Fermipy tutorial-documentation suggestions

Principe Giacomo*[1,2,3] & Leonardo Di Venere[4]

[1]Dipartimento di Fisica, Università di Trieste, Trieste, Italy
[2]Istituto Nazionale di Fisica Nucleare, Trieste, Italy
[3]INAF - Istituto di Radioastronomia, Bologna, Italy
[4]Istituto Nazionale di Fisica Nucleare, Bari, Italy

Outline

- Documentation status
- Existing tutorials in fermipy (fermipy-extra, fermi summer school, confluence)
- Existing tutorials in other packages (gammapy, ctools)
- Other available frameworks (easyFermi, google colab)
- Actions for existing tutorials & Ideas for new tutorials

Documentation - recent updates

Fermipy website

Situation in 2021:

- need to update the documentation adding also additional link to the fermitools recommendations, need to update fermipy website and some packages
- need to update the installation information (wrong for the current version)

Now - 2023

Recent updates in the documentation fermipy website are very nice (particularly the warnings) making clear to the user some recommendations

- updated the installation information (moved to mamba, missing conda installation info)
- description new features included (es. curvature test)
- updated the information of the features (adding some recommendations)

Documentation - Missing things

Still missing

- description of some features (e.g. simulate_roi, weighted likelihood)
- need to check / update table options of some functions
  (e.g. log_ebins missing in SED and extension analysis table)
- need to update the current tutorial and possibly add new ones (see second part presentation)

Documentation - Best practices

Data / model comparison

Documenting best practices for:

- fermipy:
  - customizing a model,
  - suggestions for the parameters setting in the fitting (gta.optimize, gta.fit)
  - checks on the fit
  - checklist

- General suggestions (see also Jean's recommendation catalog analysis)

This is needed also for the Fermi Tools documentation!

Documenting the best practices

slides Mattia Di Mauro presentation. (2019)

Checks on the fit

• Fit quality: A "good" fit corresponds to a value of "fit quality = 3"; if you get a lower value it is likely that there is a problem with the error matrix.
According to the Minuit documentation possible values for "fit quality" are:

0 - Error matrix not calculated at all
1 - Diagonal approximation only, not accurate
2 - Full matrix, but forced positive-definite (i.e. not accurate)
3 - Full accurate covariance matrix (After MIGRAD, this is the indication of normal convergence.)

• TS and residual map: running gta.tsmap and gta.residmap you can make the TS and residual maps.

• gta.write_roi(): This makes a series of plots such as: count map, count spectrum.

Documenting the best practices

slides Mattia Di Mauro presentation. (2019)

Checklist for your analysis

This is a list of things you should check in your analysis:

- The TS and/or residual map should not have structured or large scale residuals.
- The fit quality should be at least 2.
- Check covariance matrix and correlation coefficients.
- The count spectrum should has residuals of a few % level.

A few suggestions:

- Include sources from the catalog at least a few degrees beyond the edge of the ROI.
- If the fit is not successful (fit quality 0 or 1 and maybe also with 2), run a few times gta.optimize()
- Run the search for new sources gta.find_sources().

  An iterative source-finding algorithm that uses likelihood ratio (TS) maps of the region of interest to find new sources. After each iteration a new TS map is generated incorporating sources found in the previous iteration. The method stops when the number of iterations exceeds **max_iter** or no sources exceeding **sqrt_ts_threshold** are found.

Documenting the best practices

slides Mattia Di Mauro presentation. (2019)

If the fit is not successful

If the fit is not successful there are a few things you may try.

- Run a few time gta.optimize() and then gta.fit()
- See if there are SED parameters of some source that take crazy values (gta.get_params).
- If you are running an analysis in a small energy range (1-10 GeV, 1-100 GeV, above 10 GeV,…) substitute the SED of the sources from PLEC or LP to PL.
- Run the fit to each source leaving free to vary only the sources within a few degree.
- ……

Documenting the best practices

slides Jean Ballet presentation. (2019)

General suggestions

- *Insignificant sources* (TS < 1 in the Catalogs pipeline) should be fixed or removed before computing the covariance matrix of all other parameters.

- *Curved spectral shapes* should be avoided when they are not necessary (when a power law is just as good) because this can introduce too much freedom (for example, low-energy signal can be picked up by one source or its neighbor). Catalogs pipeline: LogParabola only when TSCurv = 2 (logLike(LogParabola) - logLike(PowerLaw)) > 9.

