Python packages installation

In order to run the python scripts developed for different lab practical sessions, you will first need to install some compulsory python libraries and packages. In what follows, we will introduce you some knowledge about these libraries and packages. We then show you a step-by-step tutorial as to how to install them. It will take 5-10 minutes to finish the process.

Introduction of some basic python libraries

In the first lab practical, you need `python 3.6`, `pip`, `juypter notebook`, `numpy` and `matplotlib`.

`python 3.6` is the programming language we used for all assignment in this module. Python is a great general-purpose programming language on its own. With the help of a few popular libraries (openCV, torch, tensorflow) it has become a powerful environment for scientific computing and rapid application development. The language design is distinguished by its emphasis on readability, simplicity and explicitness. Python code is often said to be almost like pseudocode, since it allows you to express very powerful ideas in very few lines of code while being very readable. Some people go so far as to liken it to *executable pseudocode*.

`pip` is the standard package manger for Python. It allows you to install and manage additional packages that are not part of the Python standard library.

For example, if you want to use functions from `numpy` package, you will need to install the library. With `pip` you can do something like this: `pip install numpy` `jupyter notebook` is an web based applications that allows you to create and share documents that contain code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, etc. There are other more advanced integrated development environments (IEDs) that allow you to write your Python code in a very friendly way, including Pycharm, Spyder, Visual Studio Code and Thonny. The first two are normally very popular in Python users.

`numpy` is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

`matplotlib` is a plotting library for the Python programming language and its numerical mathematics extension numpy. You are able to draw a lot of beautiful plots with this Python package.

Installation

You are now ready to install these packages with any one of computers in LG04 and UG04. Assume you are now logged in with your CS account. For example, my CS login is *duanj*. Once you are in, open the linux terminal and type in

```
module available
```

and hit enter button, you will be able to see `python/3` has been pre-installed in this computer. You will need to load `python/3` by typing in the following in your terminal

```
module load python
```

After you have done so you will be able to see *Loading python version 3* displayed in your terminal. If you further type in

```
python --version
```

you will see the Python version you used, which is *Python 3.6.3*, which is the one I introduced at the beginning. Meanwhile, you will find `pip` works as well after you load the module, which means you can use `pip` to install other packages. Let's install `numpy` by typing the following in your terminal

```
pip install numpy --user
```

Easy as it looks right. Note that there is an extra `--user` flag here, which will cause the package to be installed inside the user base binary directory. Under my login this package is now in `/home/staff/duanj/.local/lib/python3.6/site-packages`. It is obvious to see Python associated packages will be installed in`site-packages` under `python3.6`. We also note that the `--user` flag should be used for all associated packages installation. Next in terminal type in

```
pip install matplotlib --user
```

Afterwards, install `jupyter notebook` by typing in the following

```
pip install jupyter notebook --user
```

which installs the software in `/home/staff/duanj/.local/bin`. After the installation is done, type in:

```
.local/bin/jupyter-notebook
```

To see Jupyter notebook user inference, copy the web address displayed in the terminal and paste it into your firefox browser. Done all installations.

Next, you need to download `lab1_python.ipynb` and save it somewhwere in your desktop. Inside Jupyter you navigate to this file and click to open it. You are now ready to start doing the contents in the tutorial. Have fun of programming!