



cherenkov  
telescope  
array



# Introduction to scientific python

Rubén López-Coto, INFN Padova  
University of Rijeka, 15/02/21



Funded by H2020 Marie Skłodowska  
Curie FELLINI - Grant 754496



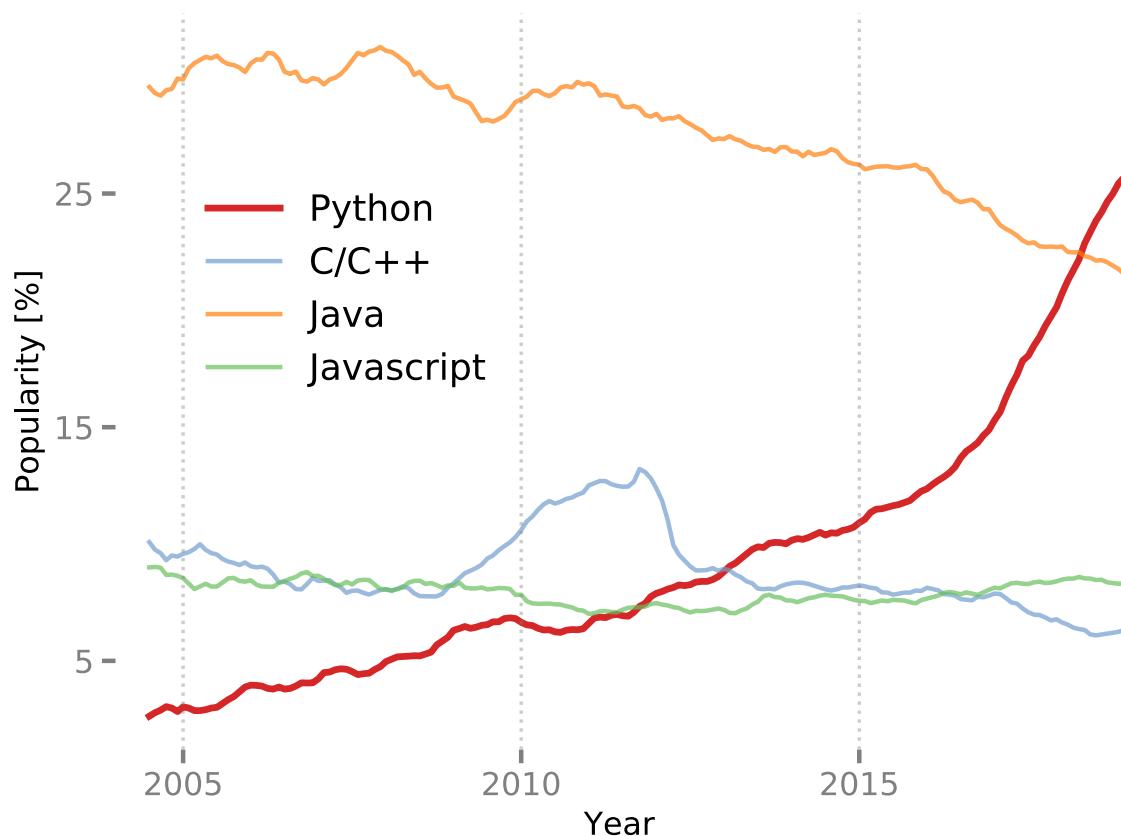


# Foreword

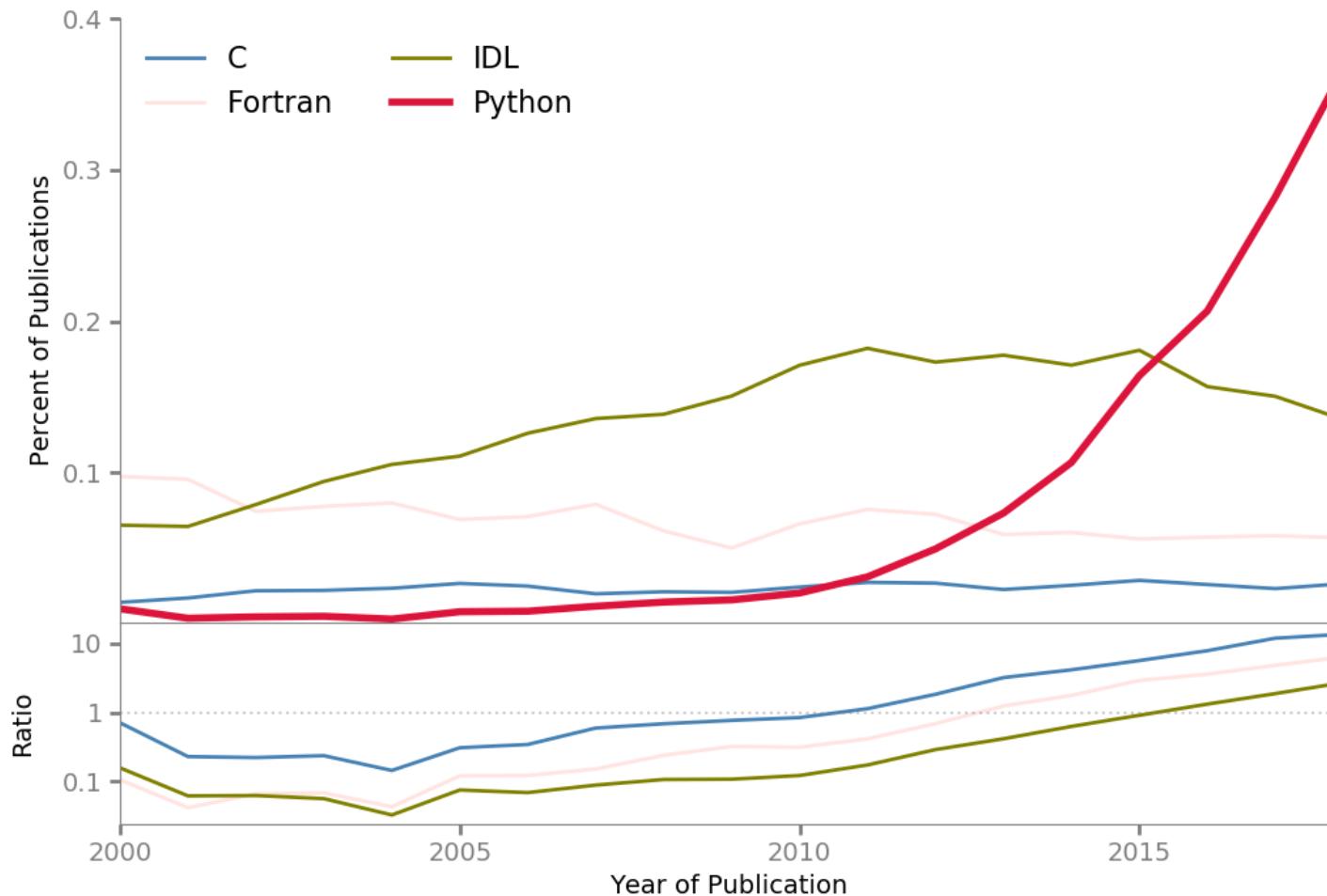
---

- This is not meant to be an intensive introduction of python or the scientific libraries that are used in the field.
- I will show four notebooks:
  - General introduction to python (`python_lesson.ipynb`)
  - Introduction to numpy (`numpy_lesson.ipynb`)
  - Fitting with spicy (`scipy_lesson.ipynb`)
  - Plotting with matplotlib (`matplotlib_lesson.ipynb`)
- But... why python?

