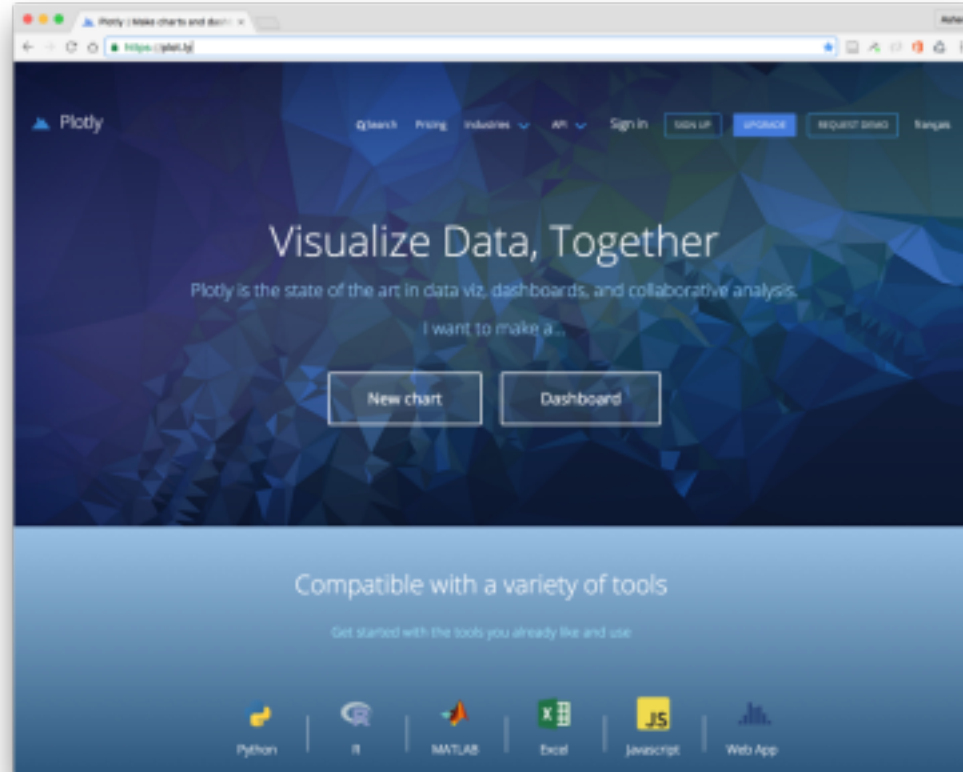


Getting Started with Plot.ly

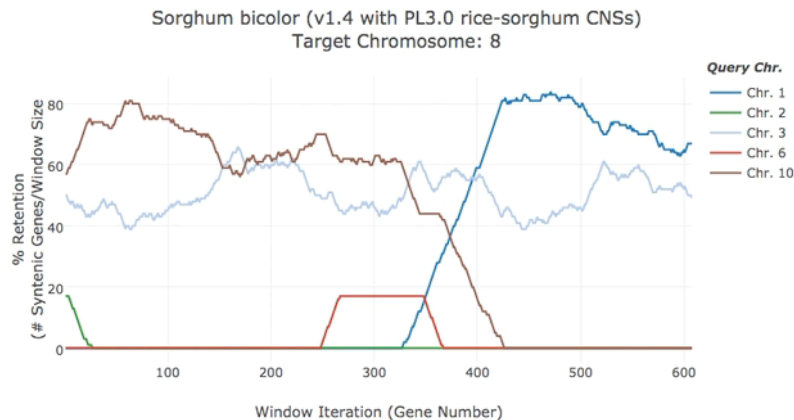


What IS Plot.ly?



Plot.ly is an open-source* data visualization tool.

- ... was built using **Python** (Django) & JavaScript.
- ... offers a web application for **visualization & analysis**.
- ... provides plotting **APIs** for many popular languages.
- ... plots are **fully interactive**, and rendered with D3.js or WebGL (for 3D).
- ... free, paid, and on-site offerings.



* well, *mostly* open-source.

Step 1: *Install Plotly Python Package*

Installing is easy...just use PIP!

```
(sudo) pip install plotly
```

Updating is also easy...YAY!

```
(sudo) pip install plotly --upgrade
```

Sidenote...BOO!

Plot.ly recommends “frequently updating” the package, as they tend to release new features/bug fixes ~1x per week.

