



PRÉPASIMPLE

prepasimple.fr

EDHEC

Mathématiques — Option économique

2012

Mathématiques appliquées

Mathématiques appliquées — Prépa ECG

Annales de concours · Corrigé

Exercice 1

1) a) On établit l'encadrement par récurrence. À l'indice 0, $u_0 = 0 \in [0, 1[$. Si

$0 \leq u_n < 1$, alors $0 \leq u_n^2 < 1$, donc :

$$\frac{1}{2} \leq u_{n+1} = \frac{u_n^2 + 1}{2} < 1.$$

La propriété est héréditaire. Ainsi $0 \leq u_n < 1$ pour tout $n \in \mathbb{N}$.

1) b) Méthode 1 — Signe de la différence.

Pour tout $n \in \mathbb{N}$:

$$u_{n+1} - u_n = \frac{u_n^2 - 2u_n + 1}{2} = \frac{(1 - u_n)^2}{2} > 0,$$

car $u_n < 1$. La suite est donc strictement croissante.

Méthode 2 — Itération d'une fonction croissante.

La fonction $h : t \mapsto (t^2 + 1)/2$ est strictement croissante sur $[0, 1[$. Comme $u_1 = 1/2 > u_0$, si $u_{n+1} > u_n$, alors les deux termes appartiennent à cet intervalle et $u_{n+2} = h(u_{n+1}) > h(u_n) = u_{n+1}$. La récurrence établit la stricte croissance.

Le calcul de la différence est ici le plus rapide ; la seconde méthode s'applique aux itérations d'une fonction croissante même lorsque la différence se simplifie moins bien.

1) c) La suite est croissante et majorée par 1, donc elle converge vers un réel $\ell \in [0, 1]$.

La fonction h est continue ; le passage à la limite dans $u_{n+1} = h(u_n)$ donne :

$$\ell = \frac{\ell^2 + 1}{2} \iff (\ell - 1)^2 = 0.$$

Par conséquent, $\lim_{n \rightarrow +\infty} u_n = 1$.

- 2) a) D'après la question 1), $v_n = 1 - u_n > 0$ et $v_n \rightarrow 0$. On réécrit d'abord la récurrence :

$$v_{n+1} = 1 - \frac{(1 - v_n)^2 + 1}{2} = v_n - \frac{v_n^2}{2} = v_n \left(1 - \frac{v_n}{2}\right).$$

Les deux termes v_n et v_{n+1} étant strictement positifs, les inverses sont définis. Ainsi :

$$\frac{1}{v_{n+1}} - \frac{1}{v_n} = \frac{v_n - v_{n+1}}{v_n v_{n+1}} = \frac{1}{2(1 - v_n/2)} \rightarrow \frac{1}{2}.$$

- 2) b) On applique le résultat admis à la suite $a_j = 1/v_{j+1} - 1/v_j$, qui converge vers $1/2$.
Le télescopage donne, pour $n \geq 1$:

$$\frac{1}{n} \sum_{j=0}^{n-1} a_j = \frac{1}{n} \left(\frac{1}{v_n} - \frac{1}{v_0} \right) = \frac{1}{nv_n} - \frac{1}{n} \rightarrow \frac{1}{2},$$

car $v_0 = 1$. Donc $1/(nv_n) \rightarrow 1/2$; comme la limite est strictement positive, on peut prendre l'inverse et obtenir $nv_n \rightarrow 2$. Ainsi :

$$v_n \sim \frac{2}{n}.$$

- 2) c) L'équivalence précédente signifie $v_n - 2/n = o(1/n)$. Puisque $u_n = 1 - v_n$, on obtient :

$$u_n = 1 - \frac{2}{n} + o\left(\frac{1}{n}\right).$$

Il s'agit d'une réécriture de l'équivalence de v_n , sans addition d'équivalents.

- 3) a) Code adapté en Python. Question adaptée au programme actuel.

Python

```
def u(n):
    valeur = 0.0
    for k in range(n):
        valeur = (valeur**2 + 1) / 2
    return valeur
```

Le paramètre n est un entier naturel. `range(n)` effectue exactement n itérations. Après k mises à jour, `valeur` représente u_k ; l'initialisation correspond à u_0 . Le résultat renvoyé est donc une approximation numérique de u_n . Pour $n = 0$, aucune itération n'est effectuée et la fonction renvoie 0.

