
Correction TP8 Python

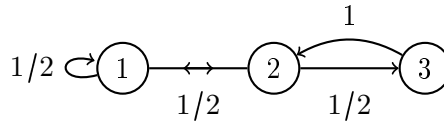
Exercice 1

1) $X_0(\Omega) = \{2\}$ donc X_0 suit une loi certaine.

$X_1(\Omega) = \{1, 3\}$ et $P(X_1 = 1) = P(X_1 = 3) = \frac{1}{2}$.

Donc X_1 suit une loi uniforme sur $\{1, 3\}$.

2)a)



$$\text{b) } A = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1 & 0 \end{pmatrix}.$$

3) $P(X_{n+1} = 1)$

$$= P_{\{X_n=1\}}(X_{n+1} = 1)P(X_n = 1) + P_{\{X_n=2\}}(X_{n+1} = 1)P(X_n = 2) + P_{\{X_n=3\}}(X_{n+1} = 1)P(X_n = 3)$$

$$= \frac{1}{2}P(X_n = 1) + \frac{1}{2}P(X_n = 2) + 0P(X_n = 3).$$

Idem pour les deux autres.

4)a) $U_0 = (P(X_0 = 1), P(X_0 = 2), P(X_0 = 3)) = (0, 1, 0)$.

$$\begin{aligned} \text{b) } U_n A &= (P(X_n = 1), P(X_n = 2), P(X_n = 3)) \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1/2 \\ 0 & 1 & 0 \end{pmatrix} \\ &= \left(\frac{1}{2}P(X_n = 1) + \frac{1}{2}P(X_n = 2), \frac{1}{2}P(X_n = 1) + P(X_n = 3), \frac{1}{2}P(X_n = 2) \right) \\ &= (P(X_{n+1} = 1), P(X_{n+1} = 2), P(X_{n+1} = 3)) \\ &= U_{n+1}. \end{aligned}$$

5)a) programme

```
import numpy as np
A=np.array(([0.5,0.5,0],[0.5,0,0.5],[0,1,0]))
U=np.array([0,1,0])
print(U)
for k in range(1,31):
    U=np.dot(U,A)
    print(U)
```

b) Le programme renvoie :

$U_0 = [0 \ 1 \ 0]$ et $U_1 = [0.5 \ 0 \ 0.5]$, on retrouve bien la loi de X_0 et X_1 .

Puis, $U_2 = [0.25 \ 0.75 \ 0]$, ce qui donne la loi de X_2 , etc...

Enfin, on observe une stabilisation des probabilités quand n est grand :

$U_{29} = [0.40042835 \ 0.39887856 \ 0.20069309]$, $U_{30} = [0.399653460 \ 0.400907270 \ 0.19943928]$.

On peut conjecturer que $\lim_{n \rightarrow +\infty} P(X_n = 1) = 0.4$, $\lim_{n \rightarrow +\infty} P(X_n = 2) = 0.4$ et $\lim_{n \rightarrow +\infty} P(X_n = 3) = 0.2$.

Cela signifie qu'après un grand nombre de déplacements, la puce a :

- 4 chances sur 10 de se trouver en $\boxed{1}$,
- 4 chances sur 10 de se trouver en $\boxed{2}$,
- 2 chances sur 10 de se trouver en $\boxed{3}$.

6)a) $\forall n \in \mathbf{N}$, $X_n(\Omega) \subset \{1, 2, 3\}$ donc $((X_n = 1), (X_n = 2), (X_n = 3))$ est un système complet.

Donc $\forall n \in \mathbf{N}$, $P(X_n = 1) + P(X_n = 2) + P(X_n = 3) = 1$.

En passant à la limite quand $n \rightarrow +\infty$, on déduit :

$$\lim_{n \rightarrow +\infty} P(X_n = 1) + \lim_{n \rightarrow +\infty} P(X_n = 2) + \lim_{n \rightarrow +\infty} P(X_n = 3) = 1.$$

De plus, $\forall i \in \llbracket 1, 3 \rrbracket$, $P(X_n = i) \geq 0$ donne par passage à la limite $\lim_{n \rightarrow +\infty} P(X_n = i) \geq 0$.

Donc Π est une loi de probabilité.

b) Un passage à la limite dans l'égalité $\forall n \in \mathbf{N}$, $U_{n+1} = U_n A$ donne :

$$\lim_{n \rightarrow +\infty} U_{n+1} = \lim_{n \rightarrow +\infty} (U_n A) = \left(\lim_{n \rightarrow +\infty} U_n \right) A, \text{ soit } \Pi = \Pi A \text{ ou } \Pi A = \Pi.$$

Donc Π est un état stable de la chaîne.

c) Récurrence immédiate. Dans l'hérédité, utiliser que $\Pi A = \Pi$.

d) En transposant l'égalité $\Pi A = \Pi$, on obtient : ${}^t(\Pi A) = {}^t\Pi$, c'est-à-dire ${}^t A {}^t\Pi = {}^t\Pi$ ou encore ${}^t A ({}^t\Pi) = 1 ({}^t\Pi)$.

Par ailleurs, ${}^t\Pi$ est un vecteur colonne non nul puisque la somme de ses composantes fait 1.

