ctools - Feature #1889

cssens sensitivity calculation for model with spectrum from file (GModelSpectralFunc)

12/14/2016 03:45 PM - Huetten M.

Status:	Closed	**Start date:**	
Priority:	High	**Due date:**	
Assigned To:	Knödlseder Jürgen	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	1.7.0		

Description

Hi!
For my thesis, I wanted to calculate the sensitivity for a model with both the spatial and spectral descriptions described by numeric values in files (extended DM subhalos with DM spectra, see appended xml file). The current cssens.py tool cannot handle such a model. It would be great to add this feature to cssens.
Moritz

Related issues:

Related to ctools - Action # 3059: cssens sensitivity calculation for a sourc...	**Rejected**	**10/29/2019**

History

#1 - 01/18/2017 02:43 PM - Knödlseder Jürgen

On what timescale would you need that functionality?

#2 - 10/29/2019 07:04 PM - Kherlakian M.

user#3 wrote:

> On what timescale would you need that functionality?

Has it been implemented? I have the same problem. I want to calculate the sensitivity for a source spectrum defined by a FileFunction, however cssens does not accept this type of spectrum, it seems.

#3 - 10/30/2019 01:36 PM - Knödlseder Jürgen

- Assigned To changed from Mayer Michael to Knödlseder Jürgen

- Priority changed from Normal to High

- Target version set to 1.7.0

Unfortunately it has not been implemented so far. I will put this feature in the backlog for the next release.

#4 - 10/30/2019 01:38 PM - Knödlseder Jürgen

- Related to Action #3059: cssens sensitivity calculation for a source with file function spectrum added

#5 - 10/31/2019 01:59 PM - Knödlseder Jürgen

- Status changed from New to Closed

- % Done changed from 0 to 100

I implemented the possibility to use other types of spectral models in csspec and cstsdist. Now both scripts can except any type of spectral model, except of a node function. I would also expect that composite or multiplicative spectral models do not work properly, at least in general.

I tested the implementing using a file function model, a constant model, a power law 2 model, and photon flux and energy flux power law models.

Code was merged into devel.

Files

subhaloFluctResponse.subhalo_model.xml	557 Bytes	12/14/2016	Huetten M.