

TP01-DynamiquePopulations-Corrige

September 16, 2019

1 TP01 - Modèles discrets de dynamique de populations

Il s'agit dans ce TP de modéliser l'évolution au cours du temps d'un organisme. Il peut s'agir d'une population bactérienne, de mammifères, d'oiseaux ou encore d'espèces végétales. Nous choisissons ici d'évacuer toute idée d'organisation spatiale pour nous consacrer exclusivement à leurs changements d'effectif.

Pour cela différents modèles se sont mis en place au cours du temps et nous allons découvrir les premiers d'entre eux, relativement simples mais essentiels pour comprendre les outils mathématiques et informatiques dont ils relèvent.

1.1 Importation préalable des outils Python nécessaires :

```
In [1]: import numpy as np
        %matplotlib inline
        %matplotlib notebook
        import matplotlib.pyplot as plt
```

1.2 I/ Le modèle malthusien.

Il semble raisonnable de supposer que, chaque unité de temps (secondes, heures, jours ou années selon la population étudiée...), l'effectif change suite à de nouvelles naissances et que, sous cette seule hypothèse, il augmente par addition d'un certain multiple f de la population lié à la fécondité. Sur cette même durée, on suppose qu'une fraction constante d de la population va mourir.

On admet en conséquence que si P_n désigne le nombre d'individus à l'instant $t = n$ ($n \in \mathbb{N}$), alors

$$\Delta P = P_{n+1} - P_n = fP_n - dP_n = (f - d)P_n = rP_n$$

où

$$r = f - d$$

On parle de **modèle linéaire**.

Question I.1 : On qualifie également ce modèle de *modèle géométrique*. Pourquoi ? Justifier pourquoi $d \in]0, 1[$, $f \in \mathbb{R}_+$ et dire à quel intervalle appartient la raison λ de ce modèle définie par :

$$P_{n+1} = \lambda P_n, \quad n \in \mathbb{N}$$

Question I.2 : Considérons la population de mouches installée dans cette classe durant l'été et dont $P_0 = 10$ individus distincts ont été dénombrés le jour de la rentrée. Les paramètres démographiques nous permettent d'estimer que $f = 0.13$ et $d = 0.07$ sur un pas de temps d'une journée.

Compléter la fonction ci-dessous qui permet de modéliser l'évolution de la population entre le 1er et le 30 septembre et retourne les effectifs quotidiens sous forme, au choix, d'une liste ou d'un tableau :

```
In [2]: def evolution1(N,n,f,d):
        # N : réel égale à la population initiale.
        # n : nombre de jours simulés.
        P = [N]
        for k in range(1,n):
            N *= (1+f-d) # 1+f-d = 1.06
            P.append(N)
        return P # P = [P0,P1,...,P29] car évolution sur 30 jours à modéliser
```

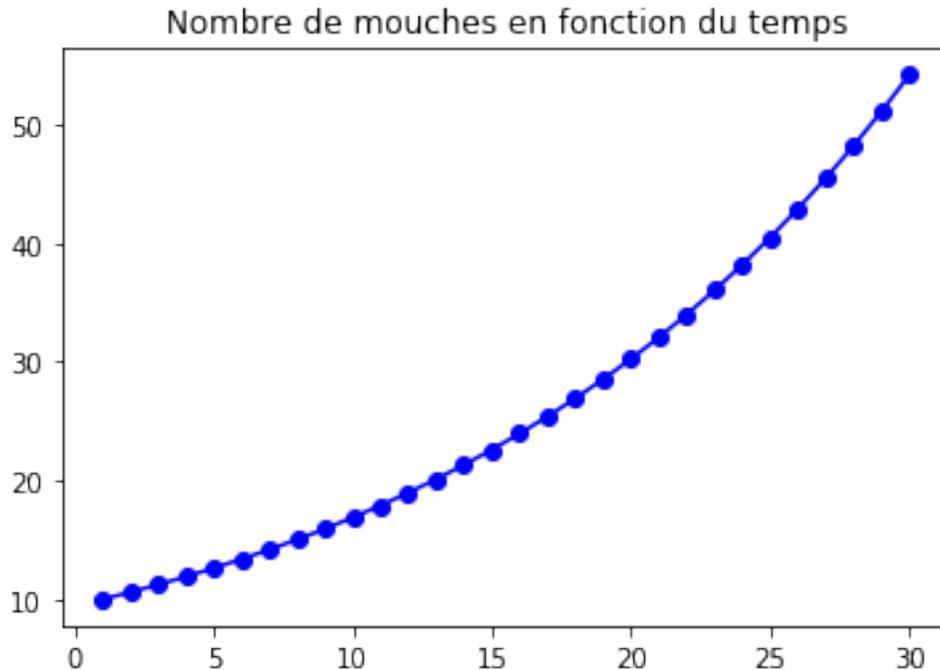
```
In [3]: def evolution1b(N,n,f,d):
        # pour l'utilisation de la forme explicite:
        P= [0]*n
        P[0] = N
        for n in range(1,n):
            P[n] = (1+f-d)**n*P[0]
        return P
```