- The *best-fit parameter value* should not be too far off the middle of the interval. (particularly check the normalization of Galactic and isotropic diffuse)

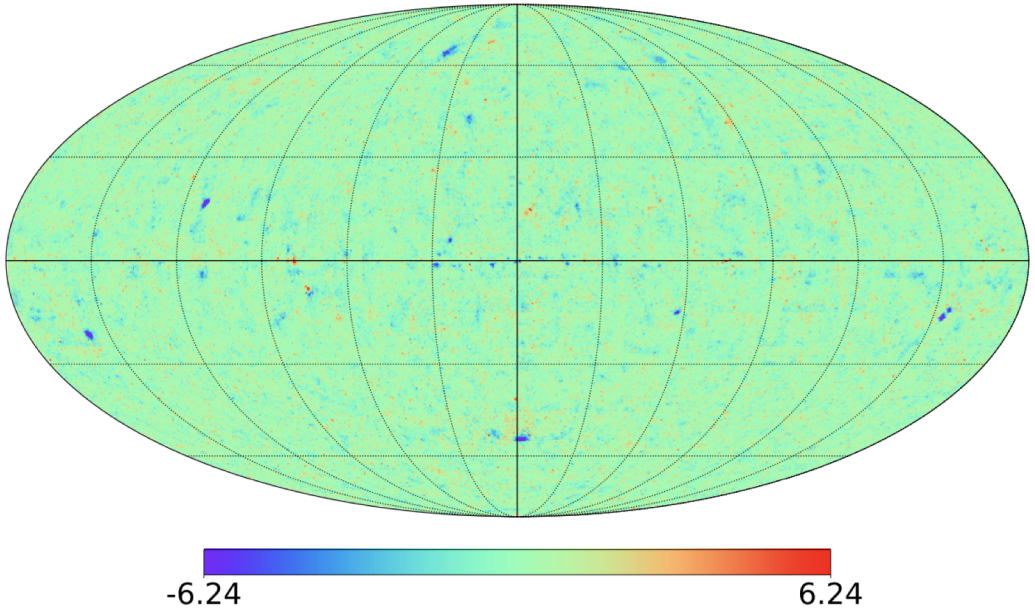
- The *reference energy* should be chosen such that the normalization parameter is not too much correlated with the shape parameters. This happens for PowerLaw or LogParabola when Scale or Eb is set to the pivot energy at which the relative error on the normalization is minimal.

gtpsmap

gtpsmap (to solve the issue with the non correct 'Gaussianity' of Fermipy residuals, Philippe Bruel presentations in confluence) arXiv link
https://fermi.gsfc.nasa.gov/ssc/data/analysis/user/gtpsmap/gtpsmap.py

- the method can take into account the likelihood weights

- Change the gta.residmap() method to have a better residual parameter?
- Is there the possibility to integrate *gtpsmap* in fermipy?



Fermipy Tutorials

Some examples already available in **fermipy-extra** (Mattia, Matthew, Eric): Fermipy-extra (GitHub)

Tutorials with full analysis chain and documentation in good shape:

- PG1553 : Basic tutorial on LAT analysis
  - firsts steps from FT1/FT2 data to likelihood fit (including residuals and SED)
    - need to add recommendations on data quality cuts
  - fit result inspection (residuals, spectra)
- IC443 : Analysis to measure the angular extension of the SNR IC443.
  - extension fit example
  - can be expanded with comparison of spatial models, usage of template models, … ?
- Draco : DM upper limit analysis of the Draco dwarf spheroidal galaxy.
  - simple likelihood analysis with upper limit calculation
  - dark matter upper limits calculation
- SMC: Tutorial complete analysis (Spectral and Extended) source analysis SMC
  - includes installation instructions, data preparation, likelihood fit, extension fitting, SED, source finding, lightcurve

Fermipy Tutorials

Tutorials with little/no documentation:

- dSph: example on fermipy pipeline to analyze dSphs
  - very little explanation
- Phased analysis: phased analysis for pulsars (Geminga)
  - requires better explanations of pulsar physics and analysis steps
- file_function: short example on how to use the FileFunction spectral shape
- gtools_model_customization : examples on how to run basic gtools and customize the spectral/spatial model
- optimize_model : likelihood fitting, results inspection
  - duplicate of other tutorials

Some a**dditional examples/scripts** collected Fermipy scripts

- FA follow-up (transients on 6-hour – 1 week time scale)
- Fermi summer school 21 (point source analysis – few months: SED + LC)

Gammapy Tutorials

Gammapy (https://gammapy.org)

many tutorial for different examples/cases
https://docs.gammapy.org/1.0/tutorials/index.html

Fermipy
We have only few tutorials/notebook with complete analysis (config file, setup, freeing sources, optimisation, localization, ..)

