NAME

enen-evts - generate enclosed count fraction and radial count density of an object

SYNOPSIS

enen-evts *[options] FITS event filename*

DESCRIPTION

enen-evts generates the enclosed count fraction (the fraction of the total counts in circular apertures as a function of aperture radius) and a one dimensional radial count density profile from a set of input event coordinates and weights. Background count densities may be determined from the data or may be explicitly specified.

If the input data consist of a single bright, concentrated, source **enen-evts** should be able to find the source, determine the center, and generate its products with little user interaction. If the source field is more complex, the user will need to provide information to help locate the source to be analyzed and limit confusion with other nearby sources.

The input data should be formatted as a FITS binary table with two columns describing event positions. A third column containing event weights may be present.

The output data are detailed in the *OUTPUT* section.

Enclosed Count Fraction Determination

The program moves through the following stages:

1.   Data input

The event locations and (optionally) weights are read from the input data file. The **--rfilter** option may be used to filter the events.

1.   The center of the source is determined.

By default, an iterative sigma-clipping algorithm is used to determine the source center. An explicit center may be provided by specifying the **--force_ctr**, **--xc**, and **--yc** options.

The iterative algorithm works by determining the weighted average of events within a circular aperture whose center is that determined by the previous iteration. The aperture's radius is calculated from the standard deviation of the radii of the events measured from the center,

```
nsigma * std_dev
```

where *nsigma* is provided by the **--nsigma** option.

The algorithm converges when the change in the determined center between two iterations is less than the absolute tolerance specified via the **--dtol** option..

An initial center and clipping radius are required. The radius is specified via the **--clip** option. By default the initial center is determined from the weighted average of the event positions.

The initial center may also be explicitly specified via the **--xc** and **--yc** options, or may be determined from the data using the **--pixcent** option.

When **--pixcent** is specified, **enen-evts** bins the data into a low resolution image and uses the location of the brightest pixel as the initial center. This process works best if there is a single bright, concentrated source. **--pixcent** takes an optional integer value which specifis the width of the (square) image in pixels; this defaults to 256 pixels.

The object, center, and the final clipping radius may optionally be displayed via **ds9** by specifying **--display**.

2.   The events are sorted by distance from the determined center.

A final clipping of the data may be performed by specifying a maximum allowable radius via the **--rmax** option.

3.   The background count density is determined.

The **--bgsub** option specifes the manner in which the background density is determined. It may take one of the following values:

`none`

> The background is set to `0`. This is the default.

number

> The background is set to the specified value, in counts per scaled squared input unit.

`ave`

> The background is determined from the count density in the annulus specified via the **--bgmin** and **--bgmax** parameters.
>
> The algorithm does not adjust the area if the annulus extends beyond the detector boundaries. Make sure that the annulus is fully on the detector!

`fit`

> The background is determined by fitting the cumulative sum of the events in the annulus specified via the **--bgmin** and **--bgmax** parameters with the function `K + bg * PI * r^2`. The events may be binned into bins with a number of counts using the **--fitbin** option.
>
> The algorithm does not adjust the area if the annulus extends beyond the detector boundaries. Make sure that the annulus is fully on the detector!

`iterfit`

> The background is determined by fitting the cumulative sum of the events in the annulus specified via the **--bgmin** and **--bgmax** parameters with the function `K + bg * PI * r^2`. The events may be binned into bins with a number of counts using the **--fitbin** option.
>
> This option differs from `fit` in that the annulus is divided into **--bgnstep** (possibly overlapping) regions of width **--bgwidth**, running from **--bgmin** to **--bgmax**. The background is fit in each of those regions, the resultant source count vs. radius curve is fit by a spline, and the background value is interpolated at the radius corresponding to the maximum of the splined source counts.
>
> The source count background value vs. region position curves are plotted.
>
> The algorithm does not adjust the area if the annulus extends beyond the detector boundaries. Make sure that the annulus is fully on the detector!
>
> (Yes, this is a kludge.)

4.   Sum up the background subtracted counts within concentric circular apertures. By default the radii of the apertures are set to the event radii. If the **--bgbin** option is specified, the radii are chosen so that there is a minimum increase in counts in each successive aperture.