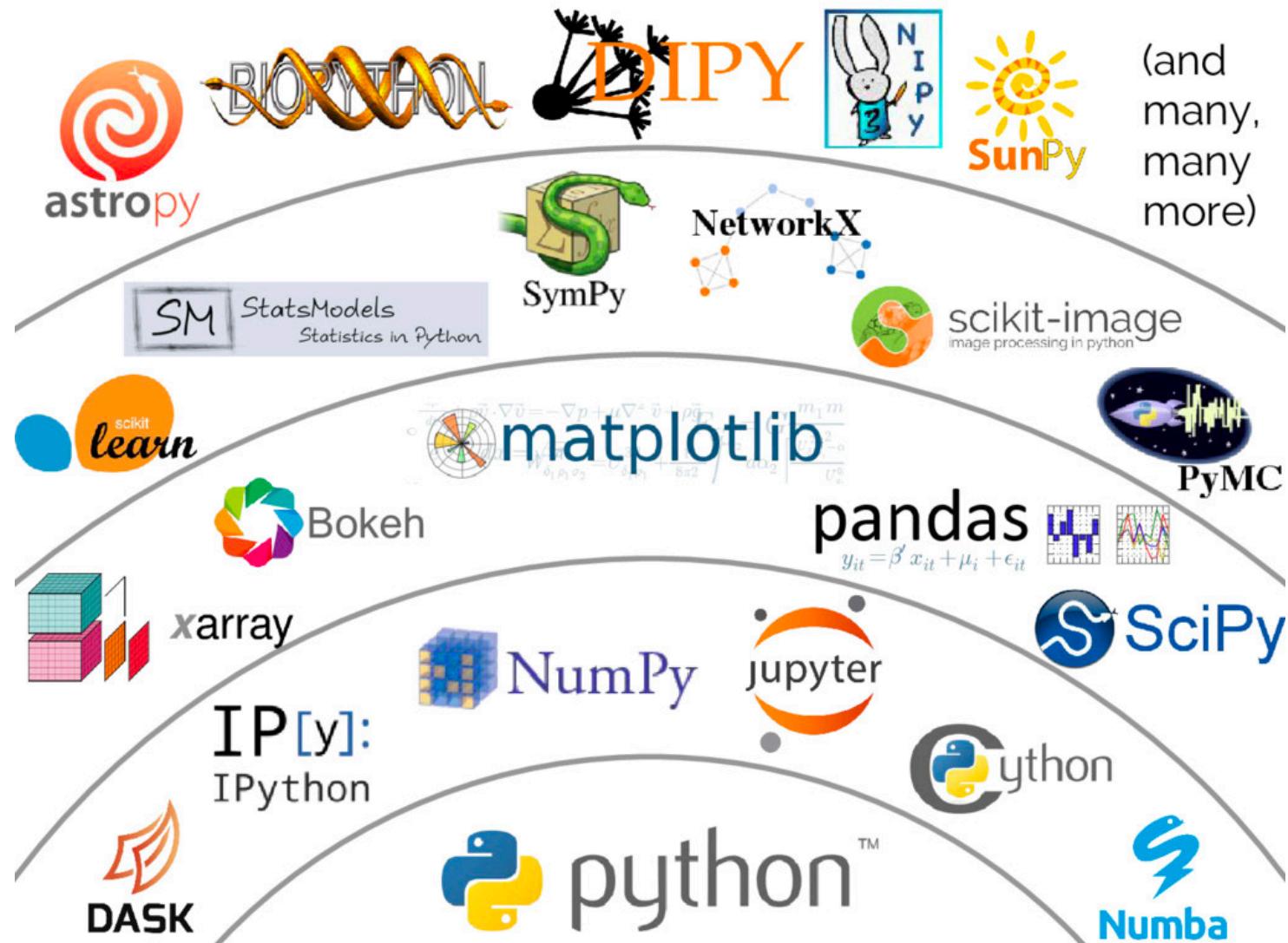
# Python popularity



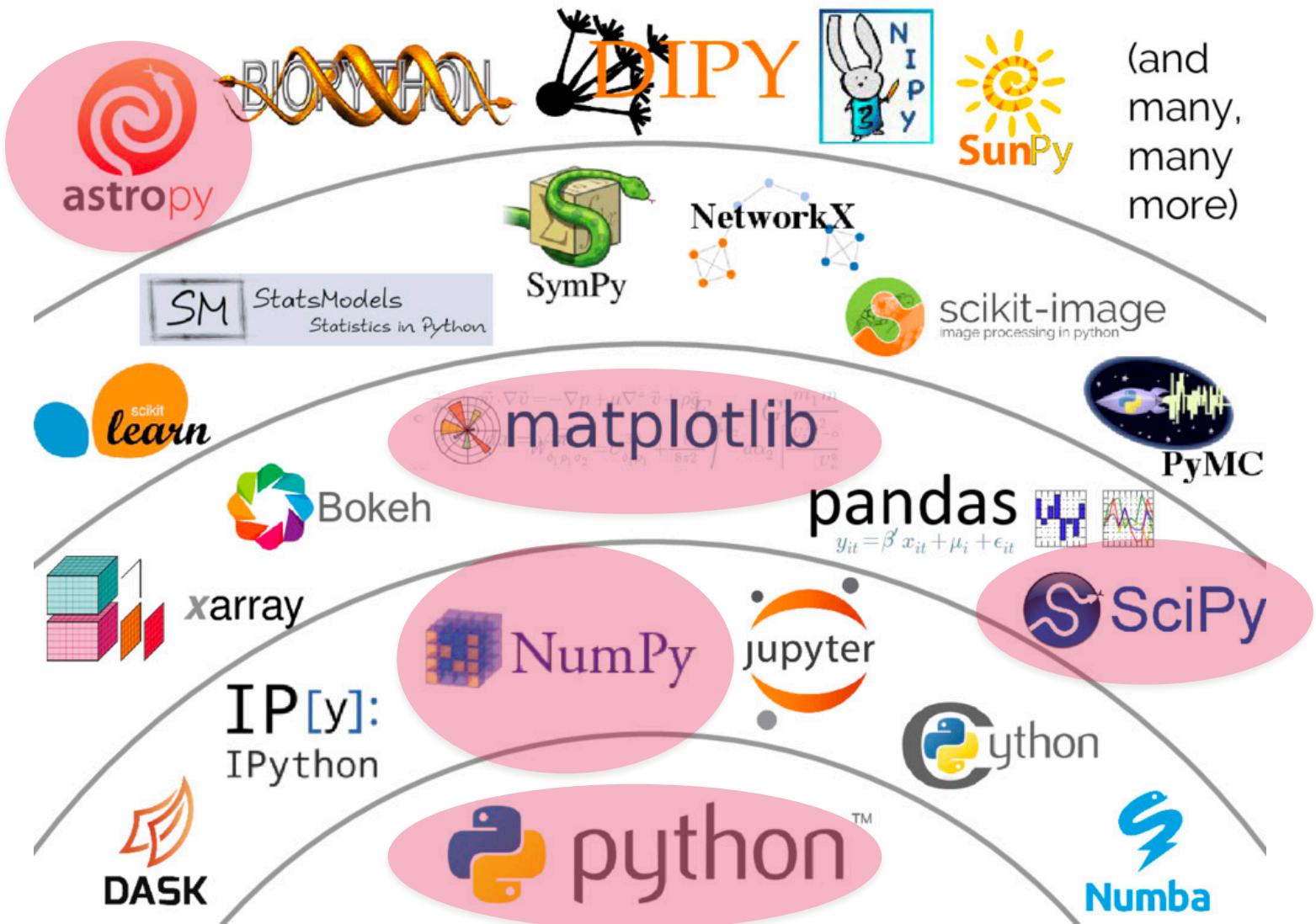
# Python popularity in astronomy



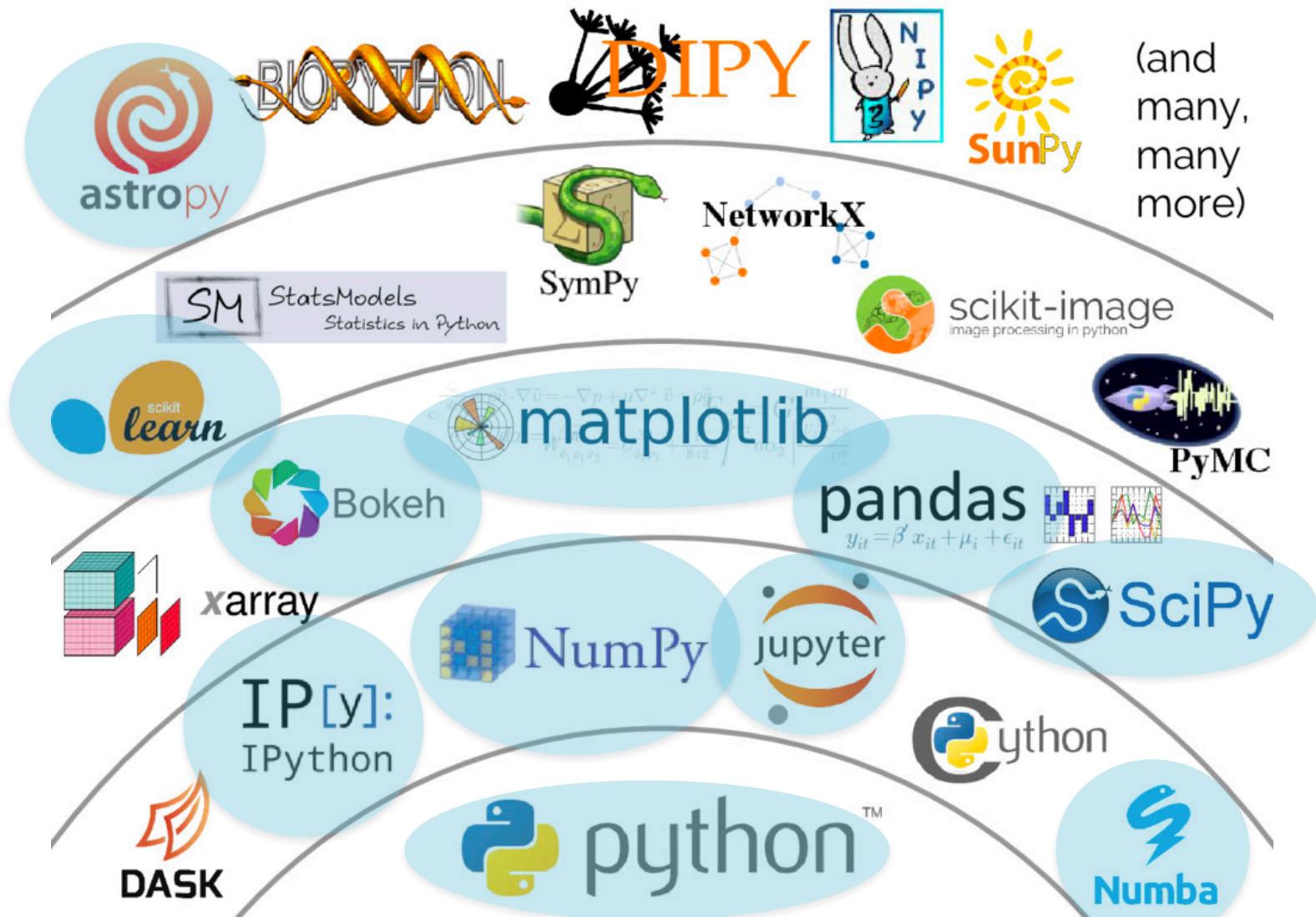
# Python libraries for scientific computing



# Python libraries for scientific computing covered in the following presentations



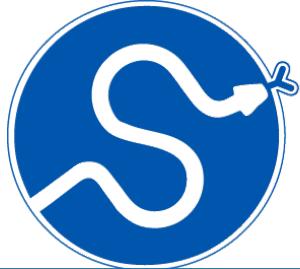
