Jupyter
[notebook, lab]

Scott Sievert
link to slides: https://github.com/stsievert/talks
Twitter, GitHub: @stsievert

jupyter Welcome to P

File Edit View Insert Cell

jupyter

Welcome to the

This Notebook Server wa

WARNING

Don't rely on this serv

Your server is hosted thar

Run some Python

To run the code below:

1. Click on the cell to se
2. Press SHIFT+ENTER

A full tutorial for using the

```
In [ ]: %matplotlib inline

        import pandas as pd
        import numpy as np
        import matplotlib
```

jupyter Lorenz Differential Equations (autosaved)

File Edit View Insert Cell Kernel Help Python 3

Code Cell Toolbar: None

Exploring the Lorenz System

In this Notebook we explore the Lorenz system of differential equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0.,360.),
                σ=(0.0,50.0),β=(0.,5), ρ=(0.0,50.0))
```

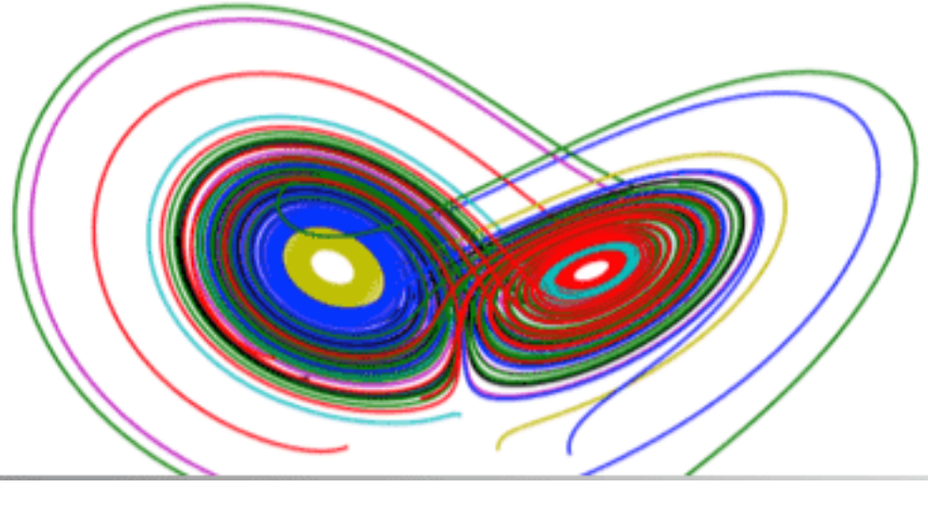
angle 308.2

max_time 12

σ 10

β 2.6

ρ 28



Use cases

- **Open science and reproducible research**
  - https://losc.ligo.org/tutorials/
- **Data science**
  - https://github.com/jakevdp/PythonDataScienceHandbook
- **Education**
  - http://jupyter.org, JupyterHub example

This talk

I'll mention

- notebook motivation
- the notebook
- the general problem Jupyter solves
- juputerlab, a new interface

Motivation