Personally, I recommend doing this in a virtual environment, just in case one of these updates would break your code.

```

root@kali:~# sudo docker exec -it virtualenv-plytl-dm /bin/bash
$ python --helpable in /Users/vahernik/virtualenv/plytl-dm/bin/python2.7
Also creating executable in /Users/vahernik/virtualenv/plytl-dm/bin/python
Installing setuptools, pip, wheel...done.
virtualenvwrapper user_scripts creating /Users/vahernik/virtualenv/plytl-dm/bin/postactivate
virtualenvwrapper user_scripts creating /Users/vahernik/virtualenv/plytl-dm/bin/postdeactivate
virtualenvwrapper user_scripts creating /Users/vahernik/virtualenv/plytl-dm/bin/preactivate
virtualenvwrapper user_scripts creating /Users/vahernik/virtualenv/plytl-dm/bin/predeactivate
virtualenvwrapper user_scripts creating /Users/vahernik/virtualenv/plytl-dm/bin/get_env_details
(virtualenv) [~/virtualenv(plytl)] $ sudo pip install plytl
Collecting plytl
  Downloading plytl-1.12.0.tar.gz (74kB)
    100% |#####| 74kB 498B/s
Collecting requests [from plytl]
  Downloading requests-2.11.1-py2.py3-none-any.whl (131kB)
    100% |#####| 131kB 576B 576B/s
Collecting six [from plytl]
  Using cached six-1.10.0-py2.py3-none-any.whl
Collecting pytz [from plytl]
  Downloading pytz-2016.6.1-py2.py3-none-any.whl (44kB)
    100% |#####| 44kB 436B/s
building wheels for collected packages: requests, six, pytz, plytl
Running setup.py bdist_wheel for plytl ... done
Stored in directory: /Users/vahernik/Library/Caches/pip/wheels/1c/bd/cb/12f3e0eb04020212a0e0b049f44f47bb043db01ed
Successfully built plytl
Installing collected packages: requests, six, pytz, plytl
Successfully installed plytl-1.12.0 pytz-2016.6.1 requests-2.11.1 six-1.10.0
(virtualenv) [~/virtualenv(plytl)] $ sudo pip install plytl --upgrade
Requirement already up-to-date: plytl in /Users/vahernik/virtualenv/plytl-dm/lib/python2.7/site-packages
Requirement already up-to-date: six in /Users/vahernik/virtualenv/plytl-dm/lib/python2.7/site-packages
Requirement already up-to-date: pytz in /Users/vahernik/virtualenv/plytl-dm/lib/python2.7/site-packages
Requirement already up-to-date: requests in /Users/vahernik/virtualenv/plytl-dm/lib/python2.7/site-packages
(virtualenv) [~/virtualenv(plytl)] $ sudo pip install

```

Step 2: Setup your API keys

...these are necessary for communicating with Plot.ly's servers.

Question: *What is an API?*

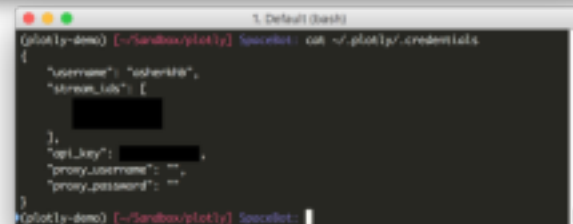
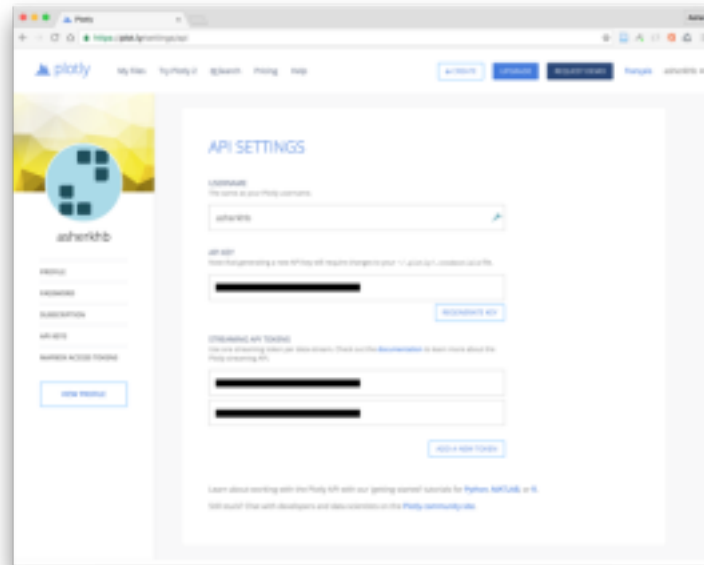
Answer: An “Application Programming Interface”

Question: ...?

