



Jupyter Notebook

A tool for well-documented, reproducible
bioinformatics workflows

Mike Trizna

Smithsonian Genomic Tools Workshop
March 24, 2016

Open Science


"Open science is the practice of making various elements of scientific research -- data & methods, code & software, and results & publications -- readily accessible to anyone."

--<http://ropensci.org/about/>

Open Science

How many times have you been reading a cool new publication, and then when you get to the Methods section, you see this term:

"In-house Perl scripts"

[Display Settings:](#) ☒ Summary, 20 per page, Sorted by Pub Date[Send to:](#) ☐**Results: 1 to 20 of 108**<< First < Prev Page **1** of 6 Next > Last >> **Limits Activated:** Published in the last 1 year [Change](#) | [Remove](#)☐ [Chado use case: storing genomic, genetic and breeding data of Rosaceae and Gossypium crops in Chado](#)

1. Sook Jung, Taein Lee, Stephen Ficklin, Jing Yu, Chun-Huai Cheng, Dorrie Main
Database (Oxford) 2016; 2016: baw010. Published online 2016 March 17. doi: 10.1093/database/baw010
PMCID: PMC4795932
[Article](#) [PubReader](#) [PDF-1.8M](#) [Citation](#)

☐ [Non-typhoidal *Salmonella* DNA traces in gallbladder cancer](#)

2. Prajish Iyer, Savio George Barreto, Bikram Sahoo, Pratik Chandrani, Mukta R. Ramadwar, Shailesh V. Shrikhande, Amit Dutt
Infect Agent Cancer. 2016; 11: 12. Published online 2016 March 3. doi: 10.1186/s13027-016-0057-x
PMCID: PMC4776363
[Article](#) [PubReader](#) [PDF-485K](#) [Citation](#)

☐ [De novo transcriptome assembly and comprehensive expression profiling in *Crocus sativus* to gain insights into apocarotenoid biosynthesis](#)

3. Mukesh Jain, Prabhakar Lal Srivastava, Mohit Verma, Rajesh Ghangal, Rohini Garg
Sci Rep. 2016; 6: 22456. Published online 2016 March 3. doi: 10.1038/srep22456



Philipp Bayer

@PhilippBayer



Following

Data was analysed using in-house Perl-scripts. They are terrible and possibly wrong so please don't ask for them. **#overlyhonestmethods**

RETWEETS

94

LIKES

35



1:52 AM - 8 Jan 2013



Source: <https://twitter.com/philippbayer/status/288538682032545792>

Jupyter Notebook



A web application that allows you to create and share documents that contain live code, equations, visualizations and explanatory text.

What is Jupyter?

Started as IPython Notebook:

- Browser-based
- Organized by "cells", which could either be Python code or Markdown
- Has its own file format ".ipynb", which can be shared

IP[y]: IPython
Interactive Computing

What is Jupyter?

From beginning, IPython Notebook had ability to run Python code in code cells.

However, more and more support was added to run shell commands, and then ability to create "kernels" for other languages to build notebooks on.

IPython kernels for other languages

Gordon Ball edited this page 3 days ago · 97 revisions

IPython/Jupyter kernels:

The Kernel Zero, is of course IPython, which you can get through `ipykernel`, and still comes (for now) as a dependency of `jupyter`. The IPython kernel can be thought as a reference implementation, here are other available kernels:

Name	Jupyter/IPython Version	Language(s) Version	3rd party dependencies	Example Notebooks	Notes
IPyKernel	Jupyter 4.0	python 2.7, >= 3.3	pyzmq		
IJulia		julia >= 0.3			
IHaskell		ghc >= 7.6			
IRuby		ruby >= 2.1			
IJavascript		nodejs >= 0.10			
JpCoffeeScript		coffeescript >= 1.7			
ICSharp	Jupyter 4.0	C# 4.0+	scriptcs		
IRKernel	IPython 3.0	R 3.2	rmq		
SageMath	Jupyter 4	Any	many		
pari_jupyter	Jupyter 4	2.8	Cython		
IFSharp	IPython 2.0	F#		Features	
Gophernotes	Jupyter 4	Go >= 1.4	zeromq 2.2.x	examples	docker image
IGo		Go >= 1.4			
IScala		Scala			
Jupyter-scala	IPython>=3.0	Scala>=2.10		example	

Source: <https://github.com/ipython/ipython/wiki/IPython%20kernels%20for%20other%20languages>

What is Jupyter?