3) b) Code adapté en Python. Question adaptée au programme actuel.

La croissance et la convergence vers 1 garantissent qu'un indice répondant à la condition existe. En examinant les indices dans l'ordre, on obtient le premier d'entre eux :

Python

```
def u(n):
    valeur = 0.0
    for k in range(n):
        valeur = (valeur**2 + 1) / 2
    return valeur

n = 0
while 1 - u(n) >= 10**(-3):
    n = n + 1
print(n)
```

La boucle s'arrête dès que l'inégalité stricte demandée est satisfaite. Le code exécuté affiche 1991. Les valeurs calculées de $1 - u_{1990}$ et $1 - u_{1991}$ sont respectivement environ 0,001000316 et 0,000999816. Ce contrôle numérique concorde avec l'ordre de grandeur fourni par l'équivalent; cet équivalent seul ne déterminerait pas le premier indice exact.

Pour éviter de recalculer tous les termes à chaque appel, on peut conserver le terme courant, ce qui ne change pas l'ordre d'examen des indices :

Python

```
n = 0
valeur = 0.0
while 1 - valeur >= 10**(-3):
    valeur = (valeur**2 + 1) / 2
    n = n + 1
```

```
print(n)
```

Les deux programmes ont été exécutés et fournissent le même indice.

Exercice 2

1) Puisque f est diagonalisable, il existe une base (a_1, a_2, a_3) et des réels $\alpha_1, \alpha_2, \alpha_3$ tels que $f(a_j) = \alpha_j a_j$ pour $j \in \{1, 2, 3\}$. Par linéarité :

$$f^2(a_j) = f(\alpha_j a_j) = \alpha_j^2 a_j.$$

Cette même base est donc une base de vecteurs propres de f^2 . Par définition, f^2 est diagonalisable. La possibilité que deux carrés α_j^2 soient égaux ne change pas cette conclusion : les trois vecteurs restent une base.

2) a) Le produit matriciel donne :

$$A^2 = B = \begin{pmatrix} 1 & -2 & 2 \\ 2 & -3 & 2 \\ 2 & -2 & 1 \end{pmatrix}.$$

Par exemple, la première ligne résulte des produits de $(0, 2, -1)$ par les colonnes de A : $4 - 3 = 1$, $-10 + 8 = -2$ et $8 - 6 = 2$. En multipliant B par lui-même, on obtient :

$$B^2 = \begin{pmatrix} 1 - 4 + 4 & -2 + 6 - 4 & 2 - 4 + 2 \\ 2 - 6 + 4 & -4 + 9 - 4 & 4 - 6 + 2 \\ 2 - 4 + 2 & -4 + 6 - 2 & 4 - 4 + 1 \end{pmatrix} = I.$$

Ainsi $A^4 = I$. Si $AX = \mu X$ avec $X \neq 0$ et $\mu \in \mathbb{R}$, alors $A^4 X = \mu^4 X = X$, d'où $\mu^4 = 1$. Sur $\mathbb{R}$:

$$\mu^4 - 1 = (\mu - 1)(\mu + 1)(\mu^2 + 1) = 0,$$

et $\mu^2 + 1 > 0$. Les seules valeurs propres possibles de A sont donc 1 et -1 . Aucune de ces possibilités n'est encore assurée par cette seule implication.

2) b) Pour $(x, y, z) \in \mathbb{R}^3$, l'équation $(A - I) \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$ s'écrit :

$$\begin{cases} -x + 2y - z = 0, \\ 2x - 6y + 4z = 0, \\ 3x - 8y + 5z = 0. \end{cases}$$

La première équation donne $x = 2y - z$; la deuxième devient $-2y + 2z = 0$, donc $y = z$ et $x = z$. La troisième est alors satisfaite. Réciproquement, tout vecteur (z, z, z) vérifie le système. Ainsi :

$$\text{Ker}(g - \text{Id}) = \text{Vect}(u), \quad u = (1, 1, 1).$$

Comme $u \neq 0$, la famille (u) en est une base et 1 est effectivement une valeur propre.

