PyLadies Stockholm | August 29th 2019

A tour of data viz in Python

Éléonore Mayola

@EleonoreMayola | github.com/eleonore9 | Elle-est-au-nord.com

hello

PyLadies Stockholm | August 29th 2019

Developer + Data scientist

Python, Clojure, JS, HTML/CSS

Freelance work

web + data

Community volunteering

P I N K
{programming}
pyladies
EMMS TECH

Endeavour

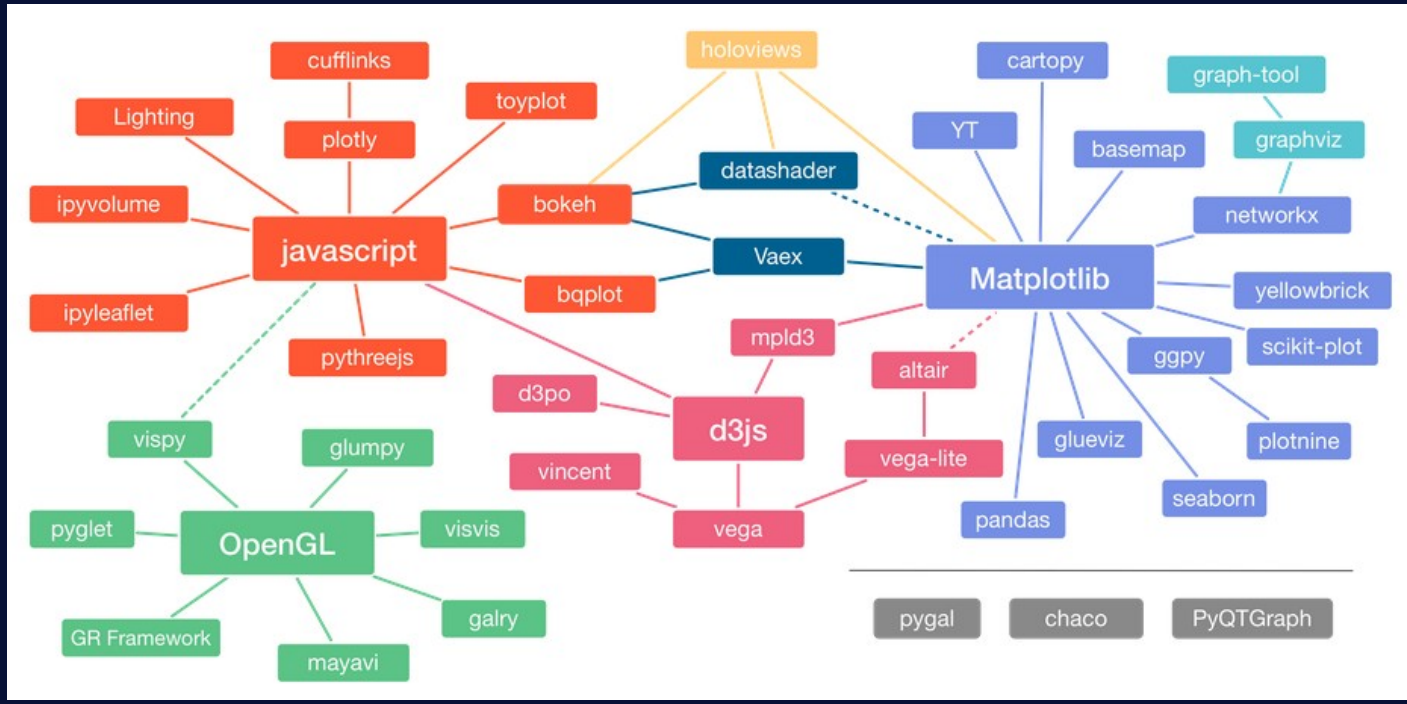
art + tech + data

@EleonoreMayola | github.com/eleonore9 | Elle-est-au-nord.com

Different needs for different use cases

* yourself
* your colleagues
* your manager
* your clients

* exploring a dataset
* writing an internal report
* writing a client report
* writing a research article