Support for other languages grew to the point that "IPython" notebook name was confusing, and project decided to re-brand as "Jupyter". This comes from the first main 3 data science languages that adopted the notebook format: **Julia**, **Python**, and **R**

Source: <http://blog.jupyter.org/2015/04/15/the-big-split/>

The Big Split™

15 APRIL 2015

IPython has grown a great deal over the years. As of 3.0, IPython includes:

- an interactive shell
- a REPL protocol
- a notebook document format
- a notebook document conversion tool
- a web-based notebook authoring tool
- tools for building interactive UI (widgets)
- interactive parallel Python based on the above REPL protocol

While all of these are part of the same story of tools for the lifecycle of a computational idea, they are increasingly becoming distinct projects that happen to live in a single repo. One significant part of the development is that pieces like the notebook and protocol are not even specific to Python, so it doesn't make sense anymore that they reside in a project called Interactive Python. This is the impetus for Project Jupyter, announced at SciPy 2014, which is the new home of language-agnostic projects that began as part of IPython, such as the notebook.

If anyone has been confused by what Jupyter is¹, it's the exact same code that lived in IPython, developed by the same people, just in a new home under a new name.

Some Examples

A 2014 article in Nature about IPython Notebook that also included a self-hosted notebook for demonstration purposes.

Source: <http://www.nature.com/news/interactive-notebooks-sharing-the-code-1.16261>



The screenshot shows the top of a Nature article page. The header includes the 'nature' logo and navigation links like 'Home', 'News & Comment', 'Research', etc. The article title is 'Interactive notebooks: Sharing the code' by Helen Shen, dated 05 November 2014. Below the title is a large illustration showing hands interacting with a digital notebook interface that displays various charts and data. The text below the illustration describes how Titus Brown, a bioinformatician, uses IPython notebooks to share and collaborate on his research, specifically mentioning RNA sequencing data. The right sidebar contains a 'Jellies' feature, a 'Recent' list of articles, and a 'Newsletter' sign-up section.

nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video | For Authors

Archive | Volume 515 | Issue 7525 | Toolbox | Article

NATURE | TOOLBOX

Interactive notebooks: Sharing the code

The free IPython notebook makes data analysis easier to record, understand and reproduce.

Helen Shen

05 November 2014

PDF | Rights & Permissions



Illustrations by The Project Twins

Flying high above the Pacific Ocean, Titus Brown is taking a deep dive into his students' research code. The long journey from Michigan State University in East Lansing to a conference in Melbourne, Australia, provides the perfect chance for the bioinformatician to scrutinize his lab's new algorithm for removing errors from RNA sequencing data.

Three years ago, Brown might have waited until he was back in his office. It is difficult to dig into other researchers' code without them being present to explain it, make changes and produce updated results. But these days, Brown can work with his lab from afar using a free, open-source software package called IPython, which helps researchers to keep a detailed lab notebook for their computational work.

Brown's students write explanatory text and intersperse it with raw code and the charts and figures that their algorithms generate. Sitting in the aeroplane with an IPython notebook downloaded to his laptop, Brown can interact with the work. He tweaks and re-runs the code, which executes directly in

Jellies



The secret
Long regard
jellyfish are
web.