2) c) L'équation $(A + I) \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$ donne :

$$\begin{cases} x + 2y - z = 0, \\ 2x - 4y + 4z = 0, \\ 3x - 8y + 7z = 0. \end{cases}$$

En substituant $x = -2y + z$, les deux dernières équations deviennent $-8y + 6z = 0$ et $-14y + 10z = 0$. En multipliant respectivement par 7 et 4, puis en soustrayant, on obtient $2z = 0$. Il vient $z = 0$, puis $y = 0$ et $x = 0$.

Par conséquent, $\text{Ker}(g + \text{Id}) = \{0\}$; -1 n'est pas valeur propre.

2) d) Méthode 1 — Dimension des espaces propres.

D'après 2) a) à 2) c), la seule valeur propre de g est 1 et son espace propre est la droite $\text{Vect}(u)$. Il ne peut donc exister une base de $\mathbb{R}^3$ constituée de vecteurs propres. L'endomorphisme g n'est pas diagonalisable.

Méthode 2 — Forme d'une matrice diagonale semblable.

Si A était diagonalisable, elle serait semblable à une matrice diagonale dont tous les coefficients diagonaux seraient égaux à son unique valeur propre 1. Cette matrice serait donc I , d'où $A = PIP^{-1} = I$. Or le coefficient de première ligne et première colonne de A vaut 0; cette égalité est impossible. Ainsi A , et donc g , n'est pas diagonalisable.

La première méthode utilise la dimension d'un espace propre; la seconde est particulièrement utile lorsqu'il n'y a qu'une seule valeur propre.

3) a) Avec la matrice $B = A^2$ calculée précédemment :

$$A^2X = -X \iff (B + I)X = 0.$$

Or les trois lignes de $B + I$ sont toutes égales à $(2, -2, 2)$. Le système se réduit donc à $x - y + z = 0$, c'est-à-dire $x = y - z$. Toutes les solutions sont :

$$(x, y, z) = y(1, 1, 0) + z(-1, 0, 1).$$

En posant $v = (1, 1, 0)$ et $w = (-1, 0, 1)$, on obtient $\text{Ker}(g^2 + \text{Id}) = \text{Vect}(v, w)$. Si $av + bw = 0$, la deuxième coordonnée donne $a = 0$ et la troisième $b = 0$. Ainsi (v, w) est une base de cet espace, de dimension 2.

3) b) Méthode 1 — Vérification directe de la liberté.

Si $au + bv + cw = 0$, alors :

$$\begin{cases} a + b - c = 0, \\ a + b = 0, \\ a + c = 0. \end{cases}$$

Les deux dernières équations donnent $b = c = -a$; la première donne alors $a = 0$, puis $b = c = 0$. La famille est libre et contient trois vecteurs dans $\mathbb{R}^3$: c'est une base.

Méthode 2 — Espaces propres de g^2 .

Puisque $g(u) = u$, on a $g^2(u) = u$. D'autre part, $g^2(v) = -v$ et $g^2(w) = -w$. Les familles (u) et (v, w) sont libres et appartiennent aux espaces propres de g^2 pour les valeurs propres distinctes 1 et -1 . Leur concaténation est donc libre, d'après la propriété d'indépendance des espaces propres associés à des valeurs propres distinctes. Elle comporte trois vecteurs : c'est une base de $\mathbb{R}^3$.

La vérification directe ne suppose aucun résultat spectral supplémentaire ; la seconde méthode exploite la structure des espaces propres déjà trouvés.

3) c) Les images calculées donnent directement :

$$\text{Mat}_{(u,v,w)}(g^2) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}.$$

Ainsi g^2 est diagonalisable, tandis que g ne l'est pas. La réciproque de l'assertion de la question 1) est donc fausse.

Exercice 3

- 1) a) Question adaptée au programme actuel : l'indépendance des lancers est explicitée dans le préambule.

Si $k = 1$, puisque $n \geq 2$, un arrêt au premier lancer a lieu exactement lorsque le premier résultat est Pile. Donc $P(T_n = 1) = p$.

Pour $2 \leq k \leq n - 1$, l'événement $[T_n = k]$ correspond à $k - 1$ résultats Face suivis de Pile. L'indépendance des lancers donne $P(T_n = k) = q^{k-1}p$. Ainsi la formule valable pour tout $1 \leq k \leq n - 1$ est :

$$P(T_n = k) = pq^{k-1}.$$

- 1) b) Pour effectuer le n -ième lancer, il faut et il suffit d'avoir obtenu Face aux $n - 1$ premiers lancers. Le dernier résultat peut être Pile ou Face et entraîne nécessairement l'arrêt. Ainsi :

$$P(T_n = n) = q^{n-1}.$$

On peut également décomposer cet événement en deux événements disjoints, de probabilités $q^{n-1}p$ et q^n , dont la somme vaut $q^{n-1}(p + q) = q^{n-1}$.