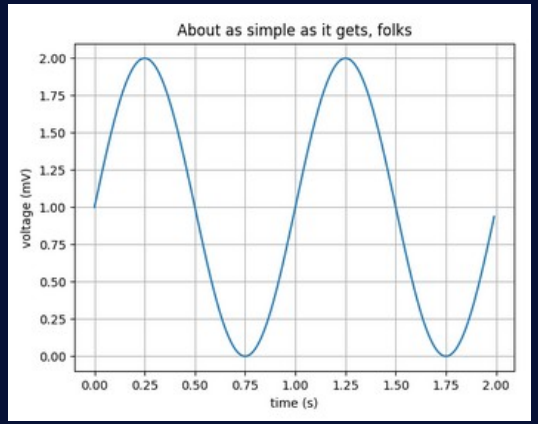


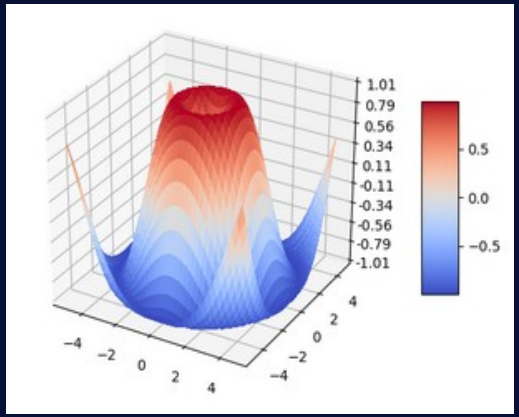
Adaptation of Jake VanderPlas graphic about the Python visualization landscape, by Nicolas P. Rougier
Source: https://pyviz.org/overviews/index.html

matplotlib

matplotlib.org

“Matplotlib tries to make easy things easy and hard things possible.”

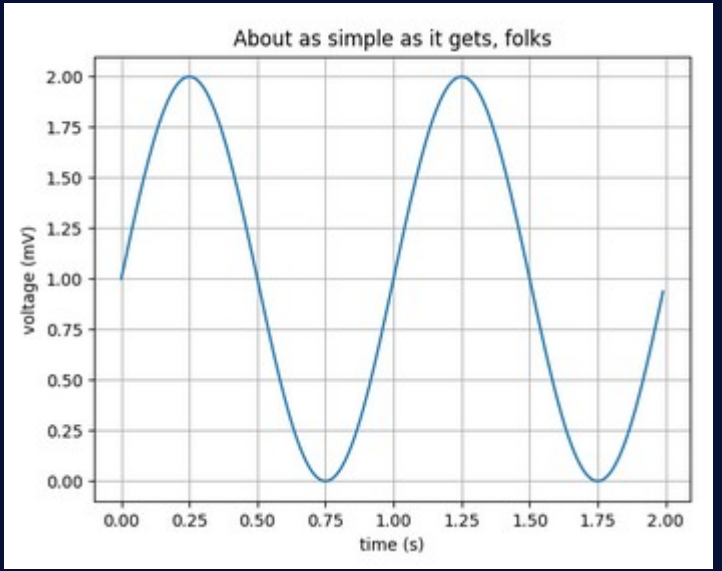




matplotlib

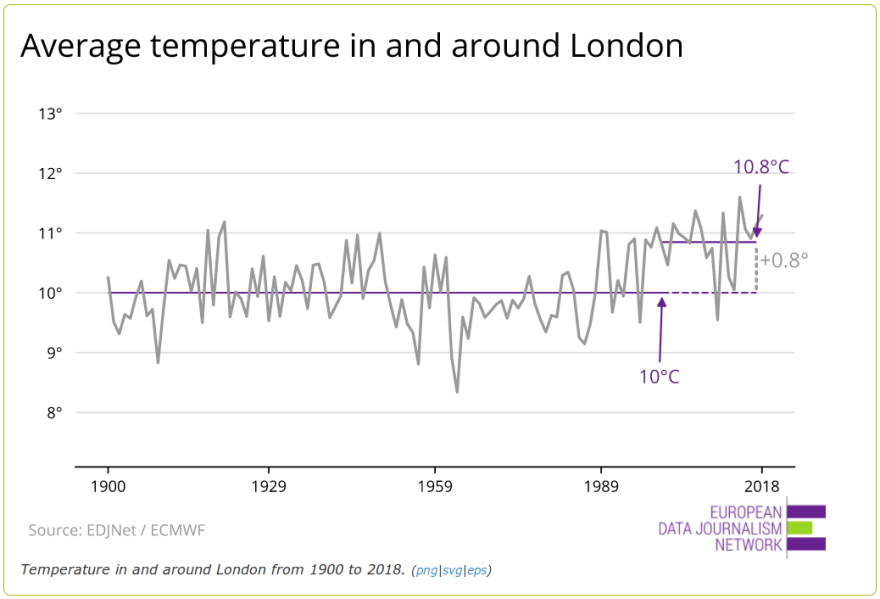
The **pyplot** module provides a MATLAB-like interface

```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np


# Data for plotting
t = np.arange(0.0, 2.0, 0.01)
s = 1 + np.sin(2 * np.pi * t)

fig, ax = plt.subplots()
ax.plot(t, s)

ax.set(xlabel='time (s)', ylabel='voltage (mV)',
       title='About as simple as it gets, folks')
ax.grid()

fig.savefig("test.png")
plt.show()
```



