Week 6 - Scientific Computing

Hongzheng Chen

Mar 22, 2020

1 Scientific Computation

Scientific Computation is such a large field that we cannot cover all the topics here. Princeton publishes an online book named *Computer Science: An Interdisciplinary Approach*, which gives a brief introduction to most of the topics of CS and also talks about some commonly-used methods and algorithms in scientific computation in Chapter 9. You can quickly get through it.

After the discussion on the seminar, you may notice scientific computing is actually an interdisciplinary field that involves linear algebra, calculus, numerical computing, optimization theory, and even computer system and architecture. Therefore, to make these scientific applications work, it is important to learn the principle of nowadays computers. To have a better understanding of how the programs work and how to make the applications run faster, I again recommend you to read CSAPP.

2 Package and Environment Management

I suppose you have installed the required software like Anaconda and pip. (If not, please refer to the material of Week 05.) To accelerate the downloading of packages, it is recommended to change the source URL of pip and conda to Tsinghua mirror, and guidance can be found here[1].

Though the Python committee has announced they will not provide support for Python 2.x from 2020, there are still many packages using Python 2.x including the Ubuntu system. Thus, to better manage your packages in different versions of Python, you'd better create a virtual environment when you begin a new project. conda has initial support for creating a new environment, please read the official document for instructions. Otherwise, you can install virtualenv via pip. Some commonly-used instructions can be found here.

Moreover, if you want to make your Python project correctly run on other computers, you should list out the third-party packages you used and install them on that computer. An automatic way is to use pipreqs to generate the dependency. The instruction can be found here.

Lastly, it is highly recommended to install Jupyter Notebook, a powerful web-based interface for data science. It has an inherent interpreter, Markdown engine, command line, and file system support, which will be very useful when you login a remote server. Tutorials and Advanced tips of Jupyter Notebook are listed. Otherwise, you can install Jupyter Lab, which is the next generation of Jupyter Notebook with more powerful programming supports.

---

[1]The Tsinghua conda mirror seems to get into trouble these days, so you can install packages by pip instead.

3 SciPy & NumPy

In our seminar, we only cover the usage of Numpy, Matplotlib, and pandas. For other packages like SciPy and SymPy, they are all very useful in mathematics and engineering. Thus, if you have spare time, you can get in and have a look. In the following, we only give some materials of NumPy. Actually, for the packages above, you can easily find the tutorials in Chinese on Baidu or Zhihu.

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 Manual. In particular, you could use NumPy basics for advanced knowledge of NumPy, and NumPy Reference in search of NumPy functions.

The Python NumPy tutorial of the class CS231n is also strongly recommended due to its conciseness and explicit statement of ideas.

4 Assignments

NumPy is the basics of further topics like machine learning and deep learning and important for you to master it. NumPy-100 servers as a good exercise for you to enhance the understanding of NumPy.

The required assignment for this week can be found in `Assignments/NumPy-Basics/main.ipynb`. Please use Jupyter Notebook to open.

Notice: Before you do the assignment, you'd better get familiar with the basic operations of NumPy.