econvolverprep

January 12, 2017

Abstract

Creates a set of PSF convolvers for use in making sensitivity maps.

1 Instruments/Modes

Instrument	Mode
EPIC MOS	IMAGING
EPIC PN	IMAGING

2 Use

This task is not designed for stand-alone use but as part of **esensitivity**.

3 Description

This task is called, together with **sensitivity**, by the perl task **esensitivity**. The descriptions for all three of these tasks should be consulted together.

The overall purpose of **esensitivity** is to create maps of the minimum count rate of a point source which would just be detectable by **eboxdetect**. The first part of the algorithm (which is described in the **sensitivity** task documentation) calculates the minimum counts within the detection box which would just be detectable; in order to convert this counts-within-a-box to total count rate of the whole source[1], **esensitivity** needs to know the ratio between the two across the width of the instrument field of view. This ratio map is calculated by **econvolverprep** followed by **asmooth**. **econvolverprep** first creates a list of small convolver images, each of which is the PSF truncated within the detection box. Each convolver corresponds to a separate section of the camera field of view. The sections are created by dividing up the field of view in a polar fashion into `naxial` by `nradial` divisions. An 'index image' relating each convolver to its appropriate section is also constructed by **econvolverprep** and transmitted to **asmooth**.

The ratio between the total count rate of a source and the expected number of counts to fall within the detection box is calculated as follows. Suppose the square detection box, of side $(b-1)/2$ for integer

[1] The detection box usually excludes a significant portion of the Point Spread Function or PSF of the source.

$b \geq 0$, is centred on image pixel (k, l). If we assume that a source detected in this box has the centre of its PSF within the central pixel, but that any position within this central pixel is equally likely, then the average number of counts C expected in the box, in proportion to the total count rate R counts s^{-1}, is given by

$$C = R \sum_{i=k-b}^{k+b} \sum_{j=l-b}^{l+b} \epsilon_{i,j} \int_{i-0.5}^{i+0.5} \int_{j-0.5}^{j+0.5} dx\ dy\ psfa(x-k, y-l).$$

I use here the convention that the continuous coordinate value (x, y) becomes equal to the pixel indices (k, l) at the centre of the pixel. The quantity $\epsilon_{i,j}$ is the exposure in seconds at pixel (i, j) and $psfa$ is the 'pixel-averaged' PSF, ie

$$psfa(x, y) = \int_{-0.5}^{+0.5} \int_{-0.5}^{+0.5} dx'\ dy'\ psf(x-x', y-y'). \tag{1}$$

The PSF (in units of pixel^{-1}) is assumed to be normalized such that

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} dx\ dy\ psf(x, y) = 1.$$

The integrals in equation 1 cannot of course be evaluated directly in a numerical context but must be replaced by sums, viz:

$$psfa(x, y) = \frac{1}{(2N+1)(2M+1)} \sum_{i=-N}^{N} \sum_{j=-M}^{M} psf\big(x - [0.5i/N], y - [0.5j/M]\big).$$

At present the task has both N and M set to 0, in other words $psf = psfa$. If this ever seems to present a major bottleneck on the accuracy of the output of **esensivity** I may decide to increase the resolution.

The PSF used at present is the medium-accuracy one obtainable from the **cal** library interface CAL_getPsf. This is returned as an array, I don't know whether of point samples or of the psf integrated across the array pixel size. The medium-accuracy PSF was obtained via **scisim** ray-tracing experiments rather than any analytical calculation, thus point samples would seem more likely. In any case neither the pixel size nor orientation of the returned array match in general the image coordinate system on which source detection is performed: the PSF image must therefore be rebinned to the source-detection system. This process unavoidably entails an additional loss of precision.

Note that the output of **econvolverprep** and **asmooth** is actually C/R, which has units of seconds. This may be considered the exposure 'seen' within the detection box by **eboxdetect**.