# Python libraries for scientific computing used in CTA





- Powerful n-dimensional arrays
- Numerical computing tools

```
>>> import numpy as np
>>> a = np.arange(15).reshape(3, 5)
>>> a
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14]])
```



**SciPy**



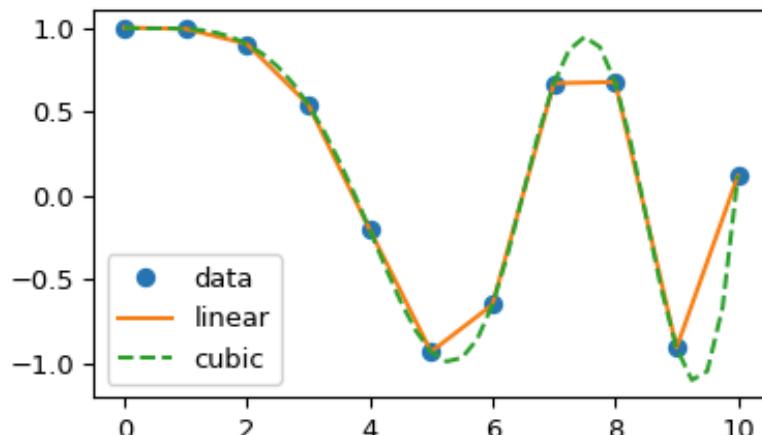
- 
- Numerical routines in python to perform tasks as:
    - Numerical integration
    - Interpolation
    - Optimization
    - Linear Algebra
    - Differential Equations solver
    - Statistics
    - Image Processing
  - Mainly written in python, although some of the underlying parts make use of C



