

GammaLib - Bug #1151

Level of fitted background

02/24/2014 01:48 PM - Martin Pierrick

Status:	Closed	Start date:	02/24/2014
Priority:	Normal	Due date:	
Assigned To:	Knödlseider Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
<p>When using a GCTAModellInstBackground in an analysis of a single point source, it looks like the background is not fitted at the right normalization (which I expected to be 1). The level is something like 1.02-1.03, which is significantly above the statistical fluctuations (I always got it >1 anyway, never something like 0.99). It may be connected to another issue that arises when using a ROI radius of 5deg, which is the same size as the radius over which the background is defined. Then, the fit is not accurate, with delta between observed and predicted counts of several 1000.</p> <p>Attached is the log of two runs: one with a ROI size of 4deg (top of the file) and one with a ROI of 5deg (bottom of the file).</p>			
Related issues:			
Related to GammaLib - Bug # 1198: Incorrect results for fitting of diffuse mo...		Closed	05/13/2014 07/01/2014

History

#1 - 02/28/2014 03:49 PM - Knödlseider Jürgen

I see that you have set the pivot energy to 1 MeV. Although the index is fixed to 0, I'm wondering whether this can lead to some numerical problems. Can you set the pivot to the TeV domain, or better use a simple spectral constant for the normalization?

#2 - 05/13/2014 11:12 AM - Martin Pierrick

- Assigned To changed from Knödlseider Jürgen to Martin Pierrick

#3 - 05/13/2014 05:40 PM - Martin Pierrick

- Status changed from New to Resolved

- Assigned To changed from Martin Pierrick to Knödlseider Jürgen

- % Done changed from 0 to 90

I did some additional tests:

- One pointing of 10,50,100h
- One point source in the field
- Unbinned analysis over 1-10TeV, also tried 100GeV-10TeV and 1-100TeV
- IRF .../data/cta/aar/bcf/DESY20140105_50h/irf_file.fits
- 50 fits in each case
- code version devel/bd092b9

Results are (model 1 is the point source):

True parameters (first background, then point source): [1.0, 0.0, 8.2e-19, -2.8]

100GeV-10TeV / 50h

Mean/deviation of background normalisation: 9.999464e-01 (9.115832e-04)
Mean/deviation of background index: 0.000000 (0.000000)
Mean/deviation of model 1 normalisation: 8.178346e-19 (1.263012e-20)
Mean/deviation of model 1 index: -2.801564 (0.011369)

1-100TeV / 50h

Mean/deviation of background normalisation: 1.000739e+00 (4.738377e-03)
Mean/deviation of background index: 0.000000 (0.000000)
Mean/deviation of model 1 normalisation: 8.105815e-19 (2.989788e-20)
Mean/deviation of model 1 index: -2.789380 (0.038785)

1-10TeV / 50h

Mean/deviation of background normalisation: 1.001022e+00 (4.984124e-03)
Mean/deviation of background index: 0.000000 (0.000000)
Mean/deviation of model 1 normalisation: 8.163553e-19 (3.259923e-20)
Mean/deviation of model 1 index: -2.800814 (0.046734)

I don't see the problem anymore. There is no impact pivot energy of the background (1MeV or 1TeV, this is not used anyway). Not sure about what changed in the code since when I reported the problem. The mean parameters of the point source converges towards the true values if I increase the observing time:

1-10TeV / 100h
Mean/deviation of background normalisation: 1.000153e+00 (2.728596e-03)
Mean/deviation of background index: 0.000000 (0.000000)
Mean/deviation of model 1 normalisation: 8.201161e-19 (2.412416e-20)
Mean/deviation of model 1 index: -2.806473 (0.036199)

So overall, it looks like it is OK. I have not done more test than that but the case can probably be closed.

#4 - 05/26/2014 04:44 PM - Martin Pierrick

- File *Unbinned_fits_50h.png* added
- File *Binned_fits_50h.png* added
- File *Binned_file_50h.png* added
- Status changed from *Resolved* to *In Progress*
- Assigned To changed from *Knödlseder Jürgen* to *Martin Pierrick*
- % Done changed from 90 to 10

FORGET ABOUT THE PREVIOUS POST !

There must have been something wrong in the previous tests and I probably tested *GCTAModelRadialAcceptance* with ASCII IRF definition instead of *GCTAModelInstBackground* with FITS definition. With appropriate tests, I confirm there is an offset.

The setup now is:
- One pointing of 50h
- No source in the field
- Unbinned analysis over 1-10TeV
- ... also tried binned with 5 bins
- IRF *.../data/cta/aar/bcf/DESY20140105_50h/irf_file.fits*
- 50 fits in each case
- code version *devel/bd092b9*

The output is :
Mean/deviation of background normalisation: 1.004012e+00 (2.674310e-03)
... and the distribution of the 50 runs is shown in *Unbinned_fits_50h.png*

The situation is even worse for a binned analysis:
Mean/deviation of background normalisation: 1.020184e+00 (3.195677e-03)
... and the distribution also is worse, see *Binned_fits_50h.png*

Comparing that to other ways to set up the background:
- A gaussian for radial dependance and a power-law for spectrum: works perfectly for both binned and unbinned (tested only 50h)
- A gaussian for radial dependance and an ASCII file for spectrum: works for unbinned but not for binned (tested only 5 bins)
For the latter case, see plot *Binned_file_50h.png*

More testes needed...

