Yaxx Chandra Quick-start Guide

Table of Contents

1   Overview

This document describes the basic steps in configuring and running *yaxx* to perform X-ray spectral analysis of *Chandra* data. It combines a step-by-step tutorial with discussion of some of the most common user options. It is assumed that *yaxx* has been successfully installed and tested as described in the installation guide.

2   Configure shell environment

The different environments used in X-ray analysis by have the potential to conflict if not initialized in the correct order (e.g. CIAO and FTOOLS). To minimize problems, *yaxx* does the analysis environment initialization internally (in the correct order) and expects that the user has *not* done any initialization.

To proceed with this quick start example, ensure that you have a "clean" environment by opening a new terminal window.

Next set the $YAXX variable to point to the *yaxx* installation directory. For instance, if you installed *yaxx* in your home directory as shown in the installation guide, you would do (for csh or tcsh):

```
set YAXX=~/yaxx
```

3   Create Analysis Root Directory

Yaxx is run from an analysis root directory that should be distinct from the $YAXX source directory. For instance:

```
mkdir ~/yaxx_analysis
cd ~/yaxx_analysis
```

4 Copy Input Chandra Data

Spectral analysis of Chandra data with *yaxx* requires (at a minimum) ACIS event data and aspect solution data, both of which are found in the primary data products for an ObsId. These input data can be stored either outside the analysis directory (e.g. in a raw data repository) or within the analysis directory in a structure that meshes with the output spectral analysis results. Here we demonstrate the latter, using data provided in the *yaxx* installation.

```
mkdir obs3102
cp $YAXX/Test/Data/Obs3102/acisf03102_evt2.fits obs3102/
cp $YAXX/Test/Data/Obs3102/pcadf*_asol1.fits obs3102/
```

The directory location of both the input Chandra data and the output analysis results are configurable.

5 Create Object List File

The list of X-ray sources to be processed by *yaxx* is specified as tabular data in the object list file. Each table row contains data such as the source ObsId and position. The format of this file can be FITS, RDB, or ASCII (with several common column delimiters supported).

To create an object list file for this example, create a new file named `sample.dat` and insert the following lines:

```
obsid   src     redshift        X       Y       object
3102    1       0.32            4167    4085    Q1250+568
```

This indicates that just one source named Q1250+568 with physical sky coordinates (4167,4085) will be extracted. Any fits that rely on a redshift will use 0.32.

6 Copy Configuration Files

The operation of *yaxx* is controlled by configuration files named `yaxx.cfg`. The key configuration files are:

System: `$YAXX/yaxx.cfg`
> System-wide configuration data. In general this should not be modified after installation.

Thread: `$YAXX/resources/<thread>/yaxx.cfg`
> Configuration data for each yaxx analysis "thread", examples of which include *Chandra* and *XMM*. These files define the specific processing steps, environments, and file naming conventions for each thread.

Project: `yaxx.cfg` (in analysis root directory)
> Configuration data for each yaxx analysis "project". This file typically includes basic run parameters, the fit models and rules, and report formatting rules. Any items in the system or thread `yaxx.cfg` files can be overridden by the project configuration file.

For the tutorial example, copy the *yaxx* installation `User/yaxx.cfg` file into the analysis directory:

```
cp $YAXX/User/yaxx.cfg ./yaxx.cfg
```

Examine this file and note some of the options specified such as the input data and output directories, mission and thread, the object list file, fit methods and statistics, source extraction parameters, fit rules, and model definitions.

Details for all parameters are given in the *yaxx* reference manual.

7 Run Yaxx

From your analysis root directory run *yaxx* by entering:

```
$YAXX/yaxx
```

If you have installed the `yaxx` launcher script somewhere in your search path you can simply run:

```
yaxx
```

You will then see *yaxx* processing ObsId 3102. At the end of a processing run *yaxx* will declare SUCCESS if all steps succeeded. The key processing steps are:

- Start a log file
- Copy or link input data to analysis output directory
- Make source and background extraction region files
- Extract spectrum (PI or PHA file) from event data
- Make image of source event data
- Fit specified models to spectral data with *sherpa*
- Make report summary pages (postscript and HTML)

Each processing step is run only if the output files for that step are non-existent or older than the input files. The typical way to force reprocessing from a particular step is to tell *yaxx* to delete certain files before processing using the C<-preclean> option. For instance, to re-do the spectral fitting:

```
yaxx -preclean fit
```

Since the report summary depends on the fit results, it will also be regenerated.

Assuming the processing is successful, examine the results with:

```
firefox report_index.html
```

where firefox can be replaced by the name of your favorite web browser. This shows the *yaxx* report index which links to report pages for the individual sources. The report page shows a summary of the source parameters, an image of the source and the extraction regions used, a table of spectral fit results, and plots of the spectral model fits. In this tutorial there is only one source, but if there are multiple sources then individual reports are linked together by the arrows in the upper left corner of each report.

A latex/postscript report is also created and can be viewed with:

```
gv obs3102/src1/report.ps
```

8 Inspect Key Analysis Data Files

The analysis data files associated with yaxx processing in this example have the structure:

```
<output_dir>/obs<obsid>/src<src>
```

The files unique to an ObsId are within the `obs<obsid>` directory, and those unique to the particular source are in the `src<src>` (i.e. the "source" directory). Some of the more useful files in the source directory are:

```
acis*.pi        : Various pi (or pha) spectral files for fitting
acis.rmf        : RMF
acis.arf        : ARF
<fit_model>.in  : Sherpa script used to fit <fit_model>, e.g. pl.in
report.html     : Final fit summary report (HTML)
report.ps       : Final fit summary report (postscript)
report.tex      : Final fit summary report (latex)
log             : Processing log.  log.<n> files are old logs
```

9 Perform Detailed Interactive Fitting

The `<fit_model>.in` file is a very useful starting point for doing more detailed or interactive spectral fitting of a particular source. For the simple powerlaw model in this example one can exactly recreate the *yaxx* fitting steps in *sherpa* by doing:

```
cd obs3102/src1
sherpa
use pl.in
```

10 Publish Results

The LaTeX report summary `report.tex` within the source directory contains a table of fit results which can be easily copied into a paper for publication. We request that a citation be given if you use *yaxx* in your analysis. A detailed citation for *yaxx* is given in the reference manual.