```
>>> X = [1, 2, 3, 4]
>>> 2*X
[1, 2, 3, 4, 1, 2, 3, 4]
>>> # whoops!
>>> [2*x for x in X]
[1, 4, 6, 8]
```

By definition, Python REPL is very *interactive*

interactive := what code runs when
and seeing output immediately

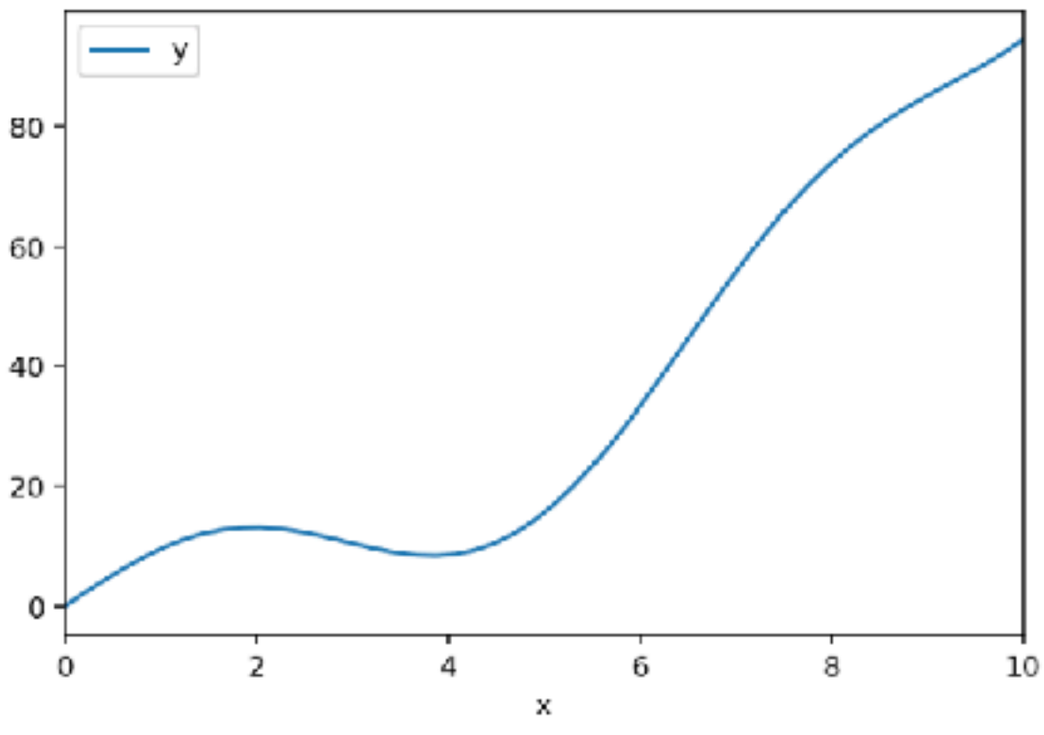
Jupyter notebooks expand REPL interactivity

Jupyter notebooks

- Introduce code *cells* for interactivity

In [21]:
```
x = np.linspace(0, 10, num=50)
y = x**2 + np.sin(x) * 10

