

**Oraux**  
corrigés et commentés

**Concours** **BCPST**

**2<sup>e</sup> édition**

**Maths**

**Agro-Véto**  
**G2E**



Vincent Devinck

# Introduction

Ce livre est destiné aux étudiants de classe préparatoire de la filière BCPST préparant les concours Agro-Véto et G2E. Il regroupe des sujets d'oraux correspondants à chacune des deux banques d'épreuves, en tenant compte de leurs particularités respectives. La première partie de l'ouvrage, qui rassemble trente sujets, est consacrée à l'épreuve de « Mathématiques pratiques et Informatique » (concours Agro-Véto). La deuxième partie correspond à l'oral de mathématiques du concours G2E et comporte vingt sujets.

Les sujets ont été choisis de manière à couvrir un large spectre du programme des deux années de classe préparatoire, conformément à ce qui est proposé à chaque session par les différents concours. Pour chaque exercice, un corrigé détaillé est proposé, le plus pédagogique possible, dans le souci d'exposer clairement les idées et les démarches du raisonnement. Les corrigés sont agrémentés de nombreux commentaires dans lesquels on y rappelle notamment des points de cours et de méthodologie. Des remarques issues des différents rapports de jury sont également apportées. On encourage le lecteur à prendre connaissance de ces rapports qui indiquent notamment les erreurs fréquemment commises par les candidats. Ceux-ci sont disponibles sur les sites internet des deux services de concours :

<https://www.concours-agro-veto.net>

et :

<https://concoursg2e.univ-lorraine.fr>

Le lecteur ne tirera profit de ce livre que s'il cherche des solutions personnelles des exercices avant d'en étudier les corrigés. Une bonne connaissance du cours est fondamentale.

Avant de préciser les spécificités de chacun des deux types d'oraux de mathématiques, faisons quelques commentaires généraux. L'oral de mathématiques est un échange entre le candidat et l'examineur : les interrogateurs posent des questions visant à évaluer les compétences du candidat, et cherchent avec bienveillance à l'aider pour compléter son argumentation ou éventuellement se rendre compte d'une erreur. L'avantage d'une telle épreuve pour le candidat est de pouvoir donner oralement de nombreux arguments pour gagner du temps. Les rapports de jury indiquent par exemple que les calculs qui ont pu être réalisés lors de la préparation ne doivent pas forcément être totalement développés au tableau pendant l'interrogation.

Les sujets proposés peuvent être de longueurs ou de difficultés variables. Ces aspects sont pris en compte dans la notation dans un souci d'équité et il est souligné dans les rapports qu'il n'est pas nécessaire d'avoir traité la totalité du ou des exercices pour obtenir

une excellente note. Il est préférable d'avoir mené un raisonnement rigoureux et argumenté, reposant sur des connaissances solides, plutôt que d'avoir survolé les questions sans donner de réelles explications.

Signalons enfin que les calculatrices ne sont pas autorisées.

## L'oral de mathématiques au concours Agro-Véto

L'intitulé de l'épreuve est « Mathématiques pratiques et Informatique ». Comme son nom l'indique, une part de l'évaluation se porte sur l'informatique. Un sujet (appelé « une planche » dans le présent ouvrage) est composé d'une question de cours suivie d'un premier exercice portant sur le programme de mathématiques et d'informatique de première ou de deuxième année de BCPST. Chaque candidat dispose de quarante minutes de préparation dans une salle dédiée où il a accès à un ordinateur. Une clé USB est fournie au début de la préparation sur laquelle il doit enregistrer les programmes relatifs à la partie informatique du sujet. Ensuite, le candidat est amené dans une salle d'interrogation, l'exposé dure également quarante minutes.

La question de cours est écrite sur le sujet donc le candidat en prend connaissance au début de sa préparation. Les seules questions de cours qui sont posées sont des énoncés de définitions ou de théorèmes et il ne s'agit pas de reproduire une démonstration. Signalons que le lecteur peut trouver une liste non exhaustive des questions qui ont été posées au cours de la session 2018 dans le rapport du jury associé (disponible sur le site du concours).

Il est conseillé au candidat de préciser dès le début de l'épreuve les questions qu'il a eu le temps d'aborder pendant la préparation, de sorte que l'examineur gère au mieux le déroulement de l'oral. Le candidat est systématiquement interrogé sur une question d'informatique, qu'elle ait été préparée ou non, afin de vérifier les compétences acquises en informatique.

Les trente premières minutes de l'oral consistent en la résolution de la question de cours (2 minutes) et de l'exercice (28 minutes). Au cours des dix dernières minutes de l'oral, le candidat prend connaissance d'un exercice d'informatique (2 ou 3 questions en général) qu'il doit traiter.

Le lecteur pourra trouver sur le site internet du Service des Concours Agronomiques et Vétérinaires (SCAV) des exemples de sujets proposés lors des différentes sessions d'oraux. Toutes les planches présentes dans ce livre sont des sujets officiels à l'exception des planches numérotées 11, 12, 17, 18, 19, 28 et 30. Dans ce qui suit, les sujets sont regroupés par thèmes : il y a au total huit planches d'algèbre, onze planches d'analyse et onze planches de probabilités.

## L'oral de mathématiques au concours G2E

Le candidat dispose de 20 minutes de préparation, suivies de 20 minutes d'exposé devant l'examineur. Le sujet proposé (appelé « couplage » dans ce livre) comporte toujours deux exercices dont un portant sur les probabilités. Le jury conseille vivement

aux candidats, pendant la préparation, de consacrer le même temps de travail aux deux exercices plutôt que de s'acharner sur le premier et de n'avoir rien à dire sur le second (les deux exercices sont obligatoires).