Recent

1. Minisynthe
Nature
2. Share
Nature
3. Prestig
contro
Nature

Newsletter

The best so
to your info

Email*

Submit

Captive or

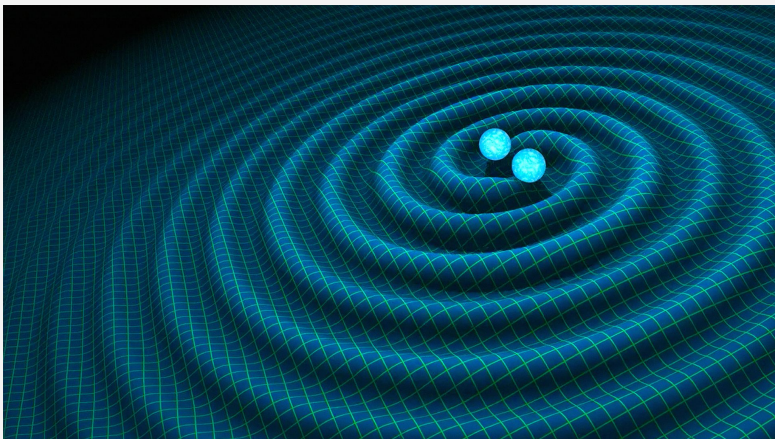


Scientists c
whales

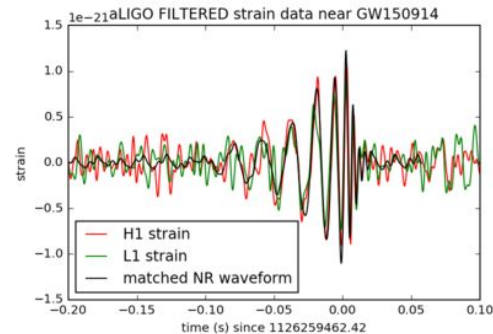
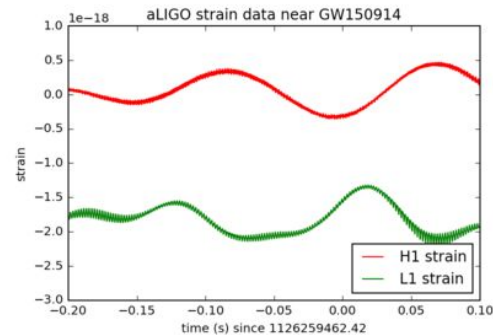
Two confid

Some Examples

The February 2016 announcement by the Laser Interferometer Gravitational-Wave Observatory (LIGO) included a Jupyter notebook to show how their data was analyzed.



```
plt.plot(time-tevent, strain_L1_files, 'g', label='L1 strain')
plt.plot(NRtime+0.002, NR_H1_filt, 'k', label='matched NR waveform')
plt.xlim([-0.2, 0.1])
plt.ylim([-1.5e-21, 1.5e-21])
plt.xlabel('time (s) since '+str(tevent))
plt.ylabel('strain')
plt.legend(loc='lower left')
plt.title('aLIGO FILTERED strain data near GW150914')
plt.savefig('GW150914_H1_strain_filtered.png')
```



Source: https://lsc.ligo.org/s/events/GW150914/GW150914_tutorial.html

Some examples

There are entire textbooks written as interactive Jupyter notebooks.

Two good examples are "[An Introduction To Applied Bioinformatics](#)" by Greg Caporaso, and "[Probabilistic Programming & Bayesian Methods for Hackers](#)" by Cam Davidson Pilon

 jupyter
nbviewer

JUPYTERFAQ

Probabilistic-Programming-and-Bayesian-Methods-for-Hackers / Chapter1_Introduction

Probabilistic Programming and Bayesian Methods for Hackers

Welcome to *Bayesian Methods for Hackers*. The full Github repository is available at [github/Probabilistic-Programming-and-Bayesian-Methods-for-Hackers](#). The other chapters can be found on the project's [homepage](#). We hope you enjoy the book, and we encourage any contributions!

Looking for a printed version of *Bayesian Methods for Hackers*?

Bayesian Methods for Hackers is now a published book by Addison-Wesley, available on [Amazon!](#)



Some examples

The IPython project maintains a curated "gallery" of interesting Jupyter notebooks, that includes examples from several different scientific fields and applications.

Source: <https://github.com/ipython/ipython/wiki/A-gallery-of-interesting-IPython-Notebooks>

A gallery of interesting IPython Notebooks

Fernando Perez edited this page a day ago · 266 revisions

This page is a curated collection of IPython notebooks that are notable for some reason. Feel free to add new content here, but please try to only include links to notebooks that include interesting visual or technical content; this should *not* simply be a dump of a Google search on every ipynb file out there.