Without binning (and even with binning) the background subtracted sums may *not* be monotonically increasing with radii in regions where the source event density is comparable to or

less than the background event density. This may be the result of too high a background rate, or due to statistical fluctuations in regions where source and background densities are comparable.

Binning the sums before background subtraction provides some protection against that noise, permitting one to delve deeper into the PSF wings.

5. Optionally "correct" the background subtracted sums.

    The background subtracted sums may still not be montonically increasing even after pre-subtraction binning. The **--enenmax** option provides several algorithms to "correct" the sums to ensure that they are motionically increasing.

    - truncation at a given radius (`--enenmax=`*float*)

    - truncation at the first maximum in the background subtracted cumulative sum (`--enenmax=fmax`)

    - deletion of bins where the background exceeds the source (`--enenmax=dneg`)

      Since the sums are cumulative, this doesn't actually change the sum, unless the last bin is deleted. What it effectively does is to increase the bin size in the noisy regions where the background exceeds source. Of course, if the background rate is too high, it'll over subtract the wings and all of the bins beyond the turn-over point will be deleted. But that's actually a feature, not a bug.

6. The sums may be binned to have a minimum increase of **--bin** events per bin.

    These sums are the *base enclosed count data*.

7. The sums are normalized by the number of source events, which is either the maximum of the "corrected" sums, or if the background has been fit (see **--bgsub**), by the fit source counts.

These normalized sums are the *base enclosed count fractions*.

Enclosed Count Fraction Output

Typically the user would like the enclosed count fraction enumerated at specific fractions, at specific radii, or at whatever comes out of the above algorithm (which is affected by the **--bgbin** and **--bin** options). **enen-evts** will produce *one or more* of these outputs, which are under control of the **--enenf**, **--enenr**, and **--enenrbin** options.

Radial Profile output

The radial profile of the source is generated by differentiating the enclosed count fraction data. *One or more options* may be used to specify the radii at which to determine the profile: **--profr**, **--profrbin**, **--profnt**, and **--profnbin**.

OPTIONS

enen-evts uses long-style options. Options which take values may be separated from the values by whitespace or the = character.

Input Options

--extname=*string*

> This specifies the extension HDU in the FITS input file from which to extract events. By default the first HDU containing a table is read.

--rfilter=*row filter*

> A CFITSIO compatible filter expression to weed out unwanted rows.

--xcol=*string*

This specifies the *X* coordinate column in the FITS file. It defaults to x.

--ycol=*string*

This specifies the *Y* coordinate column in the FITS file. It defaults to y.

--wtcol=*string*

If specified, event weights should be read from the given column in the FITS file. If not specified, events all have unit weight.

Clipping, Centering, and Scaling Options

--clip=*float*

This specifies the radius for initial clipping of the data when **enen-evts** is performing an iterative centroid.

This must also be specified if **--force_ctr** is specified and **--display** is specified (else it isn't possible to know how large to make the image).

--force_ctr

If true, the center specified by the **--xc** and **--yc** options will be used as the center. The **--bgmin** and **--bgmax** parameters must be specified in absolute units (rather than as a multiple of the image standard deviation, as this will not be determined).

--dtol=*float*

Absolute convergence tolerance for the centering algorithm.

--iter=*integer*

This specifies the number of iterations the centering routine should perform.

--nsigma=*float*

This specifies the clipping radius, in units of standard deviation, used during the iterative centering stage.

--xc=*float*

If set, this specifies an initial value (in scaled units) for the *X* coordinate of the center of the object. See **--scale**.

--yc=*float*

If set, this specifies an initial value (in scaled units) for the *Y* coordinate of the center of the object. See **--scale**.

--pixcent

If specified, choose the brightest pixel in the image as the initial center. This option optionally takes an integer argument specifying the image size in pixels (the image is square). This defaults to 256.

--rmax=*float*

After determining the center, clip all events outside of this radius (in scaled units).

Background Subtraction

--bgbin=*nevents*

If specified, the cumulative sum is binned before being background subtracted so that there is a minimum increase of *nevents* per bin.

--bgsub=*background_type*

> This option specifies which background is subtracted. It may one of `fit`, `ave`, `none`, `iterfit`, `region`, *number*. The default is `none`. See above for more details.

--bgmin --bgmax

> These take arguments which specify the inner and outer radii of the annulus in which to measure the background. The radii are specified in position units, unless they have a suffix of `s`, in which case they are specified in units of the standard deviation of the event distribution. If not specified, it defaults to `10s,11s`.

