

GammaLib - Bug #2663

GCTAOnOffObservation::apply_ebounds() cut on rmf gives strange spectral model fit results

08/16/2018 01:25 PM - Specovius Andreas

Status:	Closed	Start date:	08/16/2018
Priority:	Normal	Due date:	
Assigned To:	Knödlseeder Jürgen	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	1.6.0		
Description			
It seems like the method GCTAOnOffObservation::apply_ebounds() which accounts for the observations energy boundaries does something strange when cutting on the rmf.			
The result of a spectral plaw model fit to pks2155 HESS PDR data seem to fit well our expectations for deactivated cut on rmf, but deviates for activated cut on rmf:			
Reference (hess tool, gammapy) joint, activated cut on rmf joint, deactivated cut on rmf			
-----+-----+-----+-----			
Norm	1.802 +- 0.212 e-11	2.236 +- 0.242 e-11 (+24%)	1.814 +- 0.213 e-11 (+0.7%)
Index	3.429 +- 0.269	3.758 +- 0.265 (+10%)	3.393 +- 0.268 (-1.1%)
I am using ctools/gammalib dev Aug 04 version.			

History

#1 - 08/16/2018 01:38 PM - Specovius Andreas

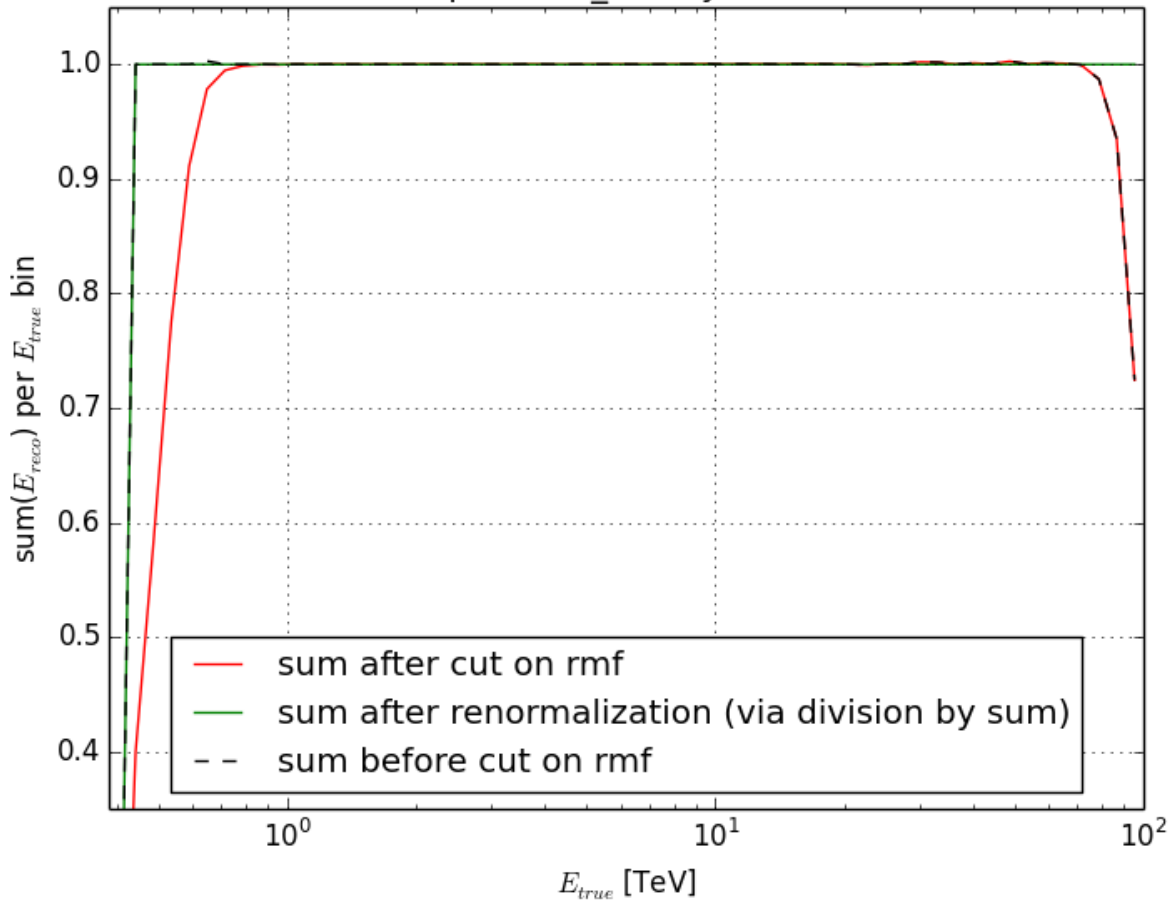
- File 47803.png added

May it be a problem, that the rmf is not re-normalized to unity after the cuts are applied?