#5 - 06/02/2014 05:31 PM - Martin Pierrick

- File *Binned_file_50h_12bins.png* added
- Status changed from *In Progress* to *Resolved*
- Assigned To changed from *Martin Pierrick* to *Knödseder Jürgen*
- Priority changed from *High* to *Normal*
- % Done changed from *10* to *90*

Following up on that...

The test of background normalization when using a gaussian for radial dependance and an ASCII file for spectrum turns out to work well provided there are enough energy bins (something like 10-12 at least). See plot *Binned_file_50h_12bins.png*.

The problem of background normalisation when using *GCTAModellInstBackground* with the current set of instrument performances remains. It is most likely due to the coarse spatial binning of these responses, which leads to problems because the events simulation and the model-fitting handle this background differently: the former uses a cumulative distribution function, which has steps, while the latter uses an interpolation. When bins are too large, it is not surprising that the same number of counts is not obtained because there is no reason for it. A possible solution would be to finely rebin the background spatial distribution using the same function that is used for interpolation in model-fitting. Or wait for more accurate performance files.

#6 - 06/06/2014 03:58 PM - Knödseder Jürgen

- Status changed from *Resolved* to *In Progress*

Put this back into "In Progress" as Stefan Ohm was also stumbling over this in his simulations of Arp220. We should really find a fix for this.

#7 - 06/06/2014 04:00 PM - Knödseder Jürgen

- File *simArp220.py* added

Attached Stefan's script that nicely produced the problem: attachment:*simArp220.py*

#8 - 06/12/2014 04:37 PM - Knödseder Jürgen

Stefan wrote:

Dear Juergen,

thanks for looking into this. I can confirm your numbers for the IFAE response files when leaving all parameters free. Indeed the > background normalisation is $\sim 0.9 \pm 0.001$ in this case. For the DESY/aar responses, however, we are off by a significant number (norm 1.35 ± 0.08 , index -2.72 ± 0.05 , background 0.82 ± 0.001).

Just let me know if I can help with testing.

Cheers, Stefan

I could confirm these values. One of the differences is the resolution of the background file. For DESY it is $14 \times 14 \times 21$ bins (in DETX, DETY, energy) for IFAE it is $32 \times 32 \times 20$ bins. The actual code makes a difference between Monte Carlo simulations (which spatially distributes the events evenly within a bin) while the model interpolates smoothly. Thus with a coarse spatial binning the situation will get worse.

One way to circumvent the problem would be to spatially rebin the background maps before using them. Implementation of feature #1212 would be needed for this.

#9 - 06/12/2014 06:28 PM - Knödlseider Jürgen

- File *Arp220-simulations.xlsx* added

I recognized that the GCTAModelBackground constructor has optional parameters that allow for a rebinning. By default these parameters are 0, but you may add nx, ny, nenergies to the constructor to get a finer internal pixelisation of the background model, e.g.

```
bck = gl.GCTAModelBackground(ob,filename,spec,100,100,100)
```

I did a couple of tests and come close to the true values. Higher resolution in nx and ny improves the source parameters, higher resolution in nenergies the background model parameters (see attached excel file).

#10 - 06/12/2014 06:30 PM - Knödlseider Jürgen

I also recognized that Stefan uses GCTAModelBackground while in the future GCTAModelInstBackground should be used that fetches directly the response from the IRF file. I still need to look how rebinning can be implemented for that case.

#11 - 06/12/2014 06:39 PM - Knödlseider Jürgen

- File *Arp220-simulations.xlsx* added

#12 - 06/12/2014 06:40 PM - Knödlseider Jürgen

- File deleted (*Arp220-simulations.xlsx*)

#13 - 06/13/2014 11:00 AM - Knödlseider Jürgen

- File *simArp220_new.py* added

- File *Arp220-10h-simulations.xlsx* added

- Status changed from *In Progress* to *Feedback*

- % Done changed from 90 to 100

I now implemented an internal rebinning to a finer grid into GCTAModelInstBackground. This gave reasonable results for the Arp220 test case. Attached an excel file and the test script used. Code is in the devel branch.

Please provide feedback and tell me if this also works for you.

#14 - 07/10/2014 11:48 PM - Knödlseider Jürgen

- Status changed from *Feedback* to *Resolved*

#15 - 07/20/2014 11:22 PM - Knödlseider Jürgen

- Status changed from *Resolved* to *Closed*

Files

BgdFitIssue.log	36.6 KB	02/24/2014	Martin Pierrick
Unbinned_fits_50h.png	19 KB	05/26/2014	Martin Pierrick
Binned_fits_50h.png	19 KB	05/26/2014	Martin Pierrick

Binned_file_50h.png	18.8 KB	05/26/2014	Martin Pierrick
Binned_file_50h_12bins.png	18.7 KB	06/02/2014	Martin Pierrick
simArp220.py	3.56 KB	06/06/2014	Knödseder Jürgen
Arp220-simulations.xlsx	41.9 KB	06/12/2014	Knödseder Jürgen
simArp220_new.py	3.4 KB	06/13/2014	Knödseder Jürgen
Arp220-10h-simulations.xlsx	50.7 KB	06/13/2014	Knödseder Jürgen