# CONCOURS AGRO-VÉTO

– MATHÉMATIQUES PRATIQUES ET INFORMATIQUE –

**Temps de préparation :** 40 minutes

**Durée de l'interrogation :** 40 minutes

*Chaque candidat a accès à un ordinateur (sur lequel est installé Pyzo, Spyder, Excel et Geogebra) pendant la préparation et lors de l'exposé.*

*La calculatrice n'est pas autorisée.*



# PLANCHE 1

## Question de cours

À quelle(s) condition(s) sur sa fonction de répartition une variable aléatoire  $X$  admet-elle une densité de probabilité ? Comment détermine-t-on alors une densité de  $X$  ?

★ ★ ★

## Exercice préparé

Soit  $E$  un  $\mathbb{R}$ -espace vectoriel de dimension 3, rapporté à une base  $\mathcal{B} = (e_1, e_2, e_3)$ . Pour tout nombre réel  $a$ , on considère l'endomorphisme  $f_a$  de  $E$  associé, défini par :

$$f_a(e_2) = 0 \quad \text{et} \quad f_a(e_1) = f_a(e_3) = ae_1 + e_2 - ae_3$$

1. Écrire en **python** une fonction **f\_a(x,y,z,a)** qui renvoie les coordonnées dans la base  $\mathcal{B}$  de  $f_a(u)$ , le vecteur  $u$  étant de coordonnées  $(x, y, z)$  dans la base  $\mathcal{B}$ .
2. (a) Déterminer une base de  $\text{Im}(f_a)$ .  
(b) Montrer que  $(e_2, e_1 - e_3)$  est une base de  $\text{Ker}(f_a)$ .
3. Écrire la matrice  $A$  de  $f_a$  relativement à la base  $\mathcal{B}$  et calculer  $A^2$ . En déduire  $f_a \circ f_a$ .
4. On pose  $e'_1 = f_a(e_1)$ ,  $e'_2 = e_1 - e_3$  et  $e'_3 = e_3$ .  
(a) Montrer que la famille  $(e'_1, e'_2, e'_3)$  est une base de  $E$ .  
(b) Déterminer la matrice  $A'$  de  $f_a$  dans cette base.  
(c) En déduire que 0 est la seule valeur propre de  $A$ . La matrice  $A$  est-elle inversible ? diagonalisable ?
5. Pour tout nombre réel  $x$  non nul, on pose  $B(x) = A - xI_3$ , où  $I_3$  désigne la matrice identité de  $\mathcal{M}_3(\mathbb{R})$ .  
(a) Justifier que la matrice  $B(x)$  est inversible pour tout nombre réel  $x$  non nul.  
(b) Exprimer  $(A - xI_3)(A + xI_3)$  puis  $(B(x))^{-1}$  en fonction de  $x$ ,  $I_3$  et  $A$ .  
(c) Pour tout  $n \in \mathbb{N}$ , exprimer  $(B(x))^n$  en fonction de  $x$ ,  $n$ ,  $I_3$  et  $A$ .

★ ★ ★

## Exercice non préparé

1. Écrire une fonction **gene(n)** qui prend en entrée un entier naturel non nul  $n$  et qui retourne une chaîne de  $n$  caractères formée aléatoirement, de façon équiprobable, des caractères 'A', 'C', 'G' et 'T'.
2. Écrire une fonction **nbAC(g)** qui prend en entrée une chaîne de caractères  $g$  formée des caractères 'A', 'C', 'G' et 'T' et qui retourne le nombre de fois où la séquence 'AC' est présente.  
Par exemple, **nb('GAGCACCTACTTGGCGCGA')** retourne 2.

# Un corrigé

## Question de cours

Soit  $X$  une variable aléatoire réelle de fonction de répartition notée  $F$ . Alors  $X$  admet une densité de probabilité si (et seulement si) :

- ★  $F$  est continue sur  $\mathbb{R}$  ;
- ★  $F$  est de classe  $\mathcal{C}^1$  sur  $\mathbb{R}$  sauf éventuellement en un nombre fini de points.

Dans ces conditions, on obtient une densité  $f$  de  $X$  en dérivant  $F$  en chaque point où elle est dérivable et en posant par exemple  $f(x) = 0$  en les autres points.

★ ★ ★

## Exercice préparé

On sait qu'un tel endomorphisme  $f_a$  de  $E$  existe et est unique car la famille  $\mathcal{B}$  est une base de  $E$ .

**1.** Commençons par déterminer l'expression analytique de  $f_a$ . Soit  $u$  un vecteur de  $E$  de coordonnées  $(x, y, z)$  dans la base  $\mathcal{B}$  de  $E$ , c'est-à-dire tel que  $u = xe_1 + ye_2 + ze_3$ . Par linéarité de  $f_a$  et d'après les données de l'énoncé, on a :

$$\begin{aligned} f(u) &= f(xe_1 + ye_2 + ze_3) = xf(e_1) + yf(e_2) + zf(e_3) \\ &= x(ae_1 + e_2 - ae_3) + z(ae_1 + e_2 - ae_3) \\ &= (ax + az)e_1 + (x + z)e_2 + (-ax - az)e_3 \end{aligned}$$

On en déduit la fonction suivante :

```
def f_a(x,y,z,a) :  
    return (a*x+a*z,x+z,-a*x-a*z)
```

## 📌 Commentaire

Cette unique question d'informatique du sujet ne pose pas de problème une fois l'expression analytique de  $f_a$  obtenue.

