

Python Data Visualization Tools

By Jordan Simpson



What is a python data visualization tool?

A python library used to create a visual such as bar graphs, charts, plots, and etc.



matplotlib

- It all started with matplotlib
- Based on Matlab -> easy python transition
- Like the “C” of Python Visual Tools



matplotlib's strengths [+]

- + Lots of Plot Export Formats:

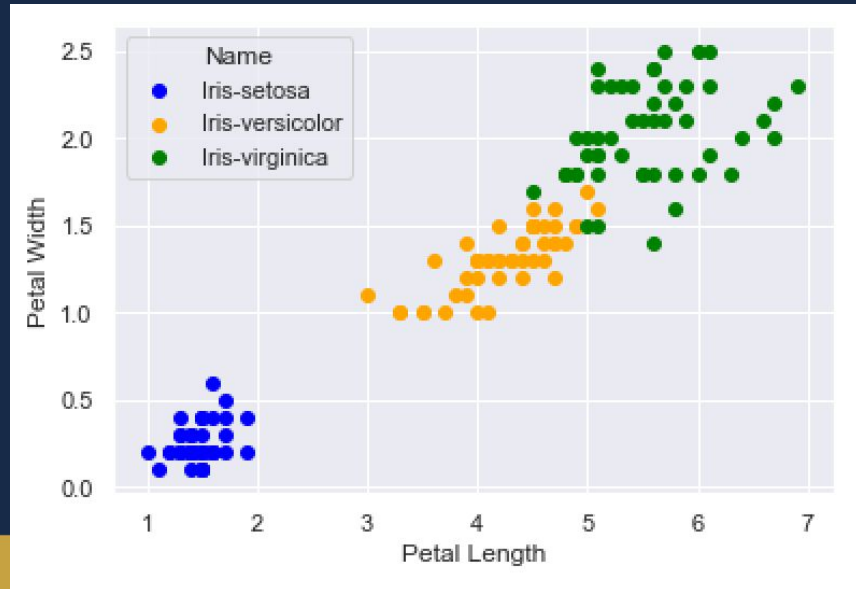
GTK	GTK3Agg	svg
GTKAgg	WebAgg	
- + Reproduce just about any plot

GTKCairo	nbAgg
MacOSX	agg
- + Well Tested Tool
(\approx 18 yrs Git commits)

Qt4Agg	cairo
Qt5Agg	gdk
TkAgg	pdf
WX	pgf
WXAgg	ps
GTK3Cairo	template

matplotlib example

```
1 import matplotlib.pyplot as plt
2 import pandas as pd
3 data = pd.read_csv('/Users/jordan/Desktop/iris.data')
4 color_map = dict(zip(data.Name.unique(), ['blue', 'orange', 'green']))
5 for name, group in data.groupby('Name'):
6     plt.scatter(group['PetalLength'], group['PetalWidth'], color = color_map[name], edgecolor =None, label = name)
7 plt.legend(frameon = True, title = 'Name')
8 plt.xlabel('Petal Length')
9 plt.ylabel('Petal Width')
```



matplotlib's weaknesses [-]

With great power comes great responsibility

- API is imperative
(need to code many aspects of the graph)
- API is overly verbose (lots of syntax)
- Poor support for web & interactive graphs
- Typically slow for large & complicated data
- *Poor default styles (subjective)*

pandas

- API library that works with matplotlib
- Aids visualization with data frame obj.
- Simple API for plots with data frames
 - less code required for many aspects of graph
 - less syntax