Donc ${}^t\Pi$ est un vecteur propre de ${}^t A$ associé à la valeur propre 1.

$$\text{e) } {}^t A = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1 \\ 0 & 1/2 & 0 \end{pmatrix} \text{ et } E_1({}^t A) = \text{Vect} \left(\begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix} \right).$$

$$\text{Il existe donc } a \in \mathbf{R} \text{ tel que } {}^t\Pi = a \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 2a \\ 2a \\ a \end{pmatrix}.$$

$$\text{Enfin, } 2a + 2a + a = 1 \text{ donne } a = 1/5. \text{ Donc } {}^t\Pi = \begin{pmatrix} 2/5 \\ 2/5 \\ 1/5 \end{pmatrix}.$$

Finalement, $\Pi = (0.4, 0.4, 0.2)$ et on retrouve les résultats observés en 5)b).

Exercice 2

$$1) A = \begin{pmatrix} 1/2 & 1/4 & 0 & 1/4 \\ 1/4 & 1/2 & 1/4 & 0 \\ 0 & 0 & 1 & 0 \\ 1/4 & 0 & 1/4 & 1/2 \end{pmatrix}.$$

A est stochastique car ses coefficients sont positifs ou nuls et la somme de ses coefficients sur chaque ligne fait 1.

2) programme :

```
A=np.array(([0.5,0.25,0,0.25],[0.25,0.5,0.25,0],[0,0,1,0],[0.25,0,0.25,0.5]))
U=np.array([1,0,0,0])
print(U)
for k in range(1,101):
    U=np.dot(U,A)
    print(U)
```

On obtient : $U_{100} = (6.63e-08 \quad 4.69e-08 \quad 9.99e-01 \quad 4.69e-08)$.

Donc $P(X_{100} = 1) = P(X_{100} = 2) = P(X_{100} = 4) \approx 0$ et $P(X_{100} = 3) \approx 1$.

Après un certain temps, la souris est presque sûre de trouver le fromage.

3) Notons $loi = (\underbrace{p_1}_{loi[0]}, \underbrace{p_2}_{loi[1]}, \underbrace{p_3}_{loi[2]}, \underbrace{p_4}_{loi[3]})$.

Comme a est un réel aléatoire de $[0, 1]$, on a les probabilités suivantes :

- $P(a < loi[0]) = loi[0] = p_1$,
- $P(a > loi[0] \text{ et } a < loi[0] + loi[1]) = (loi[0] + loi[1]) - loi[0] = loi[1] = p_2$,
- $P(a > loi[0] + loi[1] \text{ et } a < loi[0] + loi[1] + loi[2]) = (loi[0] + loi[1] + loi[2]) - (loi[0] + loi[1]) = loi[2] = p_3$,
- $P(a > 1 - loi[3]) = 1 - (1 - loi[3]) = loi[3] = p_4$.

Ainsi, la fonction *choix* renvoie les valeurs 1,2,3,4 avec les probabilités respectives p_1, p_2, p_3 et p_4 .

4)a) La commande *markov(15,1)* affiche les positions successives de la souris durant le premier quart d'heure.

On obtient par exemple : [1, 4, 1, 4, 4, 4, 4, 1, 2, 2, 2, 2, 3, 3, 3, 3]

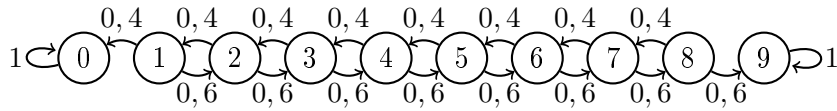
b) La fonction *fromage* renvoie le temps mis par la souris pour trouver le fromage.

c) programme :

```
def temps_moyen(pos_depart):
    S=0
    for j in range(1000):
        S=S+fromage(pos_depart)
    return(S/1000)
```

Exercice 3

1) Graphe probabiliste :



2)a) $p_{0,0} = 1$ et $p_{9,9} = 1$.

b) $\forall i \in \llbracket 1, 8 \rrbracket$, $p_{i,i-1} = 0,4$ et $p_{i,i+1} = 0,6$.

c) Dans les autres cas, $p_{ij} = 0$.

d) programme :

```
import numpy as np
A=np.zeros(shape=(10,10))
A[0,0]=1
A[9,9]=1
for i in range(1,9):
    A[i,i-1]=0.4
    A[i,i+1]=0.6
print(A)
```

3) On complète le programme précédent avec :

```
U=np.array(([0,0,0,1,0,0,0,0,0,0]))
print(U)
for k in range(1,101):
    U=np.dot(U,A)
    print(U)
```

On trouve $P(X_{100} = 0) \approx 0,277$ et $P(X_{100} = 9) \approx 0,722$.

La probabilité que Thésée sorte du labyrinthe en 100 jours au plus est donc de 27,7%.

4)a) Si au n -ième jour, Thésée possède 9 clés, au $n+1$ -ième jour, il possèdera encore 9 clés. Donc $\forall n \in \mathbf{N}$, $(X_n = 9) \subset (X_{n+1} = 9)$.

La suite $(X_n = 9)_{n \in \mathbf{N}}$ est donc croissante pour l'inclusion.

b) Thésée sort du labyrinthe si et seulement si à un jour n donné, il recueille 9 clés.

Le théorème de la limite monotone donne :

$$P(\text{Thésée sort du labyrinthe}) = P\left(\bigcup_{n=0}^{+\infty} (X_n = 9)\right) = \lim_{n \rightarrow +\infty} P(X_n = 9).$$

A l'aide du programme de la question 3), en faisant varier k de 0 à 1000, on obtient :

$$P(X_{1000} = 9) \approx 0,7225 \text{ donc } \lim_{n \rightarrow +\infty} P(X_n = 9) \approx 0,7225.$$

La probabilité pour que Thésée sorte du labyrinthe est d'environ 72,25%.