Testez votre réponse grâce aux lignes de code qui suivent :

```
In [4]: f,d = 0.13,0.07
        P0,n = 10,30
        P = evolution1(P0,n,f,d)
        print('le nombre de mouches à la fin du mois vaut : ',P[-1])
        Temps = np.arange(1,31)
        plt.figure("Evolution durant un mois")
        plt.plot(Temps,P,'bo-')
        plt.title('Nombre de mouches en fonction du temps')
```

le nombre de mouches à la fin du mois vaut : 54.18387898974738

```
Out[4]: Text(0.5, 1.0, 'Nombre de mouches en fonction du temps')
```



Vous avez dû obtenir de l'ordre de 54 mouches à la fin du mois de septembre. Il était temps qu'on rentre et qu'on ouvre les fenêtres !

Question I.3 : On s'intéresse aux femelles d'une population d'insectes dont le développement s'effectue par stades. Chaque femelle pond en moyenne 200 oeufs avant de mourir. Au moment de l'éclosion, seuls 3% survivent pour devenir à leur tour des femelles adultes, le reste devenant des mâles ou mourant.

On mesure cette fois le temps t en générations.

1. Exprimer l'équation correspondant au modèle malthusien. La population va-t-elle croître ou décroître ?
2. Supposons que nous ne connaissions pas la fécondité mais sachions seulement que la population reste stable au cours du temps. Estimer le taux de survie t_s dans ce cas.
3. quels cas ignorer le mâles pour suivre l'évolution d'une population serait une mauvaise idée ?

Réponse : En 2. le taux de survie vaut $t_s = ?$

Question I.4 : On imagine une population dont la dynamique est modélisée annuellement par l'équation :

$$P_{n+1} = 0.9P_n \text{ et } P_0 = 100$$

Dire, sans calcul, si la population va croître ou décroître.

On suppose qu'une immigration constante notée c a lieu à chaque intervalle de temps. Compléter la fonction Python `modLinIm.py` ci-dessous afin de modéliser l'évolution de cette population en fonction des valeurs de c et préciser dans un second temps à partir de quelle valeur de c la population va adopter un comportement différent.

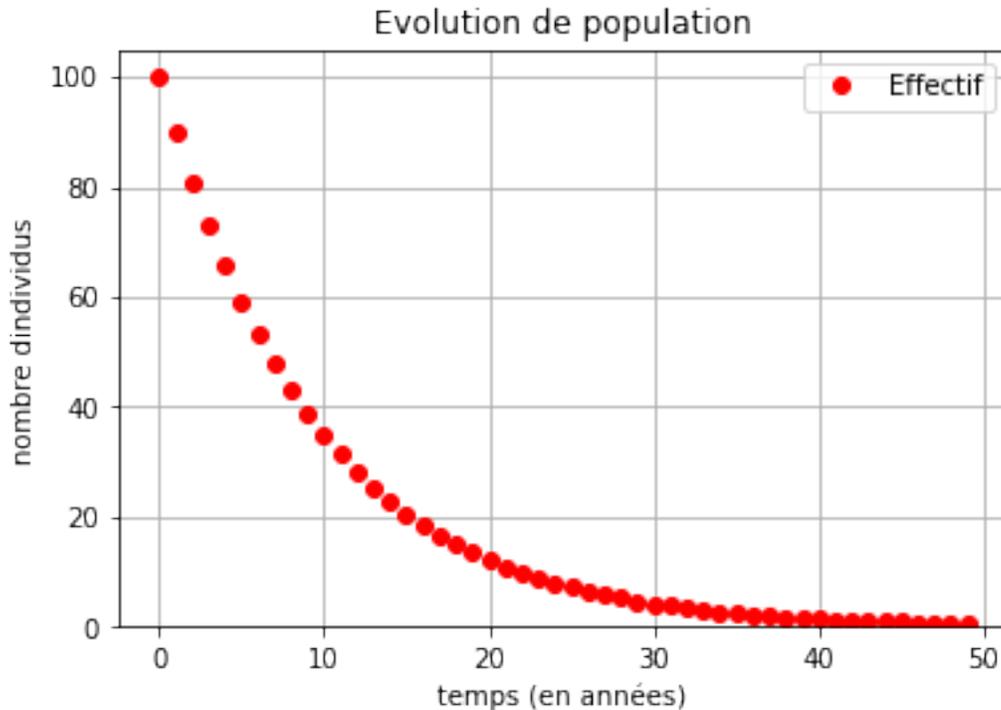
```
In [5]: def modLinIm(c,n):
        # c : réel désigne le coefficient d'immigration
        # n : entier. Nombre d'années sur lequel porte la modélisation
        P=[0]*n # initialisation d'une liste de n zéros
        N=100 # Population initiale de 100 individus
        P[0]=N
        for i in range(1,n):
            N=0.9*N+c
            P[i]=N
        return P
```

```
In [6]: c=0# Immigration initialement nulle
        n=50 #Nombre d'intervalles de temps simulés
```

```
Pop=modLinIm(c,n)
```

```
def graphe():
    Temps = np.arange(n)
    plt.figure("Phénomène migratoire")
    plt.grid()
    plt.plot(Temps,Pop, 'ro',label='Effectif')
    plt.ylim(bottom = 0.)
    plt.legend(loc='upper right')
    plt.xlabel('temps (en années)')
    plt.ylabel('nombre d''individus')
    plt.title('Evolution de population')
```

```
graphe()
```



