

Programmation Python

Tracer les courbes

1) Le module Matplotlib

Le module **Matplotlib** est chargé de tracer les courbes:



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

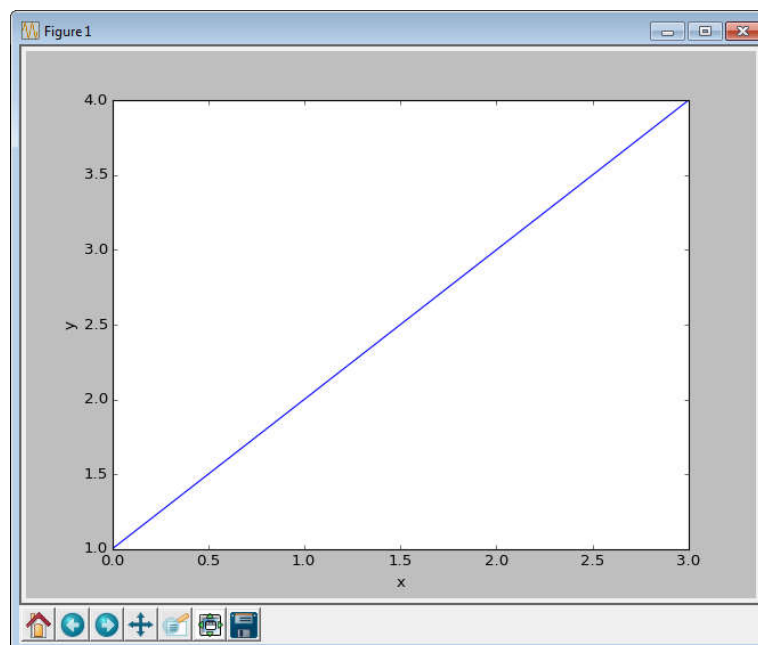
D'une manière générale les fonctions **plt.plot** attendent des vecteur/matrice, bref des tableaux de points du plan. Selon les options, ces points du plan sont reliés entre eux de façon ordonnée par des segments : le résultat est une courbe.

1- Créer son premier graphique : tracer une droite entre deux points (1,2) et (3,4)



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

```
plt.plot([1,2,3,4])  
plt.xlabel('x')  
plt.ylabel('y')  
plt.show()
```

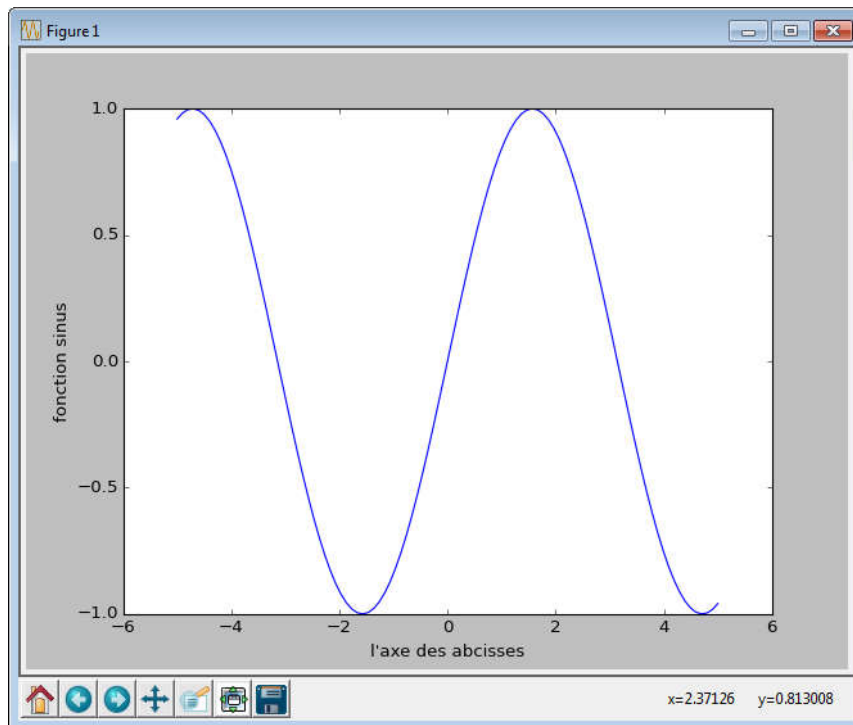


Si tout se passe bien, une fenêtre doit s'ouvrir avec la figure ci-dessus. Il est possible de jouer avec **les menus dans le bas de cette fenêtre** : zoomer, déplacer la figure, etc et surtout sauvegarder dans un format PNG, PDF, EPS, etc.

