

A Short and Incomplete Introduction to Python

Part 11: Ending remarks

Riccardo Murri <riccardo.murri@uzh.ch>,
Sergio Maffioletti <sergio.maffioletti@uzh.ch>
S3IT: Services and Support for Science IT,
University of Zurich

There is more to Python than this...

The time and scope of this course is quite limited.

Here is an (incomplete) list of Python features that you might want to look up as you become more experienced in the language:

- ▶ Object-oriented programming
- ▶ Iterators
- ▶ Decorators
- ▶ Class-level attributes, `classmethods`, `staticmethods`
- ▶ Properties and accessors
- ▶ Metaclasses

Other useful libraries

SciPy, <https://docs.scipy.org/doc/scipy/reference/>

A library of commonly used features in numerical programming (FFT, optimization, special functions, statistics, linear algebra, etc.)

skimage, <https://scikit-image.org/>

A collection of algorithms for image processing.

Other useful libraries

PyTorch, <https://pytorch.org/>

Deep Learning Framework

sklearn, <https://scikit-learn.org>

Machine Learning in Python

spaCy, <https://spacy.io/>

Natural Language Processing in Python

Other useful libraries: interactivity

Plotly, <https://plot.ly/python/>

Create interactive graphs and charts.

Dash, <https://plot.ly/products/dash/>

A Python framework for building analytics web applications entirely in Python.

Other useful libraries: performance

Dask, <http://docs.dask.org/en/latest/why.html>

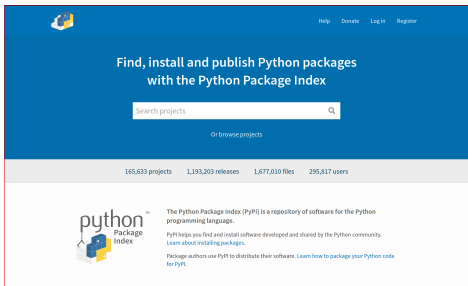
Transparently distribute NumPy and PanDas computations across multiple cores or even a batch-computing cluster.

Numba, <http://numba.pydata.org/>

A compiler to accelerate Python+NumPy functions.

Want even more? Look into PyPI

PyPI is *the* index of Python software packages. It currently indexes 165'633 packages, so the choice is really vast.



Almost all packages can be installed with a single command by running:

```
pip install packagename
```

Where to go now?

- ▶ **The Python tutorial**,
<http://docs.python.org/tutorial/>
- ▶ Dive into Python 3
<https://www.cmi.ac.in/~madhavan/courses/prog2-2012/docs/diveintopython3/index.html>
- ▶ The DataCamp courses (videos)
<https://www.datacamp.com/courses/tech:python>

For an extensive and commented list, see:
<http://python-guide-pt-br.readthedocs.io/>

Where to go now?

The Seaborn library comes with a good tutorial written by its author (note that -since Seaborn is an add-on to Matplotlib- some knowledge of Matplotlib is assumed):

<http://seaborn.pydata.org/tutorial.html>

Nicolas Rougier has written an excellent tutorial on the use of MatPlotLib: [https:](https://www.labri.fr/perso/nrougier/teaching/matplotlib/)

[//www.labri.fr/perso/nrougier/teaching/matplotlib/](https://www.labri.fr/perso/nrougier/teaching/matplotlib/)

Where to go now?

The “SciPy Lecture Notes” provide an introduction to using the scientific Python toolkits to solve a number of common problems in experimental and data science: <https://www.scipy-lectures.org/index.html>

Nicolas Rougier has also written a good tutorial on NumPy: <http://www.labri.fr/perso/nrougier/teaching/numpy/numpy.html>

Material from this course

All the material from this course is online:

<http://github.com/riccardomurri/python-for-science-intro/>

To run the notebooks and all the code from the course:

- ▶ install the **Anaconda Python distribution**; can be done on your computer, supports Linux, MacOSX, Windows:
<http://www.anaconda.com/>
- ▶ use **ElastiCluster** to build dedicated servers and clusters on the cloud:
<http://gc3-uzh-ch.github.io/elasticcluster>

Thanks!

Now go write some Python ;-)