Travaux dirigés 1 - MAP201, thème équation différentielles

Dans tout le TD on veillera à écrire un code suffisament 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.

```python
import numpy as np #importe les outils de gestion des tableaux
import matplotlib.pyplot as plt #importe les outils d'affichages matplotlib
plt.rcParams['figure.dpi'] = 120 #reglage de resolution des figures
plt.rcParams['savefig.dpi'] = 240 #reglage de resolution des figures enregistrees

N=10#declare une variable N initialisee sur la valeur 10
X=np.arange(N) #cree le tableau [0,1,2,3,4,5,....,N-1]. X[i]=i pour i=0,1,...,N-1
Y=np.random.random(N) #cree un tableau de N nombre aleatoires entre 0 et 1, numerotes de 0 a N-1
Z=np.linspace(1,3,N) #cree un tableau de N nombres qui subdivisisent l'intervalle [1,3] avec un ecart regulier

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 defaut 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
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.
plt.title("Affichage de Y,Z en fonction de X") #Donne un titre a la figure
plt.legend(loc='upper center') #localisation de l'affichage des legendes
plt.xlabel("abscisses (X)") #Nomme l'axe des abscisses
plt.ylabel("ordonnees (Y,Z)") #Nomme l'axe des ordonnees
plt.show() #affichage de la figure
```

Listing 1 – Affichage de graphes

Pour l'affichage de champs de vecteurs, on pourra utiliser ou s'inspirer de ce code.

```python
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['figure.dpi'] = 120
plt.rcParams['savefig.dpi'] = 240

def h(x,y):#definition d'une fonction de deux variables
    return (-np.sin(2*y+x),np.sqrt(1+x**2+y**4))#expression de h(x,y)
    #Rappel de notations de fonctions classiques:
    #x*y:multiplication
    #x**y: x exposant y
    #np.cos, np.sin, np.tan, np.log, np.exp, np.sqrt, np.abs: quelques fonctions classiques

def Affichage_Champs(g, xmin,xmax,ymin,ymax,N):
    #g: fonction qui definit le champs de vecteurs
    #xmin,xmax: intervalle d'abscisses affichees
    #ymin,ymax: intervalle d'ordonnees affichees
    #N: nombre de pas de discretisation
    x=np.zeros([N+1,N+1])
    y=np.zeros([N+1,N+1])
    u=np.zeros([N+1,N+1])
    v=np.zeros([N+1,N+1])#initialisation de quatres tableaux indexes sur [0,N]x[0,N]
    for i in range(N+1):#boucle for qui parcourt i=0,1,2,...,N
        for j in range(N+1):#idem pour j=0,1,2,..,N
            x[i,j],y[i,j]=xmin+(xmax-xmin)*i/N,ymin+(ymax-ymin)*j/N#abscisse et ordonnees du point associe a (i,j)
            u[i,j],v[i,j]=g(x[i,j],y[i,j])#valeur du champs de vecteur en (x[i,j],y[i,j])
    u=u*(ymax-ymin)/(xmax-xmin)#Changement d'echelle.
    u,v=u/np.sqrt(0.01+u**2+v**2),v/np.sqrt(0.01+u**2+v**2)#Normalisation du champs de vecteur: cela permet de rendre plus visible la direction.
    plt.quiver(x,y,u,v)#affichage le champs de vecteur: cette fonction trace le vecteur (u[i,j],v[i,j]) base en (x[i,j],y[i,j]), pour i,j qui parcourent [0,N]x[0,N]


Affichage_Champs(h,-2,2,-2,2,20)#Appelle la fonction ci-dessus
```

Listing 2 – Affichage de champs

Exercice 1 - Équations 1D

Pour chaque fonction f ci-dessous, tracer leur graphe sur l'intervalle conseillé $[x^{min}, x^{max}]$ grâce à la fonction plt.plot. Pour faire cela, on fixe un entier N "grand" (on pourra prendre $N = 100$), on créé X une subdivision de $[x^{min}, x^{max}]$ à N éléments (voir ligne 9 du code d'exemple), Y le tableau à N entrées tel que $Y[i] = f(X[i])$, et on affiche le graphe (X, Y) (voir ligne 11,12 du code d'exemple).

Puis, décrire de manière qualitative les solutions à $\begin{cases} \dot{x} = f(x), \\ x(0) = x^{init} \end{cases}$ pour x^{init} dans l'intervalle conseillé : on donnera le sens de variation et la limite en $t \to +\infty$ des solutions $x(t)$ selon la condition initiale x^{init}. On ne cherchera pas à donner d'expression explicite des solutions.