- 1) c) Puisque $0 < q < 1$ et $p = 1 - q$, la somme géométrique donne :

$$\sum_{k=1}^n P(T_n = k) = p \sum_{k=1}^{n-1} q^{k-1} + q^{n-1} = p \frac{1 - q^{n-1}}{1 - q} + q^{n-1} = 1.$$

Les probabilités sont positives et leur somme vaut bien 1.

- 1) d) La variable T_n prend ses valeurs dans l'ensemble fini $\{1, \dots, n\}$; elle admet donc une espérance.

Méthode 1 — Somme finie dérivée.

À partir de

$$\sum_{k=0}^{n-1} q^k = (1 - q^n)/(1 - q),$$

la dérivation de cette identité polynomiale, pour $q \neq 1$, donne :

$$\sum_{k=1}^{n-1} kq^{k-1} = \frac{1 - nq^{n-1} + (n-1)q^n}{(1-q)^2}.$$

Par conséquent :

$$E(T_n) = p \sum_{k=1}^{n-1} kq^{k-1} + nq^{n-1} = \frac{1 - nq^{n-1} + (n-1)q^n + nq^{n-1}(1-q)}{1-q} = \frac{1 - q^n}{1-q}.$$

Méthode 2 — Décomposition en indicatrices.

Pour toute valeur entière $t \in \{1, \dots, n\}$, on a

$$t = \sum_{j=1}^n \mathbf{1}_{\{t \geq j\}}.$$

Ainsi

$$T_n = \sum_{j=1}^n \mathbf{1}_{[T_n \geq j]}$$

et la linéarité d'une somme finie donne :

$$E(T_n) = \sum_{j=1}^n P(T_n \geq j) = \sum_{j=1}^n q^{j-1} = \frac{1 - q^n}{1 - q}.$$

En effet, effectuer au moins j lancers signifie avoir obtenu Face aux $j - 1$ premiers. Cette méthode compte les lancers effectivement réalisés ; elle évite le calcul d'une somme pondérée. La première méthode s'appuie directement sur la loi.

- 2) a) La variable X_n ne peut valoir que 0 ou 1, puisque le premier Pile entraîne l'arrêt. L'égalité $X_n = 0$ correspond à n résultats Face, de probabilité q^n . Donc :

$$P(X_n = 0) = q^n, \quad P(X_n = 1) = 1 - q^n.$$

Ainsi X_n suit une loi de Bernoulli de paramètre $1 - q^n$.

- 2) b) La formule de l'espérance d'une Bernoulli, ou le calcul $0 \times q^n + 1 \times (1 - q^n)$, donne :

$$E(X_n) = 1 - q^n.$$

- 3) a) Pour $0 \leq k \leq n - 1$, l'événement $[Y_n = k]$ signifie que les k premiers résultats sont Face et que le suivant est Pile. Pour $k = 0$, il s'agit simplement de Pile au premier lancer. Par indépendance :

$$P(Y_n = k) = q^k p.$$

- 3) b) La valeur $Y_n = n$ est obtenue lorsque les n premiers lancers donnent Face. Ainsi :

$$P(Y_n = n) = q^n.$$

Il n'y a pas d'autre valeur possible. Les probabilités précédentes ont pour somme $p(1 - q^n)/(1 - q) + q^n = 1$.

- 3) c) Chaque lancer donne exactement l'un des deux résultats ; on a donc l'égalité de

variables aléatoires $T_n = X_n + Y_n$. Ces trois variables sont bornées, donc intégrables. Par linéarité :

$$E(Y_n) = E(T_n) - E(X_n) = \frac{1 - q^n}{1 - q} - (1 - q^n) = \frac{q(1 - q^n)}{1 - q}.$$

- 4) Question adaptée au programme actuel : l'indexation commence à $n = 2$, conformément au préambule.

