



Set-up for documentation tutorials and datasets access

Jose Enrique Ruiz – IAA / CSIC
Gammapy Coding Sprint
Madrid - October 1st 2018



PIG 4 - Setup for tutorial notebooks and data

- Author: José Enrique Ruiz, Christoph Deil
- Created: May 16, 2018
- Accepted: tbd
- Status: draft
- Discussion: [GH 1419](#)

Abstract

For the past years, we have had tutorial notebooks and example datasets in a second `gammapy-extra` repository. The motivation was to keep the main `gammapy` code repository small. But we always had problems with code, tutorials and data changing and versions not being linked.

We propose to move the notebooks to the `gammapy` repository so that code and tutorials can be version-coupled, and to only use stable datasets in tutorials to mostly save the versioning issues. The datasets will remain in `gammapy-extra` repository.

To ship tutorials and datasets to users, we propose to add a `gammapy download` command. The `gammapy-extra` repository will remain as a repository for developers and as one place where datasets can be put, but it will not be mentioned to users.





gammapy

A Python package for gamma-ray astronomy

python awesome astronomy astropy

Python ★ 73 ⚡ 82 Updated 32 minutes ago



gammapy-docs

Gammapy docs build & deploy on Github pages

Python ⚡ 1 Updated 20 hours ago



gammapy-webpage

Repository for the Gammapy webpage

HTML ★ 1 ⚡ 6 MIT Updated 23 hours ago



gamma-cat

An open data collection and source catalog for gamma-ray astronomy

astronomy dataset gamma-ray-astronomy source-catalog

Python ★ 7 ⚡ 14 BSD-3-Clause 13 issues need help Updated on 16 Jul



gammapy

Updating astropy-helpers to 2.0.6
Remove dev/format_crab_sed.py
Mention light_curve tutorial in docs/tutorials.rst
Run black
Update downloadclass.py
Merge remote-tracking branch 'template/master'
Fix broken link in hess.ipynb
Various fixes minor to tutorials and get started docs page
Updating to the actual astropy-helpers
Update .travis.yml
Update CHANGES.rst for v0.8
Update Dockerfile
Changing all readthedocs.org to readthedocs.org
Add conda environment-dev.yml
Params for docs-building in make-docs
Update README.rst
Updating helpers to v2.0.5
Add CI testing for Python 3.7
Configure Binder
Use gammapy/scripts/jupyter/test_notebook.py
Update astropy-helpers to v2.0rc1
Update lgtn.yml
Update setup.cfg
Require regions>=0.3 in setup.py

gammapy 0.9.dev470
Search docs
Getting Started
Tutorial notebooks
Installation
Data - Data and observations
IRF - Instrument response functions
Maps - Sky maps
Image - Map image analysis
Cube - Map cube analysis
Dect - Source detection
Background - Background modeling
Spectrum - 1D spectrum analysis
Time - Time analysis
Stats - Statistics
Utils - Utilities
Scripts - Command line tools
Datasets - Dataset access
Catalog - Source catalogs
Astro - Astrophysics
Developer documentation
References
Changelog

Gammapy Docs News About CTA Contact Team Documentation

gammapy-webpage

package for gamma-ray astronomy

for gamma-ray astronomy

ray (CTA) science tools, and big gamma-ray telescopes.

started

gammapy-docs
A Python package for gamma-ray astronomy

Gammapy is a community-developed, open-source Python package for gamma-ray astronomy. It is a prototype for the CTA science tools. This page (<http://docs.gammapy.org>) contains the Gammapy documentation. The Gammapy webpage (<http://gammapy.org>) contains information about Gammapy, including news and contact information if you have any questions, want to report and issue or request a feature, or need help with anything Gammapy-related.

Getting started

Gammapy works with Python 2 and 3, on Linux, Mac OS X and Windows. To get started, we recommend you follow the installation and setup instructions in [Getting Started](#) and then learn Gammapy via the Jupyter [Tutorial notebooks](#).

