Gammapy: a Python package for gamma-ray astronomy

Maxime Regeard - on behalf of the Gammapy team

February 22, 2024

APC - AstroParticule et Comoslogie, Université Paris-cité, CNRS, Paris, France

About me

- PhD student at **APC, Université Paris-Cité, CNRS, Paris**
- Working on the pulsars at High/Very-High Energy
- Member of the **H.E.S.S.** and **CTA** collaborations
- Contributing to Gammapy
- Presenting on behalf of the **Gammapy team**

Introduction to Gammapy

Multi-wavelength Multi-instrument analysis

The Gammapy paper

What's new Gammapy ?

The Gammapy project

Outlook and Summary

Introduction to Gammapy

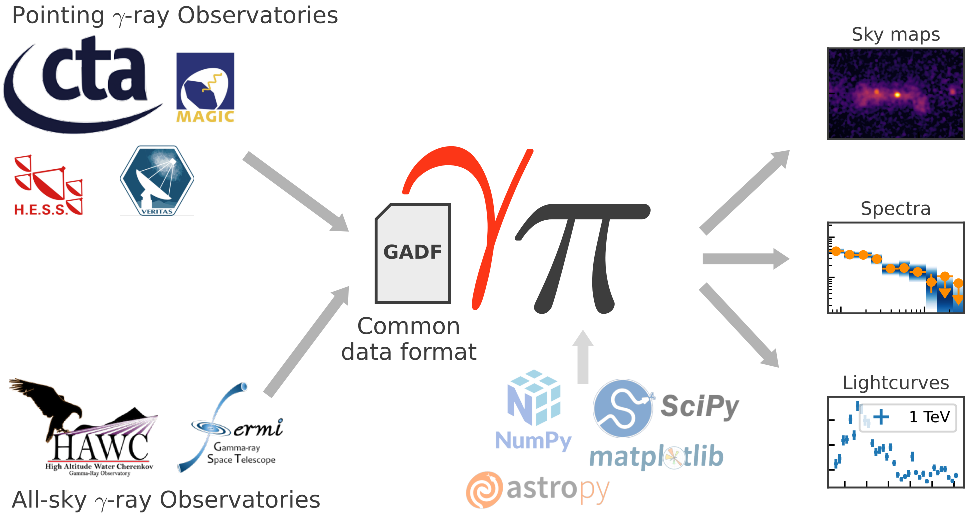
A **Python** package for **gamma-ray** astronomy

Open source, flexible, community driven, Python package

+

Selected as CTA offical science analysis tool

The idea behind Gammapy



Optional dependencies: bring in useful functionality

Optional dependencies

Required dependencies

Pydantic

Configuration

PyYAML

YAML I/O

matplotlib

Plotting, visualisation

$ click_

Command line tools

Sherpa

iminuit

Optimisation, sampling

healpy

Healpix maps

jupyter

Tutorial notebooks

astropy

Coordinates, Quantities, Tables, FITS I/O, etc.

SciPy

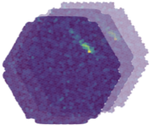
Interpolation, minimisation, FFT convolution, etc.

NumPy

ND-data structures and computations

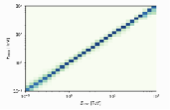
Reconstruction pipeline (instrument specific) → Science Tools (general users) →

DL 0-1	DL 2	DL 3	DL 4	DL 5
raw - calibrated	*reconstructed*	*filtered γ-like events*	*science*	*high-level*

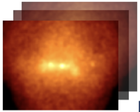


camera data

ENERGY	RA	DEC	L	B
MeV	deg	deg	deg	deg
float32	float32	float32	float32	float32
12186.642	260.45935	-33.553337	353.36273	1.7538676
25496.598	261.37506	-34.395004	353.09607	0.6520652
15621.498	259.56973	-33.409416	353.05673	2.4450684
12816.92	279.95883	-25.940391	6.43856	-4.0548879
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
13960.802	271.44742	29.615316	1.6267247	4.1431155
10477.372	266.3981			
13030.88	271.70428			

