

To Jupyter and Beyond

Presented by Tim Ribaric and Daniel Brett

First, a commercial!

The screenshot shows a ticketing page for an event. At the top, there is a navigation bar with links for 'Home', 'Code of Conduct', 'License', and 'Improve this page', along with a search bar. The main content area features the event title 'OLITA presents: Library Carpentry' and the venue 'Ryerson University'. The dates are 'April 29 - April 30, 2020' and the time is '9:00 am - 4:30 pm'. The instructors listed are 'Tim Ribaric, Kaitlin Newson' and the helpers are 'John Fink'. Below this, there is a 'Tickets' section with a dropdown menu for quantity (set to 1) and a price of '\$20.00'. At the bottom, there are icons for payment methods (MasterCard, Visa, American Express) and a green 'Order Now' button.

OLITA presents: Library Carpentry
Ryerson University
April 29 - April 30, 2020
9:00 am - 4:30 pm
Instructors: Tim Ribaric, Kaitlin Newson
Helpers: John Fink

Tickets

WED, 29 APR 2020, 9:00 AM - THU, 30 APR 2020, 4:30 PM EDT
Sales end on 30 Apr

General Admission
\$20.00

QTY: 1 CA\$20.00

Order Now

→ Agenda

What is a computational notebook?

Typical Use Cases

How to get and use Jupyter?

Live Demo

Questions / Comments?



What is a Computational Notebook?

Have you heard this phrase before?

→ What a Computational Notebook Is

First introduced in 1988 with Wolfram Mathematica, followed closely by Maple in 1989

You can think of them as an analogue to the notebook a scientist would keep track of their experiments and finds with

→ How it works

It is a virtual notebook environment used for something referred to as **Literate programming**

Basically it pairs word processing with two things called a shell and a kernel

→ What it looks like

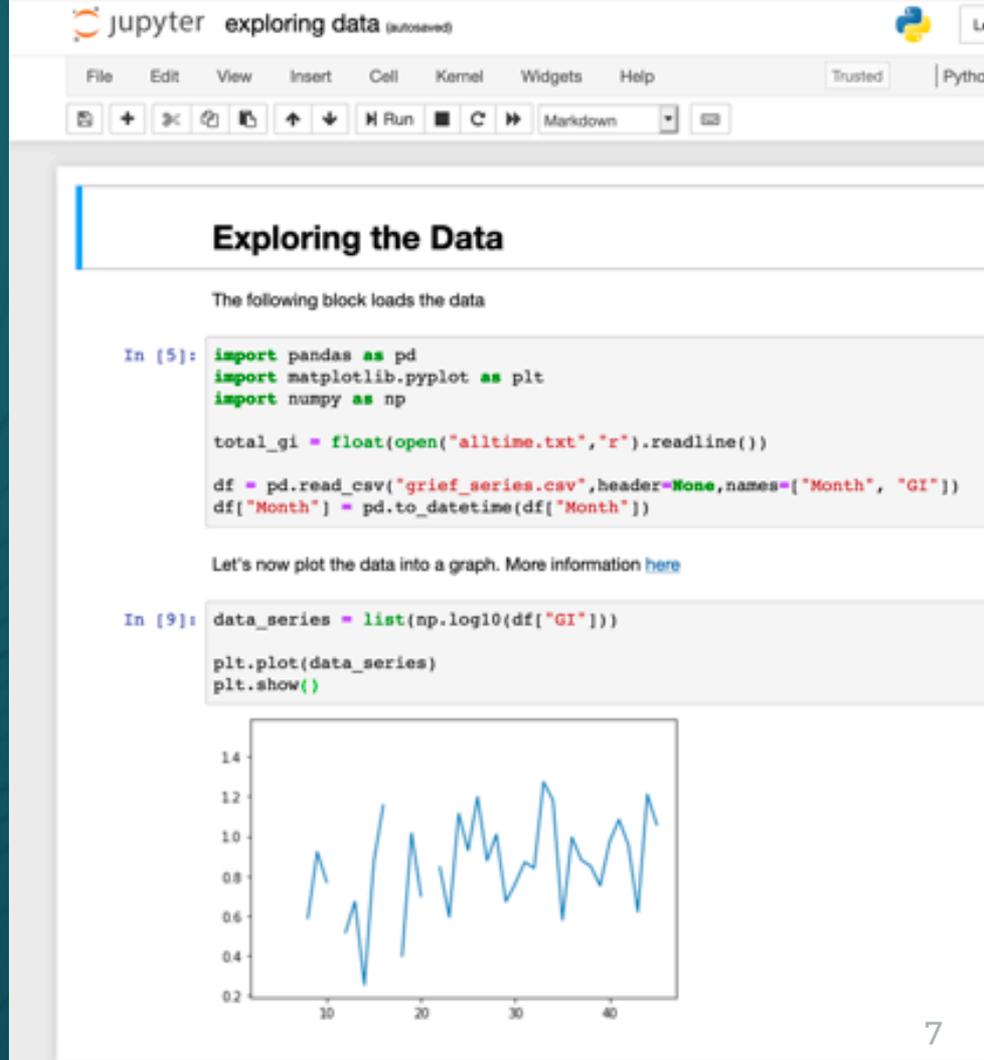
A webpage with a bunch of text boxes on it!