- point source
- extended source
- transient

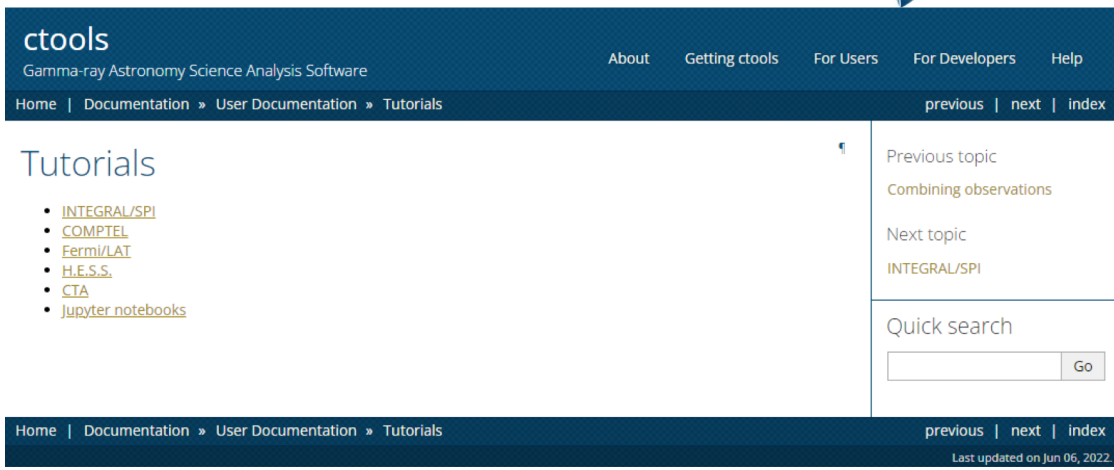
We need example exploiting the many possibilities (parameters setting) of each feature

High level interface
Low level API
Data structures
CTA with Gammapy
Fermi-LAT with Gammapy
H.E.S.S. with Gammapy
Point source sensitivity
Spectral analysis of extended sources
Flux point fitting
Spectral analysis
Spectral analysis with the HLI
Spectral analysis with energy-dependent directional cuts
1D spectrum simulation
Source detection and significance maps
2D map fitting
Ring background map
3D detailed analysis
Multi instrument joint 3D and 1D analysis
Basic image exploration and fitting
Event sampling

Flux Profile Estimation
3D map simulation
Light curves
Light curves for flares
Simulating and fitting a time varying source
Pulsar analysis
Dark matter spatial and spectral models
Source catalogs
Datasets - Reduced data, IRFs, models
Fitting
Makers - Data reduction
Maps
Mask maps
Modelling
Models
Survey Map Script

ctools Tutorials

Comparison with ctools (http://cta.irap.omp.eu/ctools/index.html)
Several tutorials for different examples/cases: http://cta.irap.omp.eu/ctools/users/tutorials/index.html



First steps with ctools

- Simulating event data
- Selecting event data
- Generating a sky map
- Binning event data
- Pre-computing the binned response
- Fitting binned data
- Inspecting the spatial fit residuals
- Inspecting the spectral fit residuals
- Generating a butterfly diagram of the source spectrum
- Deriving a source spectrum
- Doing an unbinned analysis
- Taking the energy dispersion into account
- Doing an On/Off analysis
- Using ctools from Python

First CTA Data Challenge

- Scope of the first CTA Data Challenge
- Getting the data
- Data organisation
- First analysis steps

CTA data analysis

- How to combine observations?
- How to perform a stacked analysis?
- How to handle models from Python?
- How to do advanced model manipulation and fitting in Python?
- How to perform spectral component separation?
- How to connect observations to dedicated instrument response functions?
- How to connect observations to specific models?
- How to perform spectral fitting using Xspec?
- How to generate likelihood profiles for a fitted spectrum?

