



ÉCOLE
D'INGÉNIEURS
PARIS-LA DÉFENSE

Python For Data Analysis

Imen Ouled Dlala

imen.ouled_dlala@devinci.fr
Bureau: L405





General Plan

- The basics of python for data science
- Data Science modules
 - Pandas, Numpy, Scipy
- Data analysis and visualization
 - Seaborn, Matplotlib, Bokeh
- Webscrapping
- Machine learning and Datasets
 - Sklearn, tensorflow
- API Django / Flask



Data analysis and visualization : Plan

1. Matplotlib
2. Seaborn

Matplotlib

Matplotlib function Vs OOP

Matplotlib is a plotting library for creating static, animated, and interactive visualizations in Python.

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

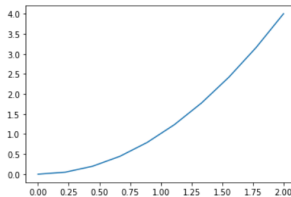
Function

```
plt.plot(x,y)  
plt.show()
```

OOP

```
fig, ax = plt.subplots()  
ax.plot(x,y)  
plt.show()
```

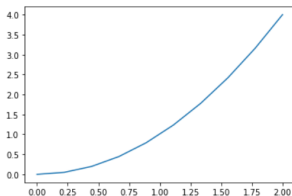
Same result



```
import numpy as np
x = np.linspace(0,2,10)
y=x**2
print(x)
```

```
plt.plot(x,y)
plt.show()
```

```
[0.          0.22222222 0.44444444 0.66666667 0.88888889 1.11111111
 1.33333333 1.55555556 1.77777778 2.]
```



x,y must have same dimensions !

Matplotlib : styles

`plot(x,y, label=...,lw=...,ls=...,c=...)`

`label` → name of the curve

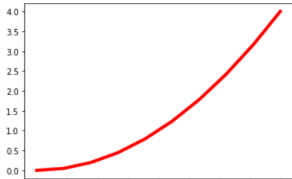
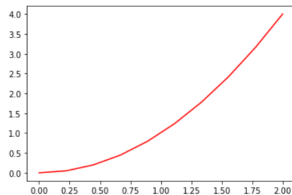
`lw` → line thickness

`ls` → line style

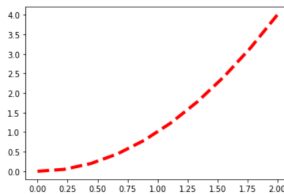
`c` → line color

```
plt.plot(x,y, c='red')
```

```
plt.plot(x,y, c='red', lw='4')
```



```
plt.plot(x,y, c='red', lw='4', ls='- -')
```



The Lifecycle of a Plot

```
plt.figure()
```

```
plt.plot(...,...)
```

```
plt.plot(...,...)
```

```
plt.title('text')
```

```
plt.xlabel('text')
```

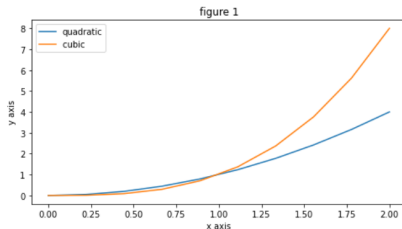
```
plt.ylabel('text')
```

```
plt.legend()
```

```
plt.show()
```

```
plt.savefig('figure1.png')
```

```
plt.figure(figsize=(8,4))  
plt.plot(x,y, label='quadratic ' )  
plt.plot(x,x**3, label='cubic')  
plt.title('figure 1')  
plt.xlabel('x axis ' )  
plt.ylabel('y axis')  
plt.legend()  
plt.show()  
plt.savefig('figure1.png')
```

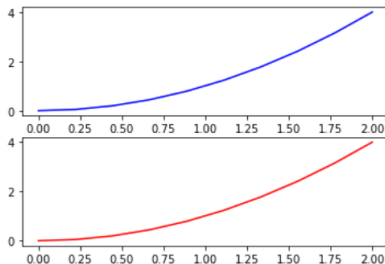


Matplotlib : subplot

The `subplot()` function allows you to organize different plots within a display grid. You must specify the number of lines, the number of columns and the number of the plot.

```
plt.subplot(2,1,2)  
plt.plot(x,y,c='red')  
plt.subplot(2,1,1)  
plt.plot(x,y,c='blue')
```

[<matplotlib.lines.Line2D at 0x2816ed08a30>]



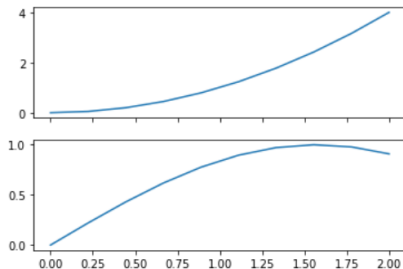
Matplotlib : Object Oriented

```
fig, ax = plt.subplots(n)
ax[0].plot(x,y)
ax[1].plot(x,y)
plt.show()
```

`fig` is an object.

`ax` is a ndarray table that contains objects.

```
fig, ax = plt.subplots(2,1, sharex=True)
ax[0].plot(x,y)
ax[1].plot(x,np.sin(x))
plt.show()
```



Matplotlib : Example

```
import numpy as np
from sklearn.datasets import load_iris

iris = load_iris()
x = iris.data
y = iris.target

names=list(iris.target_names)

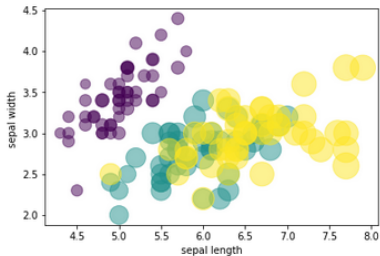
print(f'x contains {x.shape[0]} examples and {x.shape[1]} variables.' )
print(f'There is {np.unique(y).size} classes.' )
```

Output :

x contains 150 examples and 4 variables.
There is 3 classes.

Matplotlib : example

```
plt.scatter(x[ :,0],x[ :,1], c=y, alpha=0.5, s=x[ :,2]*100)  
plt.xlabel("sepal length ")  
plt.ylabel("sepal width ")
```



Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

For further details : <https://seaborn.pydata.org/>



End

Good Lecture !

This course is inspired from Romain Jouin that I thank for sharing