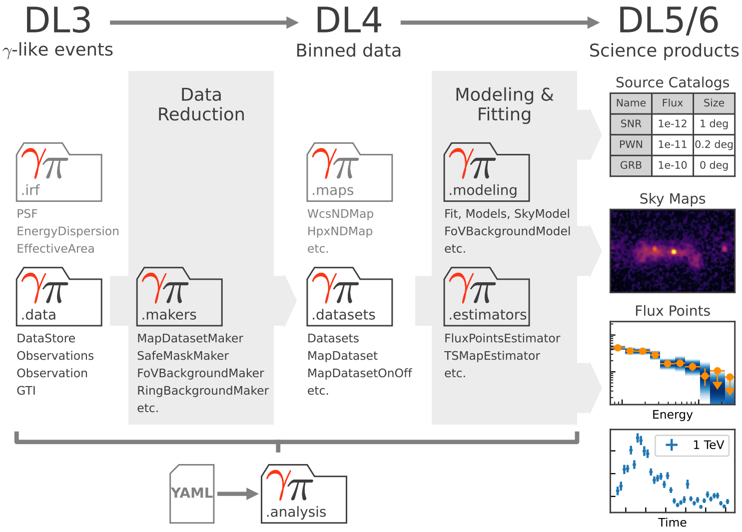


γ-like event lists
IRFs



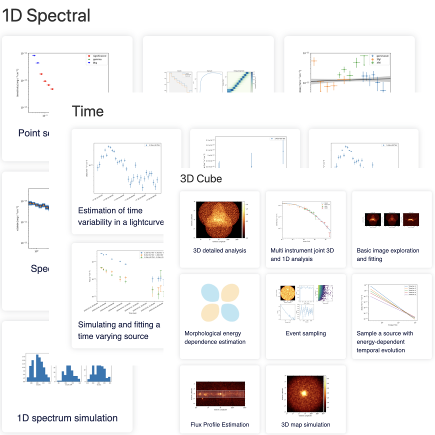
maps,
spectra,
light curves

Workflow


DL3
γ-like events
DL4
Binned data
DL5/6
Science products
Data Reduction
Modeling & Fitting
.irf
PSF
EnergyDispersion
EffectiveArea
.data
DataStore
Observations
Observation
GTI
.makers
MapDatasetMaker
SafeMaskMaker
FoVBackgroundMaker
RingBackgroundMaker
etc.
.maps
WcsNDMap
HpxNDMap
etc.
.datasets
Datasets
MapDataset
MapDatasetOnOff
etc.
.modeling
Fit, Models, SkyModel
FoVBackgroundModel
etc.
.estimators
FluxPointsEstimator
TSMapEstimator
etc.
YAML
.analysis
Source Catalogs
Name | Flux | Size
SNR | 1e-12 | 1 deg
PWN | 1e-11 | 0.2 deg
GRB | 1e-10 | 0 deg
Sky Maps
Flux Points
Energy
1 TeV
Time

Data analysis

The following set of tutorials are devoted to data analysis, and grouped according to the specific covered use cases in spectral analysis and flux fitting, image and cube analysis modelling and fitting, as well as time-dependent analysis with light-curves.