4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints

boxhalfsize	yes	integer		0 ≤`boxhalfsize`

This should match the value of –boxsize supplied to **eboxdetect**. Suppose –boxsize=b; the corresponding value of `boxhalfsize` should be $(b-1)/2$. Note that –boxsize as presently defined in **eboxdetect** is always odd.

naxial	no	integer	36	4 ≤`naxial`

The number of angles around the optic axis at which the PSF will be sampled to make a convolver. The total number of convolvers is `naxial`×`nradial`.

nradial	no	integer	10	1 ≤`nradial`

The number of radii out from the optic axis at which the PSF will be sampled to make a convolver. The total number of convolvers is `naxial`×`nradial`.

expmapset	yes	dataset		

What should be supplied here is a template image which is of the same dimensions and coordinates as the images and exposure maps which were supplied to **eboxdetect**. The supplied image must contain WCS keywords which describe the coordinate system.

convolversset	yes	dataset		

The output dataset which contains both the convolvers and the index image relating them to the camera field of view. The format of this dataset is described in the documentation for **asmooth**.

efraction	no	real	0.9	0 <`efraction`

The PSF spreads across the whole field of view, so for practical reasons a smaller, truncated core image is usually obtained. At present **econvolverprep** obtains the medium-accuracy (= ray-trace-modelled), truncated PSF function from the **cal** library. The **cal** calculates the truncation point so that the included part of the PSF contains approximately the fraction of total flux specified by the present parameter. NOTE! At present there is NO protection to ensure that the returned PSF extends beyond the boundaries of the detection box (the size of which is specified via `boxhalfsize`). However there is a warning message which relates to this case.

psfestyle	no	string	user	user—dss

eboxdetect works in general on a list of images of the same part of the sky in different energy bands. Since the PSF varies with energy, a different set of convolvers is required for each band. The mean values of the energy band can be supplied to **econvolverprep** in one of two ways: either directly via the parameter `psfenergy`, or via a data subspace (DSS) record of all event selections within the image dataset itself as supplied to the parameter `dssimageset`. NOTE that the inclusion of a DSS in XMM EPIC images is optional, so one may not be present.

psfenergy	yes	real		1 ≤`psfenergy`

If `psfestyle`='user', the task looks for the mean energy value in the present parameter.

dssimageset	yes	dataset		

If `psfestyle`='dss', the task looks for the mean energy value in the data subspace of the image set supplied to the present parameter.

5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

badPsfEnergyStyle *(error)*
The value of the parameter `psfestyle` was not recognized.

noDssPiRanges *(error)*
No filter on column `PI` was found in the data subspace.

badInstrument *(error)*
The instrument (read from the `INSTRUME` keyword of `expmapset`) is not one of the EPIC ones.

psfImageTooSmall *(warning)*
The PSF image doesn't cover all the convolver extent. You should use a larger value of `efraction`. (Note that this is not 100% foolproof since the situation in which the PSF partially overlaps some convolver cells is not detected. But it will have to do for now.)
corrective action: No corrective action.

6 Input Files

1. `expmapset`: a template image which is of the same dimensions and coordinates as the images and exposure maps which were supplied to **eboxdetect**. The dataset header must contain the following keywords:

   - `INSTRUME` (must be either EMOS1, EMOS2 or EPN).
   - `RA_PNT`, `DEC_PNT`, `PA_PNT`: the spacecraft mean attitude (should be the instrument attitude!) in decimal degrees.
   - `DATE-OBS`

   The image array header must in addition contain WCS keywords.

   Note that **esensitivity** passes to parameter `expmapset` the exposure map in the appropriate energy band which was supplied to the –expimagesets parameter of **eboxdetect**. This dataset type is expected to meet the above criteria.

2. (optional) `dssimageset`: this dataset is read if `psfestyle`='dss'. It should contain an event image in the primary array, which has an associated data subspace recording the event selections which were performed. The image in the appropriate band supplied to the –imagesets parameter of **eboxdetect** is recommended.

7 Output Files

1. `convolversset`: the format of this file is described in the task description for **asmooth**.

8 Algorithm

```
*** Not yet written
```

9 Comments

-

References