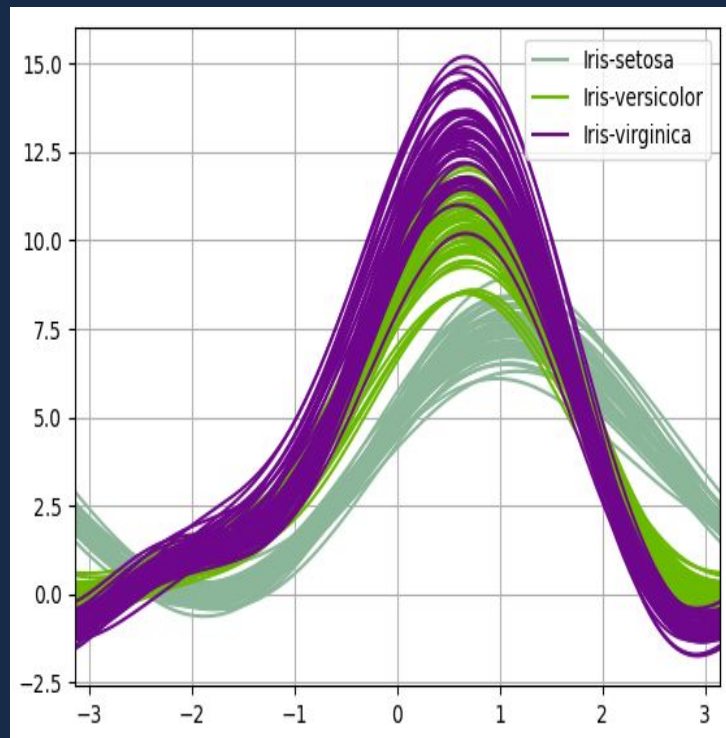


pandas's weaknesses [-] & strengths [+]

- *Poor default styles (subjective)*
- + Improves the "Coder Experience"
 - + with help from data frame object
 - + high level functions = less code & syntax

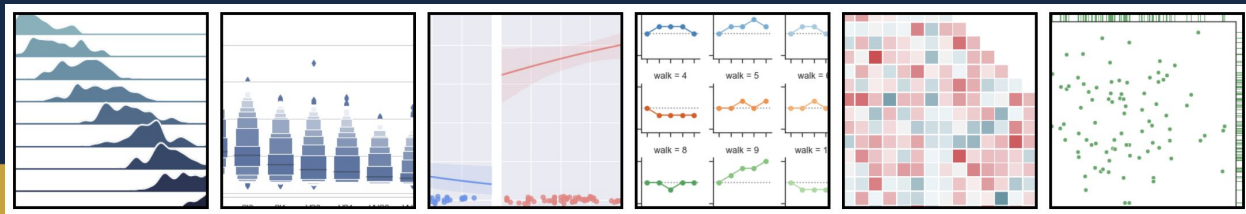
+ Various accessible visualizations

```
1 import matplotlib as plt
2 import pandas as pd
3 from pandas.plotting import andrews_curves
4 data = pd.read_csv('/Users/jordan/Desktop/iris.data')
5 plt.figure.Figure()
6 andrews_curves(data, 'Name')
```



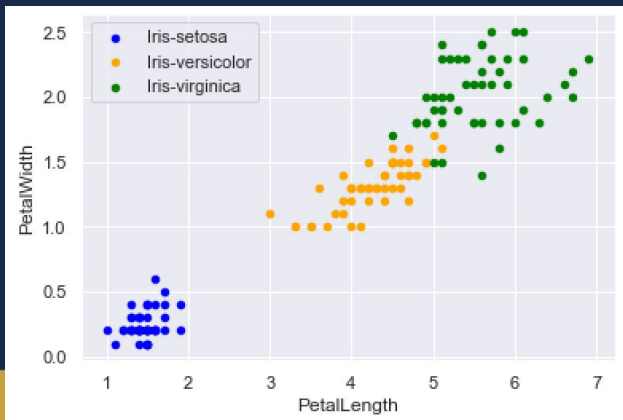
seaborn

- Similar to pandas, uses matplotlib
- Various appealing colors, plots, & styles
- Simple API for plots with data frames
 - less code required for many aspects of graph
 - less syntax
 - less code than pandas
- Excellent for data exploration

