bkgfit

November 4, 2014

Abstract

This task fits a linear combination of real-valued image components to a Poissonian, integer-valued image.

1   Instruments/Modes

Instrument	Mode
EPIC, RGS or OM	Imaging

2   Use

pipeline processing	not yet
interactive analysis	yes

3   Description

Although the task is non-XMM specific, it is primarily intended to generate maps of the background in XMM EPIC images. The task does this by fitting a linear combination of background-model component images. The task takes three main inputs (see section 6 for details): (i) the Poissonian FITS image which is to be fitted; (ii) a list of N model component FITS images; (iii) (optionally) a FITS mask image. The output is a single FITS image which represents the best-fit background model. The best-fit amplitudes $\mathbf{a}_{\mathrm{opt}}$ and the names of the component datasets are recorded in this output dataset in a binary table extension.

Clearly all the input images must have the same dimensions: call this $X \times Y$ pixels.

The fitting is done by minimizes the maximum-likelihood estimator L defined as follows:

$$L(\mathbf{a}) = -2 \sum_{x=1}^{X} \sum_{y=1}^{Y} ln[P_{x,y}(\mathbf{a})] \tag{1}$$

where $\mathbf{a}$ is the vector of N component amplitudes, and the sums are understood to be over all unmasked image pixels. Suppressing the x, y subscript for the sake of brevity, the Poissonian probability P is given by

$$P(\mathbf{a}) = \frac{B^I(\mathbf{a})\,exp[-B(\mathbf{a})]}{I!} \qquad (2)$$

where the total background model B is the linear combination of the N components b_i, viz

$$B(\mathbf{a}) = \sum_i^N a_i b_i,$$

and I represents the value at that pixel of the Poissonian image. Inserting equation 5 into 1 gives

$$L(\mathbf{a}) = -2 \sum_{x=1}^X \sum_{y=1}^Y \left\{ I_{x,y}\,ln[B_{x,y}(\mathbf{a})] - B_{x,y}(\mathbf{a}) - ln(I_{x,y}!) \right\}.$$

3.1 Bottom-up explanation:

For those whose heads began to ache at around the second equation, think of it this way: at each pixel of the image, the probability of recording I events is given by the Poisson distribution corresponding to the expectation value $< I >$. Let us take B, the sum of our model terms at that pixel, as an estimate of $< I >$. We can then assign a probability to the pixel value from equation 5. Now let us multiply all the probabilities for individual pixels together. We conjecture that this product (or, equivalently, minus its log) will be minimized exactly for $B =< I >$.

3.2 Fitting algorithm:

The vector of amplitudes $\mathbf{a}_{\mathrm{opt}}$ which minimizes L is approximated by a Newtonian method in N dimensions. In this method, the formula

$$\nabla L = \mathbf{C}\,(\mathbf{a}_{i+1} - \mathbf{a}_i)$$

where $\mathbf{C}$ is an $N \times N$ matrix of curvature terms given by

$$C_{i,j} = \frac{\partial^2 L}{\partial a_i \partial a_j}, \qquad (3)$$

is iteratively inverted until it is judged to have converged.

∇L and $\mathbf{C}$ can be expressed in closed form as

$$(\nabla L)_i = \frac{\partial L}{\partial a_i} = 2 \sum_{x=1}^X \sum_{y=1}^Y \left(b_{x,y,i} \left[1 - \frac{I_{x,y}}{B_{x,y}(\mathbf{a})} \right] \right),$$

$$C_{i,j} = \frac{\partial^2 L}{\partial a_i \partial a_j} = 2 \sum_{x=1}^{X} \sum_{y=1}^{Y} \left[\frac{b_{x,y,i}\, b_{x,y,j}\, I_{x,y}}{B_{x,y}^2(\mathbf{a})} \right].$$

3.2.1   Constraints on the model components:

The fitting procedure will only converge to a unique solution if the components b_i are linearly independent. A measure of this is the quantity

$$Q_{i,j} = \frac{\sum_{x=1}^{X} \sum_{y=1}^{Y} b_{x,y,i}\, b_{x,y,j}}{\sqrt{\left(\sum_{x=1}^{X} \sum_{y=1}^{Y} b_{x,y,i}^2 \right) \left(\sum_{x=1}^{X} \sum_{y=1}^{Y} b_{x,y,j}^2 \right)}}. \tag{4}$$