matplotlib

onedegreewarmer.eu



Customisable

Example: add labels, lines and annotations

```
matplotlib.axes.Axes.axhline
ax.axhline(y=10yrs_average, color='#6f2c91',
           linewidth=1.5)

matplotlib.pyplot.annotate
ax.annotate(f'{temperature}°C',
            Xy=(3, 1), xycoords='data',
            xytext=(0.8, 0.95),
            arrowprops=dict(facecolor='#6f2c91'),
            horizontalalignment='right',
            verticalalignment='top')
```

Look up the rich gallery of examples: **matplotlib.org/gallery/index.html**

Best for

Complex or customised plots

Challenge

Syntax can become tricky

pandas.pydata.org

"[…] high-performance, easy-to-use data structures and data analysis tools for the Python programming language."

Uses Matplotlib for plotting

```
import matplotlib.pyplot as plt
```

pandas.DataFrame.plot

DataFrame.plot(x=None, y=None, kind='line', ax=None, subplots=False, sharex=None, sharey=False, layout=None, figsize=None, use_index=True, title=None, grid=None, legend=True, style=None, logx=False, logy=False, loglog=False, xticks=None, yticks=None, xlim=None, ylim=None, rot=None, fontsize=None, colormap=None, table=False, yerr=None, xerr=None, secondary_y=False, sort_columns=False, **kwds**)**

https://pandas.pydata.org/pandas-docs/version/0.23/generated/pandas.DataFrame.plot.html

Example **Pandas** in **Jupyter** notebook

```
%matplotlib inline
```

```
import matplotlib.pyplot as plt

import matplotlib
matplotlib.style.use('ggplot')

import numpy as np

import pandas as pd
```

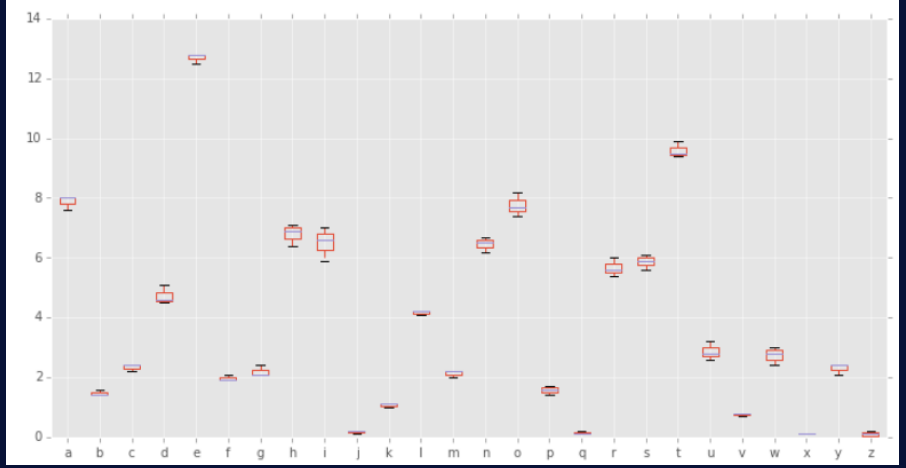
```
letters_analysis
```

	a	b	c	d	e	f	g	h	i	j	...	q	r	s	t	u	v	w	x	y	z
Alice's Adventures in Wonderland	8.0	1.4	2.4	4.5	12.5	1.9	2.4	6.4	7.0	0.2	...	0.2	5.4	5.9	9.9	3.2	0.8	2.4	0.1	2.1	0.1
Peter Pan	7.6	1.6	2.2	4.6	12.8	1.9	2.1	7.1	6.6	0.2	...	0.1	5.6	6.1	9.5	2.8	0.8	2.8	0.1	2.4	0.0
The Wonderful Wizard of Oz	8.0	1.4	2.4	5.1	12.8	2.1	2.1	6.9	5.9	0.1	...	0.1	6.0	5.6	9.4	2.6	0.7	3.0	0.1	2.4	0.2

