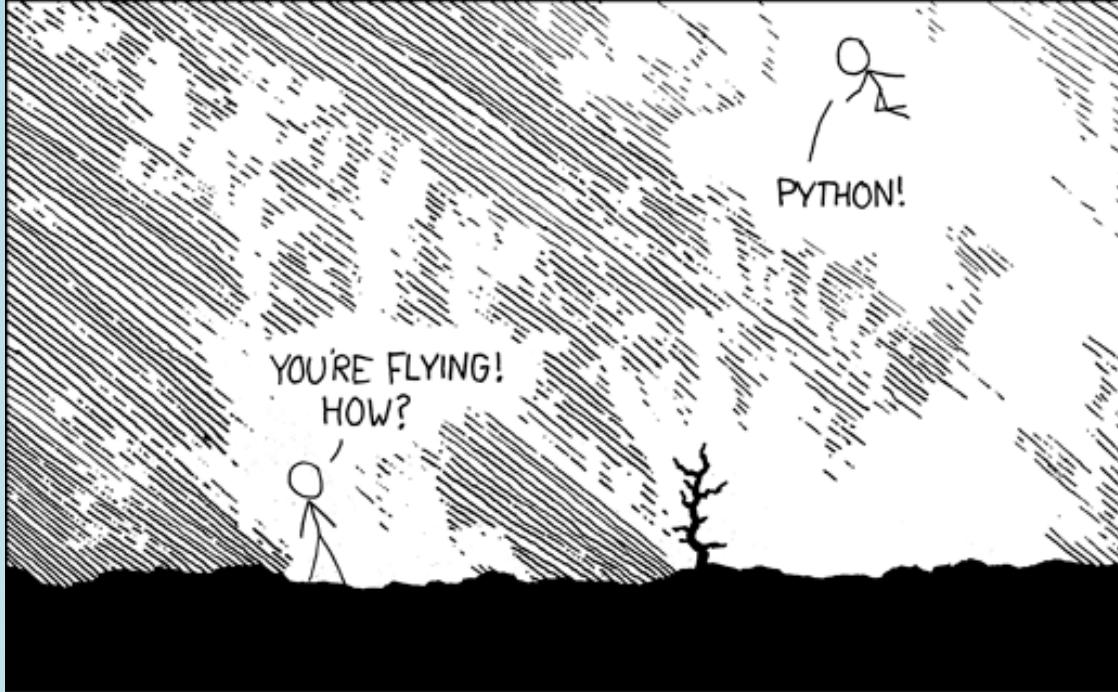




Python for data science:

Day 2



I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
/ HELLO WORLD IS JUST
print "Hello, world!"

I DUNNO...
DYNAMIC TYPING?
WHITESPACE?

COME JOIN US!
PROGRAMMING IS FUN AGAIN!
IT'S A WHOLE NEW WORLD UP HERE!

BUT HOW ARE YOU FLYING?

I JUST TYPED
import antigravity
THAT'S IT?

... I ALSO SAMPLED
EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.

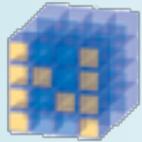
BUT I THINK THIS IS THE PYTHON.

cartoon from xkcd



Python Packages

NumPy



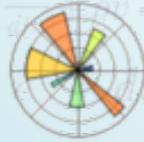
SciPy



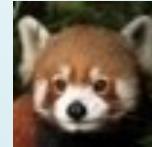
SymPy

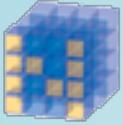


Matplotlib



Pandas





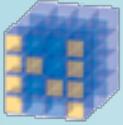
NumPy:

import numpy

```
>>> import numpy  
>>> print dir(numpy)  
['ALLOW_THREADS', 'BUFSIZE', 'CLIP', 'ComplexWarning', 'DataSource', 'ERR_CALL', 'ERR_DEFAULT
```

import numpy as np

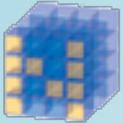
```
>>> import numpy as np
```



NumPy

`numpy.linalg.norm:`
Length of a vector

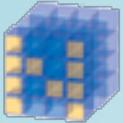
```
>>> np.linalg.norm([2,4])  
4.4721359549995796
```