df = pd.DataFrame({'x': x, 'y': y})
df.plot(x='x', y='y')
```

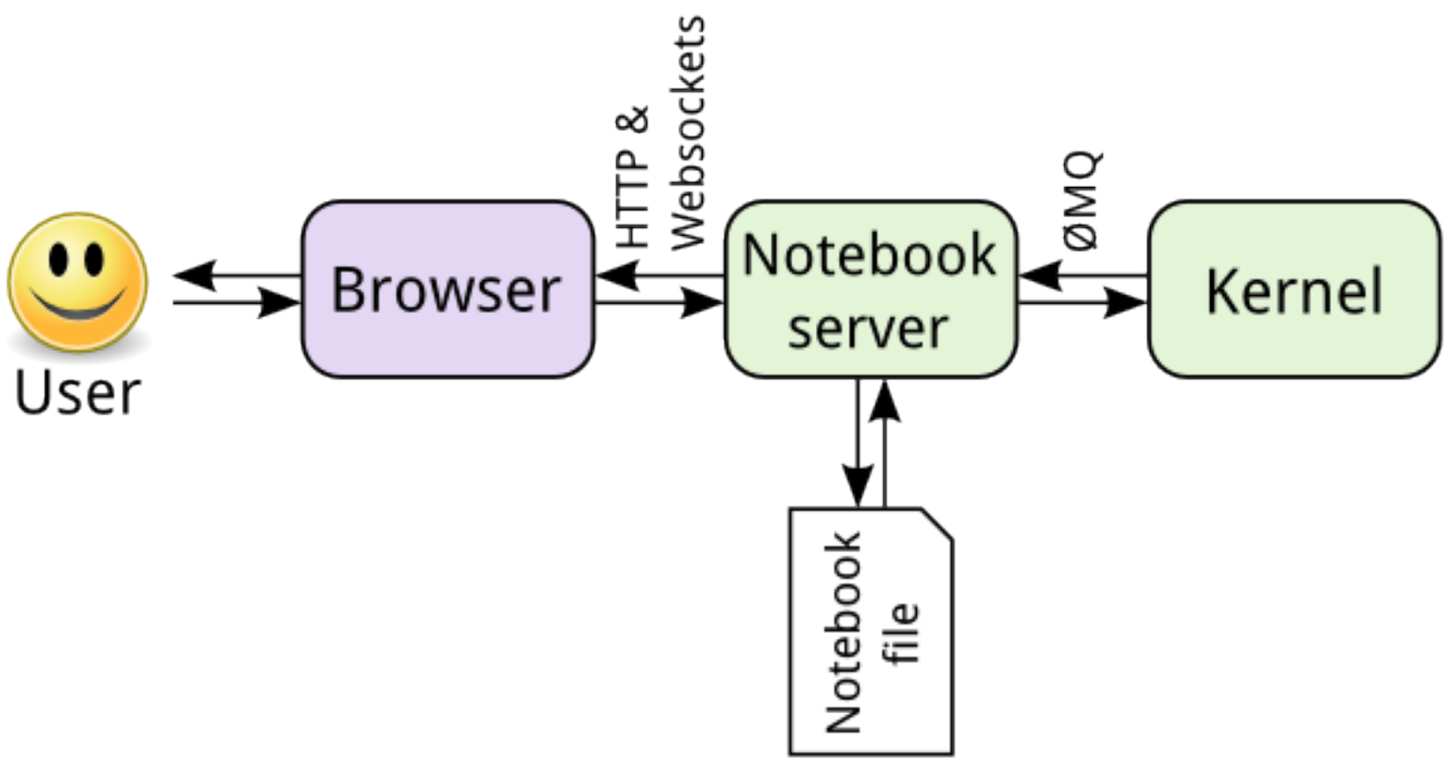
Out[21]: `<matplotlib.axes._subplots.AxesSubplot at 0x10e1c5dd8>`



Live demo

- Notebook basics: {editing, running} {code, markdown, math}
- Output persistence
- Show extensions

Jupyter is to code as HTTP is to files



Features

- **Developer productivity** (quick iteration on block of code)
- **Sharing** (file contains input and output)
- **Documentation** (markdown + math + code + outputs = useful)

Sharing

Explore GitHub

Showcases | Trending

aymericdamien / **TensorFlow-Examples** ★ Star

TensorFlow Tutorial and Examples for Beginners with Latest APIs

Jupyter Notebook ★ 16,824 5,352 Built by ★ 21 stars today

loudinthecloud / **pytorch-ntm** ★ Star

A Pytorch implementation of an NTM (Neural Turing Machine)

Jupyter Notebook ★ 21 Built by ★ 21 stars today

HTML
JavaScript
Jupyter Notebook
Python
Ruby
Swift
Vim script

Other: **Languages**

https://github.com/trending/jupyter-notebook

nbviewer

A simple way to share Jupyter Notebooks

URL | GitHub username | GitHub username/repo | Gist ID Go!

https://nbviewer.jupyter.org

Documentation

jakevdp / PythonDataScienceHandbook https://github.com/jakevdp/PythonDataScienceHandbook

<> Code | Issues 10 | Pull requests 11 | Projects 0 | Wiki | Insights