Justifier mathématiquement votre résultat : D'après le cours sur les suites arithmético-géométriques

2 II/ Les modèles non linéaires.

Les modèles malthusiens prédisent une croissance géométrique des populations. On conçoit aisément les limites d'un tel modèle qui suppose en particulier que les taux de fécondité et de mortalité sont indépendants de la taille de la population. Pourtant on sait que les paramètres démographiques d'une population dépendent notamment de la capacité d'accueil du milieu qui impose en particulier que le taux de mortalité augmente (pour des raisons de ressources par exemple) si l'effectif s'approche de cette capacité maximum.

2.1 II.1. Création d'un modèle non linéaire :

Observons la variation relative de population par individu au cours d'un pas de temps : $\frac{\Delta P}{P}$.

Dans le modèle malthusien, ce rapport est indépendant de P et vaut la constante $r = f - d$.

On suppose désormais que, lorsque l'effectif de la population est faible au regard de la capacité d'accueil de l'environnement, les ressources disponibles n'ont pas d'impact sur la dynamique qui peut être considérée comme malthusienne.

A l'inverse, pour de grandes valeurs de P , la compétition entre individus, à la fois pour de la nourriture et de l'espace rend $\frac{\Delta P}{P}$ faible. On peut même supposer que $\frac{\Delta P}{P} = 0$ pour $P = K$ où K désigne la capacité d'accueil du milieu et devient négatif pour toute valeur de P supérieure à K .

Une façon simple de représenter l'accroissement de population par individu en fonction de la taille de la population est une fois encore de considérer un modèle linéaire. le principe est le suivant :

Question II.1 : Utiliser le graphe ci-dessus pour déterminer a et b en fonction de r et de K .

Question II.2 : On s'intéresse à une population de carpes dans un étang dont la population est évaluée en tonnes.

L'équation logistique obtenue est : $P_{n+1} = P_n \cdot \left(1 + 0.7 \left(1 - \frac{P_n}{10} \right) \right)$ avec un pas de temps d'une année.

Déterminer r et K dans ce cas précis.

Question II.3 : Dans le cas général, pour tout $n \in \mathbb{N}$, on a :

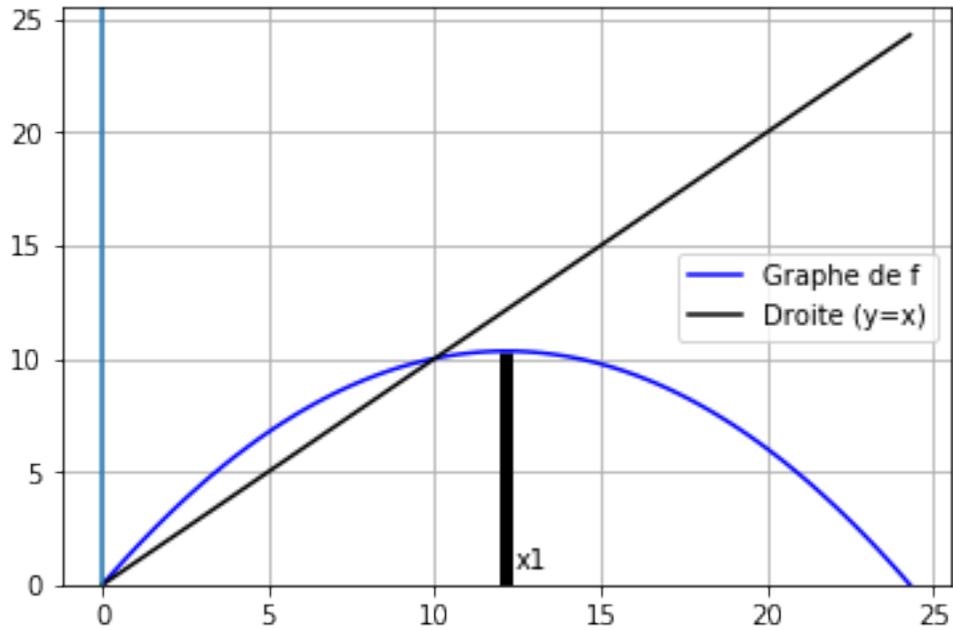
$$P_{n+1} = f(P_n) \text{ où } f(x) = x \left(1 + r \left(1 - \frac{x}{K} \right) \right)$$

Étudier succinctement la fonction f sur un intervalle $[a, b]$ qu'on précisera sur lequel elle est positive.