# SciPy

cta

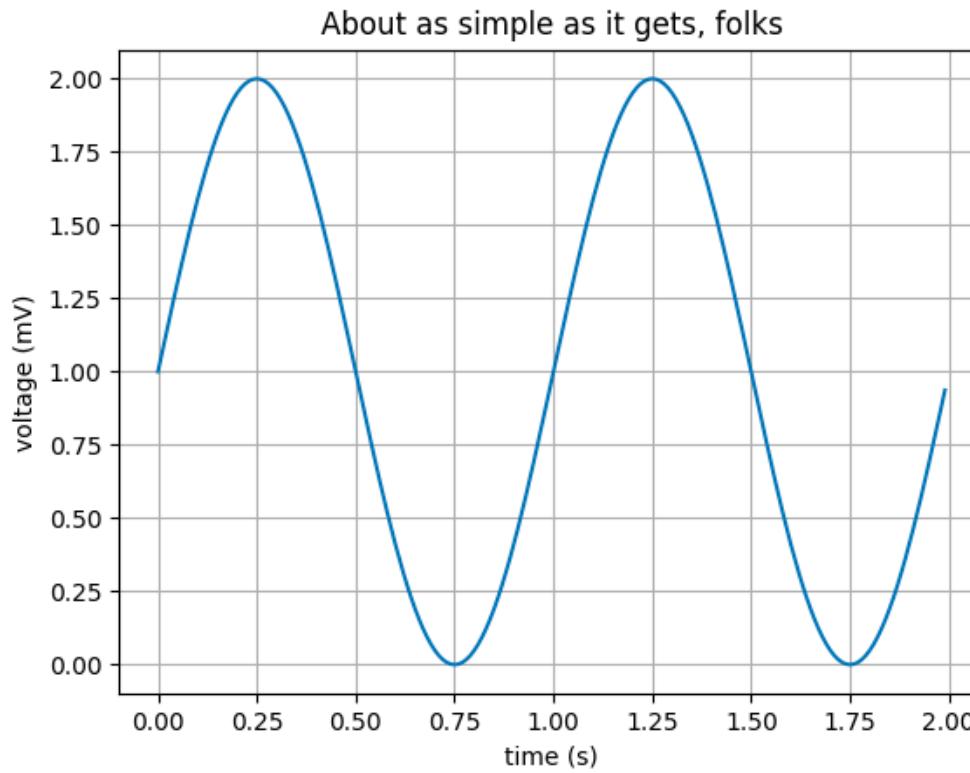
```
>>> from scipy.interpolate import interp1d  
  
>>> x = np.linspace(0, 10, num=11, endpoint=True)  
>>> y = np.cos(-x**2/9.0)  
>>> f = interp1d(x, y)  
>>> f2 = interp1d(x, y, kind='cubic')  
  
>>> xnew = np.linspace(0, 10, num=41, endpoint=True)  
>>> import matplotlib.pyplot as plt  
>>> plt.plot(x, y, 'o', xnew, f(xnew), '-', xnew, f2(xnew), '--')  
>>> plt.legend(['data', 'linear', 'cubic'], loc='best')  
>>> plt.show()
```