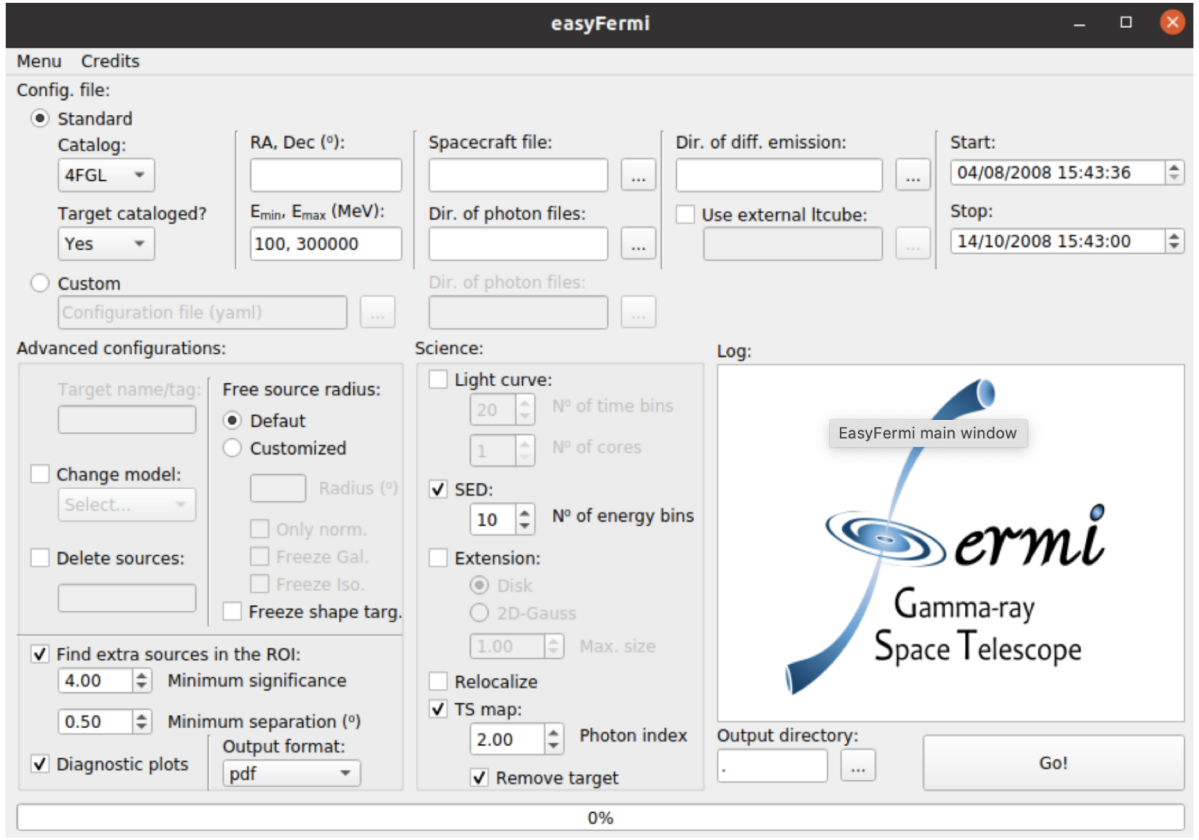
CTA first Data Challenge data analysis

- Data preparation for 1DC How-tos
- How to compute the significance of a source?
- How to generate a Test Statistic map?
- How to determine the extension of a source?
- How to compute upper limits?
- How to generate a light curve?
- How to generate a phase curve of a pulsar?
- How to exclude bins from a counts cube?
- How to display the results?
- How to use Virtual Observatory tools with ctools?

- Prepare Fermi-LAT data
- Fit a model to the data
- Generate a butterfly diagram
- Generate a spectrum
- Generate a Test Statistic map
- Derive upper flux limits
- Determine errors for a model parameters

EasyFermi

An application for standard GUI-based analysis with a ***youtube tutorial***
https://github.com/ranieremenezes/easyFermi



Google colab

Google colaboratory is a free platform to run python notebooks with dedicated worker node and memory:
https://colab.research.google.com/

It is possible to install packages, but memory is volatile → it is likeli you have to start from scratch every time you close the notebook

We have developed a few notebooks for Fermi Masterclass for High School students (in Italian):

- LAT data analysis: simple aperture photometry analysis with pre-built dataset
- LAT data explorer: FT1 and FT2 file explorer

Both notebooks are fermitools/fermipy independent for simplicity,
but it is possible to use them, as long as the installation time is reasonable

This could be a useful tool for demonstrations or for students,
not for real science analysis

Discussion

Major actions:

- All tutorials need to be updated to python3 and tested against latest fermitools/fermipy releases
- Improve description/documentation on the first group of tutorials
- harmonization of remaining tutorials
- simple examples on how setting the parameters on individual features/functions

Additional tutorials may cover:

- simulations with fermipy
  - we have existing scripts
- detailed examples of fermitools features (e.g., new spectral models)
  - synergy with the catalog modifications as well as with the Fermi tools updates
- blazar variability
  - lightcurve analysis tutorial is a good starting point
- phase-curve PSR
  - phase_analysis available tutorial is a good starting point
- stacking analysis
  - we have several scripts to be converted in documented tutorials
- weighted likelihood
- sensitivity for point-like and/or extended sources
- …?