# Jupyter & it's Moons

Shreyas Cholia, Matt Henderson, Oliver Evans (and in spirit, Fernando Pérez)

Lawrence Berkeley Lab

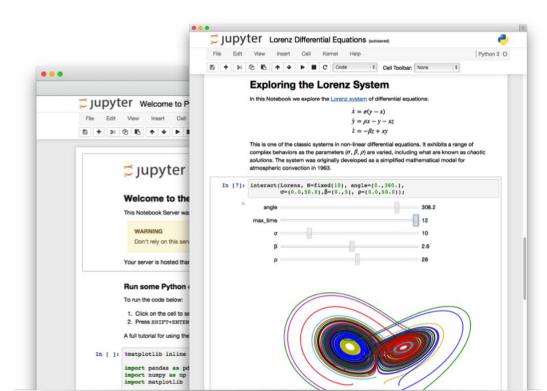
Gateways 2017 Tutorial, Ann Arbor MI

**Getting Started** 

http://tinyurl.com/jupyter-moons

### **Jupyter**

Interactive notebook environment that can combine code with visualizations, documentation and programmable widgets



### Aside

A brief history of scientific computing

- Brought to you by Jupyter

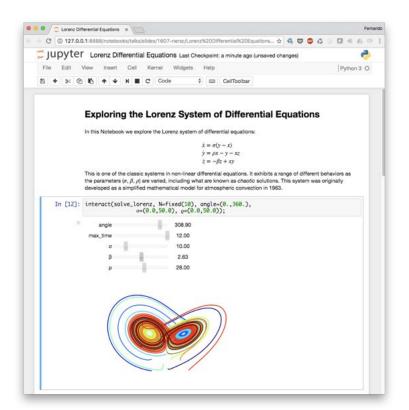
### IPython: Interactive Python, 2001

- Object Introspection (TAB!)
- OS Integration
- Rich terminal client
- GUI support (plots, ...)
- %magic commands
- Embeddable

```
1. IPython: Users/fperez (python3.5)
Python 3.5.2 |Continuum Analytics, Inc.| (default, Jul 2 2016, 17:52:12)
Type "copyright", "credits" or "license" for more information.
IPython 5.1.0 -- An enhanced Interactive Python.
         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
         -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.
Using matplotlib backend: MacOSX
Populating the interactive namespace from numpy and matplotlib
  [2]: from IPython.display import display
        from pandas_datareader import data
   ...: from datetime import datetime
   ...: stock = data.DataReader( ticker, 'yahoo', start=datetime(2012, 1, 1))
   ...: display(stock[:3])
   ...: stock['Close'].plot(title='%s Closing Price' % ticker);
                                       Low Close
                                                    Volume Adj Close
2012-01-03 26.549999 26.959999 26.389999 26.77 64731500 23.304317
2012-01-04 26.820000 27.469999 26.780001 27.40 80516100 23.852755
 2012-01-05 27.379999 27.730000 27.290001 27.68 56081400
 n [3]:
                              MSFT Closing Price
```

## The IPython/Jupyter Notebook

- Rich web client
- Text & math
- Code
- Results
- Share, reproduce.



# Why Jupyter?

- Interactivity
- Reproducibility and Collaboration
- Education

## Interactive Computing

- Code + Viz + Widgets
- Paradigm shift for exploratory computing
- Break up steps into cells

```
Factoring Polynomials with SymPy

Here is an example that uses SymPy to factor polynomials.

In [1]: from ipywidgets import interact

In [2]: from sympy import Symbol, Eq, factor

In [3]: x = Symbol('x')

In [4]: def factorit(n):
    return Eq(x**n-1, factor(x**n-1))

In [5]: factorit(12)

Interact(factorit, n=(2,40));

A Jupyter Widget
```

```
In [22]: from IPython.html.widgets import *
          t = arange(0.0, 1.0, 0.01)
          def pltsin(f):
              plt.plot(x,sin(2*pi*t*f))
              plt.show()
          interact(pltsin, f=(1,10,0.1))
                                                                 4.9
            0.5
            0.0
           -0.5
           -1.0
```

# Reproducible Research: Papers + Notebooks on Github

"An article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures."