NumPy

numpy.linalg.inv: Inverse of the Matrix

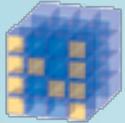
```
>>> np.linalg.inv([[2,0],[0,2]])  
array([[ 0.5,  0. ],  
       [ 0. ,  0.5]])
```



NumPy

numpy.matrix: alternative to numpy.array

```
>>> M = np.matrix([[2,0],[0,2]])
>>> M * M
matrix([[4,  0],
        [0,  4]])
>>> M.I
matrix([[ 0.5,  0. ],
        [ 0. ,  0.5]])
>>> _ * M
matrix([[ 1.,  0.],
        [ 0.,  1.]])
```

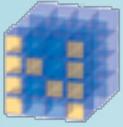


NumPy: numpy.random: Random sampling

<code>rand(d0, d1, ..., dn)</code>	Random values in a given shape.
<code>randn(d0, d1, ..., dn)</code>	Return a sample (or samples) from the “standard normal” distribution.
<code>randint([low[, high, size, dtype]])</code>	Return random integers from <i>low</i> (inclusive) to <i>high</i> (exclusive).
<code>random_integers([low[, high, size]])</code>	Random integers of type np.int between <i>low</i> and <i>high</i> , inclusive.
<code>random_sample([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>random([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>ranf([size])</code>	Return random floats in the half-open interval [0.0, 1.0).
<code>sample([size])</code>	Return random floats in the half-open interval [0.0, 1.0).

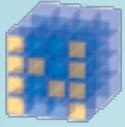
```
>>> np.random.rand(1)
array([ 0.88288734])
>>> np.random.rand(2)
array([ 0.49363292,  0.16510102])
>>> np.random.rand(1,2)
array([[ 0.91578292,  0.60660245]])
>>> np.random.rand(2,1)
array([[ 0.23357355],
       [ 0.25666865]])
>>> np.random.rand(2,2, 2)
array([[[ 0.00763289,  0.09970221],
       [ 0.68506231,  0.31527997]],
       [[ 0.94116485,  0.43140191],
       [ 0.78844403,  0.2575242 ]]])
>>> np.random.rand(2,2, 2, 2)
array([[[[ 0.57742619,  0.99551532],
       [ 0.99774305,  0.45852133]],
       [[ 0.06396083,  0.68351763],
       [ 0.96607077,  0.27239997]]],
       [[[ 0.91702878,  0.33364644],
       [ 0.34576949,  0.06507407]],
       [[ 0.20052982,  0.24095151],
       [ 0.1145685 ,  0.28035093]]]])
```

```
>>>
```



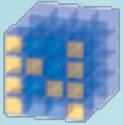
NumPy: numpy.random.permutation()

```
>>> np.random.permutation(6)
array([5, 1, 4, 0, 3, 2])
>>> sequence = np.random.permutation(10)
>>> print(sequence)
[1 2 5 9 8 3 4 6 0 7]
```



NumPy: numpy.random.uniform()

```
>>> np.random.uniform(-1,1, size=10)
array([-0.08,  0.74, -0.64, -0.98,  0.98, -0.21, -0.36, -0.34, -0.16, -0.86])
```



NumPy: numpy.random.uniform()

```
>>> np.random.uniform(-1,1, size=10000)
array([ 0.34, -0.88, -0.64, ..., -0.87,  0.09,  0.42])
>>> stats.describe(_)
DescribeResult(nobs=10000L, minmax=(-0.99984799016922654, 0.99997505667204889),
mean=-0.0017052074604612564, variance=0.33296037891169855, skewness=-0.004341077496136224,
kurtosis=-1.2054297738246007)
>>>
```



scipy.stats

Statistical functions

This module contains a large number of probability distributions as well as a growing library of statistical functions.



scipy.stats.describe

```
>>> np.random.permutation(6)
array([5, 1, 4, 0, 3, 2])
>>> sequence = np.random.permutation(10)
>>> print(sequence)
[1 2 5 9 8 3 4 6 0 7]
>>> stats.describe(sequence)
DescribeResult(nobs=10, minmax=(0, 9), mean=4.5, variance=9.166666666666661,
skewness=0.0, kurtosis=-1.2242424242424244)
```



scipy.misc.comb

```
>>> from scipy.special import comb
>>> comb(10,2)
45.0
>>> comb(3,2)
3.0
>>>
```



For many more stat related functions install the software R and the interface package rpy.