Pour tout entier fixé $k \geq 1$, dès que $n > k$, la question 1) donne $P(T_n = k) = pq^{k-1}$. Pour tout entier $k \leq 0$, $P(T_n = k) = 0$.

La fonction $k \mapsto pq^{k-1}$ sur $\mathbb{N}^*$ définit la loi géométrique de paramètre p , puisque $0 < p < 1$ et

$$\sum_{k=1}^{+\infty} pq^{k-1} = p/(1 - q) = 1.$$

Si T suit cette loi, on a donc, pour tout $k \in \mathbb{Z}$:

$$P(T_n = k) \longrightarrow P(T = k).$$

Les variables étant à valeurs dans $\mathbb{Z}$, le critère ponctuel de convergence des masses donne $T_n \xrightarrow{\mathcal{L}} T$, où T suit la loi géométrique de paramètre p sur $\mathbb{N}^*$.

- 5) Code adapté en Python. Question adaptée au programme actuel. Le paramètre p , déclaré mais non initialisé dans le programme original, est ici saisi avant les tirages.

Python

```
import numpy.random as rd

n = int(input())
p = float(input())

t = 0
x = 0
y = 0
```

```
while x == 0 and t < n:
    t = t + 1
    if rd.random() > p:
        y = y + 1
    else:
        x = 1
print(t, x, y)
```

La variable t compte tous les lancers ; x compte les Pile et y les Face. Pour un tirage uniforme sur $[0, 1[$, l'événement $\text{rd.random}() > p$ a pour probabilité $1 - p = q$: il représente Face. Dans l'autre branche, on obtient Pile et x prend la valeur 1 ; la boucle cesse alors. Elle cesse aussi après n lancers si tous les résultats sont Face. L'invariant $t == x + y$ est préservé à chaque étape.

Ce programme a été exécuté avec des entrées valides. Les parcours donnant immédiatement Pile, donnant n fois Face, et donnant Pile au dernier lancer ont aussi été contrôlés par des suites de tirages fixées. Une simulation ne prouve pas les lois établies aux questions précédentes.

Problème

1) a) Pour tout $x \in \mathbb{R}$:

$$f(-x) = \lambda | -x | e^{-\lambda(-x)^2} = \lambda |x| e^{-\lambda x^2} = f(x).$$

La fonction f est donc paire.

1) b) Sur $[0, +\infty[$, $f(x) = \lambda x e^{-\lambda x^2}$. Pour tout $R \geq 0$, une primitive donne :

$$\int_0^R f(x) dx = \left[-\frac{1}{2} e^{-\lambda x^2} \right]_0^R = \frac{1}{2} (1 - e^{-\lambda R^2}).$$

Comme $\lambda > 0$, l'exponentielle tend vers 0 quand $R \rightarrow +\infty$. L'intégrale converge donc et vaut $1/2$.

- 1) c) La fonction f est continue et positive sur $\mathbb{R}$. Sa parité et la convergence établie en 1) b) donnent la convergence de l'intégrale sur $] -\infty, 0]$ et :

$$\int_{-\infty}^{+\infty} f(x) dx = 2 \int_0^{+\infty} f(x) dx = 1.$$

Elle satisfait donc les conditions permettant d'être une densité de probabilité. Ses limites à gauche et à droite en tout point réel sont finies, conformément au cadre étudié.

- 2) a) Pour $x \geq 1$, par croissance comparée, $\lambda x^2 e^{-\lambda x^2/2} \rightarrow 0$. Il existe donc $R_0 \geq 1$ tel que, pour $x \geq R_0$:

$$0 \leq x f(x) = \lambda x^2 e^{-\lambda x^2} \leq e^{-\lambda x^2/2} \leq e^{-\lambda x/2}.$$

L'intégrale de la dernière fonction sur $[R_0, +\infty[$ converge, car $\lambda/2 > 0$. Par comparaison de fonctions continues positives, $\int_{R_0}^{+\infty} x f(x) dx$ converge. Sur $[0, R_0]$, l'intégrande est continu ; l'intégrale est finie. Cela établit la convergence demandée.

- 2) b) La fonction $|x|f(x)$ est paire et positive. D'après 2) a), son intégrale sur $\mathbb{R}$ converge et vaut $2 \int_0^{+\infty} x f(x) dx$. L'intégrale de $x f(x)$ est donc absolument convergente ; X possède une espérance.