- [Getting Started](#)
- [Tutorial notebooks](#)
- [Installation](#)

Gammapy package

As mentioned in the [Getting Started](#), the Gammapy package is structured as a series of sub-packages. We recommend that you start to learn Gammapy via the [Tutorial notebooks](#), and then consult the following pages for further information about each sub-package. Those pages also contain very detailed reference documentation for every function and class in Gammapy.

gammapy-extra

dr3-with-background
experiments Add sky image estimator prototype
figures clean up code and comments
logo Add Gammapy logo in PNG format
posters Adds poster presented at the MSSL/UCL Wavelength conference
presentations Add presentations/2017-01-13_Gammapy.pdf
test_datasets Add test_datasets/unbundled/tev_spectra/format_crab_sed.py
valhalla Add valhalla/cta_simulation
CHANGELOG
README.rst Remove



data experiments figures logo posters presentations test_datasets valhalla
clean up code and comments Add Gammapy logo in PNG format Adds poster presented at the MSSL/UCL Wavelength conference February ... Add presentations/2017-01-13_Gammapy.pdf Add test_datasets/unbundled/tev_spectra/format_crab_sed.py Add valhalla/cta_simulation
checks datasets experiments figures logo posters presentations test_datasets valhalla
move notebooks Update datasets/hess-dr3-dr1/hess-dr3-dr3-with-background.fits.gz Add sky image estimator prototype clean up code and comments Add Gammapy logo in PNG format Adds poster presented at the MSSL/UCL Wavelength conference February ... Add presentations/2017-01-13_Gammapy.pdf Add test_datasets/unbundled/tev_spectra/format_crab_sed.py Add valhalla/cta_simulation

gamma-cat

checks datasets experiments figures logo posters presentations test_datasets valhalla
move notebooks Update datasets/hess-dr3-dr1/hess-dr3-dr3-with-background.fits.gz Add sky image estimator prototype clean up code and comments Add Gammapy logo in PNG format Adds poster presented at the MSSL/UCL Wavelength conference February ... Add presentations/2017-01-13_Gammapy.pdf Add test_datasets/unbundled/tev_spectra/format_crab_sed.py Add valhalla/cta_simulation

