

Travaux pratiques 1 - Polynômes

Dans tout le TD on veillera à écrire un code suffisamment général pour être utilisé d'un exercice à l'autre. Ci-dessous, deux codes d'exemple qui rappellent quelques fonctions python élémentaires nécessaires à la création de tableaux et à l'affichage de courbes et de champs de vecteurs.

```
1 import numpy as np #importe les outils de gestion des tableaux
2 import matplotlib.pyplot as plt #importe les outils d'affichages matplotlib
3 plt.rcParams['figure.dpi'] = 120 #reglage de resolution des figures
4 plt.rcParams['savefig.dpi'] = 240 #reglage de resolution des figures enregistrees
5
6 N=10#declare une variable N initialisee sur la valeur 10
7 X=np.arange(N) #cree le tableau [0,1,2,3,4,5,...,N-1]. X[i]=i pour i=0,1,...,N-1
8 Y=np.random.random(N) #cree un tableau de N nombre aleatoires entre 0 et 1,
   numerotes de 0 a N-1
9 Z=np.linspace(1,3,N) #cree un tableau de N nombres qui subdivisent l'intervalle
   [1,3] avec un ecart regulier
10
11 plt.plot(X,Y) #Trace le graphe de Y en fonction de X, en rejoignant les points (X
   [i],Y[i]) pour i=0,...,N-1. Par default le graphe est represente en tracant les
   segments [(X[i],Y[i]), (X[i+1],Y[i+1])] pour i=0,1,...,N-2, cela peut etre
   change en explorant la documentation
12 plt.plot(X,Z,label='graphe de Z') #Trace le graphe de Z en fonction de X, avec en
   plus une legende associee a ce graphe.
13 plt.title("Affichage de Y,Z en fonction de X") #Donne un titre a la figure
14 plt.legend(loc='upper center') #localisation de l'affichage des legendes
15 plt.xlabel("abscisses (X)") #Nomme l'axe des abscisses
16 plt.ylabel("ordonnees (Y,Z)") #Nomme l'axe des ordonnees
17 plt.show() #affichage de la figure
```

Listing 1 – Affichage de graphes

On rappelle qu'un polynôme est une fonction de la forme

$$P(x) = \sum_{k=0}^n a_k x^k,$$

où les $(a_k)_{k=0,\dots,n}$ sont les coefficients du polynôme.

Exercice 1 - Evaluation de polynôme

On considère un polynôme donné sous la forme

$$P(x) = \sum_{k=0}^n a_k x^k \quad (1)$$

pour des coefficients $a_0, a_1, \dots, a_n \in \mathbb{R}$.

- 1) Combien d'opération (additions, multiplications) nécessite l'évaluation de P sous la forme (1) ?
- 2) On écrit maintenant P sous la forme

$$P(x) = a_0 + x(a_1 + x(a_2 + x(a_3 + (\dots + xa_n) \dots))) \quad (2)$$

Combien d'opérations nécessite l'évaluation ?

- 3) Ecrire une fonction prenant en entrée la liste une liste de coefficients $[a_0, \dots, a_n]$, et renvoie l'évaluation de $P(x)$ grâce à la méthode (2).

On s'assurera de son fonctionnement sur des exemples simples !

Exercice 2 - Matrices en python

Python contient plusieurs manières de gérer les matrices. Il y a plusieurs Le type *matrix* est une spécialisation du type *array*, munie des opérations matricielles usuel. Voici un code d'exemple ; on peut constater que l'opération `*` appliquée à des *array* de même dimension multiplie les éléments deux-à-deux, tandis que l'opération `@` entre matrices donne une vraie multiplication matricielle.

```
1 import numpy as np
2
3 A=np.array([[1,2],[3,4]])
4
5 print("A*A=",A*A)
6 print("A@A=",A@A)
7
8 b=np.array([[4],[5]])
9 print("A@b=",A@b)
10
11 x=np.linalg.solve(A,b)#Resolution du systeme Ax=b
12 print("Ax-b=",A@x-b)
```

- 1) Écrire une fonction prenant en entrée deux tableaux $x = [x_0, x_1, \dots, x_n]$ et $y = [y_0, y_1, \dots, y_n]$ de même taille, et renvoie des coefficients $a_0, a_1, \dots, a_n$ tels que pour tout $i = 0, \dots, n$:

$$a_0 + a_1x_i + a_2x_i^2 + \dots + a_nx_i^n = y_i.$$

On admet que tant que les (x_i) sont distincts, il existe toujours une unique solution $(a_0, \dots, a_n)$. Si les (x_i) ne sont pas distincts, on renvoie une erreur.

- 2) On note, pour tout $k = 0, 1, \dots, n$, $x_k = -1 + \frac{2k}{n}$. Tracer sur une même figure le graphe de la fonction $f(x) = \frac{1}{1+x^2}$, les points $(x_i, f(x_i))_{i=0, \dots, n}$, et le graphe du polynôme de degré inférieur ou égal à n passant par tous les points $(x_i, f(x_i))_{i=0, \dots, n}$. On note ce polynôme P_n .
On testera différentes valeurs de $n = 5, 10, 15, 20 \dots$. Comment le polynôme P_n se comporte-t-il quand n est grand ?
- 3) On note $E_n = \max_{x \in [-1, 1]} |f(x) - P_n(x)|$ l'écart entre la fonction et son polynôme d'interpolation sur $[-1, 1]$. Tracer $\log_{10} |E_n|$ en fonction de n .
- 4) Faire la même étude sur la fonction $g(x) = \frac{1}{1+(5x)^2}$. Qu'observe-t-on sur le comportement de la suite (P_n) ?
- 5) Refaire les questions précédentes avec les points de Chebychev

$$x_k = \cos\left(\frac{2k+1}{2n+2}\pi\right), \quad k = 0, \dots, n.$$