launch binder

`Download Python source code: model_management.py`

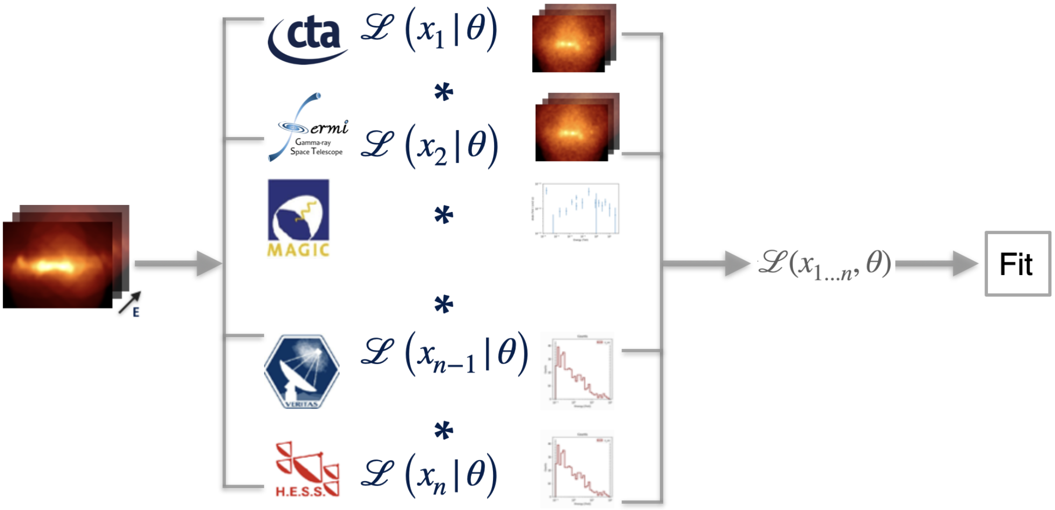
`Download Jupyter notebook: model_management.ipynb`

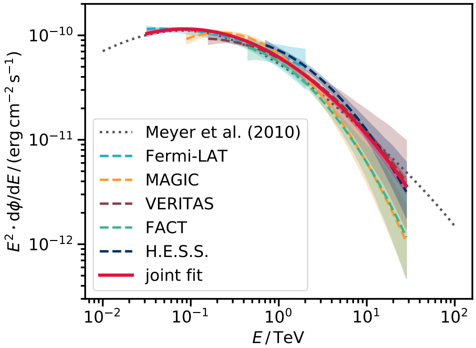
Downloadable via

`gammapy download notebooks`

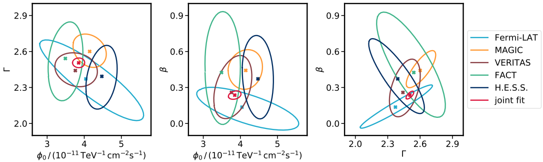
- Collection of data from different collaboration: H.E.S.S., MAGIC, CTA, Fermi-LAT, ...
- Used to generate and run the tutorials
- Used to make and run tests
- Available on a **Github repository**
- Downloadable through `gammapy download datasets`

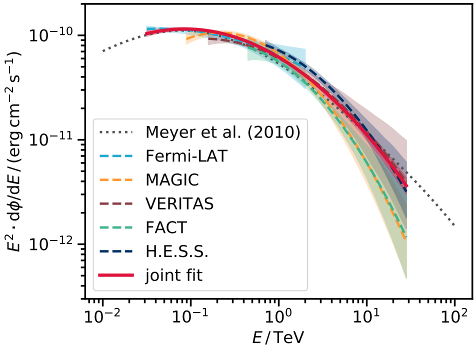
Multi-wavelength Multi-instrument analysis

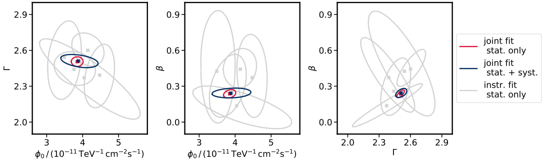

cta
$\mathscr{L}(x_1|\theta)$
*
Fermi
Gamma-ray
Space Telescope
$\mathscr{L}(x_2|\theta)$
MAGIC
*
*
$\mathscr{L}(x_{n-1}|\theta)$
VERITAS
*
H.E.S.S.
$\mathscr{L}(x_n|\theta)$
E
$\mathscr{L}(x_{1...n}, \theta)$
Fit