1) $f_1(x) = (1 - x^2)x$ sur $[-2, 2]$.
2) $f_2(x) = -\sin(x)$ sur $[-2\pi, 2\pi]$.
3) $f_3(x) = 1 - \sin(x)$ sur $[-2\pi, 2\pi]$.
4) $f_4(x) = 2 - \sin(x)$ sur $[-2\pi, 2\pi]$.

Exercice 2 - Méthode d'Euler explicite

1) On considère l'équation

$$\dot{x} = -x + \sin(t)$$

avec la condition initiale $x(0) = 1$. Écrire une fonction prenant en entrée une condition initiale x^{init}, un temps final T^{fin}, un entier $N \in \mathbb{N}^*$, et qui renvoie les tableaux $T = [t_0, t_1, \ldots, t_N]$, $X = [x_0, x_1, \ldots, x_N]$, où $t_i = \frac{iT^{fin}}{N}$ et x_i est la suite obtenue par méthode d'Euler explicite sur les temps $[t_0, \ldots, t_N]$.

On rappelle la formule de la méthode d'Euler explicite de l'équation $\dot{x} = f(t, x)$: on pose $t_i = \frac{iT^{fin}}{N}$, $\Delta t = \frac{T^{fin}}{N}$, et on définit par récurrence

$$\begin{cases} x_0 = x^{init} \\ x_{i+1} = x_i + \Delta t \ f(t_i, x_i) \end{cases}$$

2) Afficher le graphe de (T, X) obtenu pour $N = 10, 20, 50, 100$ avec un temps final $T^{fin} = 10$. On représentera les quatre sur un même graphe.

On pourra partir du premier code d'exemple donné en page 1. On pensera à mettre un label à chaque courbe, pour savoir qui est qui (voir la ligne 12 du code d'exemple)

Exercice 3 - Étude du pendule

On considère l'équation du pendule

$$\begin{cases} \ddot{x} = -\frac{g}{l} \sin(x) \\ x(0) = x^{init}, \ \dot{x}(0) = 0 \end{cases}$$

où x représente l'angle (en radian) entre le pendule et la verticale (pointant vers le bas, dans le sens de la gravité). On considère que le temps t est exprimé en secondes, g est l'intensité de la pesanteur ($g \approx 9.8$ en $m.s^{-2}$) et l est la longueur du pendule (en mètres).

On fixe $g = 9.8$ et $l = 1$.

1) Montrer/rappeler pourquoi cela revient à résoudre le système d'ordre 1 :

$$\begin{cases} \dot{x} = y \\ \dot{y} = -\frac{g}{l} \sin(x) \\ x(0) = x^{init}, \ y(0) = 0 \end{cases}$$

en identifiant $y = \dot{x}$.

2) Écrire une fonction prenant en entrée une condition initiale x^{init}, un temps final T^{fin}, un entier $N \in \mathbb{N}^*$, et qui renvoie les tableaux $T = [t_0, t_1, \dots, t_N]$, $X = [x_0, x_1, \dots, x_N]$, $Y = [y_0, y_1, \dots, y_N]$ où $t_i = \frac{iT^{fin}}{N}$ et $(x_i, y_i)_{0 \le i \le N}$ est la suite obtenue par méthode d'Euler explicite sur l'équation précédente.

3) Afficher le graphe (T, X) obtenu ci-dessus pour $N = 10000$ avec une condition initiale x^{init} prise entre 0 et $\pi/2$, et un temps final $T^{fin} = 10$.

On remarque ici que la trajectoire semble se répéter de manière périodique : x décroit jusqu'à la valeur (-x^{init}), puis croît à nouveau jusqu'à x^{init}, et répète ensuite le même mouvement. Pour estimer la période de ce mouvement, on va noter les ***pics*** *de x, c'est-à-dire les instants où $\dot{x}$ passe de positif à négatif.*

4) Écrire une fonction prenant en entrée les paramètres (x^{init}, T^{fin}, N) comme précédemment, et renvoie la liste des temps t_i ($L = [t_{i_1}, t_{i_2}, \dots]$ où $i_1 < i_2 < \dots$) pour lesquels $Y[i] > 0$ et $Y[i+1] \le 0$, c'est-à-dire pour lesquels il y a un **pic** (x passe de croissante à décroissante).