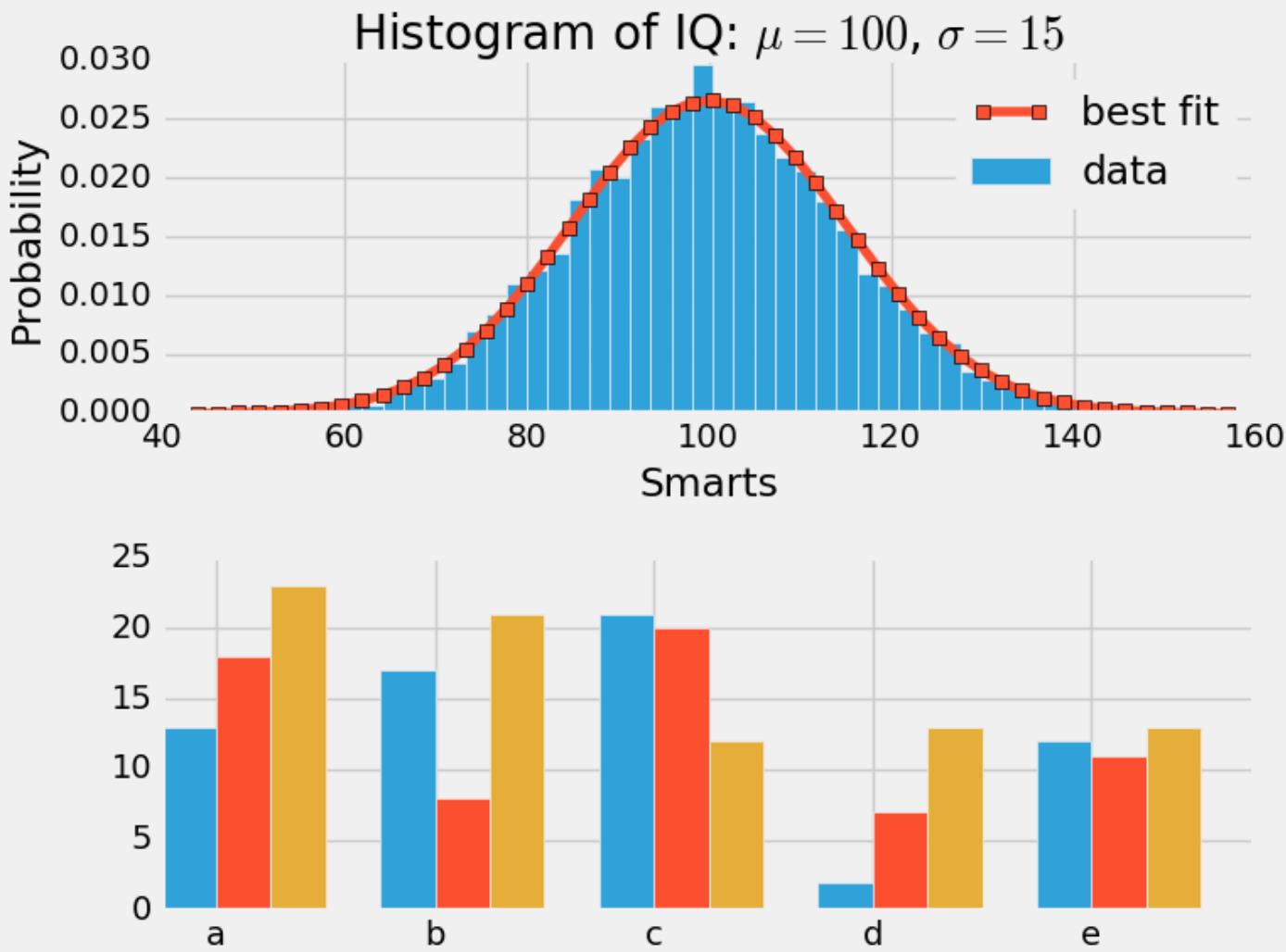


Bokeh

*Goal: a billion points, meaningfully,
interactively, in the browser*



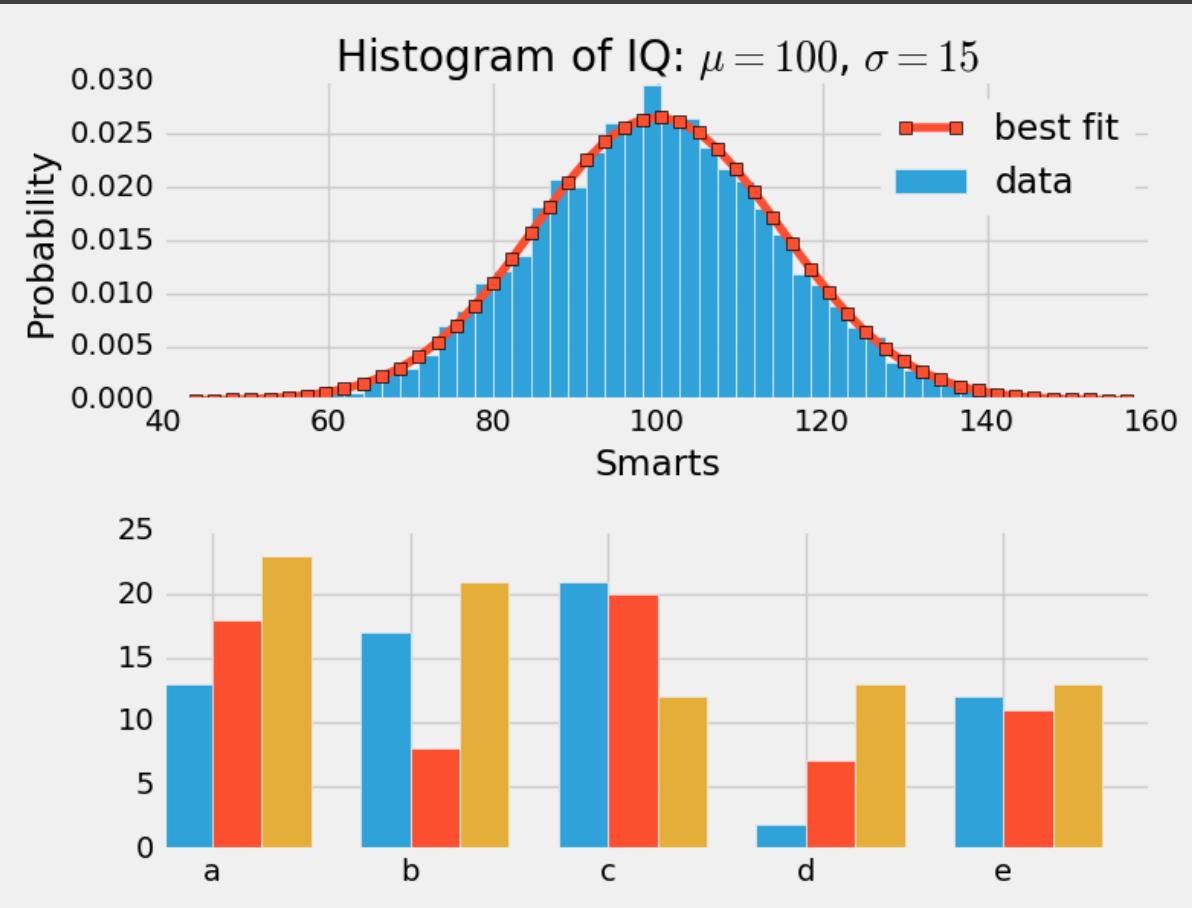
Matplotlib





Matplotlib

```
matplotlib.style.use('fivethirtyeight')
```



Without this change, most styles will default to the "jet" colormap.



Matplotlib

To run the gallery showing styles available, simply grab the source and run the package as a script:

```
$ git clone https://github.com/tonysyu/matplotlib-style-gallery.git  
$ cd matplotlib-style-gallery  
$ python -m mpl_style_gallery
```



GitHub

*"a 2 billion dollar
facebook for programmers"*



GitHub

*"15 billion active users who
learn, share, and work
together to build software"*

- Github: file repository like dropbox
- Git: version control system for software dev.