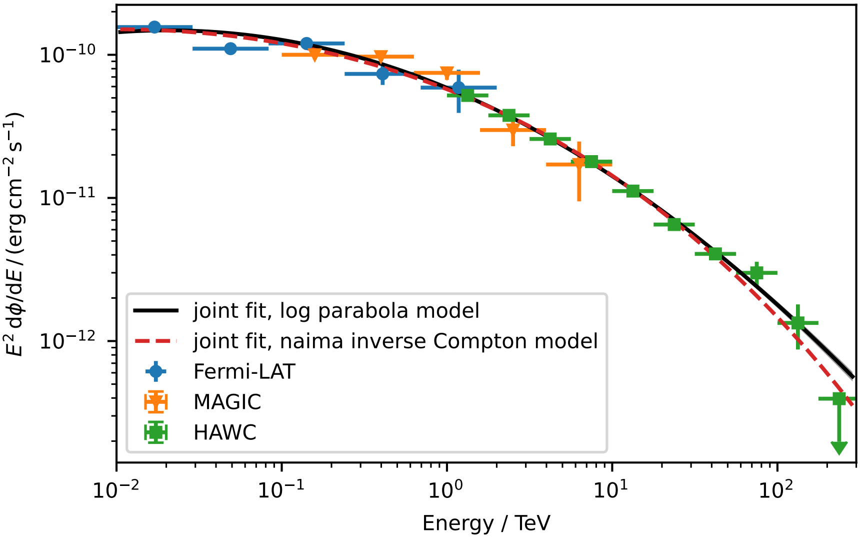
- Figures are taken from the **joint-crab** gitHub repository that contains material for **Nigro et al. (2019)**







- Figures are taken from the **joint-crab** gitHub repository that contains material for **Nigro et al. (2019)**
- The benefits of a joint likelihood fitting is clearly visible



$E^2\,\mathrm{d}\phi/\mathrm{d}E\,/\,(\mathrm{erg\,cm^{-2}\,s^{-1}})$
10^{-10}
10^{-11}
10^{-12}
joint fit, log parabola model
joint fit, naima inverse Compton model
Fermi-LAT
MAGIC
HAWC
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
Energy / TeV

The Gammapy paper

GAMMAPY V1.0 PAPER


A&A, 678, A157 (2023)
https://doi.org/10.1051/0004-6361/202346488

Astronomy & Astrophysics

Gammapy: A Python package for gamma-ray astronomy

Axel Donath[1], Régis Terrier[2], Quentin Remy[3], Atreyee Sinha[4], Cosimo Nigro[5], Fabio Pintore[6], Bruno Khélifi[2], Laura Olivera-Nieto[3], Jose Enrique Ruiz[7], Kai Brügge[8,9], Maximilian Linhoff[9], Jose Luis Contreras[4], Fabio Acero[10], Arnau Aguasca-Cabot[11,12,13,14], David Berge[15,16], Pooja Bhattacharjee[17], Johannes Buchner[18], Catherine Boisson[19], David Carreto Fidalgo[20], Andrew Chen[21], Mathieu de Bony de Lavergne[17], José Vinícius de Miranda Cardoso[22], Christoph Deil[3], Matthias Füßling[23], Stefan Funk[24], Luca Giunti[2], Jim Hinton[3], Léa Jouvin[25], Johannes King[26,3], Julien Lefaucheur[27,2], Marianne Lemoine-Goumard[28], Jean-Philippe Lenain[29], Rubén López-Coto[7], Lars Mohrmann[3], Daniel Morcuende[4], Sebastian Panny[32], Maxime Regeard[2], Lab Saha[4], Hubert Siejkowski[30], Aneta Siemiginowska[1], Brigitta M. Sipőcz[31], Tim Unbehaun[24], Christopher van Eldik[24], Thomas Vuillaume[17], and Roberta Zanin[23]

(Affiliations can be found after the references)