Text boxes are referred to as cells

Cells come in two types:

- Code

- Markdown



The screenshot shows a Jupyter Notebook window titled "exploring data (autosaved)". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, navigation, and execution. The notebook content is as follows:

Exploring the Data

The following block loads the data

```
In [5]: import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

total_gi = float(open("alltime.txt", "r").readline())

df = pd.read_csv("grief_series.csv", header=None, names=["Month", "GI"])
df["Month"] = pd.to_datetime(df["Month"])
```

Let's now plot the data into a graph. More information [here](#)

```
In [9]: data_series = list(np.log10(df["GI"]))

plt.plot(data_series)
plt.show()
```

The plot displays a line graph with the x-axis ranging from 0 to 45 and the y-axis ranging from 0.2 to 1.4. The data points are connected by a blue line, showing a fluctuating trend that generally increases over time, with several peaks and troughs.

Cells - Markdown

A syntax that makes HTML



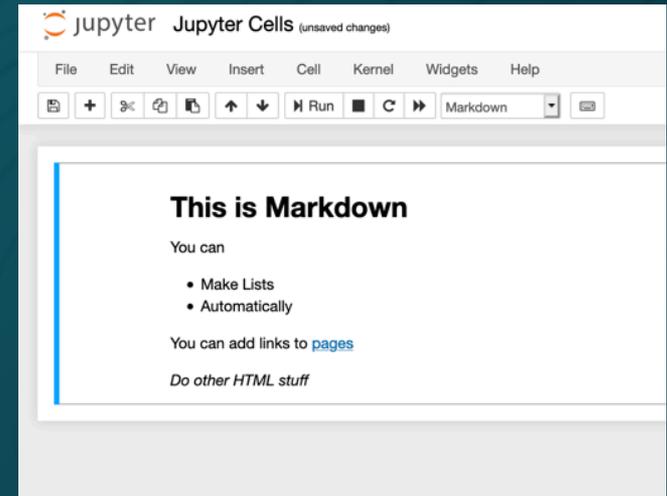
The screenshot shows a Jupyter Cell editor window titled "Jupyter Cell (unsaved changes)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The toolbar contains icons for adding, deleting, and running cells, along with a dropdown menu set to "Markdown". The cell content is as follows:

```
# This is Markdown

You can
- Make Lists
- Automatically

You can add links to [pages](http://google.com)

*Do other HTML stuff*
```



The screenshot shows the same Jupyter Cell editor window, but now displaying the rendered output of the Markdown code. The title bar reads "Jupyter Cells (unsaved changes)". The rendered content is:

This is Markdown

You can

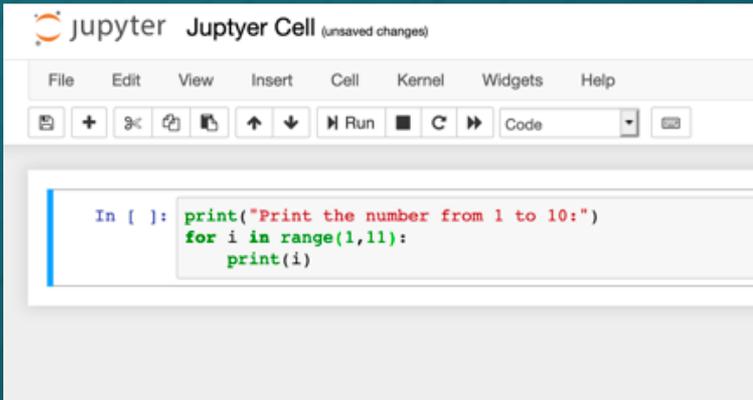
- Make Lists
- Automatically

You can add links to [pages](http://google.com)

