

# Materials: Python Basics

pppppass

January 22, 2018

The information is updated on January 22, 2018.

## 1 Installation and configuration

There are Python distributions of Windows version, but it is not recommend because it takes time to get around the differences between Windows and Unix-like systems, including line feed (`\n` in Linux but `\n\r` in Windows), path separator (`/` in Linux and `\` in Windows) and so on. Additionally, some packages like NumPy and PyTorch does not work well in Windows.

Python is included in many Linux distributions, but use Anaconda is preferred, in order to create virtual environments and avoid version conflicts.

## 2 Resources

If you are familiar with C++ or some other object-oriented language, official <https://docs.python.org/3.6/tutorial/index.html> should be sufficient as a tutorial. Details in Section 6.4, Section 9.5.1, Section 10–13, Section 15–16 may be skipped. The outline is based on this tutorial. As a official tutorial, it provides basic ideas of Python and is strongly recommended.

However, if you cannot stand so much English, Xuefeng Liao's *Python Tutorial* is also recommended. This tutorial covers more topics than the official Python Tutorial, among which some parts are too detailed as a introductory tutorial instead of a manual and beginners should omit them. The sections before IO Programming inclusive should be adequate.

It is strongly recommend to read through *PEP 20 – The Zen of Python* to get familiar with the ideas of Python. Note that *PEP 8 – Style Guide for Python Code* is also a standard documentation about Python code styles. As component of a project, this guide are detailed,

so pay attention not to be trapped in details. Note that PyCharm and some other IDEs have PEP8 code formatting support.

For detailed questions and bugs, Stack Overflow is a great website to turn to, which is a community for programmers and many Q&A can be found. It's suggested to search Stack Overflow first if bugs come into place. Recommendation about materials and packages can also be found there, and some question about environments may help at some point.

### 3 Assignment

The assignment is listed below.

- (1) **(Required)** Permutation: use a generator function to generate all the  $n$ -permutations. Files are in `Assignment/Python-Permutation`.
- (2) **(Optional)** Least Square Regression: encapsulate codes of gradient method to a least square regression into a class `Trainer`. Files are in `Assignment/Python-LeastSquare`.
- (3) **(Optional)**  $K$ -Means: implement  $K$ -Means algorithm to a specific 2-dimensional dataset. Files are in `Assignment/Python-KMeans`.