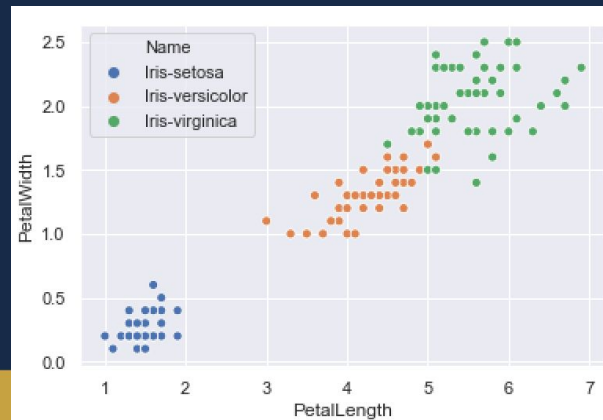


pandas vs seaborn

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 data = pd.read_csv('/Users/jordan/Desktop/iris.data')
4 fig, ax = plt.subplots()
5 colors = {'Iris-setosa':'blue',
6 'Iris-versicolor':'orange', 'Iris-virginica':'green'}
7 grouped = data.groupby('Name')
8 for key, group in grouped:
9     group.plot(ax=ax, kind='scatter', x='PetalLength',
10 y='PetalWidth', label=key, color=colors[key])
```



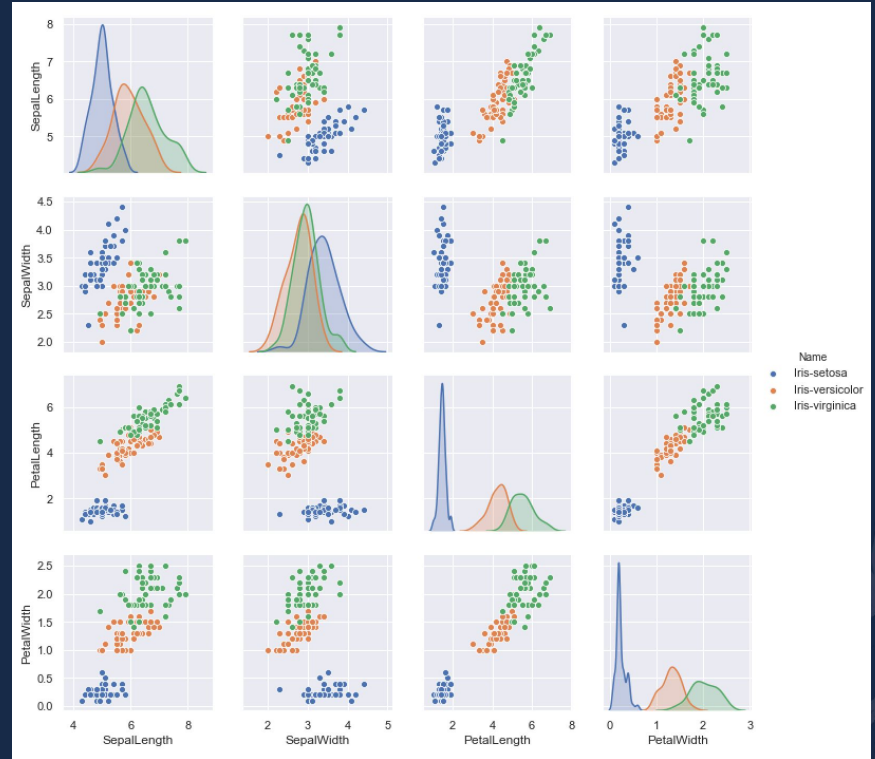
```
1 import pandas as pd
2 import seaborn as sns
3 data = pd.read_csv('/Users/jordan/Desktop/iris.data')
4 sns.set() #applies seaborn style to matplotlib graphs
5 sns.scatterplot(x = 'PetalLength', y = 'PetalWidth', 6
hue = 'Name', data = data) #produces scatterplot
```



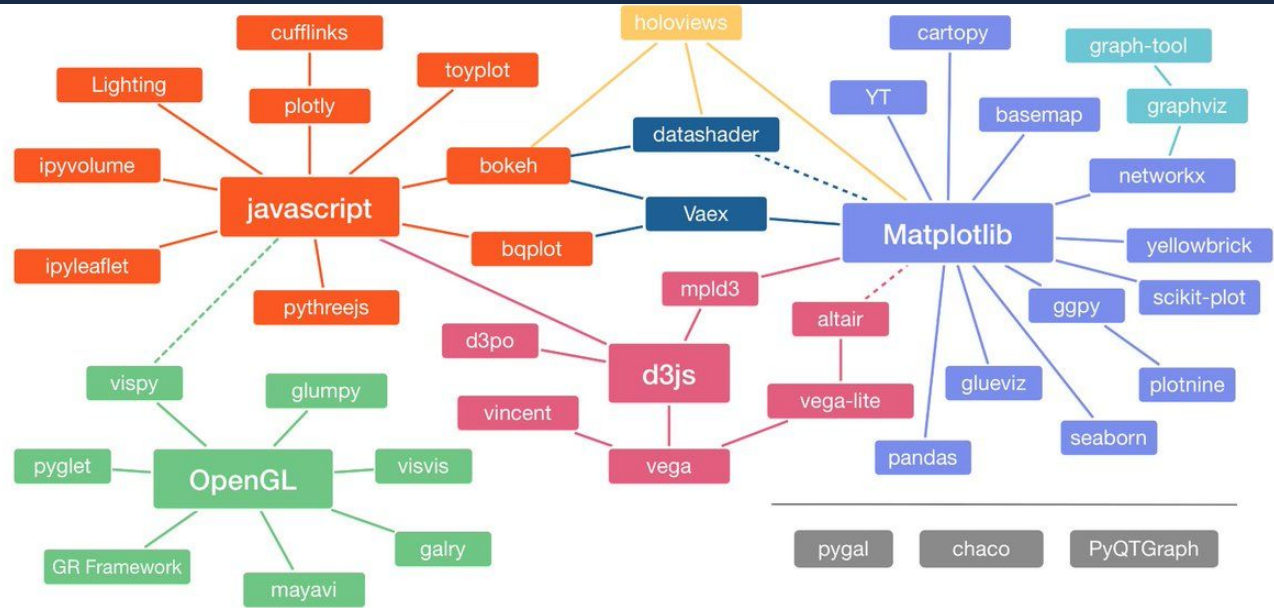
seaborn is great for data exploration

Visualize a data set's
combination of rows and
columns in 5 lines of code

