Reduction and Analysis of PCA Spectra

Recipes from the RXTE Cook Book

- Introduction
- Which configuration to analyze?
- Reduction of spectral data
- Analysis of spectral data
- Simultaneous fitting of HEXTE spectra
- Other PCA configurations

Introduction

After a brief preamble on which configuration to choose for spectral analysis, this recipe outlines all the steps in producing and analyzing spectral data from the PCA. Where appropriate, links are provided to more detailed information, the aim here being to provide a check-list of all the things that need to done.

Which Configuration to Analyze?

The Standard-2 configuration, automatically used for *all* observations, provides all the key information for accurate spectral analysis, namely:

- **PCU ID -** each PCU has a slightly different response.

- **Anode ID -** for most sources, the best signal-to-noise is to be obtained from the top-layer anodes, XL1 and XR1.

- **At least 64 PHA bins -** being a proportional counter, the PCA has somewhat crude spectral resolution: the 129 channels of Standard-2 (as opposed to the full 256) can be used without loss of resolution.

Another desirable property of Standard-2 is that the datafiles are in science array format rather than the slightly more "difficult" and voluminous science event format of Good Xenon. Moreover, the background generator pcabackest creates synthetic datafiles that look just like Standard-2, so that reducing them in parallel with the real thing is straightforward. Given these obvious advantages, **the only good reason to use a configuration other than Standard-2 for spectral analysis is to obtain temporal resolution better than the 16 seconds of Standard-2.**

This recipe, therefore, uses Standard-2 data throughout, but includes a section on other configurations like Good Xenon and Binned Burst Catcher.

Reduction of Spectral Data

Here we sketch all the steps involved in producing the files needed by xspec, namely, source and background spectra and a response. To avoid confusing generalities, this example assumes that your observation is of a flaring AGN.

1. **Use XDF to find your Standard-2 datafiles:** Instructions can be found in the *Getting Started Guide*. We recommend naming the XDF output file something like `std2.xdf`.

2. **Use XDF to find the filter file:** select the Standard Products subsystem, and at the bottom of the AppId window, select Filter_File. We recommend naming the XDF output file something like `filter.xdf`. Please note:

    - If the data you want to examine span more than one ObsId, then it's a good idea to merge the corresponding filter files. Use the ftool **fmerge**, adding `lastkey=TSTOP` to the command line. For more information, type "fhelp fmerge".

    - If you don't have a filter file, then generate one following the recipe *Creating XTE Filter Files*. In fact, to use the latest version of pcabackest, the background generator, you'll probably have to make fresh filter files anyway. [Yes, if the filter file doesn't have the columns `BKGD_THETA` and `BKGD_PHI`.]

3. **Examine the filter file and create a GTI file:** The filter file tells you the status of the satellite and its instruments during your observation and can be used to identify periods of good data. To plot it, type "fplot offset=yes" followed by the name of the filter file. Enter "Time" as the x-axis. For the y-axes, it depends what in particular we want to see:

    - How many PCUs were on? Plot the columns:

      ```
      PCU0_ON
      PCU1_ON
      PCU2_ON
      PCU3_ON
      PCU4_ON
      NUM_PCU_ON
      ```

      Periods where the number of PCUs on is different should be reduced separately and combined later - see the recipe *Combining 3-PCU, 4-PCU and 5-PCU Spectra* for details.

      Another thing you can determine from the number of PCU on is when the SAA passages occur. At these times, all PCU are turned off, so NUM_PCU_ON goes to zero.

    - Where there any Earth occultations? Plot the column `ELV`. Here, zero is defined as pointing at the horizon, negative is looking at Earth, and positive is looking at the source. We recommend filtering out data where the elevation angle is less than 10 degrees. Some data sets have shown effects of bright Earth at ~5 degrees, so you should look at your data before including anything closer than 10.

- Was the pointing stable? Plot the column `OFFSET` This should vary quite regularly on scales of ~10^-3 degrees, and these minor variations will not affect your analysis. They are mainly due to the HEXTE rocking motion. Often, however, you will see steep spikes at the very beginning or end of a given observation; this is where some small amount of slew was included in the on source data file. These must be filtered out, and we recommend using OFFSET < 0.02 as the selection criterion.

