Intro 01 Python Ecosystem

Python is programming language. We all know that.

And without drowning in details about it let’s focus on some important ones:

* Runs in all platforms and it is free, in the money and open source kind of free
* It was created a while ago, in 1991, which means is not new, but mature
* just a bit of info, we will be using python version 3 or 3.7
* It is interactive. This is, you can see results while executing the code, and you can execute parts of the code
* One key characteristic of python is the multiple data types it can handle, and does it easily

Another big advantage is that python, as many other open source software, is modular. That means that packages or modules can be added to the basic core. And many if not most of these packages are developed by the community that has a need for them, therefore a community of experts.

Stackoverflow (and online community of coders) documents trends in technology since its conception in 2008. And since then, javascipt and python show the most growth… with python being the fastest growing language in the latest years.

Coming back to the modular structure…. Python has a number of packages that specialize in scientific computing.

* At the core, is python, and on top there are libraries or modules or packages like numpy and scipy that can handle arrays and numerical operations. And also jupyter that allows and interactive interface
* Based on those, there is other modules that integrate the previous ones like scipy module (not the library), and pandas (that handle tables) and matplotlib an module to generate plots.
* Using those modules, other packages can be fuilt, like cartopy that allows plotting on maps, xarray that allows more complex scientific data handling and other packages that can do more complex statistical and mathematical methods like machine learning or principal components.
* All together constitute a ‘branch’ in the python ecosystem, or the python’s scientific ecosystem.

The good thing about it, other than the community-based development, is that you only have to install, and more important, load only the modules you need for an application. For example:

Now, before we start with python proper, a word about finding help.

No secrets here. Just google it. There large and growing python community will help out.

Google will send you to the reference, some tutorials, or stack overflow q&a. It will give you a page like this: first a question, below answers from the community.

First read that the question make sense to you, and then check different answers. People have different ways to do things. Also people use different packages (that took me a while to understand).

Finally, see links we have included at the end of each section for more detailed tutorials or info.

And now, if your jupyter notbook binders are loaded, let’s begin!