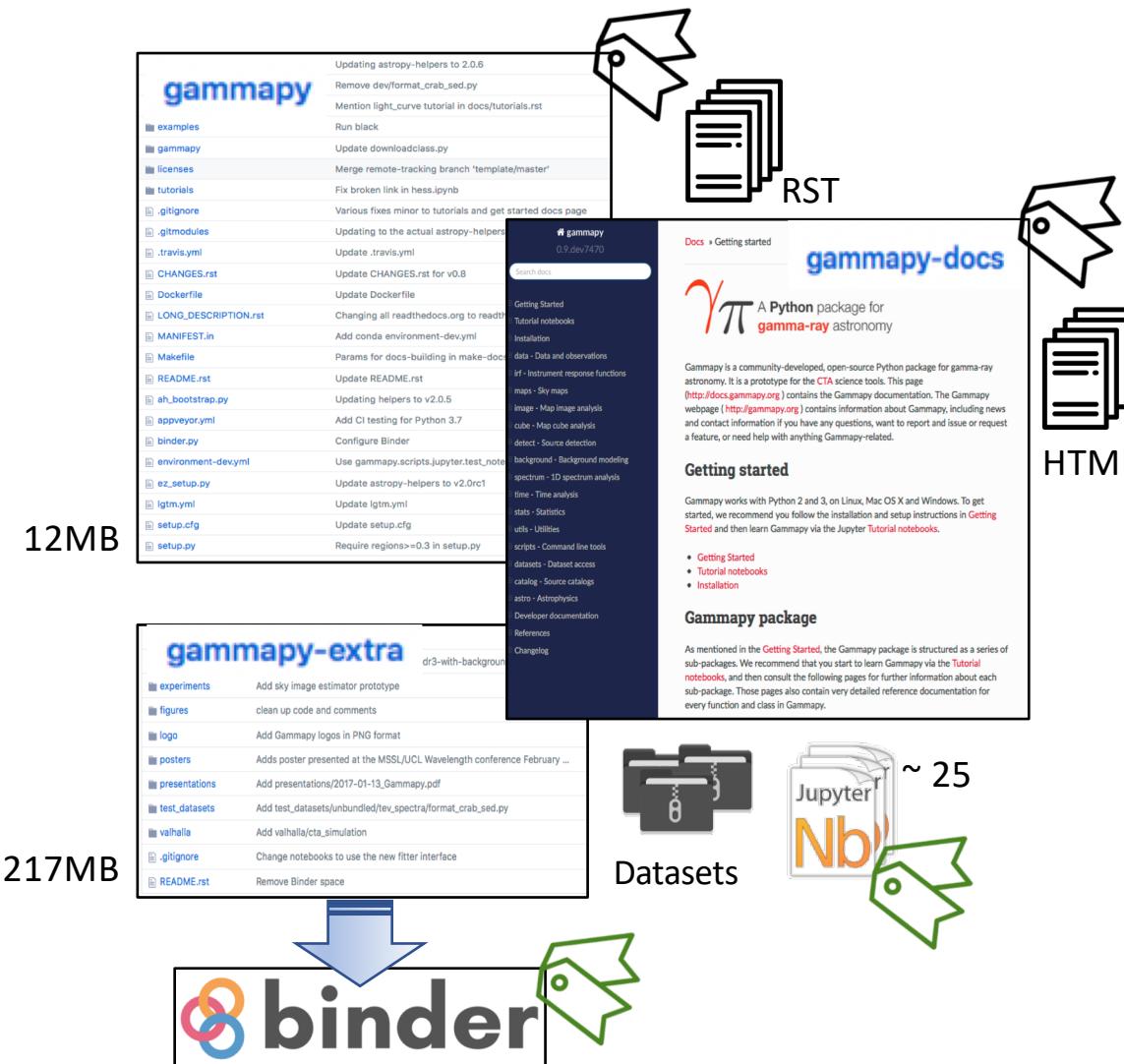
gammapy-fermi-lat-data

gammapy-fermi-lat-data

This repository contains pre-computed Fermi-LAT datasets.



Shell ★ 1 ⚡ 2 Updated 10 days ago



Documentation

- not coupled
- not versioned

- ✗ Rebuild old versions
- ✗ Reproduce notebooks

Notebooks working with old versions of Gammapy hard to recover

Binder not versioned, not accessible for old versions of the documentation



GitHub dependency

- Datasets are used in `gammapy.catalog`
- Datasets used in Tutorials
- Notebooks /Tutorials

```
git clone gammapy-extra
```

```
git clone gammapy-cat
```

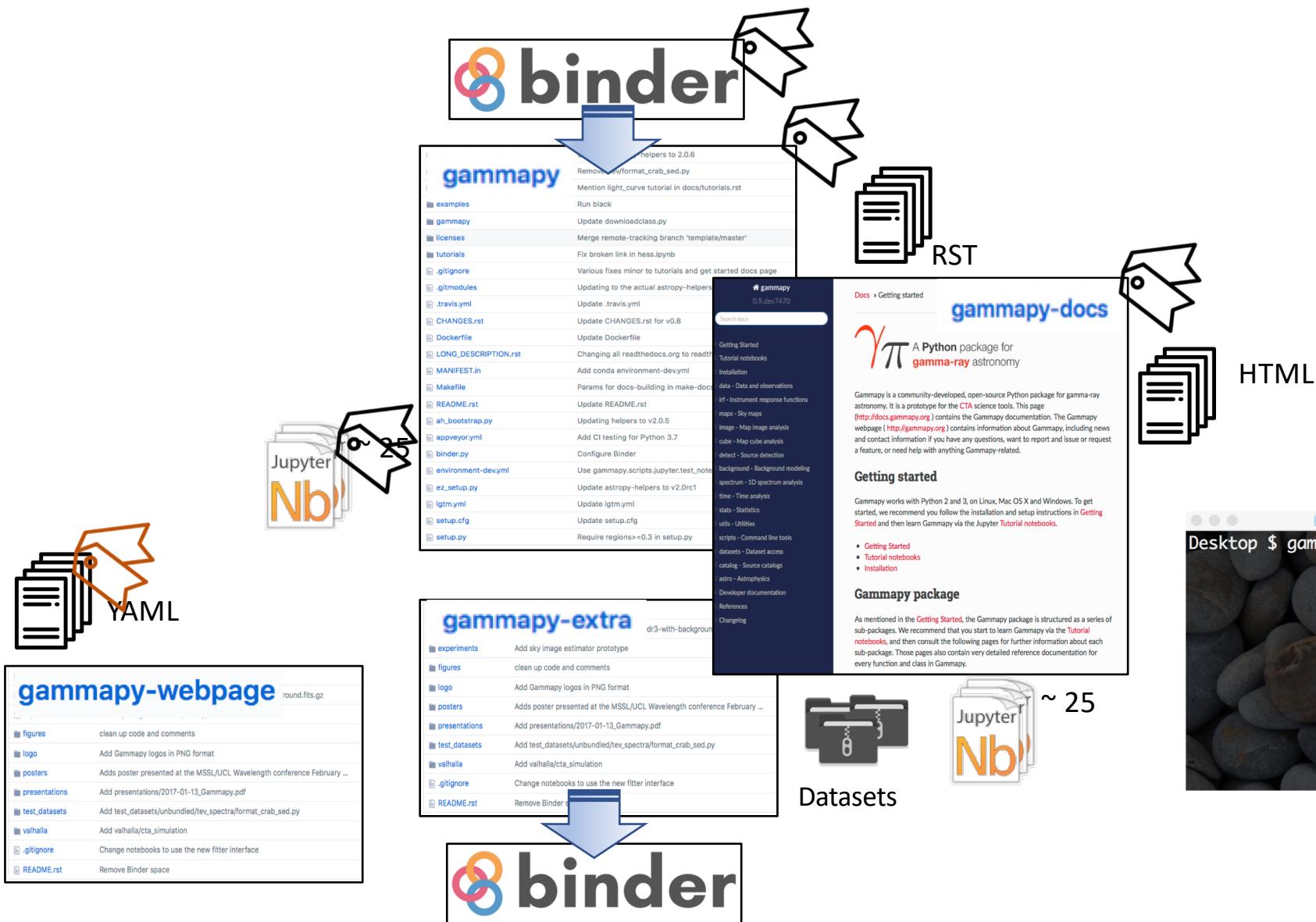
```
git clone gammapy-fermi-lat-data
```

