

OBJECTIF GENERAL L'objectif de ce TP est d'utiliser les bibliothèques : sympy, numpy et matplotlib pour calculer la limite d'une fonction en un point donné. I. INTRODUCTION Dans le but de pouvoir utiliser les différentes bibliothèques pour l'analyse et la visualisation des données, Par ces séances de travaux pratiques nous sommes appelés à utiliser certaines bibliothèques de python pour le calcul de la limite d'une fonction II. APPLICATION 1- Calcul de la limite de fonction $f(x) = x \sin(x) + 5x^2 + 1$ en 5.5 a- Utilisons sympy pour la détermination de la limite

importons la bibliothèque sympy pour le traçage de la courbe et la détermination de la limite.

```
In [139... import sympy as sp
```

Définissons la fonction dans sympy

```
In [140... x = sp.Symbol('x')
```

```
In [141... f = sp.Function('f')
```

```
In [142... f = x*sp.sin(x) + 5*x**2 + 1
```

Affichage de la fonction

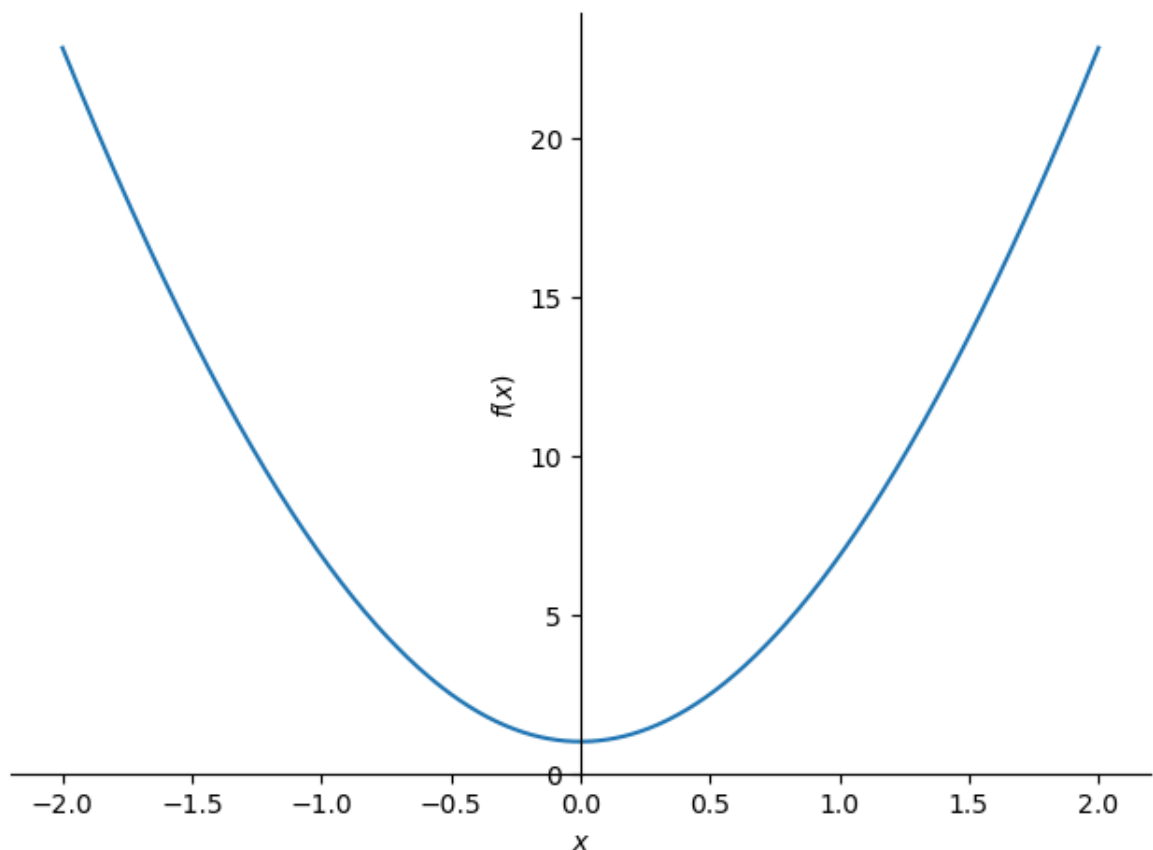
```
In [143... display(f)
```

$$5x^2 + x \sin\{\left(x \right)\} + 1$$

Traçons la courbe sur $[-2,2]$

```
In [144... display(r'la courbe de f pour x = [-2,2]')
sp.plot(f, (x, -2, 2))
```