--fitbin=*nevents*

> If specified, the cumulative sum is binned before being fit so that there is a minimum increase of *nevents* per bin.

Enclosed Count Fraction manipulation

> The following options allow manipulation of the cumulative sum after background subtraction.

--bin=*nevents*

> Bin the background subtracted enclosed count fraction data into bins which have a minimum increase of *nevents* per bin. This occurs after the sum is "corrected".

--enenmax

> This option specifies how the sum is "corrected" before being normalized. It takes one of the following values:

> *floating point number*

>> The number indicates the radius at which the sum should be truncated. The exact comparison is `<`, not `<=`.

> `fmax`

>> This indicates that the sum should be truncated at its first maximum.

> `dneg`

>> This indicates that bins where the background exceeds the source should be deleted.

> It defaults to `dneg`.

Output Options

--tag

> This specifies a string which will prefix all output files. It is required. The enclosed count fraction data will be written out to the file *${tag}_enen.rdb*. The radial profile will be written out to the file *${tag}_prof.rdb*.

Enclosed count fraction output

> If none of these options is specified, the default behavior is akin to running with the following options:

```
--enenf=0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.85,0.9,0.95,0.99
```

> More than one means of specifying the output enclosed count fraction grid may be specified.

--interpolate|--nointerpolate

> Interpolate the data to determining the enclosed count fraction. Defaults to false.

See the other options for more information on how interpolation affects their behavior.

--enenf=*list*

This specifies a list of fractional encircled energies for which the corresponding radius should be determined. See *Specifying lists of numbers* for the valid format of the list.

This is possible only if the enclosed count fraction data are montonically increasing. If interpolation is turned off, the greatest fractions less than the requested ones are output.

--enenr=*list*

This is a list of radii for which the corresponding enclosed count fraction fraction should be determined. See *Specifying lists of numbers* for the valid format of the list.

Interpolation to the exact radii is meaningful only if the encircled energy fractions are monotonically increasing with radii. If they are not, a warning is output, but the interpolation is performed.

If interpolation is not requested, the greatest radii less than the requested ones are used.

--enenrbin

If specified, output the base enclosed count fraction fractions as is. The **--bgbin** or **--bin** options effectively specify the bin width.

Radial profile output

Unless otherwise specified below, the radial profile is derived from the derivative of the background subtracted enclosed count data (the result at step 4, above).

If none of these options is specified, the default behavior is akin to running with the following options:

```
--profnbin=10
```

--profr=*list*

This specifies the outer radii of the annuli which will be used to determine the radial profile. The **--profr-opts** option specifies how those radii are to be interpreted.

See *Specifying lists of numbers* for the valid format of the list.

--profr-opts=*list*

This option specifies how the radii specified by **--profr** are interpreted. It takes a comma separated list of options.

If not specified, it defaults to interpolate.

The following options are recognized:

ecfbin

Radii are converted to the radii of the next largest bin in the uncorrected, background subtracted enclosed count data.

interpolate

Interpolate the uncorrected background-subtracted enclosed count data onto the specified radii.

keep

Normally any trailing and empty bins are removed from the output. This option keeps them.

--profrbin

Use the corrected and rebinned base enclosed count data (step 6, above). The **--bgbin** or **--bin** options effectively specify the bin width.

--profnt

Use the uncorrected background subtracted enclosed count data. The **--bgbin** option effectively specifies the bin width. This is rarely useful without some measure of binning.

--profnbin=*integer*

Begin with the uncorrected, background subtracted enclosed count data. Negative going bins are discarded (similar to setting **--enenmax** to `dneg`) and the data are rebinned such that each bin has at least the specified number of events.

Plot and Image Display Options

--plot

If specified, plots are produced. If not, plots are not produced.

--device=*PGPLOT device*

The PGPLOT device to which plots should be output. Some useful devices are:

`/xs`

This is the default, and plots to a persistent X Windows window. To have each plot show up in a separate window, use `+/xs`.

`/cps`

Generate color PostScript output sent to files.

--id=*int*

The PGPLOT `/xs` device id to start with (several are required).

--display

If specified, an image of the source and final clipping region is displayed using **ds9**.

