Proper python

Tammo Jan Dijkema & Zheng Meyer
ASTERICS – Obelics workshop Annecy

June 2018

Python and paths

- Python is an interpreted language: running a program means running `python` with that program as argument
- There can be multiple `python`'s on your system:

```
which python
```

```
> import sys; sys.executable
```

Package managers

- Installing python packages is easy, and can be done in many ways:
  - Download package, manually copy to right path (please don’t)
  - Download package, python setup.py install (please don’t)
  - `pip install <package>` (please don’t this week)
  - `conda install <package>`
- This week, we will use conda to install packages
- Check where package got installed:

```
> import numpy; numpy.__file__
```

Isolated environments

- Isolate your work from arbitrary changes by using an isolated environment (reproducibility)
- My advice: use a package manager specifically for python
  - Do not use system python
  - Do not use homebrew python
- Two main ways to isolate environments:
  - Virtualenv
  - Conda
- This week, we will use conda

Interactive python

- python
- ipython
- jupyter notebook
- jupyter lab

Developing python scripts

- Any text editor:
  - vim, emacs, notepad.exe, gedit, jupyter lab
- IDE:
  - PyCharm, Eclipse, atom, …

Plan for today

- Check everyone’s environments
- Explore some (LOFAR!) data in a jupyter notebook
- Write a script in PyCharm to do something useful to this dataset
  - Follow PEP8 coding guidelines
  - Document code
  - Test code (with high coverage)
- Put this script in an installable package

Now: PyCharm

- Demo PyCharm