Received 23 March 2023 / Accepted 7 July 2023

ABSTRACT

Context. Traditionally, TeV-γ-ray astronomy has been conducted by experiments employing proprietary data and analysis software. However, the next generation of γ-ray instruments, such as the Cherenkov Telescope Array Observatory (CTAO), will be operated as open observatories. Alongside the data, they will also make the associated software tools available to a wider community. This necessity prompted the development of open, high-level, astronomical software customized for high-energy astrophysics.
Aims. In this article, we present Gammapy, an open-source Python package for the analysis of astronomical γ-ray data, and illustrate the functionalities of its first long-term-support release, version 1.0. Built on the modern Python scientific ecosystem, Gammapy provides a uniform platform for reducing and modeling data from different γ-ray instruments for many analysis scenarios. Gammapy complies with several well-established data conventions in high-energy astrophysics, providing serialized data products that are interoperable with other software packages.
Methods. Starting from event lists and instrument response functions, Gammapy provides functionalities to reduce these data by binning them in energy and sky coordinates. Several techniques for background estimation are implemented in the package to handle the residual hadronic background affecting γ-ray instruments. After the data are binned, the flux and morphology of one or more γ-ray sources can be estimated using Poisson maximum likelihood fitting and assuming a variety of spectral, temporal, and spatial models. Estimation of flux points, likelihood profiles, and light curves is also supported.
Results. After describing the structure of the package, we show, using publicly available gamma-ray data, the capabilities of Gammapy in multiple traditional and novel γ-ray analysis scenarios, such as spectral and spectro-morphological modeling and estimations of a spectral energy distribution and a light curve. Its flexibility and its power are displayed in a final multi-instrument example, where datasets from different instruments, at different stages of data reduction, are simultaneously fitted with an astrophysical flux model.

Key words. methods: statistical – astroparticle physics – methods: data analysis – gamma rays: general

1. Introduction

Modern astronomy offers the possibility to observe and study astrophysical sources across all wavelengths. The γ-ray range of the electromagnetic spectrum provides us with insights into the most energetic processes in the Universe such as those accelerating particles in the surroundings of black holes or remnants of supernova explosions.

In general, γ-ray astronomy relies on the detection of individual photon events and reconstruction of their incident direction as well as energy. As in other branches of astronomy, this can be achieved by satellite as well as ground-based γ-ray instruments. Space-borne instruments such as the *Fermi* Large Area Telescope (LAT) rely on the pair-conversion effect to detect γ-rays and track the positron-electron pairs created in the detector to reconstruct the incident direction of the incoming γ-ray. The energy of the photon is estimated using a calorimeter at the bottom of the instrument. The energy range of instruments such as *Fermi*-LAT (Atwood et al. 2009), referred to as "high energy" (HE), goes approximately from tens of Megaelectronvolts to hundreds of Gigaelectronvolts.

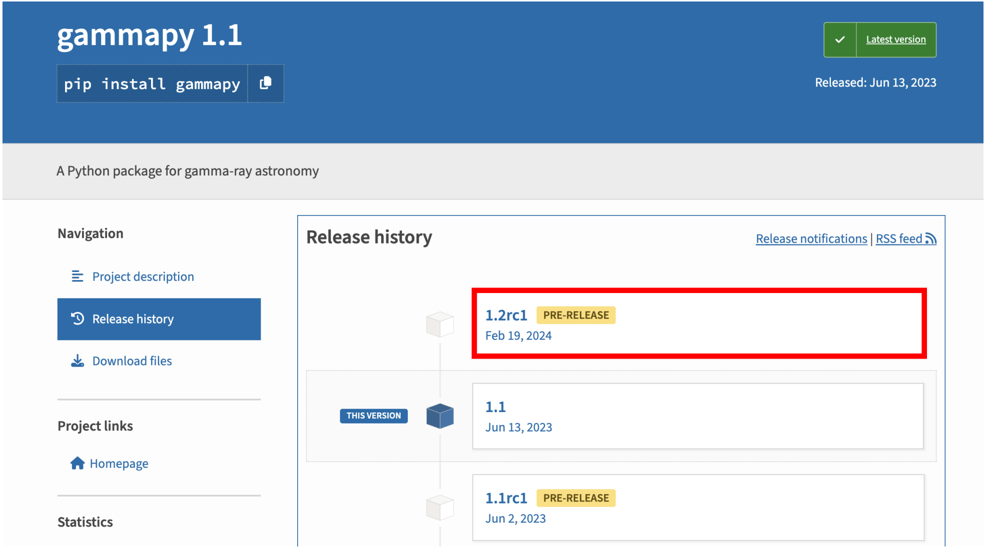
Ground-based instruments, instead, use Earth's atmosphere as a particle detector, relying on the effect that cosmic γ-rays interacting in the atmosphere create large cascades of secondary particles, so called "air showers", that can be observed from the ground. Ground-based γ-ray astronomy relies on the observation of these extensive air showers to estimate the primary γ-ray photons' incident direction and energy. These instruments operate in the so-called "very high energy" (VHE) regime, covering the energy range from a few tens of Gigaelectronvolts up to Petaelectronvolts. There are two main categories of ground-based instruments.