**2.(a)** Comme  $\mathcal{B}$  est une famille génératrice de  $E$ , on a :

$$\text{Im}(f_a) = \text{Vect}(f_a(e_1), f_a(e_2), f_a(e_3)) = \text{Vect}(ae_1 + e_2 - ae_3)$$

car  $f_a(e_1) = f_a(e_3)$  et  $f_a(e_2) = 0$ . Quelle que soit la valeur de  $a$ , le vecteur  $ae_1 + e_2 - ae_3$  est non nul (car la famille  $(e_1, e_2, e_3)$  est libre et car le nombre réel par lequel est multiplié  $e_2$ , à savoir 1, est non nul). On en déduit donc que la famille génératrice  $(ae_1 + e_2 - ae_3)$  de l'image de  $f_a$  est libre (car constituée d'un unique vecteur non nul). Ainsi :

une base de  $\text{Im}(f_a)$  est  $(ae_1 + e_2 - ae_3)$

### ✧ Commentaires

▷ On rappelle que si  $E$  et  $F$  sont deux espaces vectoriels où  $E$  est de dimension finie et si  $(e_1, \dots, e_p)$  est une famille génératrice de  $E$ , alors pour tout  $f \in \mathcal{L}(E, F)$ , on a l'égalité :

$$\text{Im}(f) = \text{Vect}(f(e_1), \dots, f(e_p))$$

▷ Une famille constituée d'un seul vecteur est libre si et seulement si le vecteur est *non nul*. Une famille constituée de deux vecteurs est libre si et seulement si les deux vecteurs ne sont pas colinéaires.

**2.(b)** Comme l'espace de départ est de dimension finie égale à 3, on sait d'après le théorème du rang que :

$$\dim(E) = \dim(\text{Im}(f)) + \dim(\text{Ker}(f))$$

donc :

$$\dim(\text{Ker}(f)) = \dim(E) - \dim(\text{Im}(f)) = 2$$

d'après la question 2.(a). Or on sait que  $f_a(e_2) = 0$  et, par linéarité de  $f_a$  :

$$f_a(e_1 - e_3) = f_a(e_1) - f_a(e_3) = 0$$

puisque  $f_a(e_1) = f_a(e_3)$ . Ainsi, les vecteurs  $e_2$  et  $e_1 - e_3$  appartiennent au noyau de  $f_a$ . Calculons le rang de la famille  $(e_2, e_1 - e_3)$  (en exprimant les coordonnées des vecteurs dans la base  $\mathcal{B}$  de  $E$ ) :

$$\begin{aligned} \text{rg}(e_2, e_1 - e_3) &= \text{rg} \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \end{matrix} = \text{rg} \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \leftarrow L_1 + L_3 \end{matrix} \\ &= 2 \\ &= \dim(\text{Ker}(f_a)) \end{aligned}$$

La famille  $(e_2, e_1 - e_3)$  est donc libre et composée de deux vecteurs dans l'espace vectoriel  $\text{Ker}(f_a)$  qui est de dimension 2. Ainsi :

une base de  $\text{Ker}(f_a)$  est  $(e_2, e_1 - e_3)$

### ✧ Commentaire

On peut aussi déterminer le noyau de  $f_a$  en utilisant l'expression analytique obtenue à la question 1.

**3.** Comme  $f_a(e_1) = f_a(e_3) = ae_1 + e_2 - ae_3$  et  $f_a(e_2) = 0$  :

la matrice de  $f_a$  exprim  e dans la base  $\mathcal{B}$  de  $E$  est  $A = \begin{pmatrix} a & 0 & a \\ 1 & 0 & 1 \\ -a & 0 & -a \end{pmatrix}$

On a :

$$A^2 = \begin{pmatrix} a & 0 & a \\ 1 & 0 & 1 \\ -a & 0 & -a \end{pmatrix} \begin{pmatrix} a & 0 & a \\ 1 & 0 & 1 \\ -a & 0 & -a \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

donc :

$f_a \circ f_a$  est l'endomorphisme nul de  $E$ , c'est-  dire  $f_a \circ f_a = 0_{\mathcal{L}(E)}$



### Commentaires

▷ Il n'est pas n  cessaire de refaire le calcul au tableau. Il suffit de signaler qu'on trouve la matrice nulle.

▷ On rappelle que si  $E$ ,  $F$  et  $G$  sont des espaces vectoriels de dimensions finies munis de bases not  es  $\mathcal{B}$ ,  $\mathcal{C}$  et  $\mathcal{D}$  respectivement, et si  $f \in \mathcal{L}(E, F)$  et  $g \in \mathcal{L}(F, G)$ , alors :

$$\text{Mat}_{\mathcal{B}, \mathcal{D}}(g \circ f) = \text{Mat}_{\mathcal{C}, \mathcal{D}}(g) \text{Mat}_{\mathcal{B}, \mathcal{C}}(f)$$

Il est souvent plus facile d'effectuer un produit matriciel qu'une composition d'applications. Pour montrer que  $f_a \circ f_a = 0_{\mathcal{L}(E)}$ , il suffit de montrer que  $A^2 = 0_{\mathcal{M}_3(\mathbb{R})}$ .

**4.(a)** Calculons le rang de la famille de vecteurs  $(e'_1, e'_2, e'_3)$  (en exprimant leurs coordonn  es dans la base  $\mathcal{B}$  de  $E$ ) :

$$\begin{aligned} \text{rg}(e'_1, e'_2, e'_3) &= \text{rg} \begin{pmatrix} a & 1 & 0 \\ 1 & 0 & 0 \\ -a & -1 & 1 \end{pmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \end{matrix} = \text{rg} \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ -a & -1 & 1 \end{pmatrix} \begin{matrix} L_1 \leftrightarrow L_2 \\ L_2 \leftrightarrow L_1 \\ L_3 \end{matrix} \\ &= \text{rg} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix} \begin{matrix} L_1 \\ L_2 \leftarrow L_2 - aL_1 \\ L_3 \leftarrow L_3 + aL_1 \end{matrix} \\ &= \text{rg} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \leftarrow L_3 + L_2 \end{matrix} \end{aligned}$$

et donc  $\text{rg}(e'_1, e'_2, e'_3) = 3$ . De plus, cette famille comporte trois vecteurs et l'espace vectoriel  $E$  est de dimension 3 donc :

la famille  $(e'_1, e'_2, e'_3)$  est une base de  $E$



### Commentaire

On ne peut pas échelonner la première matrice avec l'opération  $L_2 \leftarrow a L_2 - L_1$  car celle-ci n'est pas licite pour  $a = 0$ .