"free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency."

```
$ git clone
```

Clone a repository

```
$ git push
```

Update server with your commits across all branches that are common between your local copy and server

```
$ git fork
```

Points your repo to original via an alternates file

```
$ git pull
```

Fetch changes from server and merge into current branch



Jupyter Notebooks

"Open source, interactive data science and scientific computing across over 40 programming languages."



The Jupyter Notebook is a web application that allows you to create and share documents that contain live code, equations, visualizations and explanatory text.

Uses include: data cleaning and transformation, numerical simulation, statistical modeling, machine learning and much more.

Notebooks may be exported to a range of static formats:

- HTML (for example, for blog posts)
- LaTeX
- PDF
- Slide shows (via reveal.js)



Jupyter Notebooks

Any .ipynb notebook document available from a public URL can be shared via the Jupyter Notebook Viewer ([nbviewer](#)).



Jupyter Notebooks

The landing page of the Jupyter notebook web application, the **dashboard**, shows the notebooks currently available in the notebook directory



Jupyter Notebooks

Ok, I want to download a notebook and start working with it. What next?



Cloning notebooks from Github

- Use browser ([Github.com](https://github.com))
- Use Git



Popular repositories

SciPyCourse2016 Jupyter notebooks and data for the "Introduction to Scientific Programming in Python" Course taught at LMU in 2016 ★ 4 ● Jupyter Notebook	ratcave 3D Graphics Engine running off Python, Pyglet, and Psychopy ★ 2 ● Python
PyWavefront Forked from greenmoss/PyWavefront Python library for importing Wavefront .obj files ★ 0 ● Python	ratracker Database for Rat Lab ★ 0 ● Python
pypropixx_utils A few utility functions for interacting with Propixx projector, depends on VPixx's pypixxlib library ★ 0 ● Python	pyexperiment State Machine Python package for running simple experiments, intended for psychophysics and cognitive psychology research. ★ 0 ● Python

1,223 contributions in the last year



Nicholas A. Del Grosso
[neuroneuro15](https://github.com/neuroneuro15)

[Follow](#)

[Block or report user](#)

✉ delgrosso.nick@gmail.com
⌚ Joined on Mar 7, 2013

GitHub <https://github.com/neuroneuro15/SciPyCourse2016>

neuroneuro15 / SciPyCourse2016

Watch ▾ 1

Star 4

Fork 2

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Pulse

Graphs

Jupyter notebooks and data for the "Introduction to Scientific Programming in Python" Course taught at LMU in 2016

54 commits

1 branch

0 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

 neuroneuro15 committed on GitHub	Update README.md	Latest commit 163eb3b 29 days ago
 .ipynb_checkpoints	most of lectures, data, and homework review for lecture 8	5 months ago
 Homework Sample Data	update Lecture 6 with needed info	5 months ago
 .gitignore	a	5 months ago
 Final Project Description.odt	added final project description	6 months ago
 Homework 1 Jupyter Notebook and...	modified some lectures and homework	6 months ago
 Homework 2 Iteration, the Built-In F...	lots of updates	5 months ago
 Homework 2 Iteration, the Built-In F...	changes for lecture 5, including homework	6 months ago
 Homework 3 1D Array Analysis and ...	most of lectures, data, and homework review for lecture 8	5 months ago
 Homework 3 1D Array Analysis and ...	lots of updates	5 months ago
 Homework 4 2D Array Analysis and ...	lots of updates	5 months ago
 Homework 4 2D Arrays (DataGen).ip...	added datagen file	5 months ago



```
C:\Users\jajaco3\Documents>git clone https://github.com/neuroneuro15/SciPyCourse2016

Cloning into 'SciPyCourse2016'...
remote: Counting objects: 248, done.
Receiving objects: 100% (248/248), 7.16 MiB | 0 bytes/s, done.48Receiving objects: 98% (244/244)

Resolving deltas: 100% (126/126), done.
Checking connectivity... done.
git-lfs smudge -- 'Homework Sample Data/Homework 4/eeg_data.mat': git-lfs: command not found
error: external filter git-lfs smudge -- %f failed -1
error: external filter git-lfs smudge -- %f failed
fatal: Homework Sample Data/Homework 4/eeg_data.mat: smudge filter lfs failed
warning: Clone succeeded, but checkout failed.
You can inspect what was checked out with 'git status'
and retry the checkout with 'git checkout -f HEAD'

C:\Users\jajaco3\Documents>
```