# matplotlib



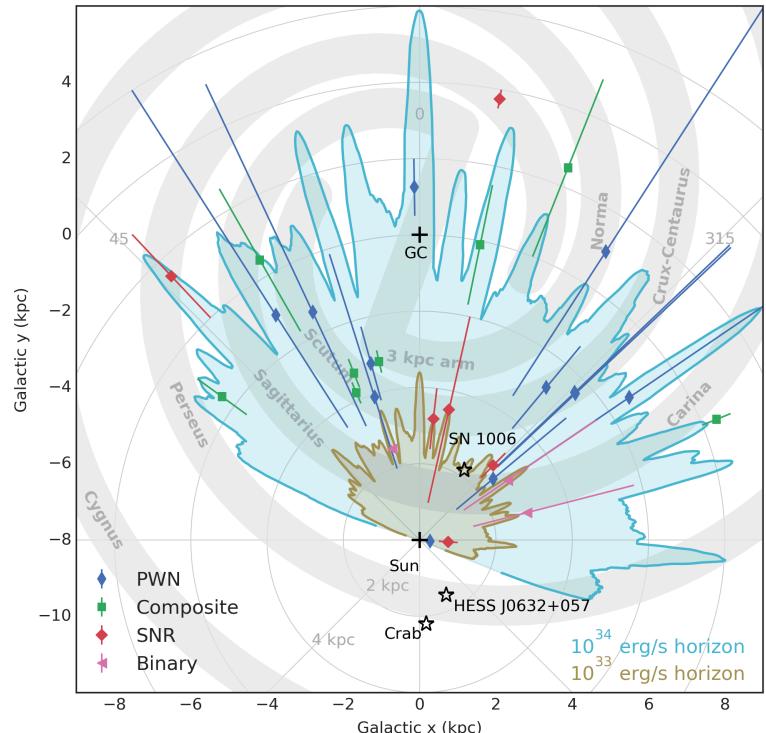
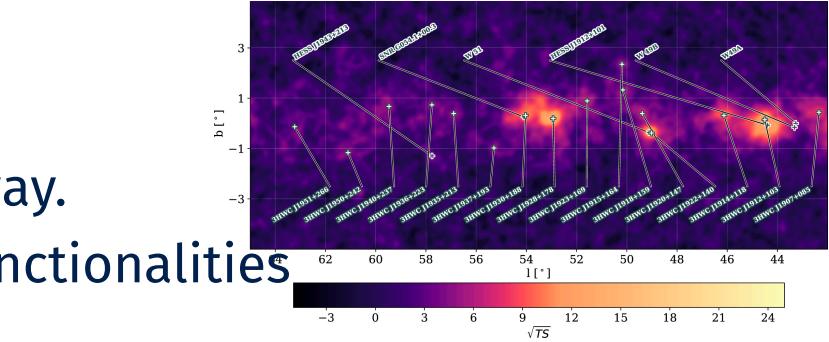
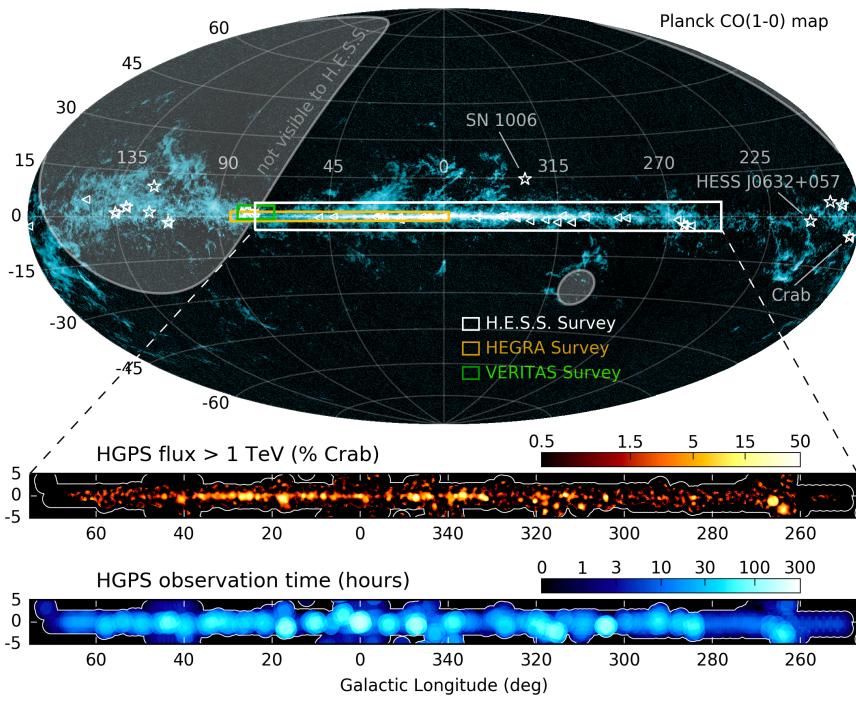
- Produce nice plots in a very easy way.