Branch: master ▾ PythonDataScienceHandbook / notebooks / 02.02-The-Basics-Of-NumPy-Arrays.ipynb Find file Copy path

jakevdp VTYPOS: various 8ae064f on Feb 21

1 contributor

1568 lines (1567 sloc) 32.1 KB Raw Blame History

This notebook contains an excerpt from the Python Data Science Handbook by Jake VanderPlas; the content is available on GitHub.

The text is released under the CC-BY-NC-ND license, and code is released under the MIT license. If you find this

https://github.com/spatialaudio/nbsphinx/

nbsphinx

Search docs

- Usage
- Markdown Cells
- Code Cells
- Raw Cells
- Hidden Cells
- Controlling Notebook Execution
- Notebooks in Sub-Directories
- Using toctree In A Notebook
- Normal reStructuredText Files
- External Links

Edit on GitHub

Jupyter Notebook Tools for Sphinx

`nbsphinx` is a Sphinx extension that provides a source parser for `*.ipynb` files. Custom Sphinx directives are used to show Jupyter Notebook code cells (and of course their results) in both HTML and LaTeX output. Un-evaluated notebooks – i.e. notebooks without stored output cells – will be automatically executed during the Sphinx build process.

Documentation (and example of use):
http://nbsphinx.readthedocs.io/

Source code repository (and issue tracker):
https://github.com/spatialaudio/nbsphinx/

Python Package Index:
https://pypi.python.org/pypi/nbsphinx/