5) Calculer cette liste de temps pour différentes condition initiale $x^{init} \in (0, \pi/2)$, avec $T^{fin} = 10$. On vérifiera que pour N assez grand, l'écart successif entre ces temps est pratiquement constant, avec une erreur <1%. On prendra cette valeur moyenne comme approximation de la **période** d'oscillation du pendule.

6) Écrire une fonction qui prend en entrée une condition initiale x^{init} dans l'intervalle $]0, \pi/2]$, et renvoie une estimation de la période associée, grâce à la méthode ci-dessus avec $T^{fin} = 3,\ N = 100\,000$.

7) Afficher le graphe (X, P), où X est une subdivision de l'intervalle $]0, \pi/2]$ en 20 intervalles réguliers, et $P[i]$ est la période associée à la condition initiale $X[i]$.

On pourra mettre en évidence le fait que pour les valeurs petites de x^{init}, la période est bien approchée par la formule

$$2\pi\sqrt{\frac{l}{g}}\left(1 + \frac{(x^{init})^2}{16}\right)$$

en affichant les deux sur un même graphe

8) (Bonus) Afficher cette fois le graphe de la période (X, P) où X est une subdivision de $]0, \pi[$ en 40 intervalles réguliers. Que se passe-t-il lorsque $x^{init} \to \pi$?

Il faut veiller à prendre un temps final supérieur aux périodes associées, que l'on ne connaît pas a priori. De plus, lorsqu'on ne trouve pas de période, il faut vérifier si c'est dû 1) au fait que la période soit supérieure au temps final qu'on s'est fixé ou 2) au fait que le nombre de pas N dans la méthode d'Euler est insuffisant.

Exercice 4 - Modèle de pêche

On considère une population de poissons x qui suit une équation logistique $\dot{x} = x(1 - x)$.

1) Rappeler l'évolution de la population pour une condition initiale $x^{init} \ge 0$. Quelle est sa limite ?

2) Écrire une fonction prenant en entrée une condition initiale x^{init}, un temps final T^{fin} et un nombre de pas N, et qui renvoie les tableaux $T = [t_0, t_1, \dots, t_N], X = [x_0, x_1, \dots, x_N]$, où $t_i = \frac{iT^{fin}}{N}$ et x_i est la suite obtenue par méthode d'Euler explicite associée sur les temps $[t_0, \dots, t_N]$, avec x^{init} comme condition initiale.

On pourra réutiliser les codes des exercices précédents.

3) On fixe $T^{fin} = 20$ et N suffisamment grand $N \ge 200$). Afficher sur un même graphe des courbes

$$(T, X^k)_{k=0,1,2,3,\dots,10}$$

où X^k est la solution associée à la condition initiale $\frac{k}{10}$.

Pour afficher plusieurs courbes sur un même graphe, il est suffisant de toutes les tracer à la suite. On donne un exemple ci-dessous

```
def ex2bis_3([arguments a inserer]):
    for k in range(11):#donc k=0,1,2,3,..,10
        T,X=[calcul de la solution associee a x_init=k/10]
        plt.plot(T,X)
    plt.xlabel('t')
    plt.ylabel('x')
    plt.show()
```

Listing 3 – Affichage de plusieurs courbes sur un même graphe. Les parties entre crochets sont à modifier.

On introduit maintenant un terme de réduction de la population lié à la pèche : le nombre de poissons pêchés entre un temps t et $t + \Delta t$ est de l'ordre de

$$h \frac{x}{0.2 + x} \Delta t$$

ce qui donne l'équation différentielle

$$\dot{x} = x(1 - x) - h \frac{x}{0.2 + x}$$

ici $h > 0$ est une constante positive.

4) Commenter l'expression du nouveau terme. Que représente la quantité h, lorsque la population x est très grande ?

5) Afficher le graphe de la solution pour $h = 0.15$, et $x^{init} = \frac{k}{10}$ pour $k = 0, 1, 2, \ldots, 10$ comme précedemment. Que constate-t-on sur le sens de variation des solutions ? Donner la valeur de $x(20)$ dans chaque cas et donner une interprétation de l'effet de la pêche sur la population selon la population initiale x^{init}.

6) Refaire la question avec $h = 0.33$, puis $h = 0.4$. Que remarque-t-on dans chaque cas ? Donner une interprétation en terme d'effet de la pêche sur l'extinction (ou non) de la population de poissons.