- Were there any SAA passages in your observation? Plot the `TIME_SINCE_SAA` column, which is a parameterization of coordinates relative to the SAA which approximately represents minutes since the peak of the last SAA passage. Since it is a coordinate parameterization, it will jump to negative values before the SAA, and in orbits where there is no SAA passage, it is defined to be 100. When TIME_SINCE_SAA is between zero and, for instance, 30 minutes, the background will increase dramatically, lowering your signal to noise. For faint sources, you may want to discard this data. See below.

- Was there significant electron contamination? Plot the ELECTRON*n* columns. (For each PCU, these numbers should be effectively the same.) For analysis of faint sources, the PCA team recommends filtering out time when ELECTRON0 is larger than 0.1. See the PCA Background Status page.

To list all the columns in a filter file, use the ftool **flcol**. For more information about plotting, please check the recipe *Plotting with PLT*.

Note that all but the earliest RXTE data files have two extensions called `GTI`, the first of which contains the times when:

1. Telemetred data are present

2. The satellite is pointing at the nominal source position, as derived from the spacecraft attitude

3. The nominal source position is not occulted by the Earth, as predicted by mission operations

4. The satellite is outside the South Atlantic Anomaly, as predicted by mission operations

The second GTI extension includes time when only the first of the above apply.

These GTI are applied by answering "APPLY" to the saextrct prompt `Input GTI files to be OR'd`. Filtering with only the GTIOR file, however, is not usually sufficient, which is why we recommend making a GTIAND file using the selection criteria discussed above. Mainly, this is because the SAA passages may not occur exactly as predicted, nor do the Earth occults; in addition, the GTIOR's criterion for determining occults does not take into account possible bright Earth effects.

To make your own GTI file to be AND'ed in the extractor, use the tool **maketime**. This tool will make a FITS GTI file from any FITS file with a time column and other columns from which to select. It will take either Fortran or C style. For example, to filter out occults and bright Earth, slews, and times where all PCU's are off during SAA and only three are on otherwise:

```
olegacy [84] [day] ~: maketime
Name of FITS file and [ext#][] FP_3c7a9f5-3c855c1.xfl
Name of output FITS file[] basic.gti
Selection Expression[] elv.gt.10.and.offset.lt.0.02.and.num_pcu_on.eq.3
Column containing HK parameter names[NAME]
Column containing HK parameter values[VALUE]
Column containing HK parameter times[TIME]
Flag, yes if HK format is compact[yes] no
```

Note that the tool is case insensitive with regard to column names and that the last input must be changed from the default 'yes' to 'no' for XTE data. This file can then be put into either saextrct or seextrct as the GTI file to be AND'ed.

For analysis of faint sources, we recommend the additional expressions `(TIME_SINCE_SAA > 30 || TIME_SINCE_SAA < 0.0) && ELECTRON0 < 0.1`, which will filter out data within thirty minutes of the peak of SAA and that with high electron contamination.

4. **Create column selection files:** After verifying which PCUs were on during your observation, create ASCII files containing the columns from which you want to extract spectra. We recommend two files: one containing all the layers of all the PCUs that were on; the other containing just the top layer of all the PCUs that were on.

   For example, if PCU0, PCU1 and PCU2 were the only PCUs on during your observation, you could create the file `pcu012.col` containing:

```
X1LSpecPcu0
X1RSpecPcu0
X2LSpecPcu0
X2RSpecPcu0
X3LSpecPcu0
X3RSpecPcu0
X1LSpecPcu1
X1RSpecPcu1
X2LSpecPcu1
X2RSpecPcu1
X3LSpecPcu1
X3RSpecPcu1
X1LSpecPcu2
X1RSpecPcu2
X2LSpecPcu2
X2RSpecPcu2
X3LSpecPcu2
X3RSpecPcu2
```

   and the file `pcu012_top.col` containing:

```
X1LSpecPcu0
X1RSpecPcu0
X1LSpecPcu1
X1RSpecPcu1
X1LSpecPcu2
X1RSpecPcu2
```

   For more information about PCU and anode selection, please check the recipe *Selecting Data by Detector and Anode*.