For testing purposes I calculated a sum in the way shown below. The sums for different locations in the code are shown in the plot.

```
for (int itrue=0; itrue<ntrue; ++itrue) {
  double sum = 0;
  for (int ireco=0; ireco<nreco; ++ireco) {
    sum += m_rmf(itrue, ireco);
  }
}
```

HESS PDR pks2155_steady run047803 rmf



If I apply a quick renormalization via dividing per etrue bin by the just calculated sum the fit result enhances a bit:

| Reference (hess tool, gammapy) | joint, activated cut on rmf, renormalized

Norm	1.802 +- 0.212 e-11	1.957 +- 0.210 e-12 (+9%)
Index	3.429 +- 0.269	3.568 +- 0.256. (+4%)

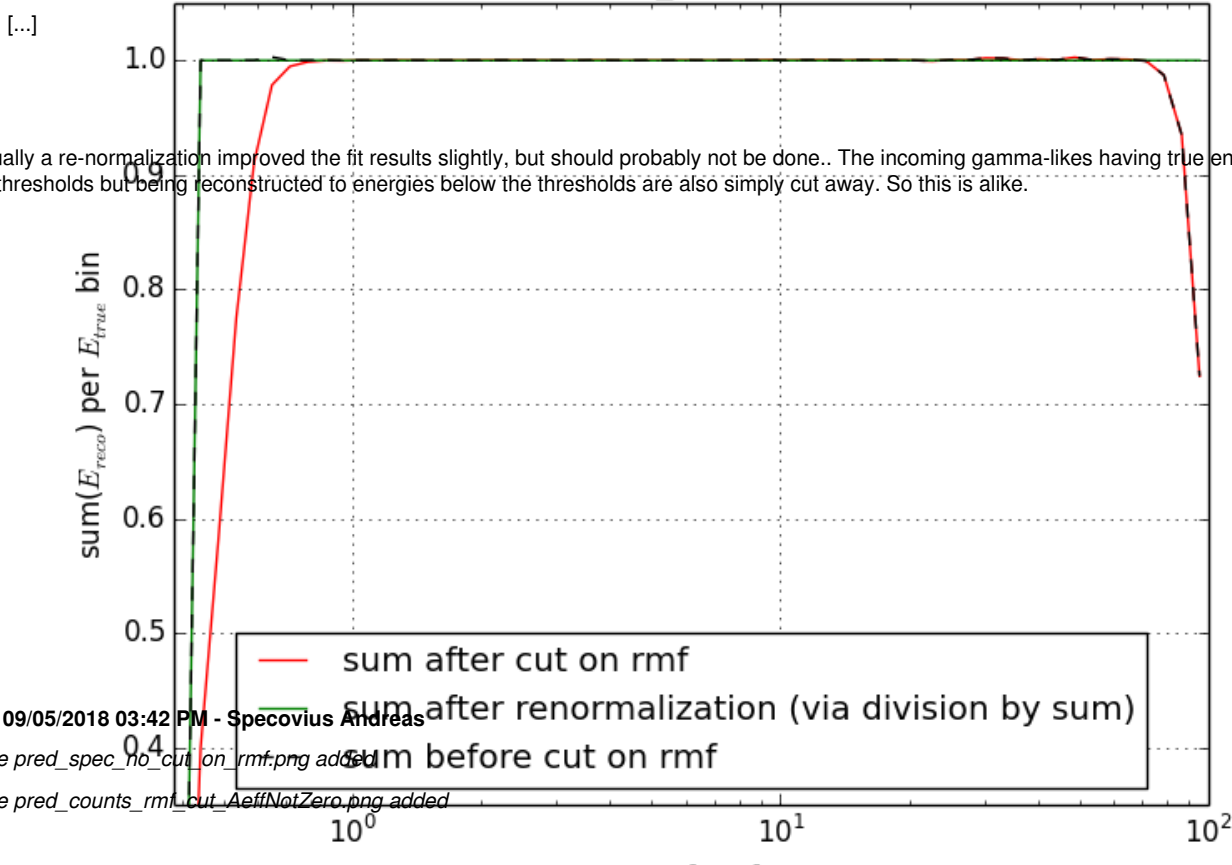
#2 - 09/04/2018 02:44 PM - Specovius Andreas

user#291 wrote:

May it be a problem, that the rmf is not re-normalized to unity after the cuts are applied?

For testing purposes I calculated a sum in the way shown below. The sums for different locations in the code are shown in the plot. [...]