If one thinks of each image b_i as a vector in an $X \times Y$ dimensional space, then $Q_{i,j}$ is the cosine of the angle between b_i and b_j. The task calculates Q for each pair of component images and fails with an error if any Q is greater than 0.99. NOTE that the sums in equation 4 are carried out on *unmasked pixels only*. Hence components may be linearly independent in masked pixels yet still fail the Q test within the task.

The task also fails if any of the components has only zero-valued pixels. Finally, since the Poisson image values are necessarily all ≥ 0, negative values are not allowed in the model components.

3.3   Truncated Poisson fitting:

Essentially this involves masking off all pixels whose values exceed a certain limit. The value of this is that bright non-background regions of an image can be easily excluded. A complication is introduced into the calculation of L because the probability of occurrence of any pixel value now obeys a truncated Poisson distribution, as follows:

$$P_{\text{trunc}}(\mathbf{a}) = \frac{\{B^I(\mathbf{a})\, exp[-B(\mathbf{a})]\}/I!}{\sum_{j=0}^{\tau} \{B^j(\mathbf{a})\, exp[-B(\mathbf{a})]\}/j!} \text{for} I < \tau, \tag{5}$$

where τ is the truncation level (rounded to the next lowest integer). Otherwise the procedure is unchanged.

3.4   Uncertainties in the coefficients:

It is known (see eg [1]) that the ML estimator L as defined in equation 1 behaves approximately like χ^2, but with an additive offset that varies with the nature of the fitting function. The variance matrix of the fitted model parameters can therefore be obtained (see [1] again, chapter 8) from twice the inverse of the Hessian or curvature matrix $\mathbf{C}$ which is defined in equation 3.

3.5   Goodness of fit:

The likelihood statistic L is just one of many ways of estimating χ^2 for Poissonian data (see [2] for other examples). It is used here in the fitting procedure because it has the advantage that the Hessian

components can be calculated analytically. It is however of no use in judging goodness of fit of the final solution, because of a large additive offset between the value of L and the value of χ^2 appropriate to the probability of occurence of any particular L value. A better estimator of χ^2 is the Pearson formula ([2]):

$$\chi^2_{\mathrm{P}} = \sum_{x=1}^{X} \sum_{y=1}^{Y} \frac{(I_{x,y} - B_{x,y})^2}{B_{x,y}}.$$

This is used in the task to estimate χ^2 and hence a probability that the fitted model B is a good predictor of the data I.

4   Parameters

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

Parameter	Mand	Type	Default	Constraints
poissonset	yes	dataset		

The name of the Poissonian image which it is desired to fit background to.

| **fitsets** | yes | dataset list | | |

List of names of datasets, each of which is an image which represents one of the components of the background model to be fitted.

| **fitstyle** | no | string | whole | whole—base—wholethenbase |

If `fitstyle`='whole', the pixels of `poissonset` may have any value, and a standard Poisson distribution is assumed. If `fitstyle`='base', only pixels which have values below a truncation level are included in the fit: in this case the probability distribution is necessarily a (renormalized) Poisson distribution which extends only from zero to the truncation level. The second type of fit is useful if there are bright patches on the image which it is desired to exclude.

| **truncationset** | yes | dataset | | |

This parameter is read if `fitstyle`='base'. It should point to a dataset which has in its primary extension an image, of the same pixel dimensions as `poissonset`, `fitsets` or any other image input to the task. The task interprets this as an image of cutoff values: only those pixels of `poissonset` which have values below the corresponding value of `truncationset` will be fitted to the task; the probability distributions at those pixels being similarly truncated.

| **withmaskset** | no | bool | no | |

If 'yes', the task reads the name of a mask image from `maskset`. Only unmasked pixels (that is, pixels for which the mask evaluates to TRUE) are fitted.

| **maskset** | no | dataset | mask.ds | |

The name of the mask image dataset.

| **withinitcoeffs** | no | bool | no | |

If 'yes', the task reads starting values of the amplitudes **a** from `initcoeffs`. If `withinitcoeffs`='no', the task estimates these itself.

initcoeffs	no	real list	1.0	

A list of starting values of the amplitudes **a**. This is only read if `withinitcoeffs`='yes'. There must be the same number of elements in this list as in `fitsets`.

outbkgmapset	no	dataset	outset.ds	

The name of the output image dataset.