'la courbe de f pour x = $[-2,2]$ '



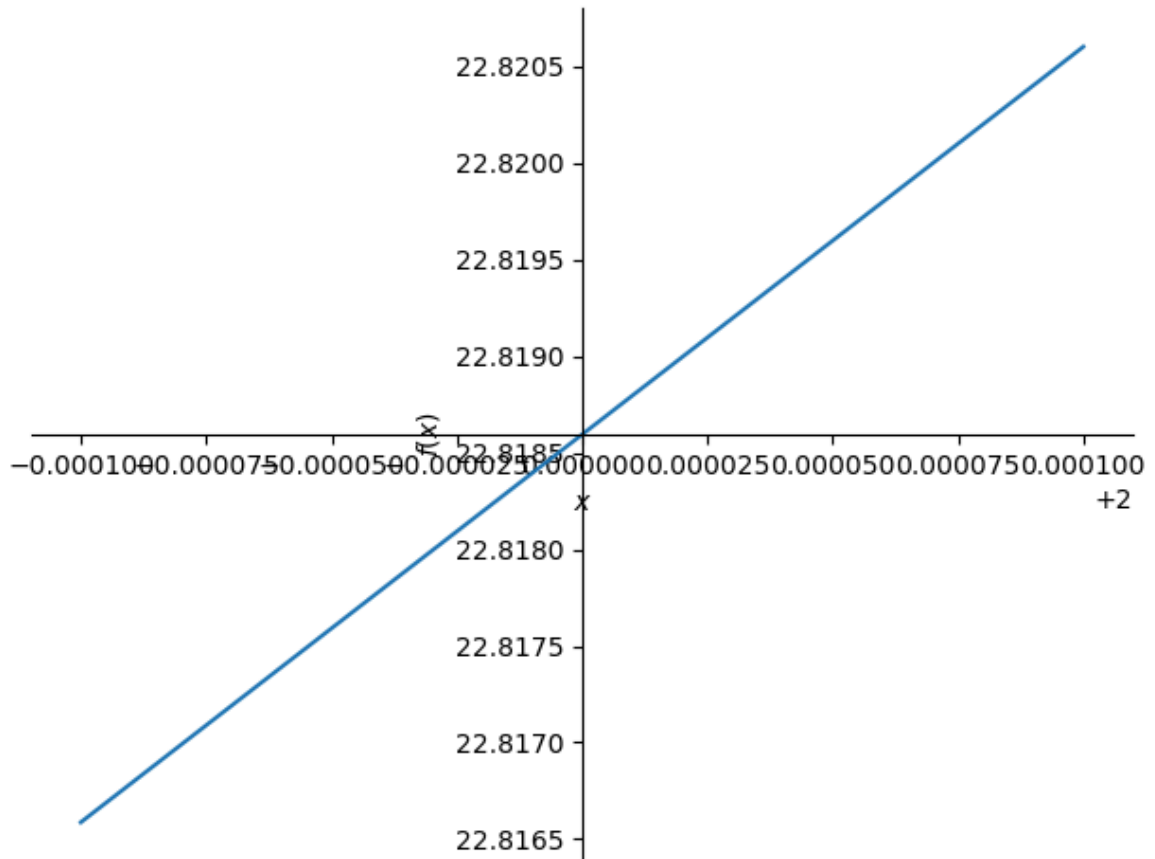
Out[144...] <sympy.plotting.backends.matplotlibbackend.matplotlib.MatplotlibBackend at 0x20e57633020>

Traçons la courbe de f sur $[1.9999, 2.0001]$

```
In [145...] display(r'la courbe de f pour x = [1.9999,2.0001]')
```

'la courbe de f pour $x = [1.9999, 2.0001]$ '

```
In [146...] sp.plot(f, (x, 1.9999, 2.0001))
```



Out[146...] <sympy.plotting.backends.matplotlibbackend.matplotlib.MatplotlibBackend at 0x20e557465a0>