Do other HTML stuff

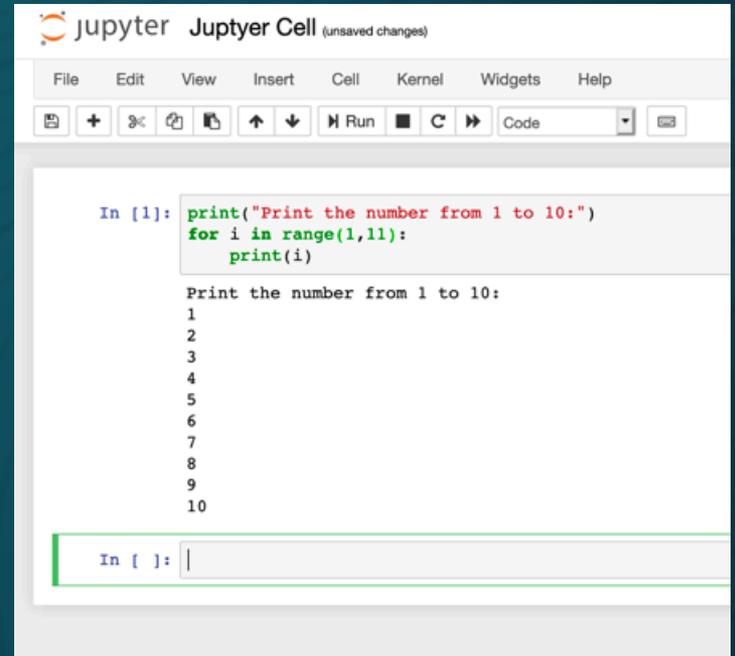
Cells - Code

This is the computer code
Simple!



A screenshot of the Jupyter interface showing a single code cell. The cell contains the following Python code:

```
In [ ]: print("Print the number from 1 to 10:")  
for i in range(1,11):  
    print(i)
```