First there are imaging atmospheric Cherenkov telescopes (IACTs), which obtain images of the atmospheric showers by

- Donath et al. (2023)
- Paper written by the Gammapy developers
- Published in **Astronomy & Astrophysics** the 23 October 2023
- Written collaboratively and openly on **a Github repository**
- Official reference for Gammapy
- Acknoledge code contributors

What’s new Gammapy ?

Version 1.2 is close !

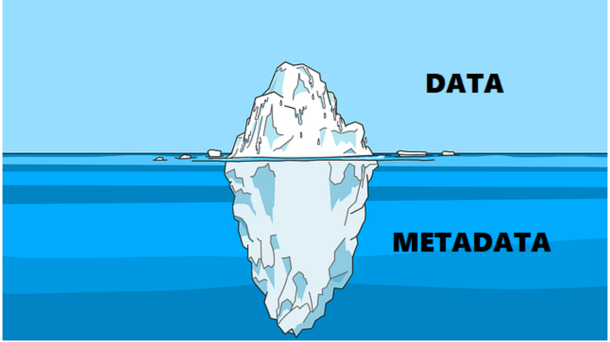


- **Improved multi-processing support**
- **Introducing Metadata**
- **Priors parameters**
- Improved performance on **Observation**
- Observation clustering method based on IRFs quality
- Utility function to test nested models hypotheses
- Estimator for morphological studies
- Tools for variablilty analysis for light curves

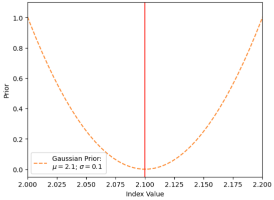
- Data reduction loop can be performed with the `DatasetsMaker` class
- Currently support `ray` as a prototype

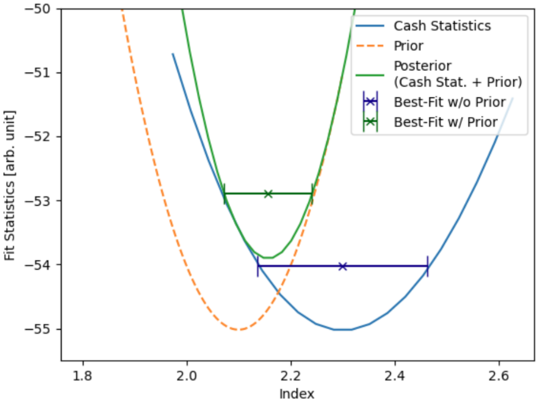
```
from gammapy.makers import DatasetsMaker

makers = DatasetsMaker(
    [dataset_maker, safe_mask_maker, fov_bkg_maker],
    n_jobs=3,
    parallel_backend="ray",
    stack_datasets=True,
    cutout_mode="partial",
)

datasets = datasets_maker.run(empty_dataset, observations)
```

- PIG 25: ”Metadata handling is crucial to correctly store information that are not directly data but are still required for processing, post-processing and serialization. They are fundamental for reproducibility.”
- Archived using **Pydantic**



- Priors on parameters that are evaluated during the fitting
- Currently only uniform and gaussian priors are supported
- Example: spectral model with index of 2.3 with a prior at 2.1

