



# specgroup

January 27, 2025

## Abstract

A procedure to group spectral channels

## 1 Instruments/Modes

Instrument	Mode
EPIC,RGS	IMAGING, TIMING

## 2 Use

pipeline processing	yes
interactive analysis	yes

## 3 Description

**specgroup** is a tool which allows a flexible, user-defined grouping of channels in a spectrum. It extends the functionality of the ftool *GRPPHA* to allow the user more options of grouping the spectrum based on statistical criteria.

Channels can be predefined as BAD (i.e. not to be used) either by using definitions from a calibration file or by specifying the channels or energy range to exclude on the command line.

The treatment of the special cases of the first and last group is user definable (section 4.4).

### 3.1 Setting bad channels

The quality of spectral channels may be set to bad by choosing the appropriate option. If set bad, channels will not be considered when grouping channels with one of the statistical methods. However, if a range or regular binning is specified which includes these bins then this will take precedence and the quality of these channels will be set to good.



The options are:

```
specgroup spectrumset=srcspec.ds setbad=CCF
```

in which case the channels to exclude from spectral fitting will be read from a calibration file.

```
specgroup spectrumset=srcspec.ds setbad='0:10,790:799'
```

in which case the channels from 0 to 10 and 790 to 799 will be set bad.

```
specgroup spectrumset=srcspec.ds setbad='0:0.2,10.0-15.0' units=KEV
```

in which case the channels with energy between 0 and 0.2 keV or 10 and 15 keV will be set to bad.

```
specgroup spectrumset=srcspec.ds setbad=no
```

in which case no channels will be excluded before the spectrum is grouped.

## 4 Grouping options and examples

### 4.1 User defined ranges

The spectrum may be grouped using a fixed set of ranges in channel space, e.g.:

```
specgroup spectrumset=srcspec.ds groupedset=SpecGrp.ds ranges="1:10,11:50"
```

which will produce two groups: the first from spectral channel 1 to spectral channel 10 and the second from spectral channel 11 to 50. The rest of the spectrum will remain ungrouped. Note that the spectral channel refers to the channel number contained in the *CHANNEL* array in the spectrum. This is not necessarily the same as the bin number of the spectrum.

The ranges can also be defined in photon energy units (KEV), e.g.:

```
specgroup spectrumset=srcspec.ds groupedset=SpecGrp.ds ranges="0.0:0.4,1.0:2.0" units=KEV  
rmfset=m1.rmf
```

which will produce a group containing spectral bins with *central* energies between 0.0 and 0.4 keV and another with central energies between 1.0 and 2.0 keV. The conversion between spectral channel and photon energy is given in the *EBOUNDS* extension of an OGIP compliant response file (see George et al...). Therefore if ranges are given in keV a redistribution matrix (RMF) must be supplied otherwise the error, *NoRMFsupplied* will be issued.

This option may be used in conjunction with the regular binning option and/or one of the statistical options. In which case the range grouping is applied first and the subsequent options are applied to the remaining ungrouped bins.



## 4.2 withRegularBins

This is a simplistic option which allows a spectrum, or part of a spectrum, to be grouped in regular intervals, e.g.

```
specgroup spectrumset=srcspec.ds groupedset=SpecGrp.ds regbinstart=1 regbinend=300 regbinwid=10
```

which would make 30 groups of 10 bins each from the initial 300 bins in the spectrum. Note that in this case the bins refer to the position in the spectrum and start from bin 1.

This option may be used in conjunction with one of the statistical options. In which case the regular bin grouping is applied first and the statistical grouping is applied to the remaining ungrouped bins.

## 4.3 Statistical grouping

One of the following statistical tests may be applied to the data to calculate spectral groups.

1. Minimum number of counts
2. Minimum signal-to-noise ratio
3. Minimum ratio above the background

Only one of these may be selected otherwise the error *MoreThanOneStatMethod* will be issued.

The chosen statistic will be applied to the spectrum starting from the lowest to the highest bin (normally lowest to highest energy) by default. This may be reversed by setting the option *hightolow=yes*.

The treatment of the last group is user selectable (see section *refspecgroup:options:lastbin*).

### 4.3.1 Minimum number of counts

With this option the spectrum is grouped such that each group contains a minimum number of (source+background) counts. This is the equivalent to the "*group min 25*" command available in the *f*tool *GRPPHA*. e.g.

```
specgroup spectrumset=srcspec.ds mincounts=50
```

which would produce a group for each set of bins which contained 50 or more counts. By default the grouping begins from the low energy end (usually the first channel). To reverse this and start from the high energy end use:

```
specgroup spectrumset=srcspec.ds mincounts=50 hightolow=yes
```

Note this grouping is based on the number of counts in the source spectrum, which may also contain background.



