

# Matplotlib

Hadrien Bodin

Mines

14 octobre 2025

# Table des matières

Matplotlib est la bibliothèque utilisée pour la représentation de données scientifique.

D'autres librairies existent en python pour cette usage, qui sont souvent dérivées de matplotlib. On peut notamment citer Seaborn et Plotly qui sont très pratiques.

Matplotlib a pour avantage d'être ultra personnalisable. A priori, tous les graphes que vous pouvez imaginer sont faisables avec matplotlib. Son principal inconvénient est la lenteur.

Les fonctions matplotlib sont, pour beaucoup, utilisables directement depuis pandas.

# Import

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

La fonction plot sert à tracer des courbes

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

Où x et y sont des listes (ou simili) de valeurs. Le résultat est alors quelconque, mais on peut l'améliorer avec énormément d'options :

- marker
- linewidth
- color
- linestyle
- etc...

La fonction `scatter` sert à tracer des nuages de points

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

Où `x` et `y` sont des listes (ou simili) de valeurs. Le résultat est alors quelconque, mais on peut l'améliorer avec énormément d'options :

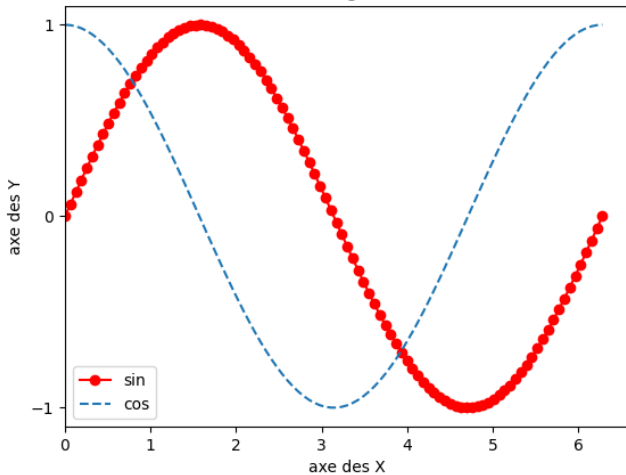
- `marker`
- `linewidth`
- `color`
- `linestyle`
- etc...



Un bloc matplotlib de base ressemble souvent à ceci :

```
x = np.linspace(0,2*np.pi,100)
y1 = np.sin(x) #coords de la premi re courbe
y2 = np.cos(x) #coords de la seconde courbe
plt.plot(x, y1, label='sin ', marker='o', color='red ')
plt.plot(x, y2, label='cos ', linestyle='dashed ')
plt.xlabel('axe des X')
plt.ylabel('axe des Y')
plt.legend()
plt.title('Trigo ')
plt.xlim(0, 2.1*np.pi)
plt.yticks([-1, 0 ,1])
plt.show()
```

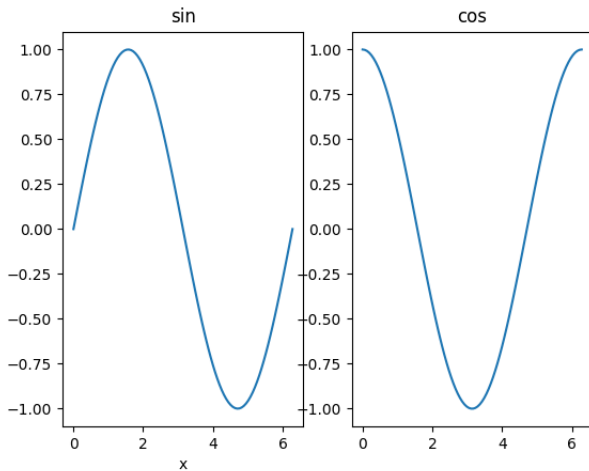
## Trigo



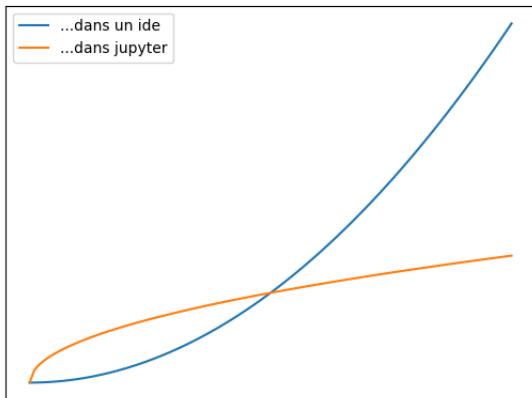
Pour combiner des figures, il faut jouer avec :

- `fig` ou `figure` qui désignent une figure globale
- `axis` qui désigne une sous figure

```
fig , (ax1 , ax2) = plt.subplots(1,2)
ax1.plot(x, y1)
ax1.set_title('sin ')
ax2.plot(x,y2)
ax2.set_title('cos ')
ax1.set_xlabel('x')
plt.show()
```



## Faire du matplotlib...



```
ax1 = plt.subplot(*args: 1, 2, 1)
```

```
ax1.
```

plot(self, args, scalex, scaley, data, kwargs)	Axes
imshow(self, X, cmap, norm, aspect, interpolation, alpha, v...	Axes
bar(self, x, height, width, bottom, align, data, kwargs)	Axes
fill(self, args, data, kwargs)	Axes
set_title(self, label, fontdict, loc, pad, y, kwargs)	Axes
title	_AxesBase
step(self, x, y, args, where, data, kwargs)	Axes
axes	Artist
acorr(self, x, data, kwargs)	Axes
add_artist(self, a)	_AxesBase
add_callback(self, func)	Artist
add_child_axes(self, ax)	AxesBase

Ctrl+Bas and Ctrl+Haut will move caret down and up in the editor [Next Tip](#)

# Exo : refaire ca

