# DST4L Class Notes: February 21, 2013

Presenter: Rahul Dave

**Rahul's notes for this class:**
https://dl.dropbox.com/u/75194/class.txt

**Additional links for this class session:**
https://dl.dropbox.com/u/75194/ch3_nltk.ipynb (Source for ch3_nltk: http://slendrmeans.wordpress.com/will-it-python/)
https://dl.dropbox.com/u/75194/NLTK-PANDAS.ipynb
https://dl.dropbox.com/u/75194/data.zip
(Note that Chrome/Windows may add hidden characters or .txt so better to save it in Firefox or Safari)
Instructions for viewing these files are provided at the end of the notes.

**Recommended Resources**

- Websites
  http://learnpythonthehardway.org
  http://ipython.org
  http://ipython.org/notebook.html See especially the links to the example collection and the notebook gallery
  http://stackoverflow.com/ The combination of Google and stackoverflow has become indispensable for learning about commands. "I can't imagine how I ever programmed before stack overflow."
- Blog post
  The Absolute Minimum Every Software Developer Absolutely, Positively Must Know About Unicode and Character Sets (No Excuses!) by Joel Spolksy. Python 3.0 is better at working with unicode.
  http://www.joelonsoftware.com/articles/Unicode.html
- Books
  *Python for Data Analysis* by Wes McKinney ("absolutely fabulous book")
  http://shop.oreilly.com/product/0636920023784.do
  *Machine Learning for Hackers* by Drew Conway and John Myles White
  http://shop.oreilly.com/product/0636920018483.do
  *Programming Collective Intelligence* by Toby Segaran
  http://shop.oreilly.com/product/9780596529321.do
  Tip: O'Reilly has frequent 50% off sales. Wait a couple weeks and get books cheaply.

**Upcoming Event**
Fernando Perez, one of the creators of IPython Notebook, will be speaking at MIT on Friday. Rahul will send around an email.

## Command Line

- Get used to using the command line! Think of it as another language.
- Using only the computer's graphical interface vs command line is like watching movies vs reading books.
- Using the command line will make you much more of an expert on your system.

## Garbage Collection

Python is a garbage-collected language. When it figures out you're not reusing copies (e.g., a temporary copy created in a for loop), it gets rid of them.

## Naming

- You should be able to read back code and understand what you did, so use natural English in naming functions, variables, etc.
- It is a convention that if you are defining a class the class begins with a capital letter

  *Example:*
  nltk.Text
  Text is a class in nltk

## Regular expressions

There are websites to test out regular expressions. Try Googling "regular expression tester"

## Subervised versus unsupervised classification

- Classification: attaching labels to data (e.g., male, female, science, notscience)
- *Supervised classifier*
  - You know the structure of some part of something and want to know structure of something else
  - You give the computer some that you've labeled, then ask the computer to classify the rest using labels you specify
- *Unsupervised classifier*
  - No labeling
  - You want to see if there's any structure to the data. For instance, if you cluster a population on heights, you are likely to see two clusters (which would correspond roughly to male and female)

## Tools for writing code

- IDLE
- SublimeText: very good text editor
- Spyderlib: http://code.google.com/p/spyderlib/

## Data frames, Pandas

- DataFrame introduced in NLTK-PANDAS.ipnyb
- Exposure to this now will help in the R class – see Wes McKinney's book
- If you save an Excel spreadsheet as a csv file you can manipulate it in Python and R.
- Pandas is almost like a relational database. Very fast.

## Imports and namespaces

- Caution: If you import everything from a module at once (e.g., from pandas import *) it can overwrite your functions. Do not use this method!
- Rahul imports pandas as pd to make calling its functions easier.

## Homework

In NLTK-PANDAS we used json.loads to load a json file. Use the opposite of json.loads to create a json file.
Will need to read this post to do the homework:
http://slendrmeans.wordpress.com/2012/12/20/will-it-python-machine-learning-for-hackers-chapter-3-naive-bayes-text-classification/

**SETUP INSTRUCTIONS FOR THOSE WHO HAVE NO PYTHON:**

ENVIRONMENT FOR VIEWING TODAY'S FILES (READ-ONLY):
Notebook viewer: http://nbviewer.ipython.org/
Copy URLs of today's files into notebook viewer one by one

ENVIRONMENT FOR TYPING PYTHON:
https://notebookcloud.appspot.com

**SETUP INSTRUCTIONS FOR THOSE WHO HAVE EVERYTHING INSTALLED:**

1) SAVE TODAY'S FILES (THE PYNB AND TXT FILES AT START OF NOTES) TO LOCAL COMPUTER

2) START IPYTHON NOTEBOOK FROM COMMAND LINE INTERFACE

ON A MAC:
1) change directory to folder where files are located (for instance, **cd /Users/[yourhomedirectoryname]/Documents/DST4L**)
2) type **/Library/Frameworks/Python.framework/Versions/7.3/bin/ipython notebook --pylab=inline**
(or use PythonWB application described at start of today's notes)

ON WINDOWS:
1) change directory to folder where files are located (for instance, **cd C:\DST4L**)
2) type full path: **C:\Python27\Scripts\ipython notebook --pylab=inline**

3) IF NECESSARY, UPLOAD TODAY'S FILES TO IPYTHON NOTEBOOK
from iPython notebook main page, click on the words "click here" to upload files
select all files saved to local computer
click "Upload" button