1. Déterminer son maximum P_{max} ainsi que ses éventuels points fixes (valeurs réelles l telles que $f(l) = l$).
Soit x_1 l'abscisse de son maximum. Utiliser les lignes de code ci-dessous pour tracer le graphe de f en distinguant trois cas, selon que $x_1 < K$, $x_1 = K$ et $x_1 > K$ (On montrera pour ça que ces trois cas peuvent se ramener à une condition sur r qu'on précisera).
2. Quelle limite au modèle proposé envisagez-vous à partir de l'étude de f ? Pourquoi désignera-t-on désormais b par P_{limite} ?
3. Pour quelle taille de la population l'accroissement est-il maximal et que vaut-il ?

```
In [8]: r = 0.7
        K = 10
        f = lambda x:x*(1+r*(1-x/K))
        a,b = 0,K*(r+1)/r
        x1 = b/2
        X = np.linspace(a,b,100)
        plt.figure("modèle logistique")
        plt.plot(X,f(X), 'b-',label='Graphe de f')
        plt.plot(X,X, 'k-',label='Droite (y=x)')
        plt.ylim(bottom=0)
        plt.axvline()
        plt.grid()
        plt.vlines(x1,0,K*(r+1)**2/(4*r),lw=5,linestyle='-')
        plt.text(x1+0.3,0.7, 'x1')
        plt.legend(loc='best')
```

```
Out[8]: <matplotlib.legend.Legend at 0x8eb37cfb70>
```



Question II.3 : On considère à nouveau l'exemple de la population de carpes dans un étang dont l'équation est donnée par :

$$P_{n+1} = P_n \left(1 + 0.7 * \left(1 - \frac{P_n}{10} \right) \right)$$

Compléter la fonction `EvolPop1.py` afin de simuler l'évolution au cours du temps de la population.

```
In [9]: r = 0.7
        K = 10.
```

```
In [10]: def EvolPop1(P0,n):
          # P0 : entier égale à la population initiale.
          # n : entier égale au nombre d'années de la modélisation
          P = np.zeros(n+1)# initialisation à 0 des n+1 valeurs de l'effectif
          P[0] = P0
          for pas in range(n):
              P[pas+1] = P[pas]*(1.+r*(1.-P[pas]/K))# Rentrer votre code ici
          return P
```

Testez votre fonction sur vingt années en prenant successivement $P_0 = 0.4346$, $P_0 = 2.3$, $P_0 = 13.245$, $P_0 = 18$, $P_0 = 10$ et $P_0 = 26$ dans la ligne de code ci-dessous. Retrouver au terme de ces essais les limites du modèle logistique.

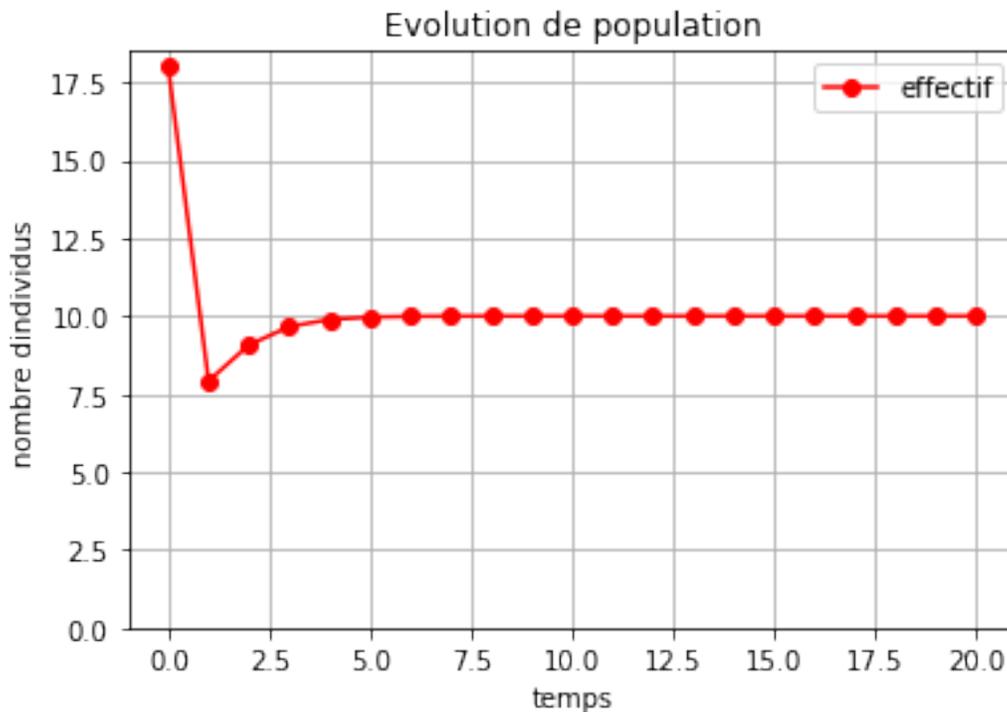
```
In [11]: n = 20
         P0 = 18
         pop = EvolPop1(P0,n)
```

```

temps = np.arange(n+1)# tableau d'entiers de 0 à n
plt.figure("Evolution avec P_0 = "+str(P0))
plt.plot(temps,pop,'ro-',label='effectif')
plt.grid()
plt.legend(loc='best')
plt.xlabel('temps')
plt.ylabel('nombre d\'individus')
plt.ylim(ymin = 0.)
plt.title('Evolution de population')