Slightly More Useful Answer: *A way to interact with someone else's program, using programming languages rather than a graphical user interface*

```
In [1] import plotly
In [2] plotly.tools.set_credentials_file( \
        username='YourAccount', \
        api key='YourKey')
```

NOTE: *You only need to do this once on your computer account, even if you use different virtual environments.*



Step 3: *Compose a Plot*

Figure() - The Plotly “Plot” Object

Composed of two parts: ‘Data’ and ‘Layout’

Data()

Contains the information to be plotted.

Composed of ‘trace’s

Layout()

Contains information about the plot

i.e. title, labels, fonts, annotations, etc.

Figure()

Combines Data() and Layout()

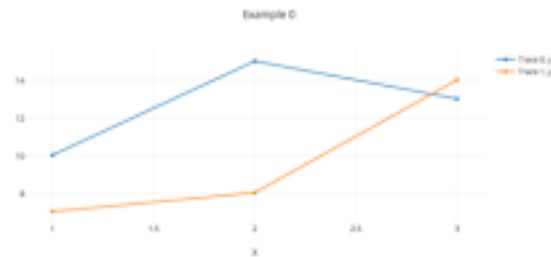
```
import plotly.plotly as py
from plotly.graph_objs import *

trace0 = Scatter(x=[1, 2, 3],y=[10, 15, 13])
trace1 = Scatter(x=[1, 2, 3],y=[7, 8, 14])
my_data = Data([trace0, trace1])

my_layout = Layout(
    title="Example 0",
    xaxis=dict(title="X"))

my_figure = Figure(
    data=my_data, layout=my_layout)

py.plot(my_figure, filename='example0')
```



<https://plot.ly/~asherkhb/462/example-0/>

Step 4: *Let's Plot!!*

Online

```
import plotly.plotly as py
```

Python

```
py.plot()
```

iPython

```
py.iplot()
```

Offline

```
import plotly.offline as pyo
```

Python

```
pyo.plot()
```

iPython

```
pyo.iplot()
```

Online vs Offline Plotting

Online Plotting

- Data is sent to Plotly's servers, plots are generated and links are returned.
- Embedded plots can be returned to iPython.

Pros

- Ability to adjust look of plots using GUI.
- Easy to share plots with other users.

Cons

- Internet connection required, need to set up API keys.
- Data is public (unless Pro account).

Offline Plotting

- Data is bundled with Plotly's library locally, made available as local .html files or divs.
- Embedded plots are available when using iPython.

Pros

- No internet connection or keys required.
- Data remains private without paying.

Cons

- Must programmatically adjust all visual parameters to achieve desired looks
- Some of Plotly's functionality not available.

Other Features & Benefits of Plotly

→ Matplotlib converter!

<https://plot.ly/matplotlib/>

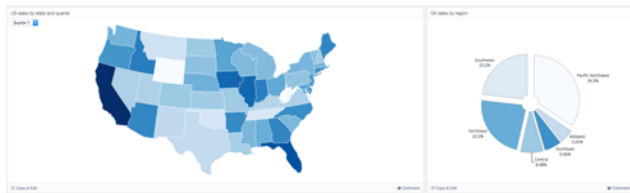


`py.plot_mpl(<mpl_fig>)`



→ Live & static dashboards!!

<https://plot.ly/dashboards/>



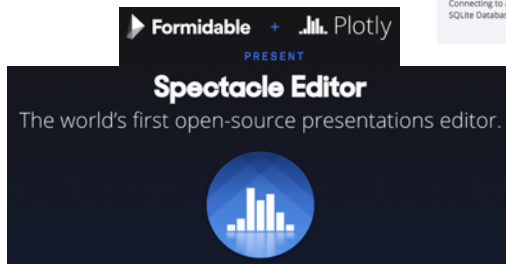
→ Database connections!*

<http://help.plot.ly/database-connectors/>



→ Presentations?!*

<https://formidable.com/open-source/spectacle-editor/>



→ Mobile Ready!



* untested by me personally.

Getting more **help**...

Check out the docs for basics & examples...

<https://plot.ly/python/>

But for more info, use the docstrings...

Python: **help(<call>)**

iPython: ***call?***

If you can't figure it out, post to the forums...

<http://community.plot.ly/>