5. **Extract a light curve:** To verify that you've removed all the bad data and to identify parts of the light curve that you might want to focus on - the flare, say - extract a light curve with saextrct, as described in the *ABC Guide*. Input `std2.xdf`, your GTI file (at the prompt `Input GTI file to be AND'd with INFILE`) and column selection file (at the prompt `Name of COLUMNS to be accumulated`).

   Plot your light curve with "fplot offset=yes", entering `TIME` as the x-axis and `RATE[ERROR]` as the y-axis. Looking at the plot:

   - Check whether there are any suspicious looking spikes or drop-outs indicative of bad data. These are usually found at the extrema of sections of good data.

   - Note down the start and stop times of the parts of the light curve you want a spectrum from, avoiding any bad data. In our example, we'll be isolating the AGN flare and a stretch of non-flare emission.

   Then run timetrans to generate time filters. Here, we'll call then `flare.tint` and `quiescent.tint`.

6. **Extract spectra:** Rerun saextrct to extract the PHA files, this time applying your time filters at the prompt `Input time intervals t1-t2, t3-t4 in seconds`. You should extract two spectra for each of your time filters: one for all the layers and one for just the top layer. In our example, we'll apply `pcu012_top.col` and `pcu012_top.col` to produce the PHA files:

   ```
   flare_all.pha   quiescent_all.pha
   flare_top.pha   quiescent_top.pha
   ```

7. **Correct for deadtime:** Deadtime exceeds 1 percent when the count rate *per PCU* exceeds 1000 counts per second. To correct for deadtime, please consult the recipe *PCA Deadtime*.

8. **Create background datafiles:** This is a fast-changing topic. Please check with the PCA Instrument Team for news of the latest version of pcabackest and the recipe Running Pcabackest.

9. **Extract background spectra:** Use exactly the same extraction parameters as in step-6 above, but with the list of pcabackest-generated datafiles as input instead of `std2.xdf`. In our example, we'd end up with, say:

   ```
   flare_back_all.pha   quiescent_back_all.pha
   flare_back_top.pha   quiescent_back_top.pha
   ```

10. **Create response matrices:** PCA response matrices should match the PHA in epoch (to ensure that the high-voltage setting is the same), and in PCU and anode. Follow the recipe *Creating PCA Responses* to make responses corresponding to the column-selection files used. For our example, we'd use pcarsp to create:

    ```
    pca_LR1_961029.rsp
    pca_xe_961029.rsp
    ```

Spectral Analysis

Here we summarize the steps and associated xspec commands used when analyzing PCA spectra. An additional subsection shows how to fit other spectra simultaneously. In this example, the files comprise:

```
All layers:
Source spectra ........ flare_all.pha
Background spectra .... flare_back_all.pha
Response ............. pca_xe_961029.rsp

Top layer only:
Source spectrum ....... flare_top.pha
Background spectrum ... flare_back_top.pha
Response ............. pca_LR1_961029.rsp
```

After starting up with "xspec":

1. **Verify the quality of the background subtraction:** At this stage of the mission (about one year after launch), the accurate, reliable estimation of background is still very much a work in progress. To check your subtraction, it's best to look at a plot of unsubtracted source and background together. Using the all-layer flare spectra as an example, the xspec commands are:

   ```
   XSPEC> data flare_all.pha flare_back_all.pha
   XSPEC> resp 1 pca_xe_961029.rsp
   XSPEC> resp 2 pca_xe_961029.rsp
   XSPEC> setplot energy
   XSPEC> plot ldata
   ```

   Look at the high-energy part of the spectra where the background predominates. If the subtraction is good, the two spectra should lie on top of one another.

   If, however, your subtraction looks bad, there are several tricks described in the pcabackest recipe under "problems and remedies" which may improve it. If you cannot adjust the modeled background to your satisfaction , you might try using real background from blank-sky observations. RXTE performed a series of blank-sky observations in November and December 1996. To obtain the data, either FTP anonymously to legacy.gsfc.nasa.gov and get the files in the directory `/xte/data/archive/AO2/P20801` or use W3Browse on the `XTEPUBLIC` catalogue to find the proposal 20801.