**4.(b)** On a :

$$f_a(e'_1) = (f_a \circ f_a)(e_1) = 0_{\mathcal{L}(E)}(e_1) = 0$$

De plus,  $e_1 - e_3 \in \text{Ker}(f_a)$  donc  $f_a(e'_2) = 0$ . Enfin :

$$f_a(e'_3) = f_a(e_3) = f_a(e_1) = e'_1$$

Donc :

la matrice de  $f_a$  dans la base  $(e'_1, e'_2, e'_3)$  de  $E$  est  $A' = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$



### Commentaire

Soient  $E$  et  $F$  deux espaces vectoriels munis de bases :

$$\mathcal{B} = (e_1, \dots, e_p) \quad \text{et} \quad \mathcal{C} = (f_1, \dots, f_n)$$

respectivement. On obtient la matrice d'une application linéaire  $f \in \mathcal{L}(E, F)$  relativement aux bases  $\mathcal{B}$  et  $\mathcal{C}$  en déterminant les coordonnées des vecteurs  $f(e_k)$  (où  $k \in \llbracket 1, n \rrbracket$ ) dans la base  $\mathcal{C}$  de  $F$ .

**4.(c)** Les matrices  $A$  et  $A'$  représentent le même endomorphisme de  $E$  (à savoir  $f_a$ ) dans des bases différentes de  $E$ . Elles ont donc en particulier les mêmes valeurs propres. La matrice  $A'$  est triangulaire (supérieure) donc ses valeurs propres sont les coefficients diagonaux de la matrice. Ainsi :

la seule valeur propre de  $A$  est 0

Comme 0 est valeur propre de  $A$  :

la matrice  $A$  n'est pas inversible

De plus (comme un endomorphisme et une matrice qui le représente ont le même rang) :

$$\text{rg}(A) = \text{rg}(f_a) = \text{rg}(A') = \text{rg} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = 1$$

et donc (d'après le théorème du rang) :

$$\dim(E_0(A)) = \dim(\mathcal{M}_{3,1}(\mathbb{R})) - \text{rg}(A) = 2 \neq 3 \quad (\text{l'ordre de la matrice } A)$$

Ainsi :

la matrice  $A$  n'est pas diagonalisable



### Commentaires

Les résultats suivants, que l'on vient d'utiliser sont très importants.

- ▷ Les valeurs propres d'une matrice triangulaire (supérieure ou inférieure) sont les coefficients diagonaux de la matrice.
- ▷ Une matrice est inversible si et seulement si 0 n'appartient pas au spectre de la matrice.
- ▷ Une matrice est diagonalisable si et seulement si la somme des dimensions de ses sous-espaces propres est égale à l'ordre de la matrice.

**5.(a)** Soit  $x \in \mathbb{R}^*$ . Alors  $x$  n'est pas une valeur propre de  $A$  (sa seule valeur propre étant 0). La matrice  $A - x I_3$  est donc de rang maximal ce qui implique qu'elle est inversible. Ainsi :

pour tout  $x \in \mathbb{R}^*$ , la matrice  $B(x)$  est inversible



### Commentaire

Il n'y a ici aucun calcul à faire par définition d'une valeur propre. En effet, si  $M \in \mathcal{M}_n(\mathbb{K})$ , alors une caractérisation du spectre  $\text{Sp}(M)$  de  $M$  est :

$$\text{Sp}(M) = \{ \lambda \in \mathbb{K} \mid \text{rg}(M - \lambda I_n) < n \}$$

**5.(b)** Soit  $x \in \mathbb{R}^*$ . Par distributivité du produit matriciel par rapport à l'addition (et en utilisant le fait que  $A I_3 = I_3 A = A$ ), on a :

$$(A - x I_3)(A + x I_3) = A^2 - x^2 I_3 = -x^2 I_3$$

car on a vu à la question 3. que  $A^2 = 0_{\mathcal{M}_3(\mathbb{R})}$ . Comme  $-x^2 \neq 0$ , il vient :

$$-\frac{1}{x^2}(A - x I_3)(A + x I_3) = I_3 \quad \text{soit encore} \quad B(x)(-x^{-2}A - x^{-2}I_3) = I_3$$

Or on a aussi  $(-x^{-2}A - x^{-2}I_3)B(x) = I_3$  (car  $A$  commute avec elle-même et avec la matrice identité) donc :

l'inverse de la matrice  $B(x)$  est  $B(x)^{-1} = -x^{-2}A - x^{-2}I_3$

## ✎ Commentaire

On rappelle qu'une matrice  $M \in \mathcal{M}_n(\mathbb{K})$  est inversible si (et seulement si) :

$$\exists N \in \mathcal{M}_n(\mathbb{K}), \quad MN = NM = I_n \quad (1)$$

Le polynôme :

$$P = (X - x)(X + x) + x^2 \in \mathbb{R}[X]$$

est un *polynôme annulateur* de  $A$ , ce qui signifie que  $P(A) = 0_{\mathcal{M}_3(\mathbb{R})}$ . Comme celui-ci a un coefficient constant non nul, on peut *isoler la matrice identité* dans la relation  $P(A) = 0_{\mathcal{M}_3(\mathbb{R})}$  afin de revenir à la définition (1) de l'inversibilité d'une matrice.