A screenshot of the Jupyter interface showing the same code cell as the left screenshot, but now with the output displayed below the code. The output is:

```
In [1]: print("Print the number from 1 to 10:")  
for i in range(1,11):  
    print(i)  
  
Print the number from 1 to 10:  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

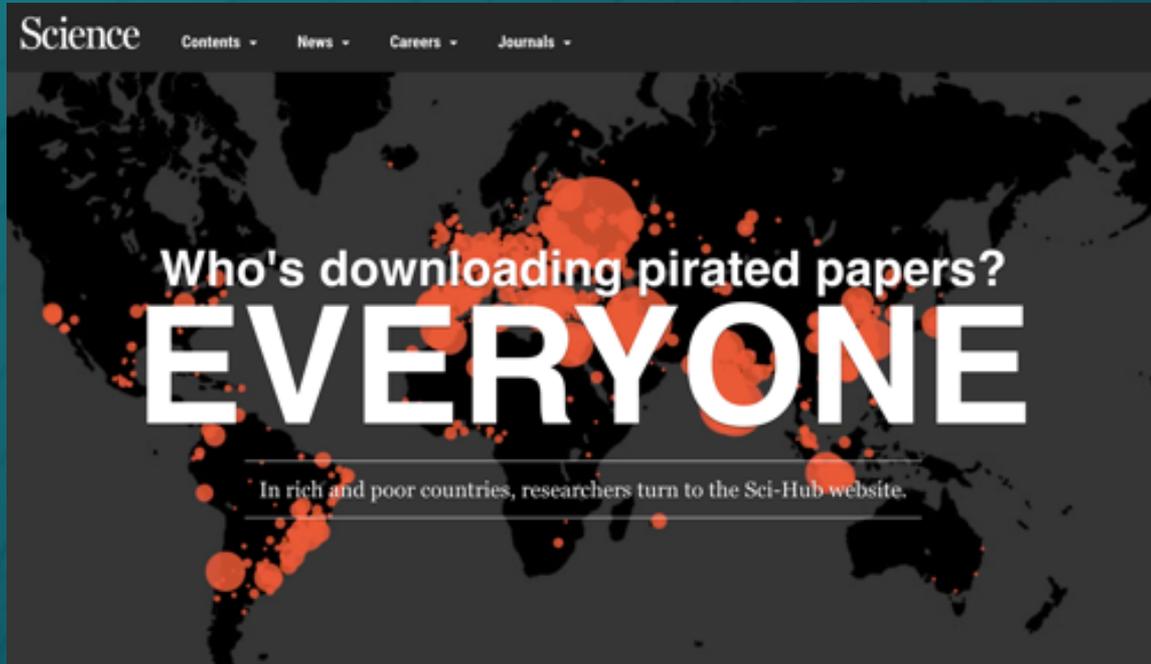


That's pretty much it!



How are they being used today?

→ Repeatable Research



→ Repeatable Research Cont.



Data from: Who's downloading pirated papers? Everyone

Bohannon, John

Elbakyan, Alexandra

Publication date: April 22, 2017

Publisher: Dryad

<https://doi.org/10.5061/dryad.q447c>

Citation

Bohannon, John; Elbakyan, Alexandra (2017), Data from: Who's downloading pirated papers? Everyone, Dryad, Dataset, <https://doi.org/10.5061/dryad.q447c>

Usage Notes

Sci-Hub download data

These data include 28 million download request events from the server logs of Sci-Hub from 1 September 2015 through 29 February 2016. The uncompressed 2.7 gigabytes of data are separated into 6 data files, one for each month, in tab-delimited text format.

scihub_data.zip

IPython Notebook for Sci-Hub raw data

IPython Notebook used to process the raw server log data (processing the GIS files into CSV, scraping DOI metadata, etc.).

Sci-Hub.html

Sci-Hub.ipynb

Sci-Hub publisher DOI prefixes

Data scraped from the CrossRef website which can be used to replicate the analysis of downloads by publisher.

publisher_DOI_prefixes.csv

References

This dataset is supplement to <https://doi.org/10.1126/science.352.6285.508>

<https://www.sciencemag.org/news/2016/04/whos-downloading-pirated-papers-everyone>

→ In the DSL

TWARC tool to harvest tweets that include #datavisualization

Ran text processing on the data in a Jupyter Notebook using R

Finally, we fed the cleaned text to Voyant to produce a word cloud



→ Use for tutorials

Low overhead for users. (Nothing to install)

At the DSL we use Jupyter Notebooks to contain our tutorials that use Python or R code

Our Python 2.0 workshop comes with a Jupyter Notebook that learners can run through

https://brockdsl.github.io/Python_2.0_Workshop/



Brock University Library
Digital Scholarship Lab

→ Netflix

Integral part of their Data Platform since 2017

“The most popular tool for working with data at Netflix” (Netflix TechBlog, 2018)

Use for creating reusable templates for frequent tasks

And much more!



→ Kaggle



Great source of free datasets

Hub where Jupyter Notebooks containing all kinds of projects and code are shared freely

High profile coding competitions with Jupyter Notebooks as the preferred submission method.

Competitions range from Passenger Screening Algorithm Challenges to solving Santa's workshop efficiency problems

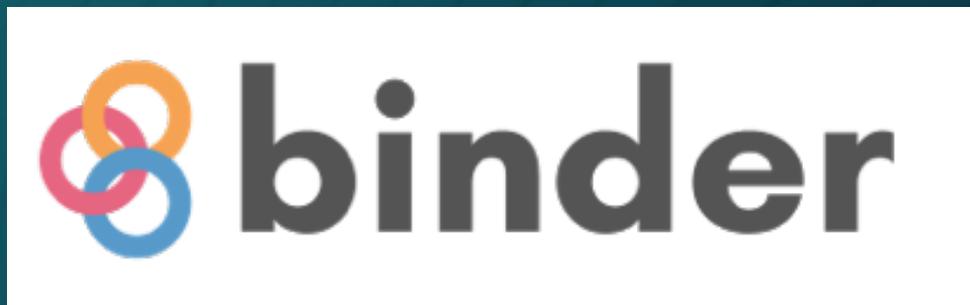


How do we get to Jupyter?

→ My Binder

A tool that turns a repository of Jupyter Notebooks on GitHub into a clean, web accessible interface.

Allows you to share Jupyter Notebooks with others without them needing Jupyter on their computers.



<https://mybinder.org/>

→ ArcGIS Notebooks

Allows you to build and run Notebooks that are connected to all of your Arc resources and your ArcGIS Online storage space.

Gives access to repeatable ArcGIS functions and mapping tools from within the notebook.

<https://www.esri.com/arcgis-blog/products/arcgis-enterprise/analytics/introducing-arcgis-notebooks/>



→ Syzygy – Via Compute Canada

A Jupyter environment accessible on the web that uses your existing single-sign-on system to allow access

Hosted by Compute Canada and Cybera

Only for Universities unfortunately



<https://syzygy.ca/>

→ Desktop Version via Anaconda

A Desktop version of Jupyter is available through a free distribution software called Anaconda

Package of statistical and mathematical tools useful for many things

All free



<https://www.anaconda.com>



Live Demo



Questions / Comments?

→ Credits

dsl@brocku.ca

<http://brocku.ca/library/dsl>

Presentation template by
[SlidesCarnival](#)

Tim Ribaric

tribaric@brocku.ca

[@elibtronic](#)

Daniel Brett

dbrett@brocku.ca