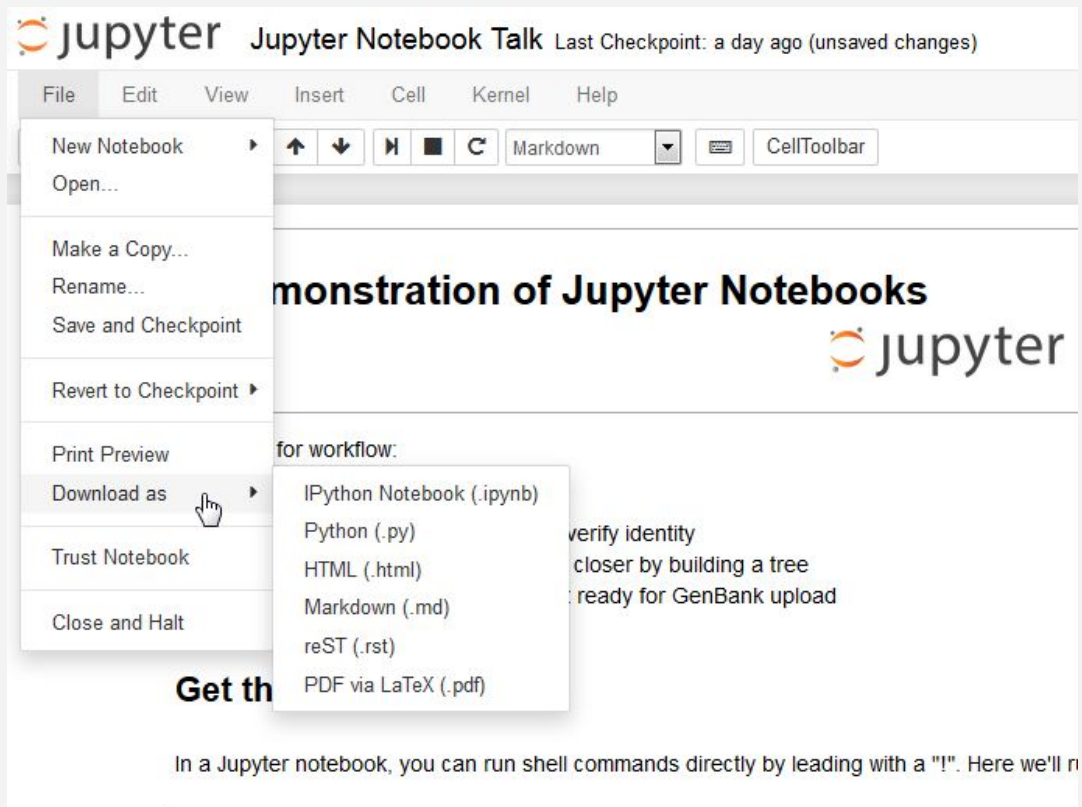
Table of Contents

1. Entire books or other large collections of notebooks on a topic
 - Introductory Tutorials
 - Programming and Computer Science
 - Statistics, Machine Learning and Data Science
 - Mathematics, Physics, Chemistry, Biology
 - Earth Science and Geo-Spatial data
 - Linguistics and Text Mining
 - Signal Processing
 - Engineering Education
2. Scientific computing and data analysis with the SciPy Stack
 - General topics in scientific computing
 - Social data
 - Psychology and Neuroscience
 - Machine Learning, Statistics and Probability
 - Physics, Chemistry and Biology
 - Economics and Finance
 - Earth science and geo-spatial data
 - Data visualization and plotting
 - Mathematics
 - Signal and Sound Processing
 - Natural Language Processing
 - Pandas for data analysis
3. General Python Programming
4. Notebooks in languages other than Python
 - Julia
 - Haskell

How can I use Jupyter Notebooks?

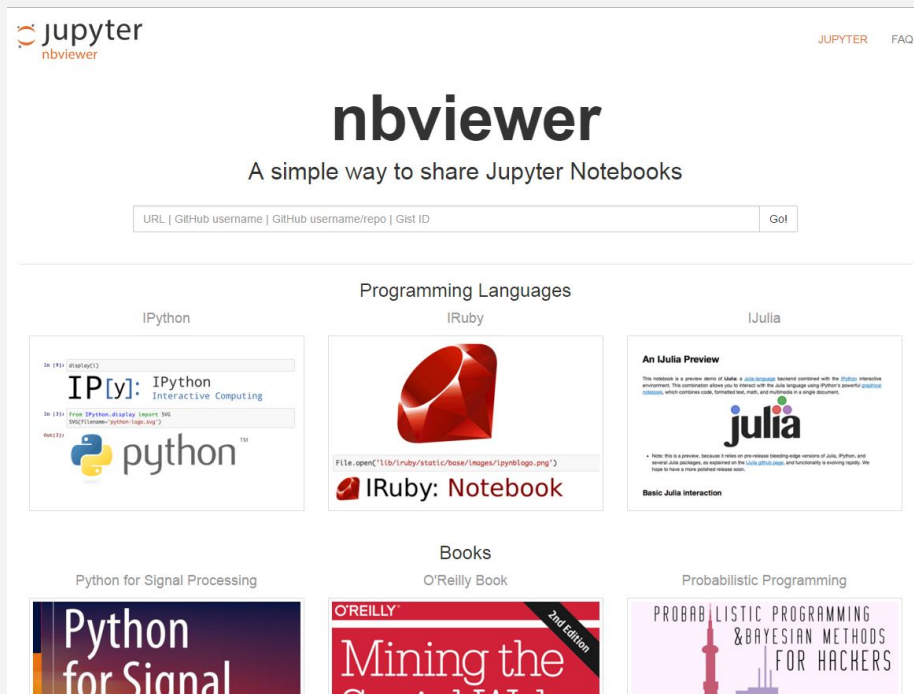
How can I use Jupyter Notebooks?

