

## TP 12 : Numpy et Matplotlib.

### I Utilisation de Numpy.

On résoudra les exercices suivants en utilisant la bibliothèque Numpy de Python.

**Exercice 1 :** Si  $A = \begin{pmatrix} -1 & -1 & -3 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$  et  $B = \begin{pmatrix} -2 & -1 & -6 \\ 1 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}$ . Calculer  $A \times B$ .

$A$  est-elle inversible ? Si c'est le cas, calculer  $A^{-1}$  (on arrondira les éléments de  $A$  à  $10^{-3}$ ).

**Exercice 2 :** si  $A = \begin{pmatrix} 0 & 0 & 0 \\ 6 & 18 & -18 \\ 6 & 18 & -18 \end{pmatrix}$ , calculer  $A^2, A^3, A^4$ . Combien vaut  $A^n$  si  $n \geq 3$  ?

**Exercice 3 :** résoudre le système 
$$\begin{cases} 2x + 2y - 3z = 2 \\ -2x - y - 3z = -5 \\ 6x + 4y + 4z = 16 \end{cases}$$

### Exercice 4 :

1°) Résoudre « à la main » le système suivant : 
$$\begin{cases} x + (10^{15} + 1)y + z = 1 \\ x + (1 + 10^{-15})y + 2z = 0 \\ 10^{15}y + z = 0 \end{cases}$$

2°) Résoudre ce système en utilisant :

- la fonction  $resolution(A, Y)$  déjà écrite dans le TP6
- Numpy

3°) Que remarque-t-on ? Pourquoi ?

### Exercice 5.

Soit le polynôme  $P(x) = x^8 + x^6 + x^4 - 3x^2 - 1$ .

1°) Grâce à Numpy, exprimer  $P^2(x)$  et  $P^3(x)$ .

2°) Exprimer la dérivée de  $P^3(x)$ .

3°) Exprimer la primitive de  $P$  valant 2 en 0.

## II Tracer des courbes avec 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.

Commençons par la fonction sinus :

```
• import matplotlib.pyplot as plt
• import numpy as np
• x=np.linspace(-5,5,100)
• plt.plot(x,np.sin(x)) #On utilise la fonction sinus de Numpy
• plt.title('Graphe de la fonction sinus')
• plt.ylabel('fonction sinus')
• plt.xlabel('axe des abscisses')
• plt.axhline(color='black')#On trace l'axe des abscisses
• plt.axvline(color='black')#On trace l'axe des ordonnées
• plt.show() #Pour afficher le graphe
```

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... et surtout sauvegarder dans un format PNG, PDF, EPS, etc.

Pour connaître toutes les options, le mieux est de se référer à la documentation de Matplotlib. Voyons ici quelques unes d'entre elles :

- **Bornes :** bornes : spécifier un rectangle de représentation, ce qui permet un zoom, d'éviter les grandes valeurs des fonctions par exemple, se fait via la commande `plt.axis([xmin,xmax,ymin,ymax])`.
- **couleur du trait :** pour changer la couleur du tracé une lettre g vert r rouge ,k noir, b bleu, c cyan, m magenta, y jaune , w blanc .  
`plt.plot(np.sin(x), 'r')` tracera notre courbe sinus en rouge.
- **style du trait :** pointillés, absences de trait, etc se décident avec `linestyle`. Au choix '-' ligne continue, '--' tirets, '-.' points-tirets, ':' pointillés, sachant que 'None', '', ' donnent "rien-du-tout"  
**Exemple :** `plt.plot(x,np.sin(x),linestyle="-", marker="o",color='g')`

### Exercice 6 :

Pour superposer plusieurs graphes de fonctions, il est possible de faire une succession de commandes `plt.plot` :

```

import matplotlib.pyplot as plt
import numpy as np
x=np.linspace(-5,5,100)
p1=plt.plot(x,np.sin(x),marker='o')
p2=plt.plot(x,np.cos(x),marker='v')
plt.title("Fonctions trigonometriques")
plt.legend([p1, p2], ["Sinus", "Cosinus"])
plt.show()

```

Rédiger ce programme sous Python et observer.

### **Exercice 7 :**

Résoudre graphiquement de façon approximative l'équation  $e^{-x^2} = \sin(x)$  sur l'intervalle  $[0,3]$ . Il est possible de zoomer sur les figures et la position du curseur de la souris s'affiche en bas à gauche de la figure.

### **Exercice 8 :**

Soit l'équation différentielle :  $\begin{cases} y(1) = 1 \\ xy'(x) - 2y(x) = 0 \end{cases}$  sur  $[1;2]$

1°) Créer un tableau de 100 points équirépartis entre 1 et 2.

2°) Soit  $f$  la solution de cette équation différentielle. Dresser un tableau des images par  $f$  des valeurs du tableau précédent (on utilisera la bibliothèque *scipy.integrate*).

3°) Tracer la courbe représentative de  $f$  grâce à Matplotlib.

### **Exercice 9 :**

1°) Tracer la courbe de la primitive de  $f(x) = \frac{1}{1+x^2}$  valant 0 en 0 en utilisant Matplotlib et Numpy.

2°) Connait-on cette fonction ?