3 rows × 26 columns

```
letters_analysis.plot(kind='box', figsize=(12, 6))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f17f5555ad0>
```



Best for

Plotting during data analysis

Challenge

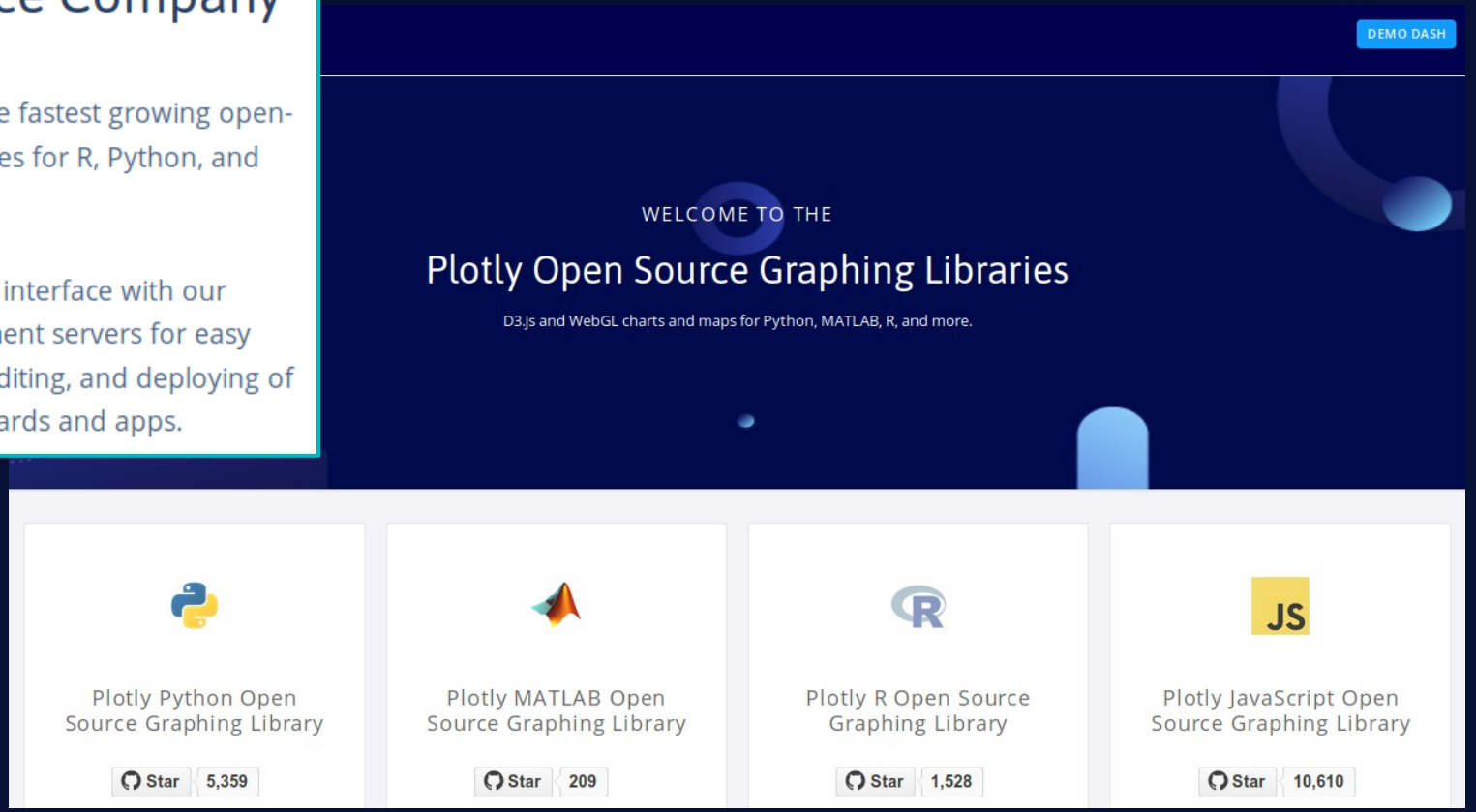
Not the most aesthetic
→ try Seaborn

plot.ly

An Open Source Company

Plotly's team maintains the fastest growing open-source visualization libraries for R, Python, and JavaScript.

These libraries seamlessly interface with our enterprise-ready Deployment servers for easy collaboration, code-free editing, and deploying of production-ready dashboards and apps.