If you have created a notebook, you can export it to a variety of different formats for people to view, such as HTML (as a webpage), Markdown or reST (which makes it easy to insert into blog posts), or PDF.



How can I use Jupyter Notebooks?

GitHub added native support for .ipynb files, which means that you can view Jupyter Notebooks posted to GitHub repositories and to Gists.



Source: <http://blog.jupyter.org/2015/05/07/rendering-notebooks-on-github/>

How can I use Jupyter Notebooks?

GitHub added native support for .ipynb files, which means that you can view Jupyter Notebooks posted to GitHub repositories and to Gists.

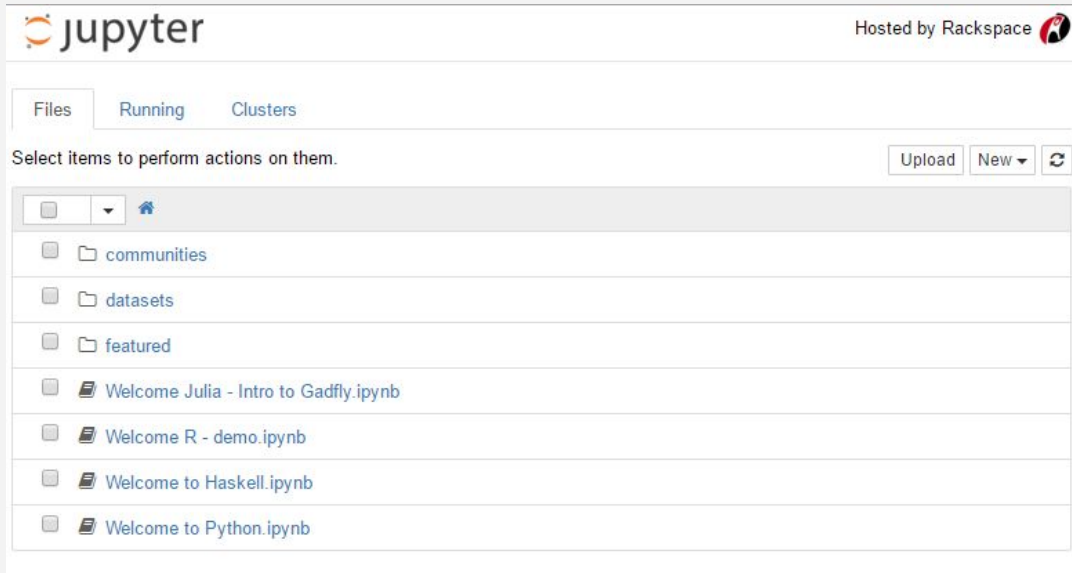


Source: <http://blog.jupyter.org/2015/05/07/rendering-notebooks-on-github/>

How can I create my own Jupyter Notebooks?

Jupyter.org has a Rackspace-hosted kernel where you can try out how Jupyter Notebooks work -- all without installing a single piece of software.

You can find this at:
<https://try.jupyter.org/>



How can I create my own Jupyter Notebooks?

If you already have Python installed on your computer, installing Jupyter is as simple as:

"pip install jupyter notebook"

If you do not have Python installed, I recommend installing Miniconda from <http://conda.pydata.org/miniconda.html>, which is cross-platform. Then you can install Jupyter using:

"conda install jupyter notebook"

Conda

Miniconda

	 Windows	 Mac OS X	 Linux
Python 2.7	64-bit (exe installer) 32-bit (exe installer)	64-bit (bash installer)	64-bit (bash installer) 32-bit (bash installer)
Python 3.5	64-bit (exe installer) 32-bit (exe installer)	64-bit (bash installer)	64-bit (bash installer) 32-bit (bash installer)

Ok, now onto the Demo...

Which you can find here:

https://github.com/MikeTrizna/jupyter_notebook_presentation/blob/master/Jupyter%20Notebook%20Talk.ipynb