Détermination d'un encadrement proche de la limite à gauche et à droite

```
In [147...] Vgauche = [1.9999, 1.99999, 1.999999]
Vdroite = [2.000001, 2.00001, 2.0001]
print('Limite à gauche')
Ugauche = [f.subs(x,i) for i in Vgauche]
print(Vgauche)
print(Ugauche)
```

Limite à gauche

[1.9999, 1.99999, 1.999999]

[22.8165871900219, 22.8183940839813, 22.8185747766513]

```
In [148...] print('limite à droite')
Udroite = [f.subs(x,i) for i in Vdroite]
print(Vdroite)
print(Udroite)
```

limite à droite

[2.000001, 2.00001, 2.0001]

[22.8186149306588, 22.8187956240564, 22.8206025907720]

Utilisons la fonction "limit" de sympy pour déterminer la limite exacte de la fonction

```
In [149... print(sp.limit(f,x,2,'-'))
```

2*sin(2) + 21

```
In [150... print(sp.limit(f,x,2,'+'))
```

2*sin(2) + 21

b- Utilisons les bibliothèques numpy et matplotlib pour déterminer la limite

importons les bibliothèques sympy et matplotlib

```
In [130... import numpy as np
import matplotlib.pyplot as plt
```

Définissons la fonction a l'aide de numpy et l'ensemble des points par lesquels passera la courbe avec les tableaux de numpy

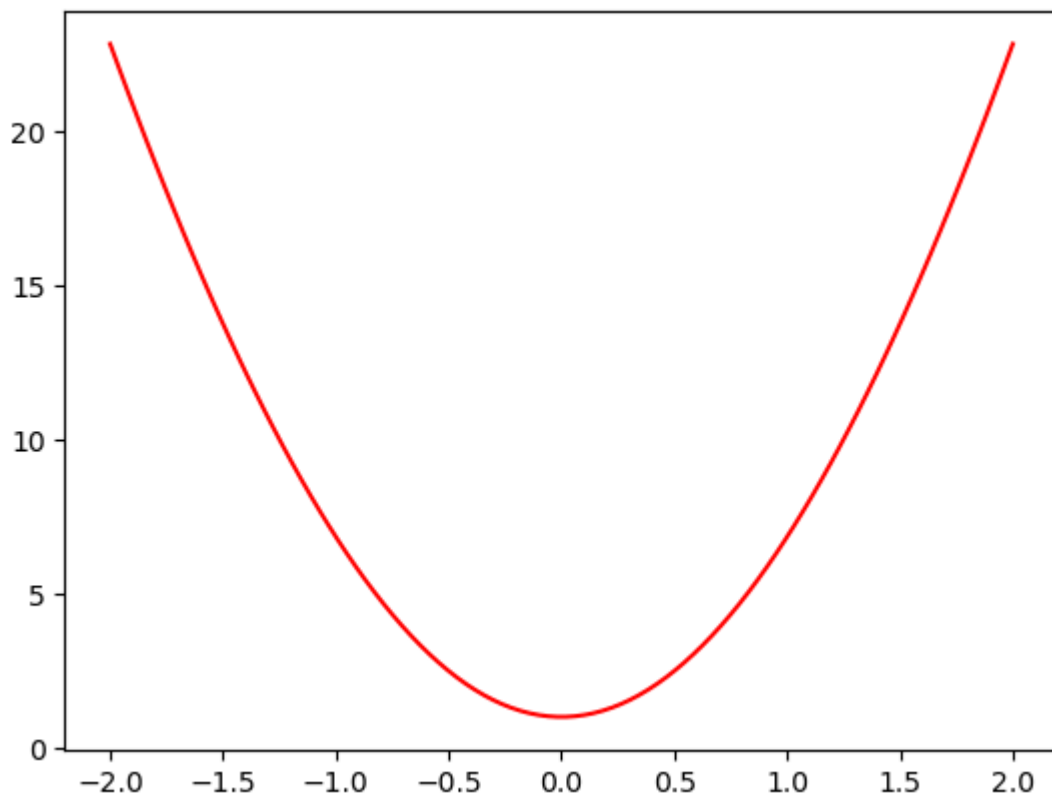
```
In [ ]: x = np.linspace(-2,2,100)
```

```
In [152... def f(x):
    return x*np.sin(x) + 5*x**2 + 1
```

Traçons la courbe de f sur [-2,2]