### 4.3.2 Minimum signal-to-noise ratio

This sets spectral groups such that each group contains a minimum signal-to-noise ratio (S/N). The calculation of the ratio will take the area-normalised background into account if a background spectrum is provided. e.g.:

```
specgroup spectrumset=srcspec.ds minSN=6.0 backgndset=bgdspec.ds
```

would make groups with a minimum S/N of 6 where S/N is given by:

$$S/N = (source - background * areanorm) / \sqrt{(source + background * areanorm^2)} \quad (1)$$

### 4.3.3 Minimum ratio above background

Defines groups where each one contains a minimum ratio of the source signal to the background counts:

$$ratio = (source - background * areanorm) / (background * areanorm) \quad (2)$$

### 4.3.4 Copy from a template spectrum

The grouping can be copied directly from another spectrum by:

```
specgroup spectrumset=bgndspec.ds grouptemplate=sourcespec.ds
```

This would group the background spectrum in exactly the same way as the source spectrum.

## 4.4 Treatment of the last bin and other ungrouped bins

In the case where the last few bins of the spectrum do not meet the required statistical criterion for a normal group it may be treated in several ways:

1. Leave the bins ungrouped and set their quality to bad:  
`lastbin="setbad"`
2. Group them and set their quality good:  
`lastbin="owngroup"`
3. Add these bins into the final group:  
`lastbin="addtogroup"`

The default is to add these bins onto the adjacent group. A different option may be selected by e.g.

```
specgroup spectrumset=srcspec.ds mincounts=25 lastbin="setbad"
```

In principle, a set of user-defined ranges could leave ungrouped bins in the middle of the spectrum. These will also be treated according to the `lastbin` parameter setting.



## 4.5 Other grouping options

### 4.5.1 Minimum resolution

If the number of counts in a spectrum is very high then a statistical grouping can dramatically oversample the instrument resolution. This can cause problems during spectral fitting because the bins are then not completely independent.

A minimum energy width of each group can be enforced by setting the parameter `oversample`. e.g.

```
specgroup spectrumset=srcspec.ds mincounts=25 oversample=3 rmfset=myrmf.ds
```

would ensure that no group was narrower than 1/3 of the full width half maximum (FWHM) resolution at the central photon energy of the group.

$$\Delta E \geq FWHM(E)$$

where the values of  $FWHM(E)$  are read for a given instrument from a calibration file (*XMM\_SPECQUAL\_nnnn.CCF* for XMM instruments).

## 4.6 Associated file keywords

The task will write the associated background, response and effective area filenames into keywords in the output spectrum header so that they can be picked up by spectral fitting software.

```
specgroup spectrumset=srcspec.ds addfilenames=yes arfset=myarf.ds  
rmfset=myrmf.ds backgndset=mybgnd.ds
```

will fill the ANCRFILE, BACKFILE and RESPFILE keywords in the spectrum header. If the background, ARF or response files are not given on the command line then these keywords will be set to the value that they had in the input spectrum or if not present in the input file they will be set to 'NONE'.

Some spectra contain the header keyword 'GROUPING=0'. If this exists in the header of the input spectrum it will be removed from the output file. All other keywords will be copied over.

## 5 Default behaviour

The command:

```
specgroup spectrumset=srcspec.ds
```

will have the following behaviour:

- Channels defined as bad for that instrument in the corresponding XMM\_SPECQUAL CCF will be set BAD.
- All keywords, ANCRFILE, BACKFILE, RESPFILE, will be set to "NONE"
- The data will be grouped such that each input channel is in a group of its own



To set the channels bad and leave the data ungrouped use the command:

```
specgroup spectrumset=srcspec.ds mincounts=0
```

## 6 Spectra made from combined instruments

It is possible to produce a spectrum from combined instruments using the tasks **rgscombine** or **epic-speccombine**. The combined spectra have an undetermined contribution from each instrument and so assumptions have to be made when grouping the spectra. For simplicity, **specgroup** uses the settings of *MOS1* if the spectrum has been made from a combination of EPIC cameras and the settings of *RGS2* if made from a combination of RGS1 and RGS2.

## 7 Parameters

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

Parameter	Mand	Type	Default	Constraints
-----------	------	------	---------	-------------

<b>spectrumset</b>	yes	dataset		
--------------------	-----	---------	--	--

The name of the input spectrum in counts v channel space.

<b>overwrite</b>	no	boolean	no	
------------------	----	---------	----	--

Whether to overwrite the input spectrum with grouping information or to produce a new file (default)

<b>groupedset</b>	no	dataset	SpecGrp.ds	
-------------------	----	---------	------------	--

The name of the grouped output spectrum

<b>backgndset</b>	no	dataset		
-------------------	----	---------	--	--

The name of the background spectrum

<b>mincounts</b>	no	int		
------------------	----	-----	--	--

The minimum number of counts (source+background) to use when grouping

<b>minSN</b>	no	real		
--------------	----	------	--	--

The minimum signal to noise ratio to use when grouping

<b>ratioabovebgnd</b>	no	real		
-----------------------	----	------	--	--

The minimum ratio of source to background to use when grouping

<b>grouptemplate</b>	no	dataset		
----------------------	----	---------	--	--

The name of a spectrum from which to copy the grouping parameters

<b>oversample</b>	no	real		
-------------------	----	------	--	--

The minimum energy resolution for each group, e.g. if oversample=3 then the minimum width of a group is 1/3 of the resolution FWHM at that energy.

