Spatial models in Gammapy

gammapy coding sprint

Paris, 02/2019

Luca Giunti, giunti@apc.in2p3.fr

Spatial models – status

The base class

```python
class SkySpatialModel(Model):
    """Sky spatial model base class."""

    def __call__(self, lon, lat):
        """Call evaluate method"""
        kwargs = dict()
        for par in self.parameters.parameters:
            kwargs[par.name] = par.quantity

        return self.evaluate(lon, lat, **kwargs)
```

The models

SkyPointSource, SkyGaussian, SkyDisk, SkyShell, SkyDiffuseConstant, SkyDiffuseMap

Planar approximation or sphere?

- Planar model are easier to normalize and implement, and for most sources small angle approximation holds
- Planar approximation is valid in limited regions of a map (e.g. near the GP), due to projection and distortion
- Planar models need to be evaluated in pixel coordinates (x, y), but gammapy models use $(\texttt{lon}, \texttt{lat})$: intrinsically on the sphere

What others have?

Astropy	p
Sherpa	p
Astromodels	p/s
Gammalib	s

Planar approximation or sphere?

Normalization: $\int_{R^2} f\, dx\, dy = 1$ vs. $\int_{4\pi} f\, d\Omega = 1$

Evaluation: e.g. $dist = \sqrt{\Delta_x^2 + \Delta_y^2}$ vs. `angular_separation()`

Name	Normalization	Evaluation
SkyGaussian	s #2034	s
SkyDisk	s	s
SkyShell	p	s
SkyEllipse #2046	s	s
SkyPointSource		
SkyDiffuseConstant		
SkyDiffuseMap		

Spatial models – to do (PIG 7, #1663)

- Settle on the planar vs. spherical models question, and implement all the models accordingly
- Implement an elongated Gaussian, and possibly other models
- Define default parameters
- Expose model parameters as attributes
- Implement an `evaluation_radius` property for all models
- Allow initialization of the spatial positions with `SkyCoord`
- Add examples to the docs