Dynamique des populations TP16

L'objectif de ce TP est de résoudre numériquement des équations différentielles modélisant l'évolution d'une population N d'individus à travers le temps (t). On tracera l'évolution la population entre les temps $t = 0$ et $t = \texttt{T\_max}$ en utilisant un nombre fixé $\texttt{Nb\_iter} + 1$ de points régulièrement espacés entre 0 et `T_max`.

Schéma :

La liste des instants considérés est donnée par :

```
list_t = np.linspace(
```

Dit différemment, le pas de temps entre deux instants t_i étant $\Delta t = \dfrac{\texttt{T\_max}}{\texttt{Nb\_iter}}$, les instants considérés sont les $t_i = i \times \Delta t$ pour $i \in [\![0, \texttt{Nb\_iter}]\!]$, on a donc aussi :

```
list_t = [
```

On commence par présenter la méthode d'Euler explicite, c'est une méthode numérique permettant de résoudre de manière approchée une équation différentielle. On teste ensuite cette méthode sur des problèmes de dynamique des populations.

Exercice 1 **Méthode d'Euler** La méthode d'Euler consiste à construire une approximation du problème de Cauchy

$$(E)\ :\ y'(t) = F(y(t)) \ \text{ avec } \ y(0) = y_0$$

entre les instants $t = 0$ et $t = \texttt{T\_max}$ de la manière suivante. On considère un pas de temps $\Delta t = \dfrac{\texttt{T\_max}}{\texttt{Nb\_iter}}$ où `Nb_iter` est un nombre fixé, et on va construire par récurrence une approximation y_i de la solution $y(t_i)$ en les points $t_i = i \times \Delta t$ pour $i \in [\![0, \texttt{Nb\_iter}]\!]$. Pour $i = 0$, on fixe $\boxed{y_0 = y(0)}$ c'est-à-dire la condition initiale donnée par le problème de Cauchy. Puis pour tout $i \in [\![0, \texttt{Nb\_iter} - 1]\!]$, on définit

$$\boxed{y_{i+1} = y_i + \Delta t \times F(y_i).}$$

Le principe de cette itération vient de la remarque suivante : si y est la solution de l'équation (E) on a, si l'intervalle de temps $t_{i+1} - t_i$ est proche de 0 :

$$\frac{y(t_{i+1}) - y(t_i)}{t_{i+1} - t_i} \simeq y'(t_i) = F(y(t_i)).$$

Comme $y_i \simeq y(t_i)$ et que $t_{i+1} - t_i = \Delta t$ on en déduit que

$$\frac{y(t_{i+1}) - y_i}{\Delta t} \simeq F(y_i)$$

et on pose donc $y_{i+1} = y_i + \Delta t \times F(y_i)$ pour approcher $y(t_{i+1})$.
Comme il y a plusieurs approximations successives dans cette méthode, il n'est pas trivial de montrer que celle-ci fonctionne c'est-à-dire qu'elle fournit effectivement une bonne approximation de la solution de l'équation différentielle. On peut toutefois d'ores et déjà remarquer que plus le pas de temps Δt est petit, meilleure est l'approximation $\dfrac{y(t_{i+1}) - y(t_i)}{\Delta t} \simeq y'(t_i)$. On peut donc s'attendre à ce que la solution soit de bonne qualité lorsque Δt tend vers 0 c'est-à-dire lorsque `Nb_iter` tend vers $+\infty$.

Schéma :

Dans la suite de cet exercice, on considère le problème de Cauchy suivant :

$$(E) \ : \ \frac{dy}{dt} = 3 - \frac{y}{2} \quad \text{avec} \quad y(0) = 1$$

et on veut tracer sa solution pour t variant de 0 à `T_max` $= 10$.

Q1 Quelle est la fonction F telle que cette équation différentielle s'écrive sous la forme $y'(t) = F(y(t))$? Écrire à la main la relation de récurrence définissant la suite (y_i) donnée par la méthode d'Euler pour l'équation différentielle (E).

Q2 Écrire une fonction `Euler_1` prenant en argument un entier `Nb_iter` et renvoyant la liste `[`y_0`,` y_1`, ...,` $y_{\texttt{Nb\_iter}}$`]` des approximations donnée par la méthode d'Euler.

Q3 Tracer la solution approchée à l'équation différentielle (E) en fonction du temps pour une valeur de `Nb_iter` de votre choix.

Q4 On sait en fait résoudre explicitement le problème (E). Le faire puis écrire une fonction `sol_exacte_1` prenant en argument $t \in \mathbb{R}$ et renvoyant la valeur de la solution exacte $y(t)$ au problème (E).

Q5 Tracer sur un même schéma la solution exacte (en vert) et la solution approchée (en bleu) pour différentes valeurs de `Nb_iter`. Que remarquez-vous ?

Exercice 2 — Modèle logistique de Verhulst

Le modèle logistique décrit l'évolution d'une population N d'individus au cours du temps (t) par l'équation différentielle suivante :

$$(E) \ : \ \frac{dN}{dt} = rN\frac{K-N}{K}$$

où r est le taux de croissance intrinsèque de la population et K la capacité limite du milieu. On suppose de plus qu'on connaît la popultation initale $N(t=0) = N_0$ et qu'on cherche à décrire l'évolution de la population jusqu'à un instant $t =$ `T_max`.

