

ctools - Bug #1439

ctobssim throws incomprehensible exception when using a binned observation as input

03/10/2015 04:54 PM - Mayer Michael

<b>Status:</b>	Closed	<b>Start date:</b>	03/10/2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assigned To:</b>	Mayer Michael	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	3.00 hours
<b>Target version:</b>	1.0.0		
<b>Description</b>			
<p>Currently, ctobssim is not intended to be used with a binned observation as input. Another tool is planned to achieve that (#544).</p> <p>Forgetting about this rule and running ctobssim with a count map as input, the error message does not reflect the actual error and appears cryptic. This should be changed.</p> <p>Here is my command line output:</p> <pre>mimayer ~/Software/ctools \$ ctobssim inobs=test/data/crab_cntmap.fits.gz Calibration database [dummy] Instrument response function [cta_dummy_irf] Model [\$CTOOLS/share/models/crab.xml] Output event data file or observation definition file [events.fits] *** ERROR encountered in the execution of ctobssim. Run aborted ... *** ERROR in GCTAEventCube::naxis(int): CTA event cube axis 208182906 is outside the valid range [0,2].</pre>			

History

#1 - 06/22/2015 10:40 PM - Knödlseider Jürgen

- Target version set to 1.0.0

#2 - 06/25/2015 04:28 PM - Knödlseider Jürgen

- Project changed from GammaLib to ctools

- Target version deleted (1.0.0)

#3 - 06/30/2015 11:56 AM - Knödlseider Jürgen

- Target version set to 1.0.0

And check this for all other tools!

#4 - 06/30/2015 02:36 PM - Mayer Michael

- Status changed from New to In Progress

- Assigned To set to Mayer Michael

#5 - 06/30/2015 02:58 PM - Mayer Michael

I think the ctools and cscripts can be separated into two kinds. There are tools that require unbinned observation as input and some that should be able to run on both (binned and unbinned):

Unbinned only:

- ctselect
- ctobssim
- ctexpcube
- ctpscube
- ctbkcube

- ctbin
- ctskymap
- csspec
- cslightcrv
- cspull
- cssens
- cstdist

Both binned and unbinned:

- ctbutterfly
- ctubemask (binned only?)
- cterror
- ctlike
- ctmodel
- cttsmap
- ctulimit
- csresmap

We should verify that each tool is able to run in the modes listed above. In addition, the error message for a wrong input observation type should be changed to be reasonably understood by a user.

**#6 - 06/30/2015 03:19 PM - Knödseder Jürgen**

Mayer Michael wrote:

- ctubemask (binned only?)

Yes

**#7 - 06/30/2015 06:59 PM - Mayer Michael**

We decided to simply skip binned observations in tools where unbinned observations are necessary and write that as warning into the logger. This has now been implemented for the following tools:

- ctselect
- ctobssim
- ctpscube
- ctexpcube
- ctbgcube
- ctskymap
- ctbin

The corresponding cscripts don't have to be changed as they just call some of the ctools above.

The tool `ctcubemask` is fine already as it skips unbinned observations already.  
In addition, tools working on the likelihood profile also seem to be alright in this matter as they can take both inputs.

Two tools are a bit different to handle:

1. `ctmodel`: First, I noticed that there is a `get_obs()` function which basically does the same as `ctool::get_observations()`. Therefore, I removed this function and use the generic `ctool` function instead. The other question is if we really want `ctmodel` to work on binned observations, which it currently does. However, it would then not be aligned with `ctbin`, which only creates a cube from the unbinned observations. Therefore the creation of a residual map could become a pain. If `ctmodel` just adds up all the model counts including binned observation (which is not considered in `ctbin`), the diagnostic might become tricky.
2. `csresmap`: This tool does not function for a binned input, as it runs `ctbin` and `ctmodel`. We could check for a binned observation and, if present, neglect the `ctbin` step. Of course, this strongly depends on how we decide that `ctmodel` should behave.

An idea for `ctmodel` would be to collapse all available unbinned observations into one binned observation but creates one model cube per given binned observation.