**5.(c)** Soient  $(x, n) \in \mathbb{R}^* \times \mathbb{N}$ . On a  $B(x) = A - x I_3$ . Les matrices  $A$  et  $-x I_3$  commutent car  $-x I_3$  est une matrice scalaire. On peut donc appliquer la formule du binôme de Newton et on a :

$$B(x)^n = \sum_{k=0}^n \binom{n}{k} A^k (-x I_3)^{n-k} = \sum_{k=0}^n \binom{n}{k} (-x)^{n-k} A^k$$

Or on sait que  $A^2 = 0_{\mathcal{M}_3(\mathbb{R})}$  donc, pour tout entier  $k \geq 2$ , on a (puisque  $k - 2 \in \mathbb{N}$ ) :

$$A^k = A^2 A^{k-2} = 0_{\mathcal{M}_3(\mathbb{R})} A^{k-2} = 0_{\mathcal{M}_3(\mathbb{R})}$$

En utilisant la relation de Chasles, on obtient pour  $n \geq 2$  :

$$B(x)^n = (-x)^n I_3 + n(-x)^{n-1} A + \underbrace{\sum_{k=2}^n \binom{n}{k} (-x)^{n-k} A^k}_{=0_{\mathcal{M}_3(\mathbb{R})}}$$

Finalement :

$$\forall (x, n) \in \mathbb{R}^* \times \mathbb{N}, \quad B(x)^n = (-x)^n I_3 + n(-x)^{n-1} A$$

car l'expression de  $B(x)^n$  obtenue reste valable pour  $n \in \{0, 1\}$  (en effet, on retrouve bien que  $B(x)^0 = I_3$  et  $B(x)^1 = B(x) = A - x I_3$ ).

## ✎ Commentaire

Pour appliquer la formule du binôme de Newton, il ne faut pas oublier de signaler, voire de justifier, que les deux matrices mises en jeu commutent. On rappelle qu'une matrice scalaire, c'est-à-dire de la forme  $\lambda I_n$  où  $\lambda \in \mathbb{K}$ , commute avec toute matrice de  $\mathcal{M}_n(\mathbb{K})$ .

★ ★ ★

## Exercice non préparé

**1.** La méthode `choice`, disponible dans le module `random`, permet de choisir aléatoirement un élément d'une liste prise en argument. On commence donc par créer la liste constituée des quatre caractères 'A', 'C', 'G' et 'T', dans laquelle nous allons choisir aléatoirement `n` fois un élément.

```
import random as rd

def gene(n) :
    L = ['A', 'C', 'G', 'T']
    chaine = ''
    for _ in range(n) :
        chaine += rd.choice(L)
    return chaine
```

### 🔦 Commentaire

On peut aussi utiliser la fonction `randint` qui nous permet ici de choisir aléatoirement un entier dans `[[0, 3]]` ; cet indice détermine le caractère choisi dans `L`.

```
L = ['A', 'C', 'G', 'T']
chaine = ''
for _ in range(n) :
    indice = rd.randint(0,3)
    chaine += L[indice]
```

**2.** On utilise une structure itérative et un compteur `compt` afin de dénombrer la séquence 'AC' dans la chaîne de caractères `g` passée en argument.

```
def nbAC(g) :
    compteur = 0
    n = len(g)
    for k in range(0,n-1) :
        if g[k] == 'A' and g[k+1] == 'C' :
            compteur += 1
    return compteur
```

### 🔦 Commentaires

▷ Il faut être vigilant quant au dépassement d'indice (dans la boucle, la dernière position visitée par `k` est `n-2`).

▷ On peut remplacer la ligne 5 de cette fonction par :

```
if g[k:k+2] == 'AC' :
```

Rappelons que si `Objet` est une liste ou une chaîne de caractères de longueur  $n > 0$  et si  $i$  et  $j$  sont des entiers tels que  $0 \leq i < j \leq n$ , alors `Objet[i:j]` est la sous-liste ou la sous-chaîne de caractères des éléments de `Objet` situés aux positions  $i$  à  $j-1$  incluses.



# PLANCHE 2

## Question de cours

Donner la définition de l'espérance d'une variable aléatoire  $X$  admettant une densité  $f$  continue sur  $\mathbb{R}$ .

★ ★ ★

## Exercice préparé

Un scientifique étudie une population de souris femelles uniquement. Il note les propriétés suivantes :

- chacune des souris donne naissance en moyenne à une femelle pendant sa première année de vie et à 8 femelles pendant sa deuxième année ;
- la probabilité pour qu'une souris femelle survive une deuxième année est de 0,25 et il n'y a aucune chance qu'elle survive au-delà de la deuxième année.

On distingue donc deux catégories de femelles : les jeunes, âgées de moins d'un an et les adultes dont l'âge est compris entre un et deux ans.

Pour tout entier naturel  $n$ , notons respectivement  $j_n$  et  $a_n$  le nombre de jeunes souris femelles et de souris femelles adultes après  $n$  années.

1. Soit  $n \in \mathbb{N}$ . Montrer que les hypothèses ci-dessus peuvent se traduire par le système suivant :

$$\begin{cases} j_{n+1} &= j_n + 8a_n \\ a_{n+1} &= 0,25j_n \end{cases}$$

On représente la population des souris femelles à l'aide du vecteur  $S_n = \begin{pmatrix} j_n \\ a_n \end{pmatrix}$ .  
Expliciter alors une matrice  $L$  telle que pour tout  $n \in \mathbb{N}$ , on ait  $S_{n+1} = LS_n$ .

2. En déduire une expression de  $S_n$  en fonction de  $L$ ,  $S_0$  et  $n$ .
3. (a) Déterminer  $(U_1, U_2)$  une base de vecteurs propres de  $L$ .  
(b) Soient  $\lambda$  et  $\mu$  les coordonnées de  $S_0$  dans la base  $(U_1, U_2)$ . Exprimer  $S_n$  en fonction de  $\lambda$ ,  $\mu$  et  $n$ .
4. On considère que la population initiale est composée de 20 jeunes souris femelles et d'aucune souris adulte femelle.  
(a) Écrire un programme informatique qui permette de retourner les listes  $(j_0, j_1, \dots, j_{10})$  et  $(a_0, a_1, \dots, a_{10})$ .  
(b) Exprimer  $j_n$  et  $a_n$  en fonction de  $n$ .  
(c) Pour tout entier naturel  $n$ , on désigne par  $t_n$  le nombre total de souris femelles après  $n$  années. Justifier que :

$$\forall n \in \mathbb{N}, \quad t_n = 15 \times 2^n + 5 \times (-1)^n$$

- (d) Déterminer la limite de  $\frac{t_{n+1}}{t_n}$  quand  $n$  tend vers  $+\infty$ . Interpréter ce résultat.
- (e) Vers quelle répartition jeunes/adultes semble tendre cette population ?

