ctools - Action #1863

Support covariance matrix output

10/04/2016 11:30 AM - Ziegler Alexander

Status:	Closed	**Start date:**	10/04/2016
Priority:	Normal	**Due date:**	
Assigned To:	Tiziani Domenico	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	1.3.0		

Description

Aim is to store the information of the covarianve matrix after running ctlike.

History

#1 - 10/04/2016 11:37 AM - Ziegler Alexander

As we discussed on the coding sprint, the best solution would be to add additional information to the model-xml output file from ctlike.
We also discussed about storing more additional info there, like e.g. LogLike value.
For the covariance matrix, we might use the VO table format as suggested by Juergen.

#2 - 10/10/2016 05:36 PM - Ziegler Alexander

- Assigned To changed from Balbo Matteo to Ziegler Alexander

- % Done changed from 0 to 80

Status is as follows:

- implemented new function save_covmat(const GFilename& filename) to GObservations class, which stores the covariance matrix to a given FITS file.
- new hidden parameter covmat for ctlike, per default set to "NONE". If valid filename is passed to this parameter, the covariance matrix is written out.

A problem is where to define the output - at the moment it is done directly after calling the optimization, because doing it in ctlike::save() is to late, as the models likelihood function is changing when ts values are required to compute. The models are set back to the original ones afterwards, but this cannot be done for the likelihood function as there is no direct access to it at that point (this caused a bit of confusion to me...).

Remaining todo:

- Update documentation to include new feature.

#3 - 10/10/2016 06:50 PM - Knödlseder Jürgen

- Target version set to 1.2.0

user#190 wrote:

> A problem is where to define the output - at the moment it is done directly after calling the optimization, because doing it in ctlike::save() is to late, as the models likelihood function is changing when ts values are required to compute. The models are set back to the original ones afterwards, but this cannot be done for the likelihood function as there is no direct access to it at that point (this caused a bit of confusion to

me...).

I think it's fine to store the covariance matrix directly after the optimization, and then let the code do other stuff for the TS computation.

#4 - 10/11/2016 11:12 AM - Ziegler Alexander

- Status changed from New to Pull request

- % Done changed from 80 to 90

I added the new covmat feature to the docu of ctlike.
I tested it with different models by hand, reading the output covariance matrix fits file again in an calculating the errors on the parameters from that output and comparing the results to the errors given in the output model xml file.
Seems like the arrangement order in the FITS file is correct and everything works fine - ready for code review.

Branches are:

- 1863_CovMat (both in gammalib and ctools repo)

#5 - 10/17/2016 05:10 PM - Knödlseder Jürgen

user#190 wrote:

> I added the new covmat feature to the docu of ctlike.
> I tested it with different models by hand, reading the output covariance matrix fits file again in an calculating the errors on the parameters from that output and comparing the results to the errors given in the output model xml file.
> Seems like the arrangement order in the FITS file is correct and everything works fine - ready for code review.
>
> Branches are:
>
> - 1863_CovMat (both in gammalib and ctools repo)

Thanks. I was busy the last days. Will look into that asap.

#6 - 10/17/2016 10:36 PM - Knödlseder Jürgen

- Status changed from Pull request to Feedback

- % Done changed from 90 to 100

I checked the code you committed.

I recognised that you did the saving of the covariance matrix in the ctlike::run() method while all saving should be done in the ctlike::save() method. I understand that you did this because the covariance matrix may eventually change after the initial fit due to the TS computation. There is however a way to accommodate this: saving the curvature matrix after the fit and restoring the curvature matrix at the end of the run method. I modified the code accordingly.

Logically, the covariance matrix should be a member of GObservations::likelihood and not GObservations. There is in fact a way to do the saving in the GObservations::likelihood since the class holds a pointer to GObservations, hence it is able to access the models. I therefore move the saving to GObservations::likelihood and called the method simply GObservations::likelihood::save(). In principle, we could add code to also save the model values, gradients, etc. into a FITS file.

I merged all changes into devel. I put the status to Feedback and you may check that the code still behaves as you expect.

#7 - 03/03/2017 10:46 AM - Knödlseder Jürgen

- Target version changed from 1.2.0 to 1.3.0

#8 - 04/06/2017 09:58 AM - Tiziani Domenico

- % Done changed from 100 to 80

Alexander an me came to the conclusion that it would be better to store the covariance matrix a csv file rather than as a fits file to enable easy access to the user. We are also not sure if this output should be used in subsequent scripts because the values of the covariance matrix can easily change their order if the according model file is changed by the user.
I am starting to implement the export to a csv file now.

#9 - 04/06/2017 11:58 AM - Tiziani Domenico

- % Done changed from 80 to 90

The output format of GObservations::likelihood::save() can now be chosen with a flag to either csv or fits. Default is csv.
The user of ctlike can not choose the format. We will update the documentation accordingly if it's OK to make the default output csv. (The fits output still works as expected but is not favored.)

An example of a csv file is given here:

RA(Crab) DEC(Crab) Prefactor(Crab) Index(Crab) Cutoff(Crab) Scale(Crab) Normalization(Crab)
5.61309599084731e-06 -1.83433791867944e-07 4.62934176274374e-24 -1.52450978816292e-06 -9.26084864350935 0 0
-1.83433791867944e-07 4.58675292270187e-06 7.65058926211812e-23 2.98909451728534e-06 -15.9355531595409 0 0
4.62934176274372e-24 7.65058926211812e-23 3.53273692750675e-36 1.92196287270621e-19 -4.25589848340144e-13 0 0
-1.52450978816292e-06 2.98909451728534e-06 1.92196287270621e-19 0.0111048626510676 -22395.023023584 0 0
-9.26084864350935 -15.9355531595409 -4.25589848340144e-13 -22395.023023584 60131839781.0409 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0

#10 - 04/06/2017 12:33 PM - Tiziani Domenico

- Status changed from Feedback to Pull request

- Assigned To changed from Ziegler Alexander to Tiziani Domenico

- % Done changed from 90 to 100

The documentation is now also updated.

The corresponding branches are:
dtiziani/gammalib:1863-covmat
dtiziani/ctools:1863-covmat

#11 - 04/10/2017 11:33 AM - Knödlseder Jürgen

I've seen that you added a flag to the new code in GammaLib to switch between CSV and FITS which is not very explicit for a user. I propose to make this more explicit.

One way could be to specify in a string the format in which the data should be saved. Another option is to extract the file type from the filename. I prefer the latter since it allows for flexibility without changing the interface.

I added a GFilename::type() method that returns fits if the filename has a .fits or .fit extension. Otherwise it returns for the moment an empty string (this method can be extended in the future for different file types).

For GObservations::likelihood is also refactored the code a bit into three different private methods which avoids having a long code with if/else in the save method:

```
void                     save_csv(const GFilename& filename) const;
void                     save_fits(const GFilename& filename) const;
std::vector<std::string> covariance_names(void) const;
```

#12 - 04/11/2017 11:05 AM - Knödlseder Jürgen

- Status changed from Pull request to Closed

- Remaining (hours) set to 0.0

I adapt the ctlike documentation to the logic mentioned above, i.e. that the file format is defined by the file type of the file name.

The change was merged into devel.