Udemy-notes (/github/jmportilla/Udemy-notes/tree/master) / Course FAQ.ipynb (/github/jmportilla/Udemy-notes/tree/master/Course FAQ.ipynb)

FAQ FOR THE COURSE

INSTALLING PYTHON

How do I install Python?

It is HIGHLY suggested you use the Anaconda distribution for this course and installing packages. You can install the Anaconda distribution here:http://continuum.io/downloads (http://continuum.io/downloads)

Pick the correct OS and note that the course is in Python 2 *not* Python 3. However, you can use either for this course, there will just be some minor changes you will have to do to follow along (e.g. print functions, range instead of xrange,integer division, etc.) For more info on the differences between Python 2 and Python 3, check out this resource:

http://sebastianraschka.com/Articles/2014_python_2_3_key_diff.html (http://sebastianraschka.com/Articles/2014_python_2_3_key_diff.html)

INSTALLING PACKAGES

How do I install packages for Python?

To install packages for Python (such as the seaborn package for statistical plotting) you have two options, using "conda install" or "pip install". It is HIGHLY recommended you use "conda install". You can install a package through the following steps:

```
1.) Open up your command line. (Linux and OSX MAC users can open up a terminal (bash) command line interface). For Windows users, open up powershell (or Anaconda Command Prompt). For windows users, search for Anaconda Command Prompt with your search tool.

2.) Once in your command line interface, type: conda install *package-name* (or pip install if you do not have the Anaconda distribution).

3.) Remember to install any dependecies for the particular package you are trying to install (conda install will do this manually)

4.) All done! Restart python (or your iPython Notebook) and you are good to go!
```

iPython Notebooks and nbviewer

How do I view the iPython Notebooks?

You have two options: go to http://nbviewer.ipython.org/github/jmportilla/ (http://nbviewer.ipython.org/github/jmportilla/) or go to https://github.com/jmportilla/Udemy-notes (https://github.com/jmportilla/Udemy-notes)

Then click on the lecture you are looking for. Note: Machine Learning Lectures are in a different Folder!

iPython

Why use iPython instead of just regular Python?

ipython is an advanced interactive python shell.

When you invoke it, you will get a python interactive interepreter (like running python), but with some features that make it much easier and more pleasant to use. The few ones I routinely use are:

```
* Tab autocompletion (on class names, functions, methods, variables)
* More explicit and colour-highlighted error messages
* Better history management
* Basic UNIX shell integration (you can run simple shell commands such as cp, ls, rm, cp, etc. directly from the ipython command line)
* Nice integration with many common GUI modules (PyQt, PyGTK, tkinter)
* very nice integration with Data Science Python packages!
```

Questions on lecture material

What do I do if I have a question?

Make sure to visit the discussion section of the particular lecture you have a question about. Many common questions have already been answered. If not, use the search box to find if a general question has already been answered.

There is so much material! Where do I start?

I suggest you start with the setup lectures. Then depending on your previous programming experience, check out Appendix A for material and resources for learning Python. Afterwards continue on on with the rest of the course in order. Check out the rest of the Appendix Resources as you see fit, for example if you are trying to use what you've learned with a SQL database at work, check out the SQL and Python appendix.

GIT AND GITHUB

What is the difference between Git and GitHub?

Git is a version control system; think of it as a series of snapshots (commits) of your code. You see a path of these snapshots, in which order they were created. You can make branches to experiment and come back to snapshots you took. GitHub, is a web-page on which you can publish your Git repositories and collaborate with other people.

Is Git saving every repository locally (in the user's machine) and in GitHub? No, it's only local. You can decide to push (publish) some branches on GitHub.

Can you use Git without GitHub? If yes, what would be the benefit for using GitHub? Yes, Git runs local if you don't use GitHub. An alternative to using GitHub could be running Git on files hosted on Dropbox, but GitHub is a more streamlined service as it was made especially for Git.

How does Git compare to a backup system such as Time Machine? It's a different thing, Git lets you track changes and your development process. If you use Git with GitHub, it becomes effectively a backup. However usually you would not push all the time to GitHub, at which point you do not have a full backup if things go wrong. I use git in a folder that is synchronized with Dropbox.

Is this a manual process, in other words if you don't commit you won't have a new version of the changes made? Yes, committing and pushing are both manual.

If are not collaborating and you are already using a backup system why would you use Git?

If you encounter an error between commits you can use the command git diff to see the differences between the current code and the last working commit, helping you to locate your error.

You can also just go back to the last working commit.

If you want to try a change, but are not sure that it will work. You create a branch to test you code change. If it works fine, you merge it to the main branch. If it does not you just throw the branch away and go back to the main branch.

You did some debugging. Before you commit you always look at the changes from the last commit. You see your debug print statement that you forgot to delete. For more info check out: http://gitimmersion.com/ (http://gitimmersion.com/)

```
In [ ]:
```