2- Tracer la fonction sinus sur l'intervalle $[-5, 5]$.



```
import matplotlib.pyplot as plt
import numpy as np
x=np.linspace(-5,5,100)
plt.plot(x,np.sin(x))
plt.ylabel('fonction sinus')
plt.xlabel("l'axe des abscisses")
plt.show()
```



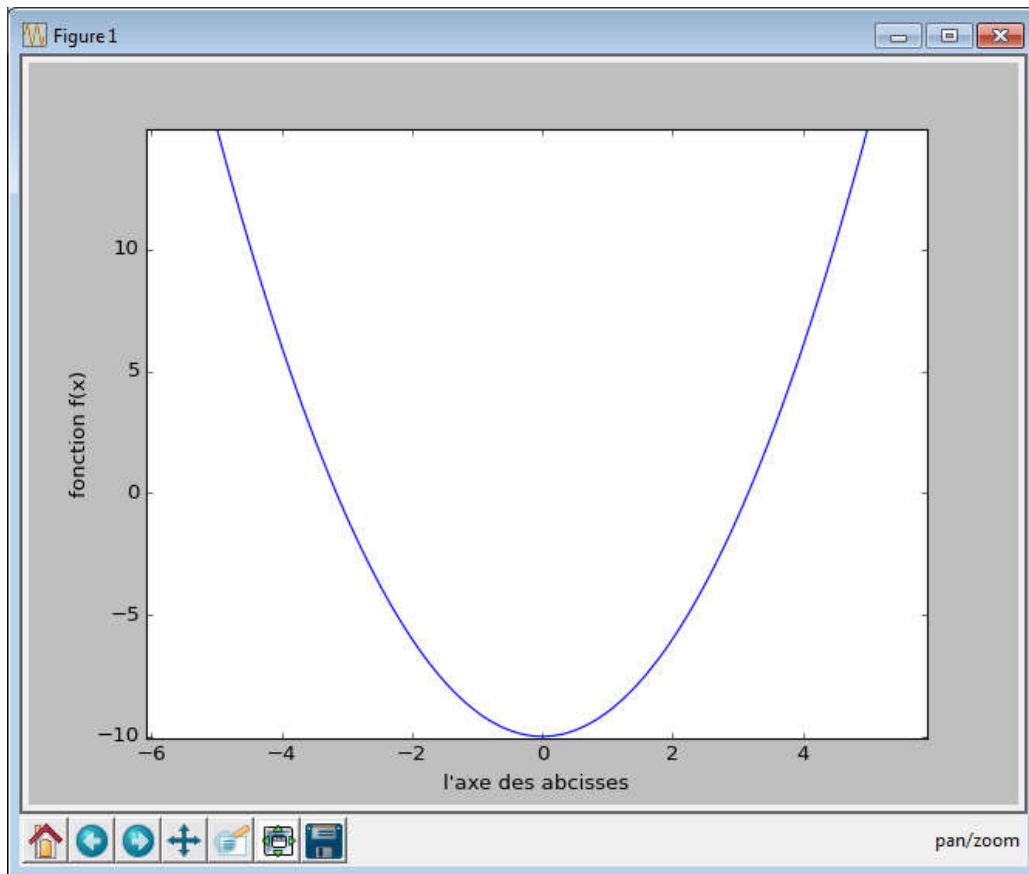
3- Tracer la fonction $f(x) = x^2 - 10$ sur $[-5, 5]$.