★ ★ ★

### Exercice non préparé

- Écrire une fonction `somme_cumul` qui prend en entrée une liste  $L$  d'entiers et qui renvoie la liste des sommes cumulées, c'est-à-dire une liste  $M$ , de même longueur que  $L$ , telle que  $M[i] = \sum_{k=0}^i L[k]$  pour tout indice  $i$  de  $L$ .
- On propose à deux joueurs  $A$  et  $F$  la résolution d'une infinité de problèmes numérotés  $0, 1, \dots, n, \dots$ .  
Ils débutent la résolution à l'instant 0 du problème numéro 0. Dès que l'un des joueurs termine la résolution du problème  $k$ , il passe au problème  $k + 1$ .  
Le numéro d'un problème est attribué au joueur qui le résout en premier. En cas de simultanéité de la résolution d'un problème par les deux joueurs, celui-ci n'est pas attribué.  
On souhaite écrire une fonction Python `répartition(duréesA, duréesF)` qui, étant données les deux listes des durées de résolution des  $n$  premiers problèmes par les deux joueurs, renvoie les listes des numéros des problèmes attribués à  $A$  et  $F$  pour les  $n$  premiers problèmes.
  - `répartition([7,1,2,1,2], [4,2,4,3,1])` renvoie `([3,4], [0,1])`. Quel couple renvoie l'exécution de `répartition([3,2,5,1,1,1], [4,2,2,4,4,2])` ?
  - Écrire la fonction recherchée.

# Un corrigé

## Question de cours

Soit  $X$  une variable aléatoire réelle admettant une densité de probabilité  $f$  continue sur  $\mathbb{R}$ . Alors  $X$  admet une espérance si et seulement si l'intégrale  $\int_{-\infty}^{+\infty} xf(x) dx$  converge absolument. Dans ce cas, on définit l'espérance de  $X$ , notée  $E(X)$ , par :

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

★ ★ ★

## Exercice préparé

**1.** Soit  $n \in \mathbb{N}$ .

- ★ Chacune des  $j_n$  jeunes souris de la  $n^e$  génération donne naissance à une souris à la  $(n+1)^e$  génération. Cela donne donc déjà  $j_n$  jeunes souris pour la nouvelle génération. D'autre part, chacune des  $a_n$  souris adultes donne naissance à 8 souris, ce qui comptabilise encore  $8a_n$  jeunes souris pour la  $(n+1)^e$  génération. Comme une souris ne survit pas à plus de deux générations, le nombre de jeunes souris à la  $(n+1)^e$  génération est bien  $j_{n+1} = j_n + 8a_n$ .
- ★ D'après l'énoncé, les souris adultes d'une génération ne peuvent provenir que des jeunes souris de la génération précédente. En moyenne, un quart de ces souris devient adulte donc  $a_{n+1} = \frac{a_n}{4}$ .

Finalement :

$$\forall n \in \mathbb{N}, \quad \begin{cases} j_{n+1} &= j_n + 8a_n \\ a_{n+1} &= 0,25j_n \end{cases}$$

Soit  $n \in \mathbb{N}$ . On a :

$$S_{n+1} = \begin{pmatrix} j_{n+1} \\ a_{n+1} \end{pmatrix} = \begin{pmatrix} j_n + 8a_n \\ 0,25j_n \end{pmatrix} = \begin{pmatrix} 1 & 8 \\ 0,25 & 0 \end{pmatrix} \begin{pmatrix} j_n \\ a_n \end{pmatrix}$$

Ainsi :

$$\text{en posant } L = \begin{pmatrix} 1 & 8 \\ 0,25 & 0 \end{pmatrix} \in \mathcal{M}_2(\mathbb{R}), \text{ on l'égalité } S_{n+1} = LS_n \text{ pour tout } n \in \mathbb{N}$$



### Commentaire

La mise en équation de l'évolution du nombre de souris est immédiate et la reformulation matricielle du système linéaire obtenu est très standard.

**2.** On utilise un raisonnement par récurrence simple. Pour tout entier naturel  $n$ , considérons la proposition  $\mathcal{P}_n$  : «  $S_n = L^n S_0$  ».

★ **Initialisation** : montrons que la proposition  $\mathcal{P}_0$  est vraie. On a :

$$L^0 S_0 = I_2 S_0 = S_0$$

donc la proposition  $\mathcal{P}_0$  est vraie.

★ **Hérédité** : soit  $n \in \mathbb{N}$  tel que la proposition  $\mathcal{P}_n$  soit vraie. Montrons qu'elle entraîne la proposition  $\mathcal{P}_{n+1}$ . On sait d'après la question 1. que  $S_{n+1} = L S_n$ . Par hypothèse de récurrence (et par associativité du produit matriciel), on a alors :

$$S_{n+1} = L(L^n S_0) = (L L^n) S_0 = L^{n+1} S_0$$

La proposition  $\mathcal{P}_{n+1}$  est donc vraie.

★ **Conclusion** : par principe de récurrence (simple), on a donc :

$$\forall n \in \mathbb{N}, \quad S_n = L^n S_0$$



### Commentaires

▷ Les suites géométriques sont réservées aux suites de nombres réels. On ne peut donc pas donner cet argument pour conclure directement que pour tout  $n \in \mathbb{N}$ , on a  $S_n = L^n S_0$ . Il faut l'établir à l'aide d'un raisonnement par récurrence, même si celle-ci est très classique et immédiate. Une bonne partie de cette récurrence doit pouvoir se faire oralement.

▷ Une erreur classique est d'écrire :

$$\ll \forall n \in \mathbb{N}, \quad S_n = S_0 L^n \gg$$

Ceci n'a aucun sens car le produit matriciel figurant à droite de cette égalité est incompatible.