The screenshot shows a GitHub interface with four repository cards:

- gammapy-extra**: Shows a single commit: "dr3-with-background.fits.gz".
 - files: `figures`, `logo`, `posters`, `presentations`, `test_datasets`, `valhalla`, `gitignore`, `README.rst`.
- gamma-cat**: Shows a single commit: "move notebooks".
 - files: `checks`, `datasets`, `experiments`, `figures`, `logo`, `posters`, `presentations`, `test_datasets`, `valhalla`.
- gitignore**: Shows a single commit: "clean up code and comments".
 - files: `checks`, `datasets`, `experiments`, `figures`, `logo`, `posters`, `presentations`, `test_datasets`, `valhalla`.
- gammapy-fermi-lat-data**: Shows a single commit: "Remove Binder space".
 - files: `checks`, `datasets`, `experiments`, `figures`, `logo`, `posters`, `presentations`, `test_datasets`, `valhalla`.

Gain control on what to provide to the end user

- How? `gammapy download` access interface + `GAMMAPY_DATA` environment variable
- What? Content declared in index files maintained by ourselves
- Where? Content and index files may be stored anywhere

$\gamma\pi$ 

gammapy download



```
$ gammapy download -h
Usage: gammapy download [OPTIONS] COMMAND [ARGS] ...
```

Download datasets and notebooks.

Download **notebooks published as tutorials and the related datasets** needed to execute them. It is also possible to download individual notebooks or datasets. The option `tutorials` will download by default the versioned file-structure of the tutorials into a `gammappy-tutorials` folder created at the current working directory. The option `datasets` will download by default the datasets used by Gammappy into a `gammappy-datasets` folder created at the current working directory. The option `notebooks` will download by default the jupyter notebook files used in the tutorials into a `gammappy-notebooks` folder created at the current working directory.



gammify download

```
$ gammify download -h
```

```
Usage: gammify download [OPTIONS] COMMAND [ARGS] ...
```

Download datasets and notebooks.

Options:

```
-h, --help Show this message and exit.
```

Commands:

```
datasets Download datasets
```

```
notebooks Download notebooks
```

```
tutorials Download tutorial notebooks and datasets
```



gammipy download

```
$ gammipy download -h
Usage: gammipy download [OPTIONS] COMMAND [ARGS] ...
```

Download datasets and notebooks.

Examples

```
-----
$ gammipy download notebooks
$ gammipy download datasets
$ gammipy download tutorials --release 0.8
$ gammipy download notebooks --src first_steps
$ gammipy download datasets --src fermi_3fhl --out localfolder/
```



gammapy download

```
$ gammapy download notebooks --src first_steps
```



```
$ gammapy download datasets --src cta-1dc
```



```
$ gammapy download tutorials --src first_steps --release 0.8
```



- datasets
- notebooks-0.8
- environment.yml

working with Jupyter notebooks



Jupyter notebooks are different after every execution
lot of hidden content and results

- diff comparison and code review almost impossible
- git repositories grow in size after every commit

```
$ du -hs gammipy-extra/  
619M gammipy-extra/  
$ du -hs gammipy-extra/.git  
444M gammipy-extra/.git
```

Let's strip the output !