Plotly Python Open Source Graphing Library

Plotly's Python graphing library makes interactive, publication-quality graphs. Examples of how to make line plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, polar charts, and bubble charts.

plot.ly/python

plotly.py

High-level, declarative charting library with over 30 chart types, including scientific charts, 3D graphs, statistical charts, SVG maps, financial charts, and more.

plotly.py

github.com/plotly/plotly.py

```
pip install plotly==4.1.0
```

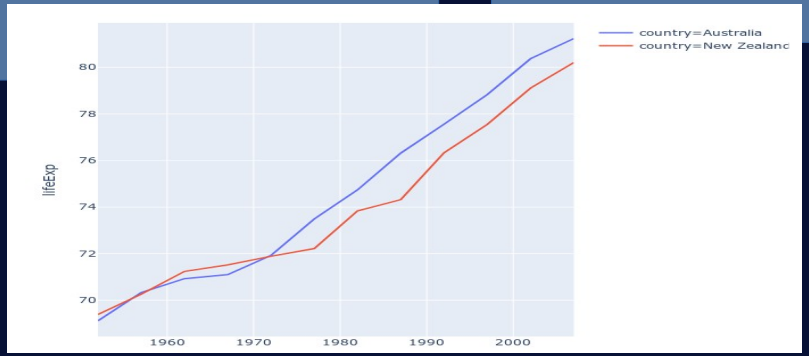
```
import plotly.graph_objects as go

Gapminder =
go.data.gapminder().query("continent=='Oceania'")

fig =
go.Figure(data=go.Scatter(x=Gapminder["year"],
y=Gapminder["lifeExp"], mode='lines',
name='country'))

fig.show()
```

Plotly Express

github.com/plotly/plotly_express

```
pip install plotly_express==0.4.1
```

```
import plotly.express as px

Gapminder =
px.data.gapminder().query("continent=='Oceania'")

fig = px.line(Gapminder, x="year", y="lifeExp",
              color='country')

fig.show()
```



Best for

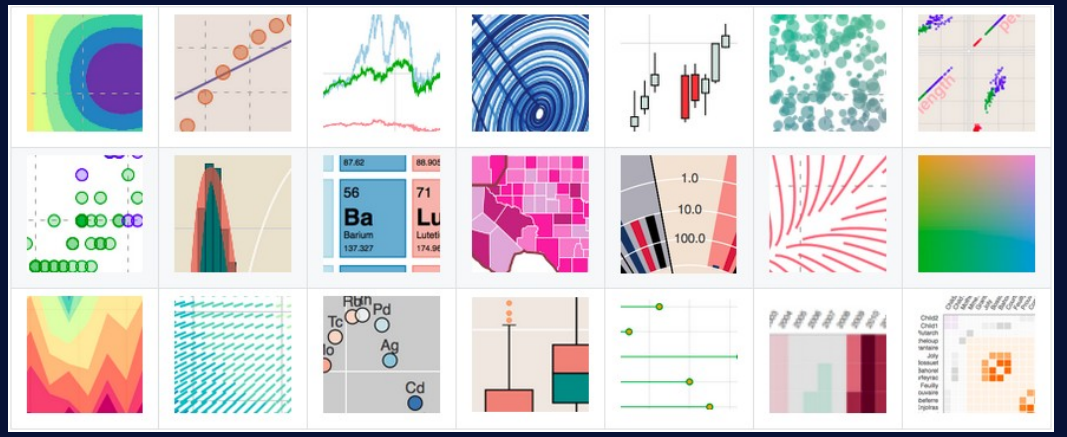
Notebook or plain html report

Challenge

Mainained by a private company

bokeh.org

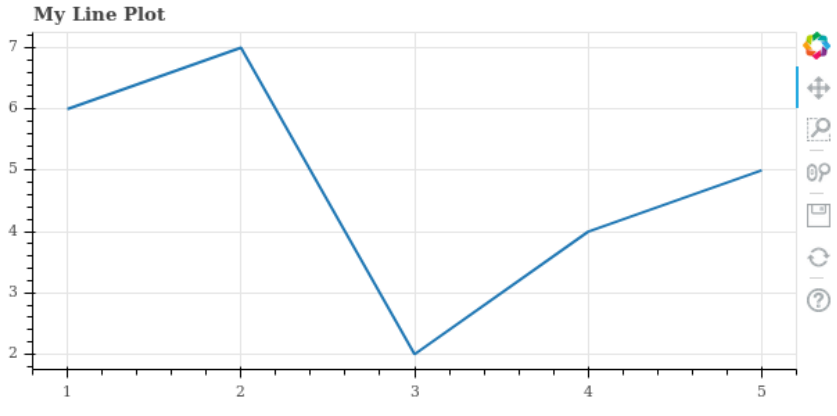
“Bokeh is an **interactive** visualization library for Python that enables beautiful and meaningful visual presentation of data in modern web browsers.”