<b>ranges</b>	no	string		
---------------	----	--------	--	--

Ranges to group over. They should be entered as integer numbers, if defined in channels, or as decimals if defined in keV (in which case the **units** parameter should be set accordingly. The beginning and end of



a range may be separated by either a colon or a dash. Multiple ranges should be separated by a comma. e.g. "2-20,63-97" or "0.2-0.5" or "0.4:3.6" etc.

<b>units</b>	no	choice	CHAN	CHAN KEV
--------------	----	--------	------	----------

The units used for ranges - can be channel number or keV.

<b>regbinstart</b>	no	int		
--------------------	----	-----	--	--

First channel to use if grouping done by regular bins

<b>regbinend</b>	no	int		
------------------	----	-----	--	--

Last channel to use if grouping done by regular bins

<b>regbinwid</b>	no	int		
------------------	----	-----	--	--

Number of channels to combine in each group if grouping done by regular bins

<b>rmfset</b>	no	dataset		
---------------	----	---------	--	--

The name of a redistribution matrix used to define the energy of the spectral channels.

<b>hightolow</b>	no	boolean	false	
------------------	----	---------	-------	--

Whether to group from low to high (false) or high to low (true) energies.

<b>addfilenames</b>	no	boolean	true	
---------------------	----	---------	------	--

Whether to set the ANCRFILE, RESPFILE, BACKFILE keywords in the spectrum header.

<b>arfset</b>	no	dataset		
---------------	----	---------	--	--

The name of an effective area (ARF) file. This is purely used as a string to write into the ANCRFILE keyword. The file is not opened.

<b>lastbin</b>	no	choice	addtogroup	addtogroup setbad owngroup
----------------	----	--------	------------	-------------------------------

What to do about ungrouped channels, typically the channels at the end of the spectrum which don't meet the statistical criterion.

<b>setbad</b>	no	string	CCF	CCF no ranges_of_values
---------------	----	--------	-----	----------------------------

Whether to set some channels BAD and exclude them from grouping. Options are: "CCF" - to read the BAD channels from a calibration file, "no" - meaning 'take no action' or a range such as "0-10,750-799" may be entered to explicitly specify which channels to set BAD. The range may be defined in channels or keV, consistent with the setting of the **units** parameter.

## 8 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.

### NoRMFSupplied (*error*)

The input units for ranges have been set to keV but a redistribution matrix hasn't been supplied.

**MoreThanOneStatMethod** (*error*)

Only one statistical method, e.g. minimum counts, minimum S/N or ratio above background, can be specified

**InvalidRangeString** (*error*)

The ranges must be entered as '1-10' or '1:10' if specified in channels or as '2.0:4.53' or '2.0-4.53' if specified in keV.

**EnergyOutOfRange** (*error*)

The specified energy range is outside the range of energies contained in the spectrum

**RegBinRange** (*error*)

The start bin for regular binning must be  $i=0$ .

**IncompatibleTemplate** (*error*)

The template spectrum has a different number of channels from spectrum to be grouped.

**AmbiguousGrouping** (*error*)

There has been a clash between the ranges and regular binning requested.

**AlreadyGrouped** (*warning*)

A bin which has been requested to be set bad by the `setbad` option has already been grouped and so will stay as a good channel.

*corrective action:* leave channel as good

**NoMergeGroup** (*warning*)

After grouping has completed and ungrouped channels are treated according to the `lastbin` parameter selection. Sometimes it is impossible to find a contiguous group to add a given channel to. Contiguous channels which fall in this category are used to form a new group.

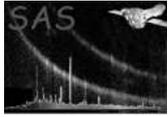
*corrective action:* create a new group

## 9 Input Files

- an EPIC spectrum consisting of counts vs P.I. channel.
- a redistribution matrix (RMF) corresponding to the source in question
- an optional EPIC background spectrum consisting of counts vs P.I. channel.

## 10 Output Files

- A spectrum with grouping applied. The grouping is specified in the column `GROUPING` where a number 1 indicates the first channel in a group and -1 indicates a member of the current group. The quality of each channel is given in the `QUALITY` column; where 0 = good quality, 1 is bad quality and 2 means dubious.



## 11 Algorithm

Set the grouping ranges using the following order

1. User defined ranges
2. Regular user defined bins
3. Set bad, anything defined as bad

With whatever is left, if set, group with one of the following

- a) Minimum no of counts
- b) Min S/N
- c) Min ratio above background

Convolve these with the options

- i) Minimum bin width as fraction of energy resolution
- ii) Start from low or high energy end
- iii) Treatment of last bin

Sort groups

## 12 Comments

## References