Buckheit and Donoho, WaveLab and Reproducible Research, 1995

1833 lines (1833 sloc) 276 687 M Multi-tiered genomic analysis of head and neck cancer ties TP53 mutation to 3p loss HNSCC HPV- Cohort tun, Jannifer R Grandis, Oupon T Nguyan & Trep lifeter remain unbiased in our screen for molecular coordates of survival, we do have much more information on TP53 In Poets, a TP53 mutation is labeled as disruptive if it is either a stop mutation, or if is located at a binding site and induces a change in polarity of the encoded amino acid. Interestingly, we found that the polarity of the substitution had little effect on prognosis and that patients with a mutation to the L2 binding site had worse outcomes than patients with a mutation to the L3 binding site. In addition, within the context of the framework we set forth for biomarker discovery, we chose to ignore the classification of mutations (past silent/non-silent) in order to keep sample size high at the risk of false positives. For these reasons we elected to simply display the functional assignment of the mutations in Figure 1 rather than obscure these results with a classification For full list of data and packages imported see the imports notebook In [1]: import NotebookImport from Imports import 00000 importing IPython notebook from Imports.ipynb Populating the interactive namespace from numpy and matplotlib changing to source directory populating namespace with data **TP53 Mutation Clinical Coorelates** In [2]: p53 mut = mut.df.ix['TP53'].ix[keepers.o].dropng().astype(int) In [3]: survival\_and\_stats(p53\_mut, surv, figsize=(5,4), order=[2,1,8]) SI DESCRIPTION Software Overview attifetthat.com : \$5 of head and neck cancer ties TPS3 mutation to 3o loss by Gross et al. In general code for datarecorded in Electron Notabooks. The analysis for this porient was relatively non-linear and has thus been sold into a number of notehoods, as described in Analysis Simployers, but results should be able to be replicated by running these notebooks

TOBATPS3 exploration : ×

- C A Giffuls, Inc. (US) https://github.com/thwandygross/TCGA/biob/master/Analysis\_Note

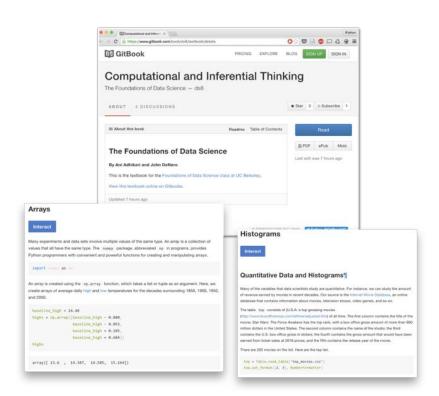
Branch: master - TCGA / Analysis Notebooks / TP53 exploration.ipvnl

Q 0 0 0 0 0 0 E

#### Education

- New curriculum aimed at all freshmen at UC Berkeley
- Interactive textbook is Jupyter Notebooks
- Course deployment is JupyterHub

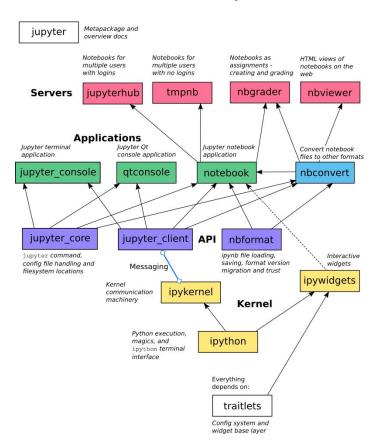
http://data8.org



### Lots of things have emerged in the ecosystem

- Multi-user environments (Jupyterhub)
- Interactive Widgets
- Jupyter Platform (Jupyterlab)
- Sharing
- Classroom platform (nbgrader)
- And more ...

## Visual overview of the ecosystem

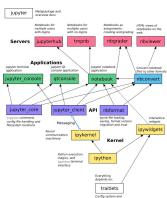


### We're not going to cover all of this!

This is really an introductory survey tutorial. We won't cover writing your own kernel or Jupyter Protocol level communications

But we hope to help you get started

- Google, Github etc. are your friends



### Outline

- Installation and Running
- Introduction and History
- Jupyter Browser View
- Notebook Tour
- Super Short Intro to Python
- Jupyter Architecture
- Scientific Computing Pandas, Numpy, MPL, Sympy etc.
- Widgets
- Sharing notebooks
- JupyterLab
- JupyterHub
- Kernels and Customizing Your Environment
- Other moons