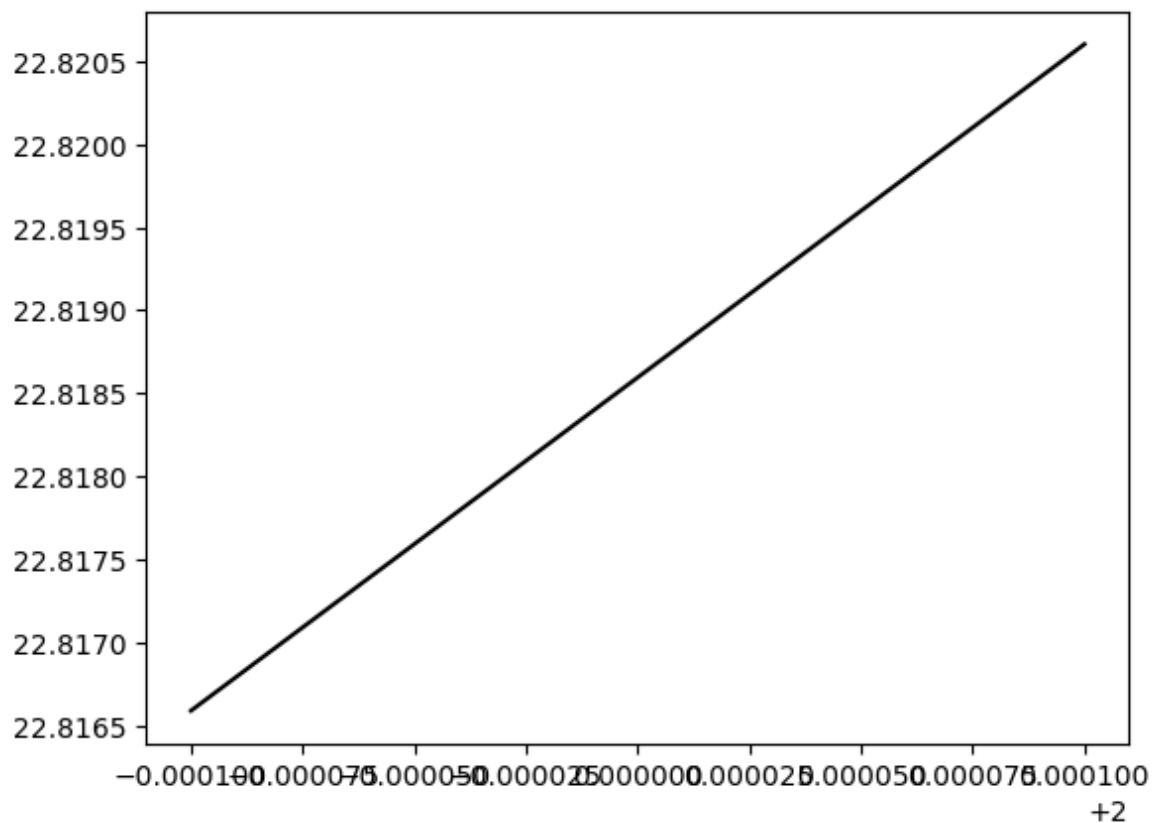
```
In [153... print(r'la courbe de f sur [-2,2]')
plt.plot(x,f(x),"r")
plt.show()
```

la courbe de f sur [-2,2]



Réduisons l'intervalle sur laquelle nous traçons la courbe à [1.9999,2.0001]

```
In [134... x = np.linspace(1.9999,2.0001,10)
plt.plot(x,f(x),"k")
plt.show()
```



Détermination d'un encadrement proche de la limite à gauche et à droite

```
In [135... Vgauche = np.array([1.9999,1.99999,1.999999])
print("limite à gauche")
Ugauche = f(Vgauche)
print(Vgauche)
print(Ugauche)
```

```
limite à gauche
[1.9999  1.99999  1.999999]
[22.81658719 22.81839408 22.81857478]
```

```
In [136... Vdroite = np.array([2.000001,2.00001,2.0001])
print("limite à droite")
Udroite = f(Vdroite)
print(Vdroite)
print(Udroite)
```

```
limite à droite
[2.000001 2.00001  2.0001 ]
[22.81861493 22.81879562 22.82060259]
```