Puisque f est paire, $x f(x)$ est impaire. Les intégrales sur les deux demi-droites sont opposées, ce qui donne :

$$E(X) = \int_{-\infty}^{+\infty} x f(x) dx = 0.$$

La symétrie n'aurait pas suffi sans la vérification préalable de l'intégrabilité absolue.

- 3) a) On effectue l'intégration par parties sur le segment $[0, R]$, avec $u(x) = x^2$ et $v'(x) = \lambda x e^{-\lambda x^2}$, donc $v(x) = -\frac{1}{2} e^{-\lambda x^2}$:

$$\int_0^R x^2 f(x) dx = -\frac{R^2}{2} e^{-\lambda R^2} + \int_0^R x e^{-\lambda x^2} dx = -\frac{R^2}{2} e^{-\lambda R^2} + \frac{1}{2\lambda} (1 - e^{-\lambda R^2}).$$

Par croissance comparée, $R^2 e^{-\lambda R^2} \rightarrow 0$. En passant à la limite, on établit à la fois la convergence et la valeur :

$$\int_0^{+\infty} x^2 f(x) dx = \frac{1}{2\lambda}.$$

3) b) La fonction $x^2 f(x)$ est paire et positive. Son intégrale sur $\mathbb{R}$ est donc finie et vaut $1/\lambda$. Le théorème de transfert donne $E(X^2) = 1/\lambda$. Comme $E(X) = 0$, la formule de Kœnig-Huygens donne :

$$V(X) = E(X^2) - (E(X))^2 = \frac{1}{\lambda}.$$

4) a) Si $y < 0$, alors $F_Y(y) = 0$, puisque $Y = X^2 \geq 0$. Si $y \geq 0$, on a :

$$[Y \leq y] = [-\sqrt{y} \leq X \leq \sqrt{y}].$$

Comme X admet une densité, les probabilités des événements ponctuels sont nulles. Par conséquent :

$$F_Y(y) = \begin{cases} 0 & \text{si } y < 0, \\ F_X(\sqrt{y}) - F_X(-\sqrt{y}) & \text{si } y \geq 0. \end{cases}$$

4) b) Méthode 1 — Dérivation de la fonction de répartition.

Pour $y > 0$, F_X est dérivable aux points $\sqrt{y}$ et $-\sqrt{y}$, car f est continue. La dérivation composée donne :

$$F'_Y(y) = \frac{f(\sqrt{y}) + f(-\sqrt{y})}{2\sqrt{y}} = \frac{2\lambda\sqrt{y} e^{-\lambda y}}{2\sqrt{y}} = \lambda e^{-\lambda y}.$$

Pour $y < 0$, la dérivée est nulle. La continuité en 0 résulte de $F_X(0) - F_X(0) = 0$ et de la continuité de F_X . On peut donc choisir :

$$f_Y(y) = \begin{cases} \lambda e^{-\lambda y} & \text{si } y \geq 0, \\ 0 & \text{si } y < 0. \end{cases}$$

C'est la densité de la loi exponentielle de paramètre λ . La division par $\sqrt{y}$ n'a été effectuée que pour $y > 0$; la valeur de la densité en 0 est choisie séparément.

Méthode 2 — Calcul direct des probabilités d'intervalles.

Pour $y \geq 0$, la parité et le calcul de 1) b) donnent :

$$F_Y(y) = \int_{-\sqrt{y}}^{\sqrt{y}} f(t) dt = 2 \int_0^{\sqrt{y}} \lambda t e^{-\lambda t^2} dt = 1 - e^{-\lambda y}.$$

Avec $F_Y(y) = 0$ pour $y < 0$, on reconnaît directement la répartition exponentielle. La première méthode illustre le transfert par la fonction de répartition; la seconde exploite une primitive déjà disponible.

- 4) c) La loi exponentielle obtenue donne $E(Y) = 1/\lambda$. Puisque $Y = X^2$, on retrouve $E(X^2) = 1/\lambda$. Avec $E(X) = 0$, on conclut à nouveau $V(X) = 1/\lambda$.
- 5) a) Pour $U \in [0, 1[$, $1 - U \in]0, 1]$, donc $W \geq 0$. Pour $t < 0$, $F_W(t) = 0$. Pour $t \geq 0$, les équivalences suivantes respectent le changement de sens produit par la multiplication par $-\lambda < 0$:

$$W \leq t \iff \ln(1 - U) \geq -\lambda t \iff U \leq 1 - e^{-\lambda t}.$$

Le réel $1 - e^{-\lambda t}$ appartient à $[0, 1[$. La répartition uniforme donne donc $F_W(t) = 1 - e^{-\lambda t}$. Ainsi W suit la loi exponentielle de paramètre λ , la même loi que Y .