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

def f(x):
    return x*x-10

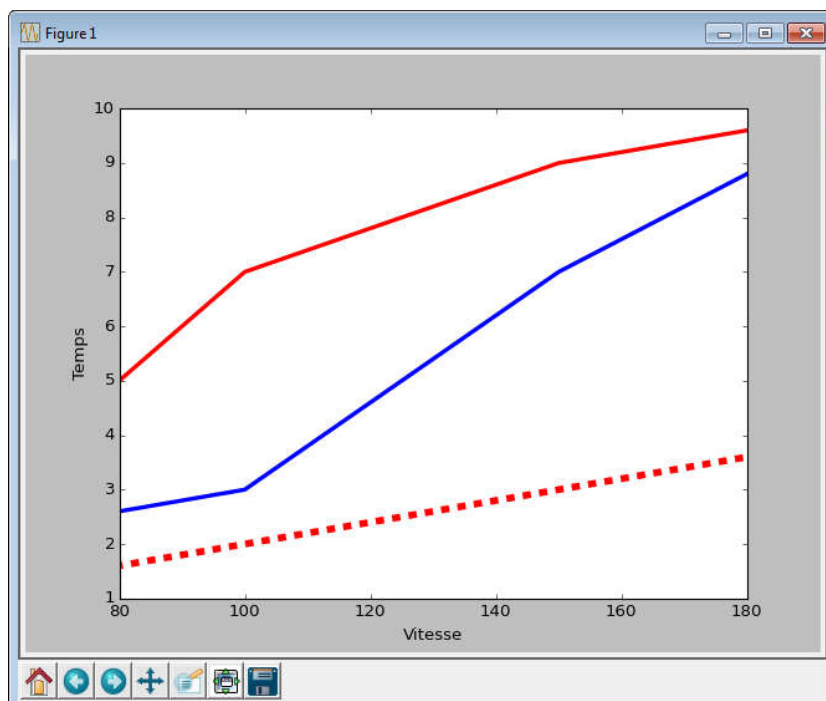
x=np.linspace(-5,5,100)
plt.plot(x,f(x))
plt.ylabel('fonction f(x)')
plt.xlabel("l'axe des abscisses")
plt.show()
```



4- Plusieurs courbes



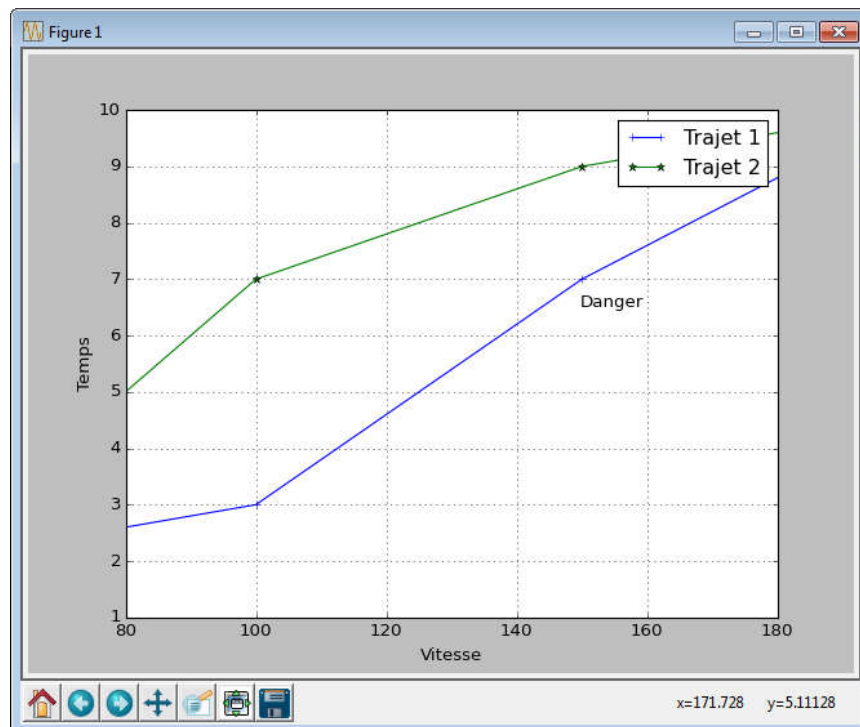
```
import matplotlib.pyplot as plt
plt.plot([50,100,150,200], [1,2,3,4], "r--", linewidth=5)
plt.plot([50,100,150,200], [2,3,7,10], "b", linewidth=3)
plt.plot([50,100,150,200], [2,7,9,10], "r", linewidth=3)
plt.xlabel('Vitesse')
plt.ylabel('Temps')
plt.axis([80, 180, 1, 10])
plt.show()
```



5- Ajouter une légende pour les courbes



```
import matplotlib.pyplot as plt
plt.grid(True)
plt.plot([50,100,150,200], [2,3,7,10], "b",
         linewidth=1, marker="+", label="Trajet 1")
plt.plot([50,100,150,200], [2,7,9,10], "g",
         linewidth=1, marker="*", label="Trajet 2")
plt.axis([80, 180, 1, 10])
plt.text(150, 6.5, 'Danger')
plt.xlabel('Vitesse')
plt.ylabel('Temps')
plt.legend()
plt.show()
```



6- Les histogrammes

Pour créer un histogramme on utilise la méthode `hist`. On peut lui donner des données brutes et il s'occupera de faire les calculs nécessaires à la présentation du graphique.

On a pris ici l'exemple de 1000 tirage au sort (random) de valeur entre 0 et 150 et voici le résultat:



```
import matplotlib.pyplot as plt
import random
# 1000 tirages entre 0 et 150
x = [random.randint(0,150) for i in range(1000)]
plt.hist(x,50,normed=1,facecolor='b',alpha=0.5)

plt.xlabel('Mise')
plt.ylabel(u'Probabilité')
plt.axis([0, 150, 0, 0.015])
plt.grid(True)
plt.show()
```