2. **Subtract the background:** Read in the data again, this time specifying the background file with the "back" command:

   ```
   XSPEC> data flare_all.pha
   XSPEC> back flare_back_all.pha
   XSPEC> resp pca_xe_961029.rsp
   XSPEC> plot ldata
   ```

   The plot should give you an indication of the useful pass-band. There are no useable data below 2.0 keV, so these should always be excluded. Where, at high energies, to curtail the pass band depends on where the source spectrum meets the background. Assuming that the background predominates above 40 keV:

   ```
   XSPEC> ignore 0.0-2.0 40.0-**
   ```

3. **Specify and fit model:** Models in xspec are entered with the "model" command after which you list the model components followed by the initial parameter values. For more information, please consult the xspec manual or type "help commands model".

4. **Examine fit:** The command "plot data resid" will display the data, the model and the residuals of the fit. Using "ldata" instead of data will plot a log-log plot. Alternatives to the residuals are "chi" for bin-by-bin contributions to chi-squared and "ratio" for the bin-by-bin ratio of data to model. Type "help commands plot" for more details, or look up the xspec section in the recipe *Plotting with PLT*.

5. **Derive physical quantities:** The "error" command, followed by parameter number, will give you the 90 percent confidence range of the best-fitting parameter value. The "flux" command, followed by a pass band specified in keV will integrate the best-fitting model to give the flux in units of erg/cm2/s. And if your model includes an emission line, the "eq" command will return its equivalent width in eV when followed by the model component number - that's the "3" in this example:

```
XSPEC> mo wa po ga
 mo = wabs[1] (powerlaw[2] + gaussian[3])
```

6. **Compare top-layer and all-layer spectra:** Whether the all-layer spectrum or the top-layer spectrum provides the best signal-to-noise depends on source strength and hardness. It's best to verify this by direct comparison of derived results, although for some observations simply looking at the two sets of background-subtracted spectra could be enough.

Simultaneous Fitting of HEXTE Spectra

Let's assume we have the HEXTE files (Clusters 0 & 1) corresponding to the PCA spectrum `flare_all.pha`, i.e. extracted with the same time filter `flare.tint`:

```
Source spectra ....... flare_cl0.pha, flare_cl1.pha
Background spectra ... flare_back_cl0.pha, flare_back_cl1.pha
Response ............. hexte_mt5_pwa_archive.rsp, hexte_mt5_pwb_archive.rsp
```

The only "knack" to simultaneously analyzing multiple spectra lies in specifying the PHA files. When you want to apply exactly the same model with exactly the same parameters to all your spectra, then you can simply list all the PHA files after a single "data" command. Sometimes, however, you want to fit the same model to all your spectra with some parameters shared but with other parameters taking on different values depending on the spectrum. In this case, the "data group" is included in the "data" command.

The simultaneous fitting of PCA and HEXTE spectra falls into this latter category, since, at this stage of the mission, the two instruments are not accurately cross-correlated. By using data groups, you can allow the PCA and HEXTE normalizations to vary separately while sharing the parameters than define the spectral shape.

In this example, we'll assume, *for the sake of convenience*, that the PCA is accurately calibrated and allow the normalization of HEXTE to vary separately. Further, we'll assume that any differences in response between the two HEXTE Clusters are fully accounted for in their respective responses.

1. **Read in the data:** We'll take advantage of the "group" concept:

   ```
   XSPEC> data 1:1 flare_all.pha 2:2 flare_cl0.pha 2:3 flare_cl1.pha
   ```

   Each datafile is prefixed with a group specifier. The first number gives the group, the second the file. This means, for example, that `2:3 flare_cl1.pha` is in the second group and is the third file overall.[Note that if we're not sure of the Cluster-0/Cluster-1 cross-calibration, we could assign `flare_cl1.pha` to a third group.] To associate background and response files with individual files, use the file number, as before (the group doesn't matter here):