Find the directory containing SciPyCourse2016-master



```
C:\Users\jajaco3\Documents>dir
Volume in drive C is Windows
Volume Serial Number is E88B-1894

Directory of C:\Users\jajaco3\Documents

11/03/2016  04:52 PM    <DIR>          .
11/03/2016  04:52 PM    <DIR>          ..
11/01/2016  05:10 PM    <DIR>          .ipynb_checkpoints
10/05/2016  03:16 PM    2,873   .Rhistory
09/11/2016  07:13 PM    <DIR>          btabibian.github.io
09/11/2016  07:16 PM    <DIR>          btabibian.github.io-master
08/17/2016  03:46 PM    <DIR>          Custom Office Templates
11/01/2016  09:25 PM    <DIR>          GitHub
09/27/2016  11:17 AM    <DIR>          LabData
09/11/2016  08:31 PM    33,070  Lecture1-Copy1.ipynb
11/01/2016  11:29 AM    10,731  Lecture1.ipynb
08/18/2016  03:03 PM    <DIR>          Python Scripts
11/01/2016  05:41 PM    <DIR>          pythonexamplefolder
08/22/2016  09:21 AM    <DIR>          R
11/03/2016  04:37 PM    <DIR>          SciPyCourse2016-master
09/02/2016  09:02 PM    <DIR>          Turning
11/03/2016  10:03 AM    <DIR>          TurningPoint 5
11/01/2016  05:10 PM    2,699   Untitled1.ipynb
                           4 File(s)        49,373 bytes
                           14 Dir(s)   196,888,305,664 bytes free
```



Open Jupyter notebook

```
C:\Users\jajaco3\Documents>jupyter notebook
[W 16:55:15.381 NotebookApp] Unrecognized JSON config file version, assuming version 1
[I 16:55:17.400 NotebookApp] [nb_conda_kernels] enabled, 4 kernels found
[I 16:55:18.250 NotebookApp] nbpresent HTML export ENABLED
[W 16:55:18.253 NotebookApp] nbpresent PDF export DISABLED: No module named nbrowserpdf.e
[I 16:55:18.263 NotebookApp] [nb_conda] enabled
[I 16:55:18.414 NotebookApp] [nb_anacondacloud] enabled
[I 16:55:18.767 NotebookApp] Serving notebooks from local directory: C:\Users\jajaco3\Document
[I 16:55:18.770 NotebookApp] 0 active kernels
[I 16:55:18.772 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 16:55:18.776 NotebookApp] Use Control-C to stop this server and shut down all kernels (twic
```



Open Jupyter notebook

```
C:\Users\jajaco3\Documents>jupyter notebook
[W 16:55:15.381 NotebookApp] Unrecognized JSON config file version, assuming version 1
[I 16:55:17.400 NotebookApp] [nb_conda_kernels] enabled, 4 kernels found
[I 16:55:18.250 NotebookApp] nbpresent HTML export ENABLED
[W 16:55:18.253 NotebookApp] nbpresent PDF export DISABLED: No module named nbbrowserpdf.e
[I 16:55:18.263 NotebookApp] [nb_conda] enabled
[I 16:55:18.414 NotebookApp] [nb_anacondacloud] enabled
[I 16:55:18.767 NotebookApp] Serving notebooks from local directory: C:\Users\jajaco3\Document
[I 16:55:18.770 NotebookApp] 0 active kernels
[I 16:55:18.772 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 16:55:18.776 NotebookApp] Use Control-C to stop this server and shut down all kernels (twic
```



Open any .ipynb file and you are on your way to learning python using Jupyter notebooks and Github!

Thank you!



To obtain these slides as well as the jupyter notebooks that were presented, clone:

`github.com/jeremyallenjacobson/PythonForDataScience`