3D analysis

This tutorial shows how to run a 3D map-based analysis using three example observations of the Galactic center region with CTA.

Setup

```
In [ ]: %matplotlib inline  
import matplotlib.pyplot as plt  
  
In [ ]: import numpy as np  
import astropy.units as u  
from astropy.coordinates import SkyCoord  
from gammapy.datasets import MapDataset  
from gammapy.data import DataStore  
from gammapy.lrf import EnergyDispersion  
from gammapy.maps import WcsGeom, MapAxis, Map  
from gammapy.cube import MapMaker, MapKernel, MapFit  
from gammapy.modeling import Models, SNeModel  
from gammapy.spectrum.models import PowerLaw  
from gammapy.image.models import SkyGaussian, SkyPointSource  
from regions import CircleSkyRegion
```

```
In [ ]: gammipy info --no-envvar --no-dependencies --no-system
```

Prepare modeling input data

Prepare input maps
We first use the DataStore object to access the CTA observations and retrieve a list of observations by passing the observations IDs to the .obs_list() method:

```
In [ ]: # Define which data to use  
data_store = DataStore.from_dir("%GAMMAPY_DATA/cta-1dc/index/gps/")  
obs_ids = [11038, 11110, 11119]  
obs_list = data_store.obs_list(obs_ids)
```





Jupyter notebooks in the repositories **MUST BE EMPTY**

The output is only visible in the HTML documentation or after local execution

```
gammapy jupyter --src mynotebook.ipynb strip
gammapy jupyter --src myfolder/ strip
```

Jupyter notebooks should have well formatted code cells

```
gammapy jupyter --src mynotebook.ipynb black
gammapy jupyter --src myfolder/ black
```

Jupyter notebooks/tutorials should not be broken

```
gammapy jupyter --src mynotebook.ipynb test --tutor
gammapy jupyter --src myfolder/ test --tutor
```