- 5) b) Code adapté en Python. Question adaptée au programme actuel.

Comme $|X| = \sqrt{Y}$ et que W a la loi de Y , on simule $|X|$ par $\sqrt{W}$:

Python

```
import numpy as np

import numpy.random as rd

def vax(lam):
    u = rd.random()
    w = -np.log(1 - u) / lam
    return np.sqrt(w)
```

Le paramètre lam est strictement positif. Puisque u appartient à $[0, 1[$, l'argument du logarithme est strictement positif et w est positif ou nul. Le résultat possède bien la loi de $|X|$: pour $r \geq 0$, sa probabilité d'être au plus r est $P(W \leq r^2) = 1 - e^{-\lambda r^2} = P(|X| \leq r)$. Pour $r < 0$, les deux probabilités sont nulles. Cette fonction a été exécutée pour plusieurs paramètres positifs.

5) c) La densité est paire et $P(X = 0) = 0$. La question 1) b) donne :

$$P(X > 0) = \int_0^{+\infty} f(t) dt = \frac{1}{2} = P(X < 0).$$

Pour justifier complètement le tirage du signe, prenons une variable R de loi $|X|$ et une variable S , indépendante de R , telle que $P(S = -1) = P(S = 1) = 1/2$. Pour $t \geq 0$:

$$P(SR \leq t) = \frac{1}{2} + \frac{1}{2}P(R \leq t) = \frac{1}{2} + \frac{1}{2}(1 - e^{-\lambda t^2}) = 1 - \frac{1}{2}e^{-\lambda t^2} = F_X(t).$$

La dernière égalité vient de l'intégration de f sur $[t, +\infty[$. Pour $t < 0$, seule la branche $S = -1$ contribue ; comme R n'a pas d'atome :

$$P(SR \leq t) = \frac{1}{2}P(R \geq -t) = \frac{1}{2}e^{-\lambda t^2} = F_X(t),$$

la dernière égalité résultant de la parité de f et de sa primitive sur $[0, +\infty[$. Ainsi SR a la loi de X .

Code adapté en Python. Question adaptée au programme actuel.

Python

```
import numpy as np

import numpy.random as rd

def vax(lam):

    u = rd.random()

    w = -np.log(1 - u) / lam

    return np.sqrt(w)

def x(lam):

    valeur = vax(lam)

    signe = rd.randint(2)

    if signe == 0:

        return -valeur

    else:

        return valeur
```

La valeur absolue et le signe utilisent des tirages distincts modélisés comme indépendants. Chaque signe a probabilité $1/2$; le code a été exécuté pour plusieurs valeurs positives de lam . Les deux branches ont été contrôlées avec des tirages fixés.

- 6) a) Puisque tous les x_k et λ sont strictement positifs, les facteurs de L sont strictement positifs. En posant

$$s = \sum_{k=1}^n x_k > 0,$$

on a :

$$L(\lambda) = \prod_{k=1}^n \lambda e^{-\lambda x_k} = \lambda^n e^{-\lambda s} > 0,$$

puis :

$$\ln(L(\lambda)) = n \ln \lambda - \lambda s = \varphi(\lambda).$$

La fonction est la vraisemblance de l'échantillon indépendant observé. Aucune formule générale supplémentaire sur les estimateurs n'est nécessaire.

6) b) Sur $]0, +\infty[$, la fonction φ est dérivable et :

$$\varphi'(\lambda) = \frac{n}{\lambda} - s = \frac{n - s\lambda}{\lambda}.$$

Comme $s > 0$, cette dérivée est strictement positive pour $0 < \lambda < n/s$, nulle en n/s , et strictement négative pour $\lambda > n/s$. La fonction est donc strictement croissante puis strictement décroissante et possède un unique maximum global en :

$$z = \frac{n}{\sum_{k=1}^n x_k}.$$

Puisque l'exponentielle est strictement croissante et $L = e^\varphi$, ce même réel est l'unique point de maximum de L . Le nombre z est une estimation obtenue à partir des observations ; la variable aléatoire Z_n de la question suivante est l'estimateur correspondant.