**3.(a)** Commençons par déterminer les valeurs propres de la matrice  $L$ . Soit  $\lambda \in \mathbb{R}$ . Alors :

$$\begin{aligned} \lambda \text{ est valeur propre de } L &\iff \operatorname{rg}(A - \lambda I_2) < 2 \iff \begin{vmatrix} 1 - \lambda & 8 \\ 0,25 & -\lambda \end{vmatrix} = 0 \\ &\iff \lambda^2 - \lambda - 2 = 0 \\ &\iff \lambda = -1 \text{ ou } \lambda = 2 \end{aligned}$$

Les valeurs propres de  $L$  sont donc  $-1$  et  $2$ . Or :

$$L + I_2 = \begin{pmatrix} 2 & 8 \\ 0,25 & 1 \end{pmatrix}$$

et (en notant  $C_1$  et  $C_2$  les colonnes de la matrice) on a  $4C_1 - C_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ . Donc le vecteur

$U_1 = \begin{pmatrix} 4 \\ -1 \end{pmatrix}$  (qui est non nul) est un vecteur propre de  $L$  associé à la valeur propre  $-1$ .

De même :

$$L - 2I_2 = \begin{pmatrix} -1 & 8 \\ 0,25 & -2 \end{pmatrix}$$

et on observe que  $8C_1 + C_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$  donc  $U_2 = \begin{pmatrix} 8 \\ 1 \end{pmatrix}$  (qui est non nul) est un vecteur propre de  $L$  associé à la valeur propre  $2$ .

La famille de vecteurs  $(U_1, U_2)$  est libre dans  $\mathcal{M}_{2,1}(\mathbb{R})$  car  $U_1$  et  $U_2$  ne sont pas colinéaires. La famille comporte de plus deux vecteurs, ce qui coïncide avec la dimension de  $\mathcal{M}_{2,1}(\mathbb{R})$ , d'où :

la famille  $(U_1, U_2)$  est une base de  $\mathcal{M}_{2,1}(\mathbb{R})$  composée de vecteurs propres de  $L$  (associés respectivement aux valeurs propres  $-1$  et  $2$ ) où :

$$U_1 = \begin{pmatrix} 4 \\ -1 \end{pmatrix} \quad \text{et} \quad U_2 = \begin{pmatrix} 8 \\ 1 \end{pmatrix}$$

### ✎ Commentaire

Le calcul standard des sous-espaces propres de la matrice  $L$  est tout aussi immédiat.

**3.(b)** Notons  $P$  la matrice de passage de la base canonique vers la base  $(U_1, U_2)$ . On a l'égalité  $L = PDP^{-1}$  où on a posé  $D = \begin{pmatrix} -1 & 0 \\ 0 & 2 \end{pmatrix}$ .

À l'aide d'un raisonnement par récurrence (simple), montrons que pour tout entier naturel  $n$  la proposition  $\mathcal{P}_n$  : «  $L^n = PD^n P^{-1}$  » est vraie.

★ **Initialisation** : on a

$$PD^0 P^{-1} = PI_2 P^{-1} = PP^{-1} = I_2 = L^0$$

donc la proposition  $\mathcal{P}_0$  est vraie.

★ **Hérédité** : soit  $n \in \mathbb{N}$  tel que la proposition  $\mathcal{P}_n$  soit vraie. Montrons qu'elle entraîne la proposition  $\mathcal{P}_{n+1}$ . D'après l'hypothèse de récurrence (et par associativité du produit matriciel), on a :

$$\begin{aligned} L^{n+1} &= L^n L = (PD^n P^{-1})(PDP^{-1}) = PD^n P^{-1} PDP^{-1} \\ &= PD^n I_2 DP^{-1} \\ &= PD^{n+1} P^{-1} \end{aligned}$$

Ainsi, la proposition  $\mathcal{P}_{n+1}$  est vraie.

★ **Conclusion** : par principe de récurrence simple, on a bien :

$$\forall n \in \mathbb{N}, \quad L^n = PD^nP^{-1}$$

On en déduit que (puisque  $P$  est inversible et d'après la question 2.) :

$$\forall n \in \mathbb{N}, \quad P^{-1}S_n = D^n P^{-1}S_0$$

Ici,  $P^{-1}S_0$  est la matrice des coordonnées du vecteur  $S_0$  exprimé dans la base  $(U_1, U_2)$ , notée  $P^{-1}S_0 = \begin{pmatrix} \lambda \\ \mu \end{pmatrix}$ . On a (puisque la matrice  $D$  est diagonale) :

$$\forall n \in \mathbb{N}, \quad P^{-1}S_n = \begin{pmatrix} (-1)^n & 0 \\ 0 & 2^n \end{pmatrix} \begin{pmatrix} \lambda \\ \mu \end{pmatrix} = \begin{pmatrix} (-1)^n \lambda \\ 2^n \mu \end{pmatrix}$$

Nous avons obtenu la matrice des coordonnées du vecteur  $S_n$  exprimé dans la base  $(U_1, U_2)$ . Finalement :

$$\forall n \in \mathbb{N}, \quad S_n = (-1)^n \lambda U_1 + 2^n \mu U_2 = \begin{pmatrix} 4(-1)^n \lambda + 8 \times 2^n \mu \\ (-1)^{n+1} \lambda + 2^n \mu \end{pmatrix}$$

### 🔦 Commentaires

▷ La récurrence établie ici est très classique.

▷ Soit  $n \in \mathbb{N}$ . La matrice du vecteur  $S_n$  exprimé dans la base canonique de  $\mathcal{M}_{2,1}(\mathbb{R})$  est  $S_n = \begin{pmatrix} j_n \\ a_n \end{pmatrix}$ . Par contre, la matrice de ses coordonnées dans la base  $(U_1, U_2)$  est  $\begin{pmatrix} \lambda \\ \mu \end{pmatrix} = P^{-1}S_n$  par définition même de  $P$  (la matrice  $P^{-1}$  est la matrice de passage de la base  $(U_1, U_2)$  vers la base canonique de  $\mathcal{M}_{2,1}(\mathbb{R})$ ).