   ```
   XSPEC> back 1 flare_back_all.pha
   XSPEC> resp 1 pca_xe_961029.rsp
   XSPEC> back 2 flare_back_cl0.pha
   XSPEC> resp 2 hexte_mt5_pwa_archive.rsp
   XSPEC> back 3 flare_back_cl1.pha
   XSPEC> resp 3 hexte_mt5_pwb_archive.rsp
   ```

2. **Specify the model:** The group comes into play when the model parameters are specified, for each group requires its own set of parameters. When you get to specify the parameters for the second group, hit return for those you want tied to their counterparts in the first group, but enter arbitrary values for those you want to be varied separately. In our case of a simultaneous observation with different instruments, the only parameters that we'll want to vary separately are the normalizations of all the additive components. When the fit converges, the screen will display the values of the shared parameters and the two different normalizations.

Other aspects of multi-group fitting:

- To turn a separately varying parameter into a shared one, use the "newpar" command. For example, to tie parameter 10 to parameter 3:

  ```
  XSPEC> newpar 10 = 3
  ```

  Yes, the spaces are obligatory.

- To turn a shared parameter into a separately varying one, use the "newpar" command to give the parameter a new value. For example:

  ```
  XSPEC> newpar 10 1.04
  ```

- For a simple, graphical representation of the difference in normalizations, plot the unfolded spectra:

  ```
  XSPEC> plot ufspec
  ```

  There should be an obvious discontinuity where the pass-bands overlap.

- For models with more than one additive component, the "const" multiplicative model should be used to ensure that the flux ratios of the components are the same within each group. For example, suppose your model comprises an absorbed power-law with a Gaussian emission line. The usual way to specify it would be:

```
XSPEC> mo wa po ga
```

Instead, for more than one group, use:

```
XSPEC> mo wa ( (po const) + (ga const) )
 mo = wabs[1] (constant[3] (powerlaw[2]) + constant[6] (gaussian[5]))
```

This time, when you enter the parameter values, the normalizations should be treated as shared parameters while the constants should vary separately. The additional step of tying the two constants together within each group is done with the newpar command before fitting. In our three-group example, we'd say:

```
XSPEC> newpar 8 = 4
XSPEC> newpar 16 = 12
XSPEC> newpar 24 = 20
```

where, the two constants are parameters 4 & 8 in the first group, 12 & 16 in the second, 20 & 24 in the third.

Other PCA Configurations

As mentioned above, the only good reason to extract spectra from a configuration other than Standard-2 is to obtain better time resolution than 16 seconds. Whilst this presents no special problems for extracting source spectra, generating background is not quite as straightforward as for Standard-2.

The data used by pcabackest to generate synthetic background have an intrinsic resolution of 16 seconds. The full model, therefore, cannot be applied without interpolation (which doesn't work well at present). However, time-series analysis of background data shows that the background light curve does not contain much power above 0.1 Hz. Practically, this means that the same 16-second pcabackest spectrum can be used as background for, say, all four 4-second extracted from the same time interval - although at present it's hard to quantify how much additional error this entails.

A second area that requires extra attention is channel binning of the background data. In its current version, pcabackest can output data with two binnings: Standard-2 with 129 bins, and, via linear interpolation of the former, Good Xenon with the full 256 bins. Choose the latter binning scheme, then:

- **If your data are Good Xenon** then extract your source specturm (with sefilter) and background spectra (with saextrct), and generate a response in the usual way.
- **If your data are *not* Good Xenon** then:
    1. Extract source and background spectra in the usual way.
    2. Run the ftool rddescr to parse the channel descriptor in the *source* PHA file and write it to a file called, by default, `chan.txt`

3. Rebin the *background* PHA file with the ftool rbnpha adding `binfile=chan.txt` to the command line.

4. Generate a response in the usual way, using the *source* PHA file as input.

Finally, remember that for all configurations except Standard 1 & 2 and Good Xenon the EDS applies gain and offset corrections before telemetry to align the gains of the individual PCUs to a common scale. Since you generate background and response *after* telemetry, you have to apply the same gain alignment on the ground. For the background, this is simply a matter of answering "yes" to the pcabackest question:

```
Correct for PCU Xenon gain variations?
```

For the response, it's even simpler, as pcabackest automatically recognizes whether the input PHA file requires the gain alignment and applies it to the PHA file and response.