Dans tout l'exercice, on suppose que les constantes K, r, N_0 et `T_max` sont fixées. On utilisera donc des *variables globales* pour ces quantités, c'est-à-dire des variables définies en début de script, en dehors des fonctions. On propose tout d'abord de prendre les valeurs suivantes :

```
r = 0.7
Tmax = 10
K = 300
N0 = 10
```

Q1 Écrire à la main la relation de récurrence définissant la suite (N_i) donnée par la méthode d'Euler pour l'équation différentielle (E).

Q2 Écrire une fonction `Euler_2` prenant en argument un entier `Nb_iter` et renvoyant la liste `[`N_0`, `N_1`, ..., `$N_{\texttt{Nb\_iter}}$`]` des approximations donnée par la méthode d'Euler.

Q3 Tracer la solution approchée à l'équation différentielle (E) en fonction du temps pour une valeur de `Nb_iter` de votre choix.

Q4 Même si cette équation différentielle n'est pas linéaire, on peut en fait résoudre mathématiquement l'équation logistique (E) de manière explicite. Pour cela, posons $y = \dfrac{1}{N}$.

1. Montrer que y est solution de l'équation $(E') \ : \ y' + ry = \dfrac{r}{K}$.
2. Résoudre l'équation (E') et en déduire l'expression de $N(t)$ en fonction d'une constante d'intégration λ.
3. En utilisant la valeur initiale $N(t=0) = N_0$, montrer que, finalement, N est donnée par :

$$N(t) = \frac{K}{1 + (\frac{K}{N_0} - 1)e^{-rt}}$$

4. Écrire une fonction `sol_exacte_2` prenant en argument t et renvoyant la valeur exacte de la solution $N(t)$.

Q5 Tracer sur un même schéma la solution exacte et la solution approchée pour différentes valeurs de `Nb_iter`. Que remarquez-vous ?

Q6 Faites varier les paramètres r, K et N_0.

Exercice 3 Modèle proie-prédateur de Lotka-Volterra

Le modèle de Lotka-Volterra décrit l'évolution de deux populations N (les proies) et P (les prédateurs) au cours du temps (t) par le système d'équations différentielles suivant :

$$(S)\ :\ \begin{cases} \dfrac{dN}{dt} = rN - CNP \\ \dfrac{dP}{dt} = -mP + gCNP \end{cases}$$

où r est le taux de croissance intrinsèque des proies, m le taux de mortalité des prédateurs, g le taux de croissance des prédateurs selon la densité des proies et C une dernière constante. On suppose de plus qu'on connaît les popultations initales $N(t=0) = N_0$, $P(t=0) = P_0$ et qu'on cherche à décrire l'évolution des populations jusqu'à un instant $t = \texttt{T\_max}$.

Comme précédemment, toutes les constantes seront représentées en Python par des variables globales. On propose tout d'abord de prendre les valeurs suivantes :

```
r = 0.1
m = 0.1
C = 0.001
g = 0.7
Tmax = 400
N0 = 400
P0 = 100
```

Contrairement à précédemment, on doit maintenant prendre en compte deux populations, N et P, donc construire deux suites (N_i) et (P_i) approximant les valeurs de $N(t_i)$ et $P(t_i)$. Le principe de la méthode d'Euler s'applique de la même manière, on définit donc depuis les initialisations à N_0 et P_0 :

$$\forall i \in [\![0, \texttt{Nb\_iter - 1}]\!],\ \begin{cases} N_{i+1} = N_i + \Delta t \times (rN_i - CN_iP_i) \\ P_{i+1} = P_i + \Delta t \times (-mP_i + gCN_iP_i) \end{cases}$$

Q1 Écrire une fonction `Euler_3` prenant en argument un entier `Nb_iter` et renvoyant les deux listes `[`N_0`, `N_1`, ..., `$N_{\texttt{Nb\_iter}}$`]` et `[`P_0`, `P_1`, ..., `$P_{\texttt{Nb\_iter}}$`]` données par la méthode d'Euler.

Q2 Tracer sur le même graphe l'évolution de N (en bleu) et de P (en rouge) en fonction de t pour une valeur de `Nb_iter` de votre choix.

Q3 Cette fois-ci, il n'est pas possible de résoudre explicitement le système différentielle (S). On utilise donc la fonction Python `odeint` qui calcule par des techniques évoluées une très bonne approximation de la solution. Dans la suite, on confondra le résultat donné par `odeint` avec la solution exacte. Téléchargez sur cahier de prépa le fichier `odeint_Lotka_Volterra`, copiez et exécutez le code qu'il contient dans votre script.

Q4 Tracer sur le même graphique les solutions exactes et approchées pour différentes valeurs de `Nb_iter`. Que remarquez-vous ?

Q5 Tracer l'évolution du point $(N(t), P(t))$ au cours du temps. Comparer les solutions exactes et approchées.

Q6 Faites varier les paramètres N_0, P_0, r, m, g, C, `T_max`.