```

Out[11]: Text(0.5, 1.0, 'Evolution de population')



Justifier mathématiquement ces différents comportements en étudiant la suite récurrente $(P_n)_{n \in \mathbb{N}}$ selon les valeurs de P_0 .
On pourra s'aider pour ça de la fonction `suiteRec.py` ci-dessous (qui n'est pas à savoir écrire) et du graphe de f tracé à la question II.3.

```

In [12]: # Fonction de récurrence :
r = 0.7 # valeur fixée
K = 10
f = lambda x: x*(1+r*(1-x/K))
# Paramètres du modèle :
P0 = 0.5
n = 10
P_limite = K*(1+r)/r

```

```

# Valeurs utiles pour le tracé de f :
X=np.linspace(0,P_limite,100)
Y=f(X)

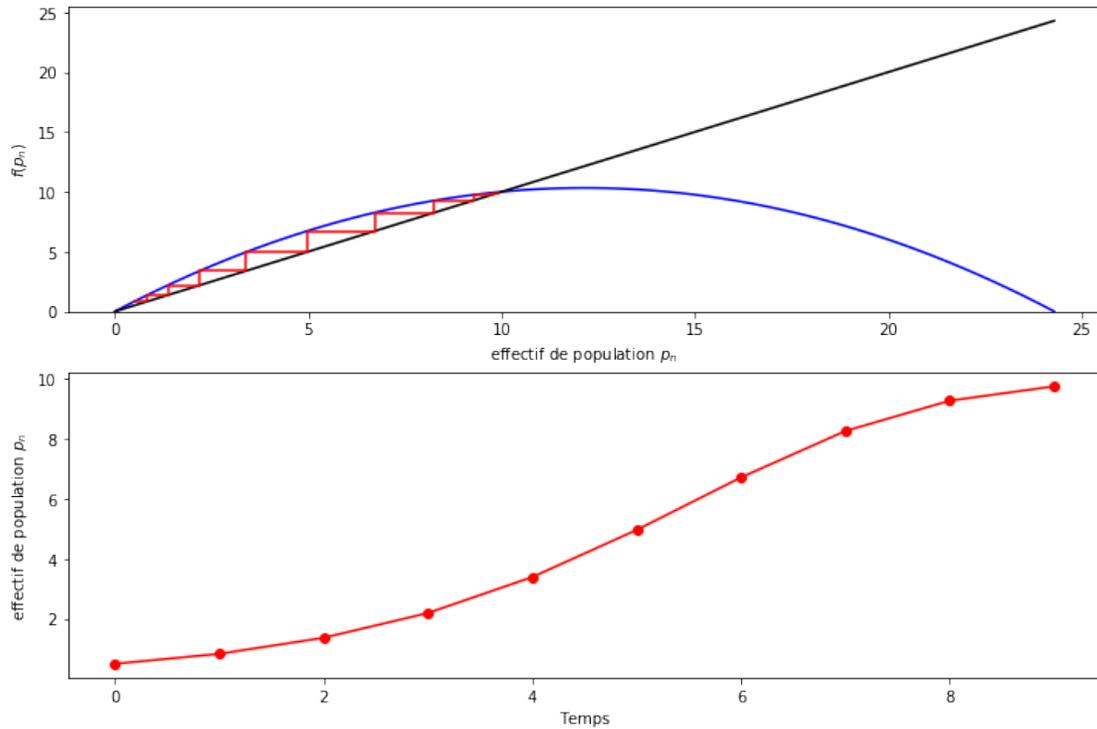
def suiteRec(u,n):
    U = [u]
    v = f(u)
    A, O = [u], [v] # initialisation des Abscisses et des Ordonnées
    for k in range(1,n):
        U.append(v)
        A.extend([v,v])
        u = v
        v = f(u)
        O.extend([u,v])
    A.append(v)
    O.append(v)
    return A,O,U

Abs,Ord,U=suiteRec(P0,n)

def trace_graphe():
    #plt.figure("suite récurrente pour P0 = "+str(P0))
    plt.subplots(figsize=(12, 8))
    plt.subplot(211)
    plt.plot(X,Y, 'b-',X,X, 'k-',Abs,Ord, 'red',label='$p_{n+1}=f(p_n)$')
    plt.ylim(ymin = 0.)
    plt.xlabel('effectif de population $p_n$')
    plt.ylabel('$f(p_n)$')
    plt.subplot(212)
    plt.plot(np.arange(n),U, 'ro-')
    plt.xlabel('Temps')
    plt.ylabel('effectif de population $p_n$')
    plt.show()

trace_graphe()