If I apply a quick renormalization via dividing per true bin by the just calculated sum the fit result enhances a bit:



Actually a re-normalization improved the fit results slightly, but should probably not be done.. The incoming gamma-like having true energies above the thresholds but being reconstructed to energies below the thresholds are also simply cut away. So this is alike.

#3 - 09/05/2018 03:42 PM - Specovius Andreas

- File pred_spec_no_cut_on_rmf.png added
- File pred_counts_rmf_cut_AeffNotZero.png added

I think the underlying problem is something different and disabling the cut on the rmf and/or renormalizing the energy dispersion are just something that patched the actual issue (maybe by chance to reasonable fit results):

We discussed in our group here in Erlangen and we think that the effective area should not be zero beyond the energy threshold (at least for the case where energy dispersion is enabled!). As this seems to be a different major topic I opened a new issue: <https://cta-redmine.irap.omp.eu/issues/2674>

A zero effective area beyond the threshold prevents the model flux with true energies below the threshold from contributing to the predicted counts above and in the vicinity of the threshold although their probability of being reconstructed to energies above the threshold may be >0 (depends on specific edisp!).

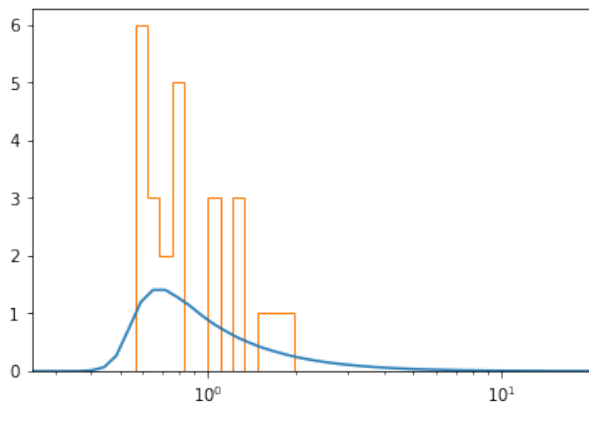
This would mean that the predicted counts in the vicinity of the threshold are systematically underestimated.

why did renormalization improve results?

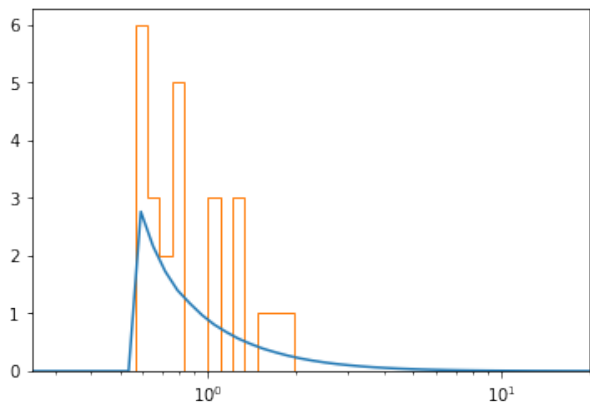
Hence, indeed renormalization of the energy dispersion after the cut on the rmf was applied improves the fit results as the predicted counts in the vicinity of the thresholds, which are underestimated, are "manually" scaled up.

why did disabling the cuts on the rmf improve results?

This is pretty sure due to the full contribution of the energy dispersion to the predicted flux also beyond thresholds. See the following plot showing the predicted flux expanding beyond thresholds when the cut on the rmf is not done (here, effective area is zero beyond thresholds, how it is implemented in ctools).



For comparison the following plot shows the predicted flux when the cut on the *rmf* is done but the effective area is not set to zero beyond thresholds.



#4 - 09/05/2018 04:11 PM - Knödseder Jürgen

Thanks for doing this detailed study. I have current a few important things to do on my desk, and the CTAC meeting is coming closer, but I will help you with that as soon as I find some time.

#5 - 09/10/2018 04:20 PM - Knödseder Jürgen

- *Assigned To set to Knödseder Jürgen*
- *Target version set to 1.6.0*

I fixed the application of the energy threshold in #2674. Does this also fix this issue?

#6 - 09/11/2018 08:57 AM - Specovius Andreas

Indeed. This issue seems to be closed!

#7 - 09/11/2018 09:39 AM - Knödseder Jürgen

- *Status changed from New to Closed*
- *% Done changed from 0 to 100*

Files

47803.png	50.4 KB	08/16/2018	Specovius Andreas
pred_spec_no_cut_on_rmf.png	6.1 KB	09/05/2018	Specovius Andreas
pred_counts_rmf_cut_AeffNotZero.png	6.53 KB	09/05/2018	Specovius Andreas