way of tracking non-Python dependencies

C/C++ Extensions

Sherpa's extensions depend on third-party dependencies (FFTW, WCSSubs, CIAO libraries).

Sherpa's source code includes them:

- for convenience

- for lack of a native Python way of tracking non-Python dependencies

Users can edit `setup.cfg` to point to their own builds of the dependencies (e.g. optimized FFTW).

Summary

The evolution of Sherpa tracks the evolution of Computing

Summary

The evolution of Sherpa tracks the evolution of
Computing

Infrastructure Updates vs New Features

Summary

The evolution of Sherpa tracks the evolution of
Computing

Infrastructure Updates vs New Features

Integration with Astropy, Matplotlib, IPython/Jupyter