```



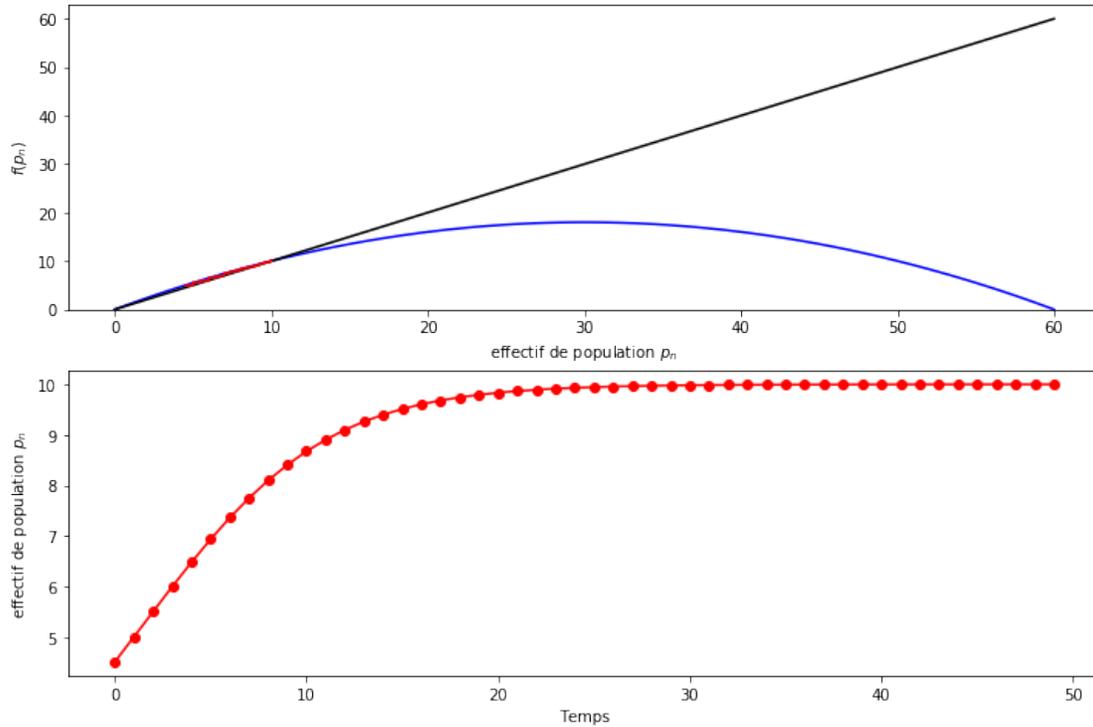
On se pose maintenant la question de savoir ce qui se passe si on fait varier r et notamment quel est l'impact sur l'évolution de la suite $(P_n)_{n \in \mathbb{N}}$ de la distinction de cas observée à la question II.2 : $r < 1$ et $r > 1$

Question II.4: On considère cette fois l'équation logistique $\Delta P = rP(1 - P/10)$.

Utiliser comme ci-dessous `suiteRec.py` et `trace_graphe()` écrites à la question II.3 afin d'étudier sur le long terme ($n = 50$ années) et selon les valeurs de P_0 qu'on fera varier le comportement de la population pour $r = 0.2, 0.8, 1.9, 2.2, 2.5, 2.55$ et 2.95 successivement.

```
In [14]: # Fonction de récurrence :
r = 0.2 # valeur
f = lambda x:x*(1+r*(1-x/10))
# Paramètres du modèle :
P0 = 4.51
n = 50
P_limite = 10*(1+r)/r
# Valeurs utiles pour le tracé de f :
X=np.linspace(0,P_limite,100)
Y=f(X)

Abs,Ord,U=suiteRec(P0,n)
trace_graphe()
```



Question II.5 : Estimation des paramètres.