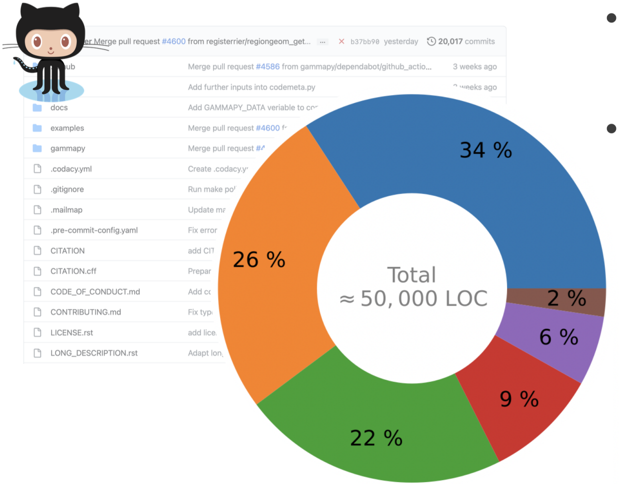




The Gammapy project

Some code statistics

- We have **achieved a healthy distribution of approximately 1/3 code, 1/3 docs and 1/4 tests,** however docs and test could still improve…
- There is a **small amount of Cython code** that re-implements certain masked Numpy expressions to **speed up critical parts of the likelihood fit**…



Hosted and **openly developed on GitHub**: https://github.com/gammapy/gammapy

GitHub actions used to run CI on each PR and a release pipeline

codecov.io used for monitoring of code test coverage

Docs

Docs are build and deployed using GitHub actions: https://docs.gammapy.org/

Sphinx used to build the documentation, Sphinx Gallery for the tutorial gallery

Pytest used for testing

Defined as pre-commit hooks

Black code formatting used for a consistent code format.

Isort to automatically **sort and format imports**

Flake 8: Flake 8 to **check PEP8** standard

Community

- Slack: Use for quick questions, help
- GitHub issues: Feature requests and bug reports
- GitHub discussions: Users and developpers discussions

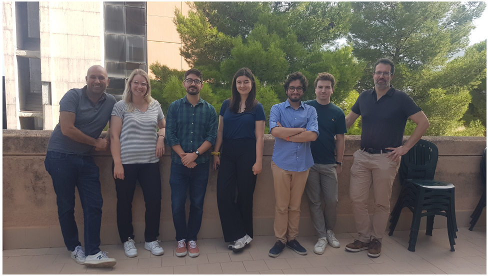
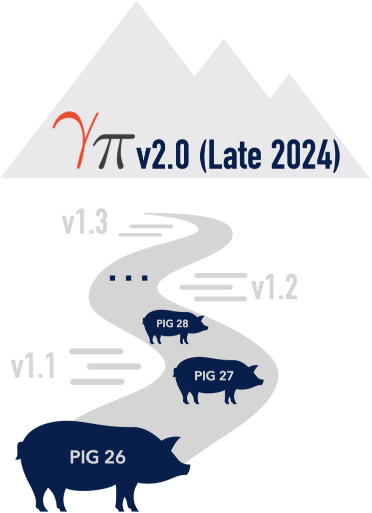


Figure 1: Last coding sprint in Palermo, October 2023

- Coding sptrint $\sim$ 2-3 times a year
  - Focus on specific subject
  - Work before realease
  - Proven to be effective
- Dev meeting every friday at 2 pm (Paris time)
  - Every day work
  - Discuss PRs and review code

Outlook and Summary

- On the roadmap of v2.0 (draft)
  - Internal data model separation
  - Event type handling
  - Handling of systematic effect
  - Split out `gammapy.maps` ?
- Is Gammapy a good tool for KM3Net ?



Summary

- Gammapy is an open source Python library for gamma-ray astronomy
- Importance of joint analysis and how Gammapy is a great tool for that
- Version 1.2 will be release in a few days, with a lot of exciting new features !
- However, we need man power ! Everybody is more than welcome to contributes

- The Gammapy paper: Donath et al. (2023)
- A. Donath presentation at the **SciPy conference**:
- **gammapy-presentations**: GitHub repository where the gammapy community put presentations

References

Donath, A., Terrier, R., Remy, Q., et al. 2023, A&A, 678, A157

Nigro, C., Deil, C., Zanin, R., et al. 2019, A&A, 625, A10