Features

Share notebooks

Notebooks can be shared with others using email, Dropbox, GitHub and the Jupyter Notebook Viewer.

Interactive widgets

Code can produce rich output such as images, videos, LaTeX, and JavaScript. Interactive widgets can be used to manipulate and visualize

Language of choice

The Notebook has support for over 40 programming languages, including those popular in Data Science such as Python, R, Julia and Scala.

Big data integration

Leverage big data tools, such as Apache Spark, from Python, R and Scala. Explore that same data with pandas, scikit-learn, ggplot2, dplyr, etc.

https://github.com/jupyter/jupyter/wiki/Jupyter-kernels

Live demo

interactive widgets

Jupyter Lab

Building blocks

File browser

Notebooks

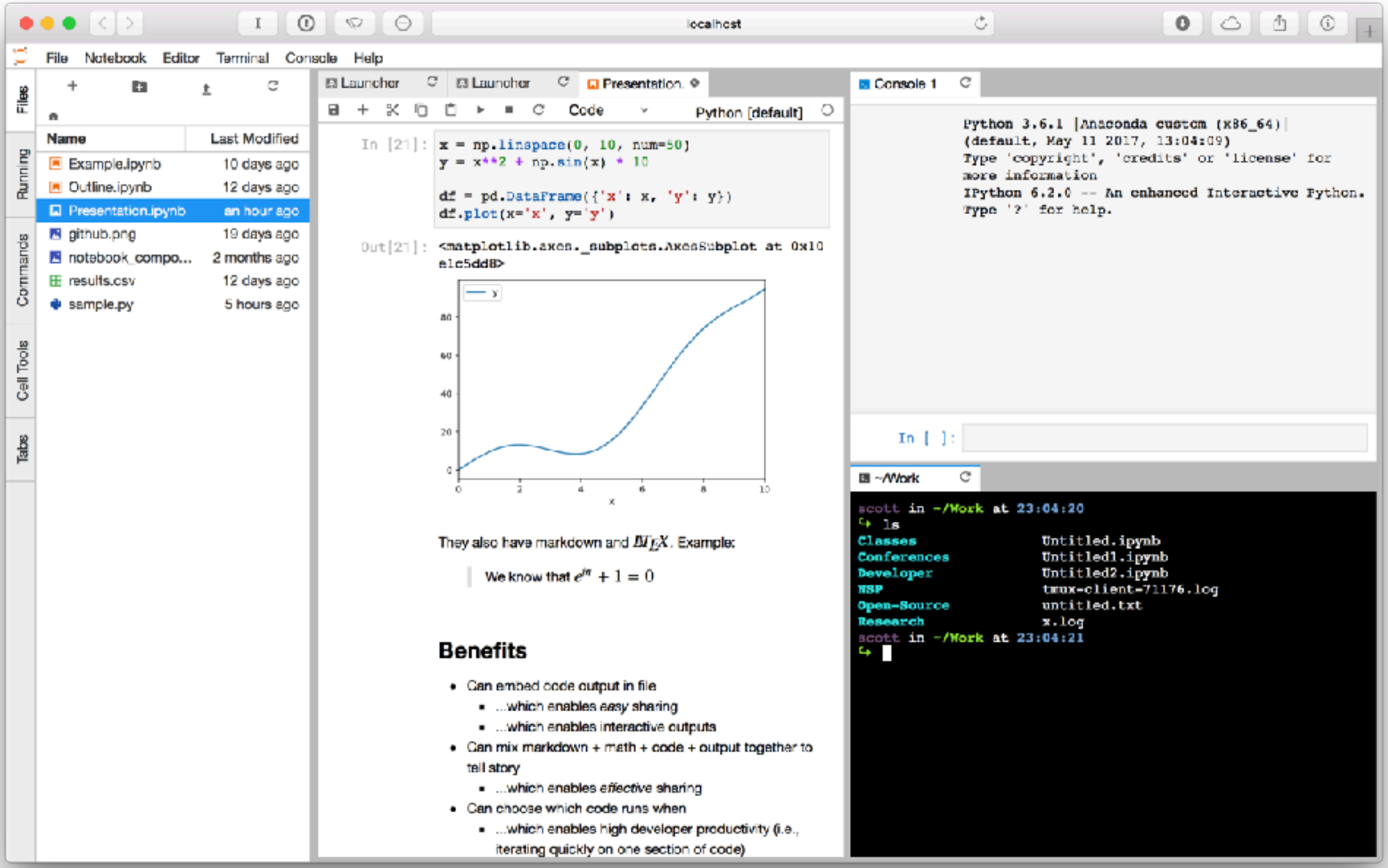
Terminal

Text

Kernels

Output

Jupyterlab


localhost
File Notebook Editor Terminal Console Help
Files
Running
Commands
Cell Tools
Tabs
Name
Last Modified
Example.ipynb 10 days ago
Outline.ipynb 12 days ago
Presentation.ipynb an hour ago
github.png 19 days ago
notebook_compo... 2 months ago
results.csv 12 days ago
sample.py 5 hours ago
Launcher
Launcher
Presentation
Code
Python [default]
In [21]:
x = np.linspace(0, 10, num=50)
y = x**2 + np.sin(x) * 10
df = pd.DataFrame({'x': x, 'y': y})
df.plot(x='x', y='y')
Out[21]:
<matplotlib.axes._subplots.AxesSubplot at 0x10e1c5dd8>
y
x
They also have markdown and LaTeX. Example:
We know that $e^{i\pi} + 1 = 0$
Benefits
Can embed code output in file
...which enables easy sharing
...which enables interactive outputs
Can mix markdown + math + code + output together to tell story
...which enables effective sharing
Can choose which code runs when
...which enables high developer productivity (i.e., iterating quickly on one section of code)
Console 1
Python 3.6.1 |Anaconda custom (x86_64)|
(default, May 11 2017, 13:04:09)
Type 'copyright', 'credits' or 'license' for
more information
IPython 6.2.0 -- An enhanced Interactive Python.
Type '?' for help.
In [ ]:
~/Work
scott in ~/Work at 23:04:20
ls
Classes
Conferences
Developer
NSP
Open-Source
Research
Untitled.ipynb
Untitled1.ipynb
Untitled2.ipynb
tmux-client-71176.log
untitled.txt
x.log
scott in ~/Work at 23:04:21

Live demo

Benefits

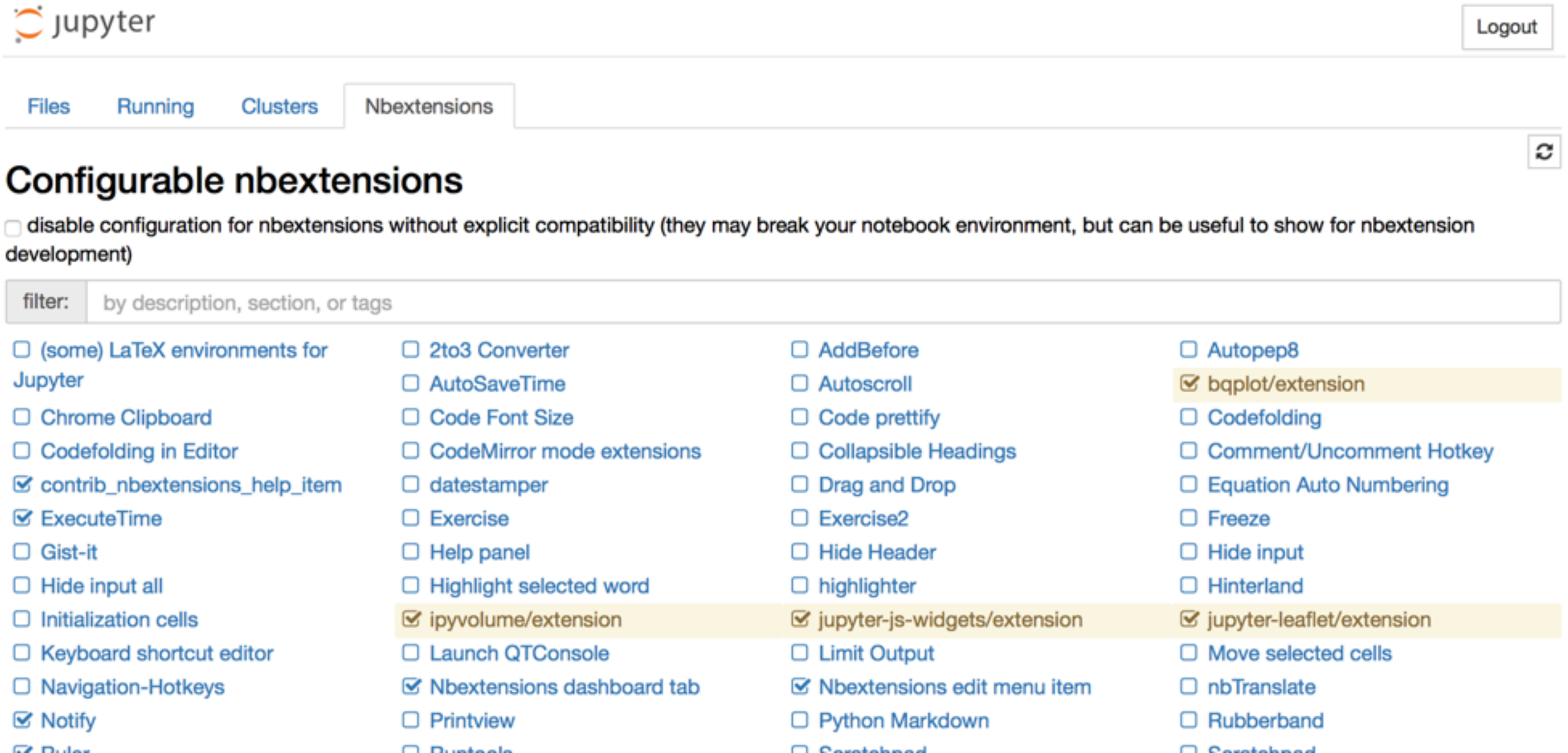
- Makes development cycle interactive
    - Quicker to see and check outputs
- Easy to develop plugins
    - e.g., live markdown preview is super easy
    - full tutorial: https://github.com/jupyterlab/jupyterlab/blob/master/docs/xkcd_extension_tutorial.md
- Jupyterlab has powerful access to files
    - Live updating

Tools

- **nbdime**: git diff/merge
- **nbflow**: Data input/output for different notebooks