7) a) Question adaptée au programme actuel : la famille de densités particulières f_m est explicitée jusqu'à $m = 1$, ce qui inclut le cas $n = 2$; une valeur de Z_n est fixée lorsque le dénominateur s'annule.

Posons

$$S_n = \sum_{k=1}^n Y_k.$$

Les Y_k sont positifs ou nuls et $P(Y_1 = 0) = 0$, donc $[S_n = 0] \subset [Y_1 = 0]$ est de probabilité nulle. Le quotient définit donc Z_n presque sûrement ; la convention du sujet fournit une valeur sur tout l'espace sans modifier sa loi ni son espérance éventuelle.

Résultat admis dans cette version adaptée : S_n admet pour densité f_n , où, pour $t > 0$,

$f_n(t) = \lambda^n t^{n-1} e^{-\lambda t} / (n-1)!$. La famille donnée vérifie $\int_0^{+\infty} f_m(t) dt = 1$ pour $m \geq 1$.

Ce résultat particulier permet le transfert qui suit ; aucune loi supplémentaire n'est invoquée comme connaissance de cours.

Pour $t > 0$ et $n \geq 2$, on vérifie l'identité :

$$\frac{n}{t} f_n(t) = \frac{n\lambda^n}{(n-1)!} t^{n-2} e^{-\lambda t} = \frac{n\lambda}{n-1} f_{n-1}(t).$$

L'intégrande est positif. Comme f_{n-1} est normalisée, cette identité établit la convergence absolue de l'intégrale nécessaire au transfert, y compris pour $n = 2$ où $f_{n-1} = f_1$. La fonction $t \mapsto n/t$ est continue sur $]0, +\infty[$. Le théorème de transfert s'applique donc et donne :

$$E(Z_n) = \int_0^{+\infty} \frac{n}{t} f_n(t) dt = \frac{n\lambda}{n-1} \int_0^{+\infty} f_{n-1}(t) dt = \frac{n}{n-1} \lambda.$$

7) b) Question adaptée au programme actuel : la notion d'estimateur sans biais est définie dans le sujet.

La question précédente fournit une espérance finie. Posons :

$$Z'_n = \frac{n-1}{n} Z_n.$$

Cette variable ne dépend que de l'échantillon et de n , sans utiliser la valeur inconnue de λ ; elle constitue donc un estimateur de λ . Par linéarité :

$$E(Z'_n) = \frac{n-1}{n} E(Z_n) = \frac{n-1}{n} \frac{n}{n-1} \lambda = \lambda.$$

Ainsi Z'_n est sans biais pour toute valeur $\lambda > 0$. Sur l'événement $S_n > 0$, sa formule est $Z'_n = (n-1)/S_n$; sur l'événement nul $S_n = 0$, la convention précédente donne $Z'_n = 0$.

Relevé des adaptations

- Exercice 1, 3) a) et 3) b) : Code adapté en Python ; conservation de la récurrence et de la recherche du premier indice.

- Exercice 3, préambule de 1) : indépendance des lancers explicitée pour justifier les produits de probabilités.
- Exercice 3, 4) : indexation de la suite à partir de $n = 2$, conformément au domaine fixé au début de l'exercice ; les deux premiers termes éventuels n'interviennent pas dans la convergence.
- Exercice 3, 5) : Code adapté en Python, trois instructions à compléter conservées et documentation des commandes fournie ; saisie de p ajoutée, car le programme original utilisait ce paramètre sans l'initialiser.
- Problème, 5) b) et 5) c) : Code adapté en Python, paramètre nommé `lam` et commandes documentées ; indépendance du tirage du signe et de la valeur absolue explicitée pour préserver la loi simulée.
- Problème, 7) a) : famille particulière f_m explicitée pour tout $m \geq 1$, notamment f_1 nécessaire au cas $n = 2$; convention ajoutée pour le quotient sur l'événement de probabilité nulle où la somme est nulle.
- Problème, 7) b) : définition d'un estimateur sans biais fournie, afin que la question repose sur le seul calcul de l'espérance établi juste avant.