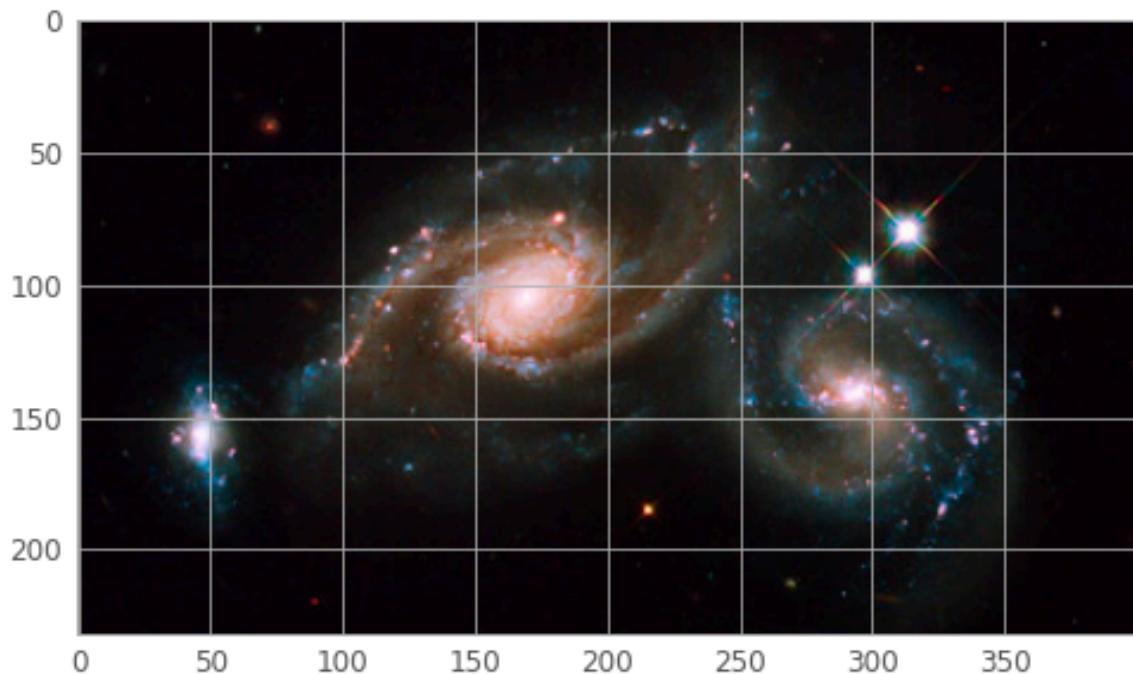
# matplotlib



- Produce nice plots in a very easy way.
- And some other more advanced functionalities



- (Demonstration Tomorrow)
- More specific software focused on astronomy.





# Hands on

- Run notebooks in the **notebooks/scientific\_python** folder
  - [python\\_lesson.ipynb](#)
  - [numpy\\_lesson.ipynb](#)
  - [scipy\\_lesson.ipynb](#)
  - [matplotlib\\_lesson.ipynb](#)

The screenshot shows a GitHub repository interface. At the top, there's a navigation bar with 'main' (dropdown), 'intro-iact-analysis / notebooks / scientific\_python /', 'Go to file', 'Add file', and a three-dot menu. Below the navigation is a commit history table.

Author	Commit Message	Date	Action
rlopezcoto	add missing files	fda2b7b on 6 Jan	History
..			
figures	add figures	last month	
README.md	Update README.md	last month	
matplotlib_lesson.ipynb	strip output from notebooks	last month	
numpy_lesson.ipynb	strip output from notebooks	last month	
python_lesson.ipynb	strip output from notebooks	last month	
scipy_lesson.ipynb	strip output from notebooks	last month	

The screenshot shows the content of the README.md file. It contains the following text:

**Introduction to Scientific Python**

Info from several workshops/presentations.

Notebooks based on the ones included in: [Scientific Python Notebooks](#) from Max Noethe

MPIK Feb. 2017 Coding sprint