```
1 import pandas as pd
2 import seaborn as sns
3 data = pd.read_csv('/Users/jordan/Desktop/iris.data')
4 sns.set() #applies seaborn style to matplotlib graphs
5 sns.pairplot(data = data, hue = 'Name') #produces pairplot
```



More Tools in the Toolbox



Tools to tryout (Recommendations)

- ✂ bokeh/plotly - most developed tools for web & 3D graphs
- ✂ datashader/Vaex - big data real time aggregator (pixel heat maps)
- ✂ altair - declarative grammar API (streamlined "sentence" describing code)

Common Cluster Characteristics

- ✂ Keep matplotlib's strengths, export formats, & improve "coder experience"
- ✂ Web interactivity via plot serialization (often JSON in Jupyter Notebooks) (netlogo)
- ✂ Large data visuals
- ✂ Declarative grammar API
- ✂ Large 3D data sets

A picture is worth thousand words

know how to paint that picture in python

- produce a quick & dirty visual
- Discover new data connections visually

CS

- Excellent skill for data science
- Use an easy API & language to make visuals
- Make Dmitry proud and do some data science for fun!

Math

- Adventure beyond Excel & LaTeX (Learn a easy programming language)
- Streamline Excel data entry (comma separated file required!)
- Streamline graph production



Source
Code
[.py]
slides
[PDF]
(GitHub)



Email
Contact



In-depth
video on
Python
Visual
Tools
(YouTube)