--log_enen | **--nolog_enen**

Take (or don't take) the log of the radius when plotting the encircled energy function. Defaults to **--log_enen**.

--pix_size=*float*

The size of a pixel in scaled units. Defaults to the value of **--scale**. This is only used when displaying the centered image with **ds9**. See **--scale**.

--scale=*float*

This specifies a scale factor to be applied to the event positions (they are multiplied by this factor). See **--unit**. It defaults to 1.

--unit=*unit*

This specifies positional units. It defaults to `pixel`.

Miscellaneous Options

--version

Print the version and exit.

--help

> Print a short help message and exit.

--usage

> Print a longer help message and exit.

Specifying lists of numbers

Several options (**--enenf**, **--enenr**, **--profr**) have values which are used to generate lists of numbers.

If the first character of the value is the @ character, the rest is the name of a file containing numbers, one per line. Otherwise, the value should be a comma separated list of numbers.

Numbers may be either simple (just a floating point number), or may describe a sequence. Sequences may have one of the following forms:

```
[min]:max:[op]step
[min]:max:[op]step:nstep
```

where elements in square brackets are optional. If *min* is omitted, the last number (either generated or provided by the user) is used as the starting point. *op* may be one of -, +, or *. If it is omitted, it defaults to +. The sequence ends either when the number exceeds max or when nstep numbers are generated.

Some examples and their output:

```
1:10:1   => 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
1:10:1:5 => 1, 2, 3, 4, 5
1:10:*2  => 1, 2, 4, 8
10:1:-1  => 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
```

OUTPUT

Plots

The following plots are produced:

Fractional enclosed count fraction

> The uncorrected, background subtracted enclosed count fraction is plotted, along with the annular background regions (see **--bgmin** and **--bgmax**). Hardcopy is written to files with the prefix ${tag}_enen.

Cumulative Sum

> The cumulative, non-background subtracted sum is plotted, as well as the source+background fit if specified with **--bgsub**. Hardcopy is written to files with the prefix ${tag}_sum.

Fit residuals

> The residuals to the fit to the source+background are plotted, if the background was fit (via **--bgsub**). Hardcopy is written to files with the prefix ${tag}_resid.

Iterative Fit Plots

> If iterative fitting is requested, plots of the fits are generated. Hardcopy is written to files with the prefixes ${tag}_bg_iter_cts and bg_iter_bg.

Radial Profile

> The requested (via **--profr**, **--proft**, and **--profnt** radial profiles are plotted. Hardcopy is written to files with the prefix ${tag}_prof.

Data Files

The enclosed count fraction and radial profile data are written out as RDB tables.

enclosed count fraction data

The RDB table has the following columns

fraction

The enclosed count fraction

radius

The radius of a circle enclosing the fractional energy.

counts

The background subtracted number of counts inside the circle enclosing the fractional energy.

error

The statistical error in the enclosed count fraction. This does not take into account the error in the background determination. If events are weighted, this does not take into account the background at all.

req

The option which resulted in the output data. It is one of

```
enenf enenrbin enenr
```

Radial Profile data

The RDB table has the following columns

surfbri

The background subtracted surface brightness

surfbrierr

The statistical error in the surface brightness

radius

The average radius of the annulus

rmin, rmax

the lower and upper bounds of the annulus

counts

The counts used to determine the surface brightness.

req

The option which resulted in the output data. It is one of

```
profr proft profnt
```

Examples

```
enen-evts --profr=0.01:20:1.1:60 \
    --force_ctr --xc=16267.5 --yc=16443.5
```

```
        --tag=test --verbose \
        --bgmin=500 --bgmax=600 --bgsub=fit hrcf01385N004_evt2.fits


    enen-evts --tag=test \
        --bgmin=200 --bgmax=400 --bin=500 --profrbin --enenrbin \
        --scale=7.59 --unit=arcsec --clip=10 --fitbin>=50 \
        hrcf01385N004_evt2.fits


    enen-evts --tag=sim --verbose --bgsub=none \
        --pix_size=0.024 --dtol=0.001 --unit=mm  \
        --extname=raytrace --xcol=rt_y --ycol=rt_z \
        --display \
/data/fifi/dj/axaf/studies/PSF/arlac/01385/2001-04-23/many/rays/096_sim.fits
```

AUTHOR

Diab Jerius (djerius@cfa.harvard.edu)