

Materials: NumPy

Yixuan Wang ppppass

Update on March 4, 2018

The information is updated on March 4, 2018.

1 Installation and configuration

To install NumPy, it is suggested that you should use the distribution of Python, Anaconda, for it includes all the key packages. Alternative ways are to install NumPy via Pip, or via package managers such as Ubuntu or Homebrew. You can refer to *installing instructions on the official website of SciPy* for detailed instructions.

2 Resources

For beginners, it is strongly recommended to visit this website of *quickstart tutorial of NumPy*. It is easy to read and comprehend, in the meantime would prepare sufficient knowledge for you to handle preliminary tasks of NumPy.

However, for a thorough understanding of how NumPy functions, you'd better go through the *NumPy v1.14 Manual*. In particular, you could use *NumPy basics* for advanced knowledge of NumPy, and *NumPy Reference* in search of NumPy functions.

A Chinese documentation, *NumPy 的详细教程* is also listed here. In fact, it is simply a Chinese version of the official documentation *quickstart tutorial of NumPy*.

From my personal perspective, the Python NumPy tutorial of the class *CS231n* is strongly recommended due to its conciseness and explicit statement of ideas.

If you are familiar with Matlab, refer to *NumPy for Matlab users* for a quick start of NumPy. This piece of material identifies several crucial differences and similarities between NumPy and Matlab.

Of course, you can always turn to Stack Overflow for discussion and some specific problems you have encountered.

3 Assignment

It is recommended to find supplementary materials of your own to enhance your command of NumPy. Just for reference, this repository *100 numpy exercises* on GitHub serves as a good example.

The assignment for week 2 is listed below.

(1) **(Required)** Polar Coordinates: Consider a random 10×2 matrix representing cartesian coordinates, convert them to polar coordinates. See [Assignment/NumPy-PolarCoordinates](#) for details.

(2) **(Optional)** Rank Computation: Compute the matrix rank of a random 10×10 matrix without invoking the `rank(numpy.linalg.matrix_rank())` method. See [Assignment/NumPy-RankCompu](#) for details. (Hint: use SVD)

(3) **(Optional)** Extracting Submatrix: Extract all the contiguous 3×3 blocks from a random 10×10 matrix. See [Assignment/NumPy-ExtractingSubmatrix](#) for details.

(4) **(Optional)** Intensive Broadcasting: Solve a system using intensive boardcasting. See [Assignment/NumPy-IntensiveBroadcasting](#) for details.

(5) **(Optional)** From Function: Use `numpy.fromfunction()` and `numpy.indices()` to create two Laplacian matrices and *explain how `numpy.fromfunction()` works and the relation between `numpy.fromfunction()` and `numpy.index()`*. See [Assignment/NumPy-FromFunction](#) for details.

(6) **(Optional)** Random Choice: Use Boolean slice and `numpy.random.choice()` respectively to perform random choice over a given matrix. See [Assignment/NumPy-RandomChoice](#) for details.

(7) **(Required)** Expectation Maximization: Use Expectation Maximization Algorithm to handle a Gauss Mixture Model. See [Assignment/NumPy-ExpectationMaximization](#) for details.