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.

noComponents *(error)*
> The list `fitsets` is empty.

inconsistentComponents *(error)*
> A pair of `fitsets` component images was found with differing dimensions.

badComponentDimensions *(error)*
> The dimensions of the components `fitsets` are not the same as those of `poissonset`.

badMaskDimensions *(error)*
> The dimensions of `maskset` are not the same as those of `poissonset`.

negativePoissonValues *(error)*
> Some of the pixels of `poissonset` have negative values.

negativeValuesModelImage *(error)*
> One of the model components has some negative-valued pixels.

modelImageAllZero *(error)*
> One of the model components has been found with all zero pixel values.

maskImageAllFalse *(error)*
> No true-valued mask pixels were found.

imagesTooSimilar *(error)*
> Two of the model components have been found to be too similar (ie, Q for this pair exceeds the maximum permitted value). See section 3.2.1.

badFitStyle *(error)*
> The value of `fitstyle` was not recognized.

wrongNumCoeffs *(error)*
> The number of elements in the list `initcoeffs` was not the same as the number in the list `fitsets`.

badLuSolveExitStatus *(error)*
> The **ssclib** routine solveLinearEquations exited with a non-zero status, which indicates an error.

fitIsDiverging *(warning)*
>   The fitting routine is no longer converging.
>   *corrective action:* The fitting routine terminates.

maxIterationsExceeded *(warning)*
>   The fitting routine has exceeded a set maximum number of iterations without achieving the
>   convergence criterion.
>   *corrective action:* The fitting routine terminates.

zeroChiSquared *(warning)*

>   *corrective action:* The best-fitting chi-square value is zero, which in practice should never
>   happen for the real data.

6   Input Files

There are three types of FITS image input, listed below. All input images must share the same dimensions.

1. A Poissonian image I, that is, an image for which the values are random, non-negative integers which obey a Poisson probability distribution about a conjectural image of expectation values $< I >$. This image may have any numeric data type, but it is converted within the task to 16-bit integer.

2. A list of N model component images $\mathbf{b}$. The task **bkgfit** calculates amplitudes $\mathbf{a}$ such that $\sum_i^N a_i b_i$ approximates $< I >$, the expectation value of I. The component images may have any numeric datatype but are converted within the task to 32-bit reals. Negative values are not permitted in any component, no component image may be all zero, and components which are too similar should be avoided.

3. (Optional) a mask image. This may have any numeric data type but is converted within the task to 1-bit logical. The conversion rule is that values > 0 give TRUE, all others FALSE. Only those pixels for which the mask is TRUE are fitted.

7   Output Files

The output dataset contains a 32-bit-real-valued image which is that linear combination $\sum_i^N a_i b_i$ of the input model components b_i which most closely approximates the expectation value $< I >$ of the (masked) image I. The best-fit amplitudes $\mathbf{a}_{\mathrm{opt}}$ and the names of the component datasets are stored in the image array header in a table named `COMPDATA` which has columns as follows:

- `WEIGHT`: stores the values of $\mathbf{a}_{\mathrm{opt}}$.

- `FILENAME`: stores the corresponding element of `fitsets`.

8   Algorithm

Not yet written.
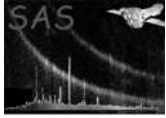
9   Comments

References

[1] P. R. Bevington. *Data Reduction and Error Analysis for the Physical Sciences*. McGraw-Hill, 1 edition, 1969.

[2] K. J. Mighell. Parameter estimation in astronomy with poisson-distributed data. i. the $\chi_\gamma 2$ statistic. *Ap. J.*, 518:380–393, 1999.