

Install Python using MacPorts (from beginning)

Titipat Achakulvisut

October 13, 2014

Hello Daneil, here I write some documentation how to install python (2.7) and its packages in Mac OSX. It is not an official document but probably an easy way to get start with Macports! Macports depends mostly by its community and ones who update the port. Some ports does not work very well e.g. they may have not updated the port when Apple change operating system. From my experience, it is normal to have some port problem with macports when new Mac operating system like Maverick was announced or new XCode version were updated. It will take sometime (like a week to a month) for them to update the new port that are compatible with new operating system. But yeah, most ports work very well even though OS changes since the community still big enough and most people still use it to install program!

Macport

In order to install Macports. At first, we need to install Xcode (see also the version of Xcode you have is not too old) and its command line tools, X11 and some other things first. Then we will install Macports from the [macports](#) website (see your OS and download the right one). All available ports are at this [site](#). Note that the port name that are related to python27 will be in the form of py27-... e.g. py27-ipython, py27-tornado, py27-numpy, py27-scipy, py27-pandas, py27-scikits-learn. The name is different from pip install in those who use Linux operating system. More note is that make sure that we have installed the command line in XCode. In order to check whether we have installed command line in XCode, you can type as follow:

```
> sudo xcode-select --install
> sudo port -v selfupdate
```

As I have seen from now selfupdate should not take that long time. If it takes too long, probably something was wrong such as XCode version and Macports that we installed.

Python 2.7

After we get Macports, there are almost 18000 ports that we can install from git, emacs, vim to video downloader like youtube-dl and music player like cmus. Sometimes, it is nice to play music from yourterminal :P. A lot of ports will have dependencies (we can check on the website above). When we download one port, its dependencies will come together. However, when we want to uninstall each port, we again need to uninstall its dependencies first before uninstall it. For python27 (or python version 2.7), I have some recommendation command line as follow:

```
> sudo port install vim
> sudo port install emacs
> sudo port install git
> sudo port install python27
```

```
> sudo port install py27-matplotlib py27-numpy py27-scipy py27-ipython +notebook
> sudo port install opencv +python27
> sudo port install pip
```

Emacs and vim are classic text editor and also in this case, we use it to prove that our installation before works. Pip is a package management to help you install python packages. We can use pip to install python library and update library. And to check recent version, we can type

```
> pip list --outdated
```

```
> sudo pip install 'package name'
> sudo pip install --upgrade 'package name'
```

Note that the ipython notebook may fail to install with the version of py27-tornado as I faced one time (version of tornado <= 3.1.0). This can be solved by download the source file from tornado and then we can create the folder to store that file or we can use git to clone the source file: `git clone https://github.com/tornadoweb/tornado`. Suppose we save the source file at desktop, we can change directory to that folder where they have `setup.py` code and install as follow:

```
> cd Desktop/tornado
> python setup.py build
> sudo python setup.py install
```

To check the tornado version type `python` in command line then `import tornado` and then type `tornado.version` to see if the version is compatible for ipython notebook i.e. 3.2.0 We also have other ports that related to python27 that can be easily installed such as:

```
> sudo port install youtube-dl
> sudo port install py27-pandas
> sudo port install py27-scikits-learn
> sudo port install py27-pyside
```

If you want to see more, just go to the available port sites. And some errors can be solve by googling! Moreover, we sometimes will install python3.3 in the future. We can choose the version of python that we have installed as follows:

```
> sudo port select --set python python27 (or python33 in the future!)
> sudo port select --set ipython ipython27
```

To uninstall the port just use, for example,

```
> sudo port uninstall 'portname'
> sudo port uninstall --follow-dependents 'portname'
```

It is good to check at directory that we have PATH of Macports included in `.bash_profile` file (e.g. if port is not recognized). We change directory and check by `typels -la` and use text editory

```
> cd ~
> ls -la
> emacs .bash_profile
```

In the file, we should have this kind of path included:

```
export PATH=/opt/local/bin:/opt/local/sbin:$PATH
```

Run Python

We can run python by typing given command line in the terminal:

- `ipython` to run ipython on terminal
- `ipython notebook` to run ipython notebook on specific folder that we are in right now in terminal

Some Useful Command Line

We have some common command line that we will use a lot to check the port and see whether which one are currently active since we will install a lot of them. We also need to know how to upgrade the port itself and to install new updated library/ packages we have installed.

- `port installed`
This one will print the list of port that we have installed and will note active if the port are currently in used
- `sudo port selfupdate` or `sudo port -v selfupdate`
this command line is used to update the Macport itself. A lot of flag can be added such as `-d`, `-v`, `-f`, to see how it works just google them.
- `sudo port upgrade outdated`
this command line is used to update the port that we have installed. So the new macports will get updated once in a while. This way is very nice way to install new updated port without keep track of the newest version all the time. If you do not want to keep older version of ports, use `sudo port -u upgrade outdated` instead.
- `sudo port uninstall inactive`
macports suggests to run this line once in a while to uninstall inactive ports.
- `sudo port -fp uninstall installed`
I know sometime we mess up the port and have no way to fix it. This is how to uninstall all ports and lets start installing all again!

Some UNIX Command line

- `history`
This line is to see the type history that you used before.
- `cd ...`
Simple change the directory. `cd ..` is use to change directory back to former folder
- `which python/ which ipython`
See the directory of `python` that we used, or `ipython` that we currently use
- To check all the path that included in your python, you can open `python` or `ipython` and then use `sys` library to print the path included in your python

```
import sys
sys.path
```

Emacs Shortcuts

Here is the [link](#) to Emacs shortcut which you may find it helpful in typing text/code in terminal application. If you don't like terminal, I also recommend iTerm which is the terminal application but more beautiful and you can open multiple tabs using it. And some trick in terminal, you can hold **option** key and click in terminal page to go to that point/line without using keyboard too!

My Comments on Macports

A lot of code are developed in Linux as you know. Therefore, most of the time, installer in Linux mostly works. However, as a Macports user, I still got problems sometime installing programs via Macports. But, on the bright side, Mac is still good for most of the thing other than programming development comparing to distribution of GNU/Linux such as *Ubuntu*, *Mint* which can be spy by NSA as Richard Stallman **claim** (I'm kidding).

People also compare MacPorts with Anaconda. In my opinion, Anaconda packs all the libraries that people needed in **python** together. However, MacPorts are broadly used **not** only as python installer but other programs as well. With Macports, it is easy to install and manipulate version of packages you installed and give you more freedom to handle Python.