On suppose avoir relevé les variations d’une population d’insectes au cours du temps. Les données obtenues sont les suivantes :

n	0	1	2	3	4	5	6	7	8	9	10
P_n	0.97	1.52	2.31	3.36	4.63	5.94	7.04	7.76	8.13	8.3	8.36

1. Représenter par des points l’évolution de cette population.
2. On souhaite simuler cette évolution en utilisant le modèle logistique pour lequel $r = 0.8$ et $K = 10$. Donner les dix premières valeurs obtenues à l’aide de l’équation logistique correspondante et les tracer sur le même graphe que précédemment. Ce modèle vous semble-t-il adapté ?
3. Proposez des valeurs de r et de K permettant d’approcher au mieux les valeurs expérimentales. Justifier votre démarche et la confirmer par une représentation graphique.

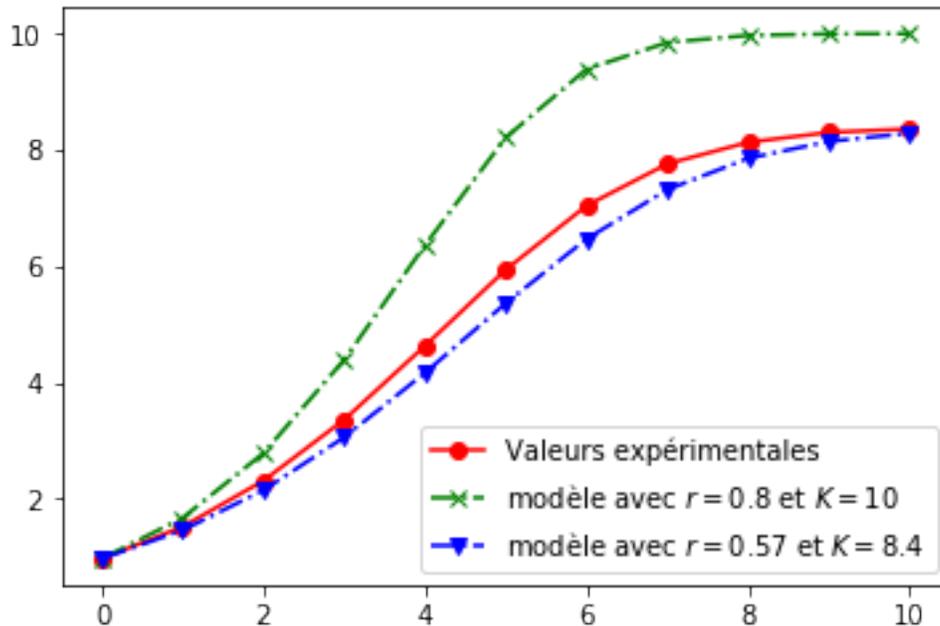
```
In [15]: X = np.arange(11)
P = np.array([0.97, 1.52, 2.31, 3.36, 4.63, 5.94, 7.04, 7.76, 8.13, 8.3, 8.36])
plt.figure("Modèle hypothétique et estimé")
plt.plot(X,P, 'ro-', label='Valeurs expérimentales')
# Question 2. ----- :
r = 0.8
K = 10.
```

```

P_mod1 = EvolPop1(0.97,10)
plt.plot(X,P_mod1,'gx-.',label='modèle avec $r=0.8$ et $K=10$')
# Question 3. ----- :
r,K = 0.57,8.4
P_mod2 = EvolPop1(0.97,10)
plt.plot(X,P_mod2,'bv-.',label='modèle avec $r=0.57$ et $K=8.4$')
plt.legend(loc='best')

```

Out [15]: <matplotlib.legend.Legend at 0x8eb3dba860>



3 III. Analyse des modèles logistiques non linéaires

3.1 3.1 Introduction.

Nous avons pu constater dans les sections précédentes que l'évolution du modèle était sensible à la fois aux conditions initiales (P_0) mais aussi aux paramètres choisis, notamment r .

Dans l'exemple du suivi de la population de carpes dont le modèle était :

$$P_{n+1} = P_n \left(1 + r \left(1 - \frac{P_n}{10} \right) \right) = f_r(P_n) \text{ avec } r = 0.7$$

toute valeur de P_0 conduisait à un comportement à long terme identique, à savoir, la convergence vers $P^* = 10 = K$.

A la question II.4. on a constaté que pour toute valeur de $r < 2$ cette même valeur P^* d'équilibre semble être atteinte.

Notons pour la suite :

$$f_r(x) = x \left(1 + r \left(1 - \frac{x}{K} \right) \right)$$

Conformément au cours sur les suites récurrentes, la fonction f_r étant continue sur \mathbb{R}_+ , en cas de convergence de la suite (P_n) sa limite vérifie $f_r(P^*) = P^*$.

C'est un point fixe de f_r .

Définition 1 : On appelle **état stationnaire** toute solution constante du modèle. On parle aussi de valeur d'équilibre ou de point fixe du modèle.

Toute valeur d'équilibre se situe donc graphiquement à l'intersection de la courbe représentant f_r et de la droite d'équation $(y = x)$. Pour le modèle logistique on dénombre donc deux états stationnaires : $P_1^* = 0$ et $P_2^* = K$.