The updated code is available in

- `gammalib` on [1439-change-usage-of-binned-observations-in-GCTACube-classes](#)
- `ctools` on [1439-adapt-ctools-to-consistently-handle-observation-types](#)

#### #8 - 06/30/2015 07:00 PM - Mayer Michael

- Status changed from *In Progress* to *Feedback*
- % Done changed from 0 to 90
- Estimated time set to 3.00

#### #9 - 06/30/2015 08:03 PM - Knödseder Jürgen

Mayer Michael wrote:

Two tools are a bit different to handle:

1. `ctmodel`: First, I noticed that there is a `get_obs()` function which basically does the same as `ctool::get_observations()`. Therefore, I removed this function and use the generic `ctool` function instead. The other question is if we really want `ctmodel` to work on binned observations, which it currently does. However, it would then not be aligned with `ctbin`, which only creates a cube from the unbinned observations. Therefore the creation of a residual map could become a pain. If `ctmodel` just adds up all the model counts including binned observation (which is not considered in `ctbin`), the diagnostic might become tricky.

It would be nice to have `ctmodel` also working for binned observation, but I understand your point of inconsistent behavior. What about the following: if `ctmodel` gets an observation definition file or an event list it will work in the mode where it only uses the unbinned data. If however a counts cube is provided on input, it will compute a model for exactly that cube.

I was wondering whether it would make sense to rename the tool to fit the names of the other tools, e.g., `ctmodcube`? This also raises the question whether `ctbin` should in fact be renamed. `ctbin` has the "historic" name from the Fermi Science Tools, but to fit in the suites of the other tools one could think about calling that tool `ctevtcube` or `ctcntcube`. Maybe something that we should discuss all together.

1. `csresmap`: This tool does not function for a binned input, as it runs `ctbin` and `ctmodel`. We could check for a binned observation and, if present, neglect the `ctbin` step. Of course, this strongly depends on how we decide that `ctmodel` should behave.

Would it make sense to do the following:

- if an observation definition file is provided, use only the unbinned observations in a way similar to ctbin, etc.
- if an event list is provided, to as for observation definition file
- if a counts cube is provided, skip the ctbin step and use it directly in ctmodel

**#10 - 07/01/2015 12:24 PM - Mayer Michael**

- Status changed from Feedback to Pull request

- % Done changed from 90 to 100

I made the proposed changes on the branches listed above.

For csresmap, I made the parameter algorithm unhidden and further allowed to two hidden parameters: cntcube and modcube. If the user specifies those two on construction, the tool will automatically skip ctbin and ctmodel and compute the residual directly (similar to farith).

**#11 - 07/01/2015 06:49 PM - Knödlseider Jürgen**

Mayer Michael wrote:

- gammalib on *1439-change-usage-of-binned-observations-in-GCTACube-classes*
- ctools on *1439-adapt-ctools-to-consistently-handle-observation-types*

I did the *1439-change-usage-of-binned-observations-in-GCTACube-classes* branch, enhancing a bit the code so that also information about non-CTA observations is logged.

I will now proceed with the ctools branch.

**#12 - 07/02/2015 11:51 AM - Knödlseider Jürgen**

I homogenized the output and tested the following tools with success:

- ctobssim
- ctselect
- ctbin
- ctexpcube
- ctpsfcube
- ctbkgcube
- ctskymap

**#13 - 07/02/2015 12:55 PM - Knödlseider Jürgen**

I also checked ctmodel and removed the cntcube parameter from csresmap. If a counts cube is specified as inobs, the script will ask for the modcube parameter first. If it is given it simply computes the residual, otherwise it will continue querying the response parameters.

**#14 - 07/02/2015 05:24 PM - Knödlseider Jürgen**

- *Status changed from Pull request to Feedback*

I think we can close this now.

**#15 - 07/02/2015 05:26 PM - Mayer Michael**

I agree. I will probably run some further checks next week. But any occurring problem should then result in a new issue.

**#16 - 07/02/2015 05:30 PM - Knödlseider Jürgen**

- *Status changed from Feedback to Closed*