Introduction to Scientific Python

Blake Fleischer, PhD

Research Scientist

Partnership for an Advanced Computing Environment (PACE)

Objectives :

- Showcase python as a viable alternative to scientific packages like R, MatLab

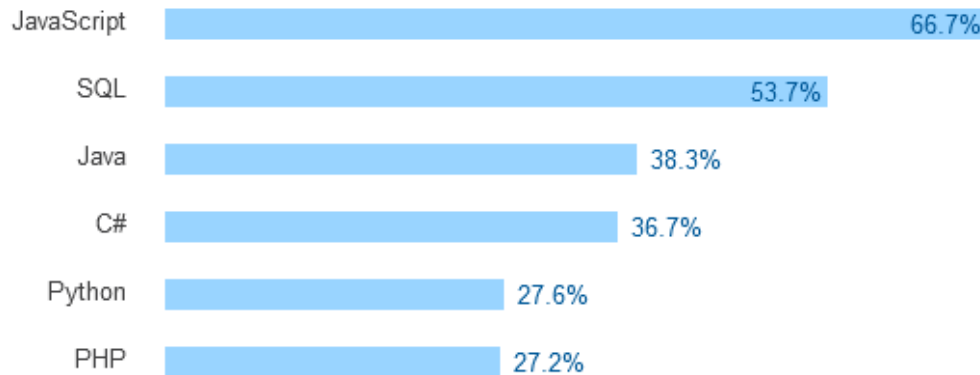- Overview of some of Python's scientific functions

Why python for Science?

- Easy to learn

- Well Supported

- Useful after graduation (free and open source)

- Extremely popular outside of academia

People use Python

2017 StackOverflow Survey

JavaScript	66.7%
SQL	53.7%
Java	38.3%
C#	36.7%
Python	27.6%
PHP	27.2%

Google searches for language + 'tutorial' (PYPL)

Worldwide, Aug 2017 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Java	22.7 %	-0.7 %
2		Python	16.3 %	+3.7 %
3		PHP	8.9 %	-1.1 %
4		C#	8.3 %	-0.5 %

Python Packages

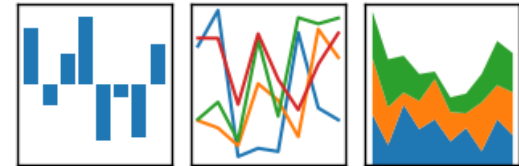- Extending beyond simple programming, additional functionality comes from packages

- Package discovery:
  - Finding the packages that suit your needs

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$

Tools of the Trade: Git

- "*Version control* is a system that records changes to a file or set of files over time so that you can recall specific *versions* later."

- Git incorporates free web hosting for publicly visible files

- Backbone of the developer world

- Developed to support the Linux kernel

- https://try.github.io/levels/1/challenges/1

- http://rogerdudler.github.io/git-guide/

- How to undo things:
  - https://github.com/blog/2019-how-to-undo-almost-anything-with-git

Tools of the Trade: Anaconda

- Package manager for python – useful for managing Python!

- From the author of Python's numpy and scipy packages

- To set up:
  - Download anaconda from https://www.continuum.io/downloads
  - Install packages for this course:
  - `conda install python=3 scipy numpy matplotlib jupyter notebook pandas pytables hdf5`
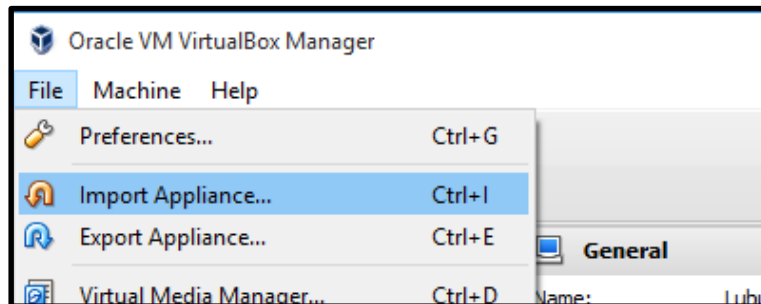
Tools of the Trade: Jupyter

- Framework for interactive programming

- Jupyter Notebook: a web interface to interactive programming
  - Interface for coding, viewing data, and plotting
  - Reproducible science

- Uses ipython
  - Access docstrings (help on commands)
  - Tab completion

Materials for this Course:

- Download VirtualBox (free)
  - https://www.virtualbox.org/wiki/Downloads

- Download the .ovf image for the course (1.8 GB compressed, + ~5GB when added to virtualbox)
  - http://pace.gatech.edu/sites/default/files/Lubuntu32.ova

- Alternatively, install anaconda on your local machine and pull the materials from:
  - https://github.com/blakeflei/IntroScientificPythonWithJupyter

Overview

- Part I:
  - Jupyter Notebook overview
  - Input/Output in Python
  - Basic Math
  - Indexing/Array Operations
  - Plotting
  - FittingPlot

- Part II
  - Some basic statistics
  - Signal Processing
  - Ordinary Differential Equations
  - Pandas
  - Gotchas and Optmization

Do some science!