Les équilibres, pour autant, peuvent présenter différentes caractéristiques. Ainsi, dans l'exemple ci-dessus, une population proche de 0 tend à s'éloigner de cet effectif nul tandis qu'une population proche de K tend à se rapprocher de K . On dira que $N^* = 0$ est un état stationnaire **instable** tandis que $N^* = K$ est un état stationnaire **stable**.

Définition 2 : On dit qu'un état stationnaire N^* d'un modèle logistique est **stable** lorsque toute suite (P_n) solution de ce modèle ayant pour condition initiale N_0 proche de N^* tend vers N^* lorsque n tend vers l'infini. Un état stationnaire qui n'est pas stable au sens de cette définition sera dit **instable**.

Tout système écologique étant soumis à de légères perturbations, on peut estimer que ce sont les équilibres stables que nous devrions observer dans la nature et que l'étude de nos modèles devra se focaliser sur ces états stationnaires s'ils existent.

3.2 Principe de linéarisation.

L'objectif est de déterminer analytiquement si un état stationnaire est stable ou instable.

On pose $P_n = P^* + \epsilon_n$ où ϵ_n désigne la perturbation par rapport à l'équilibre à l'instant $t = n$.

La relation $P_{n+1} = P^* + \epsilon_{n+1} = f_r(P_n) = f_r(P^* + \epsilon_n)$ pour tout $n \geq 0$ permet d'obtenir ϵ_{n+1} en fonction de ϵ_n .

Si $|\epsilon_{n+1}| > |\epsilon_n|$ alors P_{n+1} s'est éloigné de l'équilibre, sinon il s'en est rapproché. Il sera dit stable.

Exemple : Considérons à nouveau l'exemple de la population de carpes pour lequel $r = 0.7$ et $K = 10$

Dans le cadre du modèle logistique associé, posons $P_1^* = 0$ et $P_2^* = 10$ et commençons par observer l'état stationnaire P_2^* .

Dans ce cas : $P_n = P_2^* + \epsilon_n = 10 + \epsilon_n$.

Alors :

$$\begin{aligned} P_{n+1} = 10 + \epsilon_{n+1} &= f_r(P_n) = (10 + \epsilon_n) \left(1 + 0.7 \left(1 - \frac{10 + \epsilon_n}{10} \right) \right) \\ &= (10 + \epsilon_n)(1 - 0.07\epsilon_n) = 10 + 0.3\epsilon_n - 0.07\epsilon_n^2 \end{aligned} \quad (1)$$

D'où

$$\epsilon_{n+1} = 0.3\epsilon_n - 0.07\epsilon_n^2$$

Mais comme on s'intéresse à de petites perturbations autour de N_2^* , on a ϵ_n proche de zéro et donc

$$\epsilon_{n+1} \underset{\epsilon_n \rightarrow 0}{\sim} 0.3\epsilon_n$$

ou encore

$$|\epsilon_{n+1}| < |\epsilon_n|$$

On peut donc conclure à une compression des perturbations, ce qui confirme que $N_2^* = 10$ est un état stable.

Question 3.1 : Vérifier qu'au voisinage de $P_1^* = 0$, $|\epsilon_{n+1}| > |\epsilon_n|$ lorsque ϵ_n proche de zéro. Conclure que les perturbations sont augmentées, ce qui assure le caractère instable de $N_1^* = 0$.

Argument général : L'instabilité ou non d'un état stationnaire repose sur le fait que $\left| \frac{\epsilon_{n+1}}{\epsilon_n} \right|$ est supérieur ou pas à 1 lorsque ϵ_n tend vers 0.

Une autre caractérisation de la stabilité devient donc possible :

$$\frac{\epsilon_{n+1}}{\epsilon_n} = \frac{P_{n+1} - P^*}{P_n - P^*} = \frac{f_r(P_n) - f_r(P^*)}{P_n - P^*}$$

puisque P^* est un point fixe.

Dès lors, f_r étant dérivable sur \mathbb{R}_+ :

$$\lim_{\epsilon_n \rightarrow 0} \frac{\epsilon_{n+1}}{\epsilon_n} = \lim_{P_n \rightarrow P^*} \frac{f_r(P_n) - f_r(P^*)}{P_n - P^*} = f_r'(P^*)$$

Théorème 1 : Si un modèle $P_{n+1} = f_r(P_n)$ possède un état stationnaire P^* , alors $|f_r'(P^*)| > 1$ implique que P^* est instable tandis que $|f_r'(P^*)| < 1$ implique que P^* est stable.

Remarque : Si $|f_r'(P^*)| = 1$ alors cette information n'est pas suffisante pour conclure sur la stabilité de cet état.

Question 3.2 : Vérifier ce théorème au regard des évolutions modélisées à la question II.4.