github.com/bokeh/bokeh

Interactive notebook tutorial

```
In [1]: import numpy as np # we will use this later, so import it now

        from bokeh.io import output_notebook, show
        from bokeh.plotting import figure

In [2]: output_notebook()

        BokehJS 1.2.0 successfully loaded.

In [3]: # create a new plot (with a title) using figure
        p = figure(plot_width=600, plot_height=300, title="My Line Plot")

        # add a line renderer
        p.line([1, 2, 3, 4, 5], [6, 7, 2, 4, 5], line_width=2)

        show(p) # show the results
```

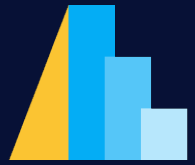


Best for

Interactive plots or dashboards

Challenge

Tricky to create dashboard apps

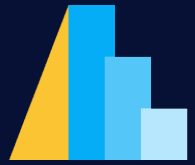


altair-viz.github.io

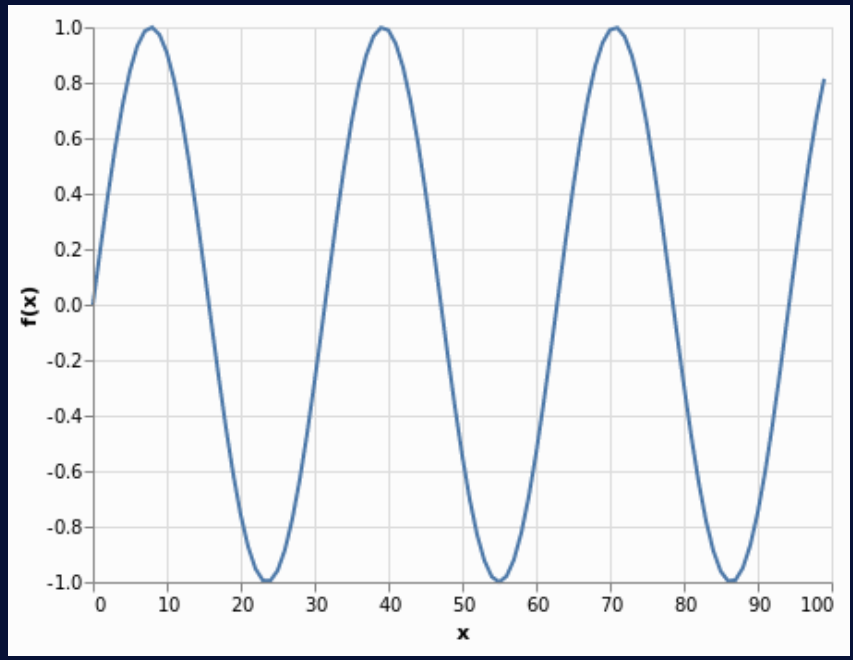
vega.github.io/vega

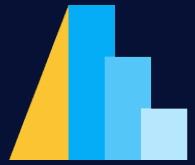
“Altair is a declarative statistical visualization library for Python, based on Vega and Vega-Lite.”

“Vega is a visualization grammar, a declarative language for creating, saving, and sharing **interactive** visualization designs.”



```
import altair as alt
import numpy as np
import pandas as pd


x = np.arange(100)

source = pd.DataFrame({
  'x': x,
  'f(x)': np.sin(x / 5)
})

alt.Chart(source).mark_line().encode(
    x='x',
    y='f(x)'
)
```





Best for

Interactive plots and maps

Challenge

One main maintainer

	Best for	Challenges
matplotlib	Complex or customised plots	Syntax can become tricky
pandas	Plotting during data analysis	Not the most aesthetic → try Seaborn
plotly	Notebook or plain html report	Mainained by a private company
Bokeh	Interactive plots or dashboards	Tricky to create dashboard apps
	Interactive plots and maps	One main maintainer

Thank you

github.com/Eleonore9/tour_dataviz_python

* Exhaustive list of Python tools for data viz: **pyviz.org/tools.html**

* Libraries mentioned:
  - **matplotlib.org**
  - **pandas.pydata.org**
  - **plot.ly**
  - **bokeh.org**
  - **altair-viz.github.io**