- **nbextensions**: Many useful tools
- **jupyterhub**: Allows *easy* to cluster
- **nbgrader**: allows students to submit HW with JupyterHub

nbextensions

- Easy install, many extensions
- https://github.com/ipython-contrib/jupyter_contrib_nbextensions



nbdime

- Git diff/merge of Jupyter notebooks
- http://nbdime.readthedocs.io/en/latest/

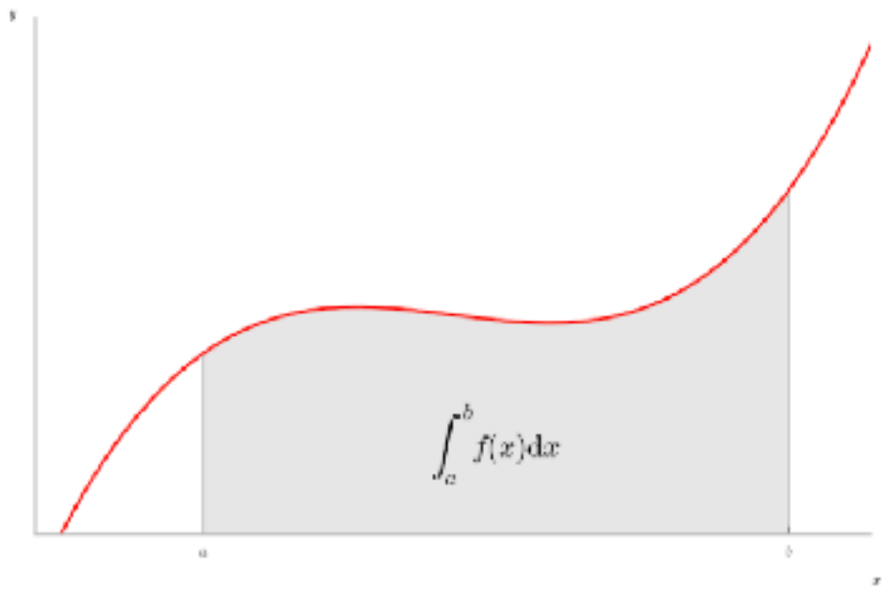
In [4]:

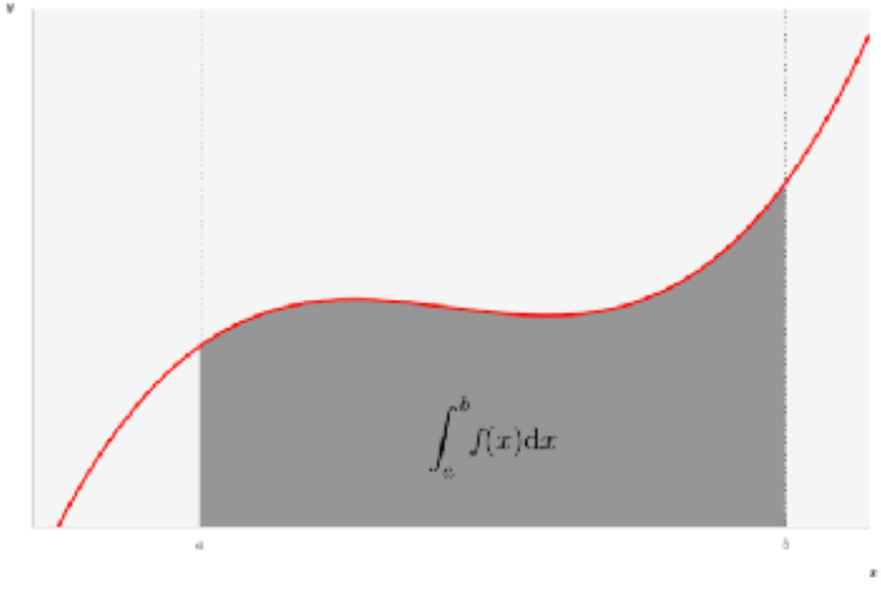
```
(...)
iy = func(ix)
verts = [(a, 0)] + list(zip(ix, iy)) + [(b, 0)
poly = Polygon(verts, facecolor='0.9', edgecol
ax.add_patch(poly)

(...)
```

In [4]:

```
(...)
iy = func(ix)
verts = [(a, 0)] + list(zip(ix, iy)) + [(b, 0)
poly = Polygon(verts, facecolor='0.6', edgecol
ax.add_patch(poly)

(...)
```

Outputs changed





nbflow

- Define workflows with data files. i.e., .py => .csv => .ipynb => .png => .tex => .pdf
    - Can work with more complicated DAGs

- Useful when writing papers, and many input/output files for every other file

- https://github.com/jhamrick/nbflow

JupyterHub + nbgrader

- Easily allow students to code up assignments
- Zero software install needed on their part
- Interact and submit homework through web application

JupyterHub Interface

1. Go to URL
2. Enter password (and optionally a username too)
3. Click a big blue button
4. Profit!