Jupyter notebooks may be executed from the terminal

```
gammapy jupyter --src mynotebook.ipynb run
gammapy jupyter --src myfolder/ run
```

```
gammapy jupyter -h
```

```
usage: gammapy jupyter [-h] [-f FILE] [-d DIRECTORY]
Perform a series of actions on Jupyter notebooks.

The chosen action is applied for every Jupyter notebook present in the
current working directory. Option --file allows to chose a single file
while option --fold allows to choose a different folder to scan. These
options are mutually exclusive, only one is allowed.

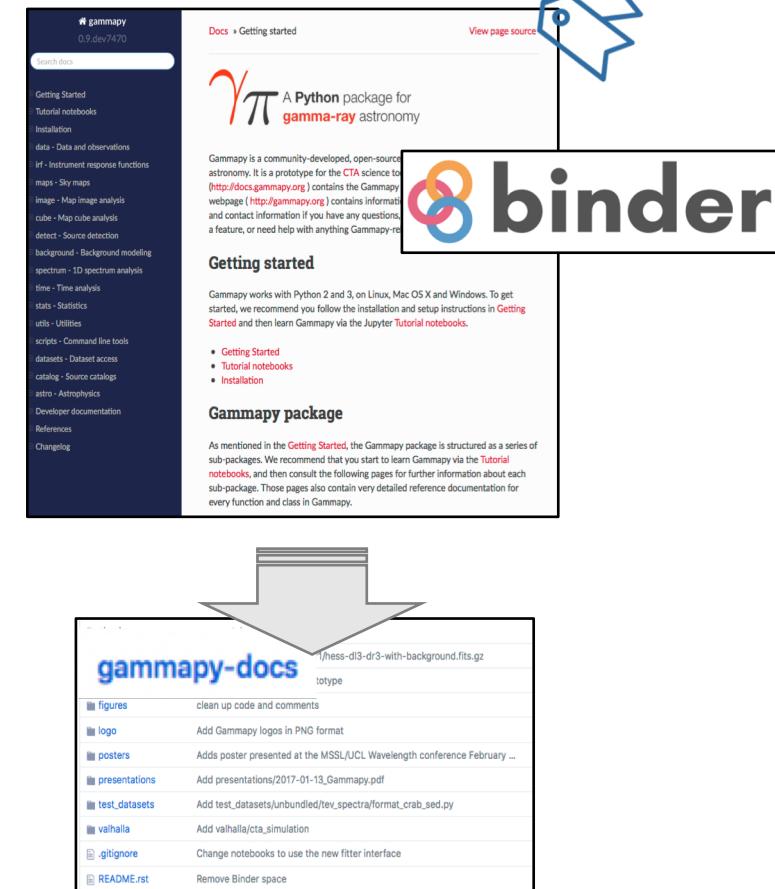
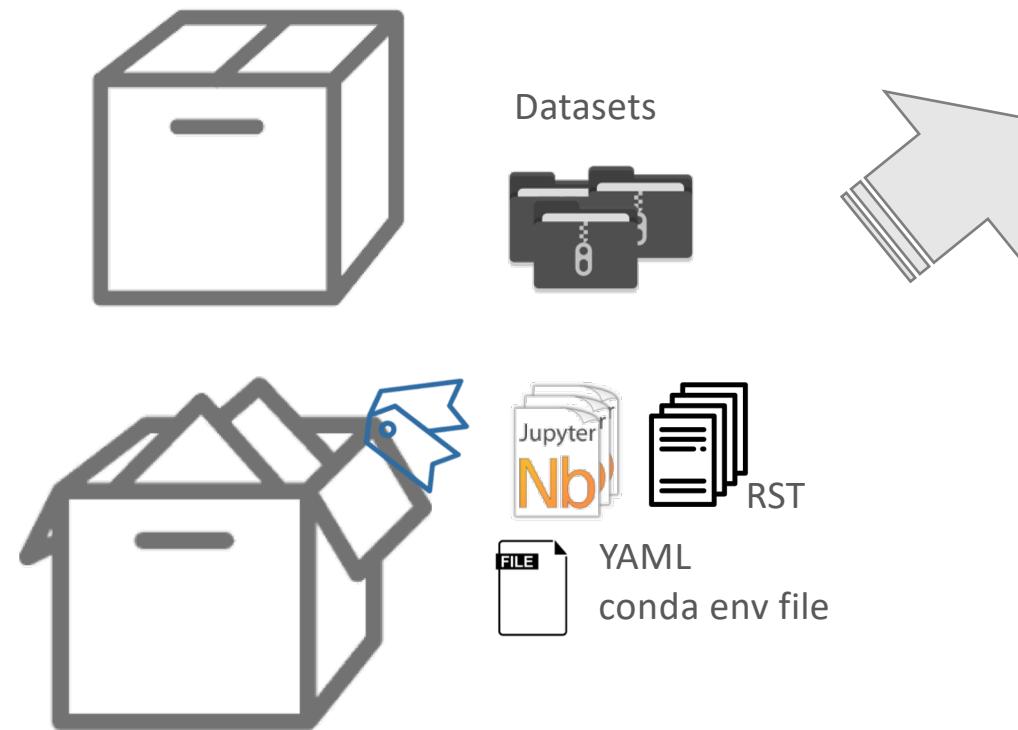
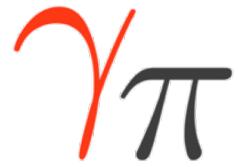
Examples
-----
$ gammapy jupyter strip
$ gammapy jupyter --src mynotebooks.ipynb run
$ gammapy jupyter --src myfolder/tutorials test
$ gammapy jupyter black

Options:
--src TEXT  Local folder or Jupyter notebook filename.
-h, --help   Show this message and exit.

Commands:
black Format code cells with black.
run Execute Jupyter notebooks.
strip Strip output cells.
test Check if Jupyter notebooks are broken.
```



documentation building





Choose and create a virtual environment for your documentation

```
curl -O http://gammify.org/download/install/gammify-0.8-environment.yml  
conda env create -f gammify-0.8-environment.yml  
conda activate gammify-0.8  
export GAMMIFY_DATA=/your/local/gammify/data/folder
```

Check out the version for the RST files and tutorials

```
git checkout v0.8
```

Build the documentation with tutorials

```
make docs-all release=v0.8  
  
strip output from notebooks  
black format code cells in notebooks  
test/run notebooks  
launch Sphinx
```

Fix notebook

```
make docs-all src=/path/to/notebook.ipynb release=v0.8
```

Fix RST file

```
make docs-all nbs=false
```

Questions

$\gamma\pi$