▷ Plus généralement, on rappelle que si  $\mathcal{B}$  et  $\mathcal{C}$  sont deux bases d'un même espace vectoriel  $E$  de dimension finie alors, en notant respectivement  $X$  et  $Y$  les matrices des coordonnées d'un vecteur  $x$  de  $E$  dans les bases  $\mathcal{B}$  et  $\mathcal{C}$ , on a l'égalité :

$$Y = P_{\mathcal{C}, \mathcal{B}} X$$

où  $P_{\mathcal{C}, \mathcal{B}}$  désigne la matrice de passage de la base  $\mathcal{C}$  vers la base  $\mathcal{B}$  de  $E$ .

**4.(a)** La fonction suivante prend en entrée le nombre  $n$  de générations :

```
def souris(n) :
    j = 20 #nombre de jeunes au début
    a = 0 #nombre d'adultes au début
    J = [20] #liste des jeunes
    A = [0] #liste des adultes
    for k in range(n) :
        j, a = j+8*a, 0.25*j
        J.append(j)
        A.append(a)
    return J, A
```

Un appel à la fonction `souris` avec la valeur  $n = 10$  fournit les deux listes :

[20,20,60.0,100.0,220.0,420.0,860.0,1700.0,3420.0,6820.0,13660.0] (jeunes)

et :

[0,5.0,5.0,15.0,25.0,55.0,105.0,215.0,425.0,855.0,1705.0] (adultes)

### 🔦 Commentaire

On rappelle que la commande `a, b = b, a` permet de faire une affectation simultanée. L'enchaînement :

```
j = j+8*a
a = 0.25*a
```

dans la boucle `for` ne donne pas le résultat escompté.

**4.(b)** Soit  $n \in \mathbb{N}$ . Déterminons les coordonnées ( $\lambda$  et  $\mu$ ) du vecteur  $S_0 = \begin{pmatrix} 20 \\ 0 \end{pmatrix}$  dans la base  $(U_1, U_2)$ . Par définition de  $U_1$  et  $U_2$ , on a :

$$S_0 = \lambda U_1 + \mu U_2 = \begin{pmatrix} 4\lambda + 8\mu \\ -\lambda + \mu \end{pmatrix} \iff \begin{cases} \lambda + 2\mu = 5 \\ -\lambda + \mu = 0 \end{cases} \\ \iff \lambda = \mu = \frac{5}{3}$$

On applique maintenant le résultat obtenu à la question 3.(b) :

$$\begin{pmatrix} j_n \\ a_n \end{pmatrix} = \begin{pmatrix} \frac{20}{3}(-1)^n + \frac{40}{3}2^n \\ \frac{5}{3}(-1)^{n+1} + \frac{5}{3}2^n \end{pmatrix}$$

et donc (par identification) :

$$\forall n \in \mathbb{N}, \quad j_n = \frac{20}{3}(-1)^n + \frac{40}{3}2^n \quad \text{et} \quad a_n = \frac{5}{3}(-1)^{n+1} + \frac{5}{3}2^n$$

### 🔦 Commentaire

Une autre méthode pour déterminer le terme général des deux suites est de remarquer que la suite  $(j_n)_{n \in \mathbb{N}}$  est récurrente linéaire d'ordre deux. En effet, en exploitant le système obtenu à la question 1., on obtient :

$$\forall n \in \mathbb{N}, \quad j_{n+2} = j_{n+1} + 2j_n$$

**4.(c)** Soit  $n \in \mathbb{N}$ . Le nombre  $t_n$  de souris femelles après  $n$  années est :

$$\begin{aligned} t_n = j_n + a_n &= \frac{20}{3}(-1)^n + \frac{40}{3}2^n + \frac{5}{3}(-1)^{n+1} + \frac{5}{3}2^n \\ &= \left(\frac{20}{3} - \frac{5}{3}\right)(-1)^n + 15 \times 2^n \end{aligned}$$

Finalement :

$$\forall n \in \mathbb{N}, \quad t_n = 15 \times 2^n + 5(-1)^n$$

**4.(d)** Pour tout entier naturel  $n$ , on a  $t_n \neq 0$  et :

$$\frac{t_{n+1}}{t_n} = \frac{30 \times 2^n - 5(-1)^n}{15 \times 2^n + 5(-1)^n} = \frac{30 - 5\left(\frac{-1}{2}\right)^n}{15 + 5\left(\frac{-1}{2}\right)^n}$$

en factorisant par  $2^n$ . Comme  $-\frac{1}{2} \in ]-1, 1[$ , on a  $\lim_{n \rightarrow +\infty} \left(\frac{-1}{2}\right)^n = 0$  et donc :

$$\lim_{n \rightarrow +\infty} \frac{t_{n+1}}{t_n} = 2$$

On peut donc en déduire que :

à long terme, la population de souris femelles tend à doubler d'année en année

### 🔦 Commentaire

Il est important de préciser que la suite géométrique converge vers 0 car la raison (qui vaut ici  $-\frac{1}{2}$ ) appartient à l'intervalle  $] -1, 1[$ .

**4.(e)** Soit  $n \in \mathbb{N}$ . Alors  $a_n \neq 0$  et :

$$\frac{j_n}{a_n} = \frac{20(-1)^n + 40 \times 2^n}{5(-1)^{n+1} + 5 \times 2^n} = \frac{40 + 20\left(-\frac{1}{2}\right)^n}{5 - 5\left(-\frac{1}{2}\right)^n}$$

Comme  $\lim_{n \rightarrow +\infty} \left(\frac{-1}{2}\right)^n = 0$ , on a  $\lim_{n \rightarrow +\infty} \frac{j_n}{a_n} = \frac{40}{5} = 8$ . Ainsi :

la population tend vers une répartition de 8 jeunes pour 1 adulte

## Exercice non préparé

**1.** La variable `s` dans la fonction ci-dessous permet de calculer les sommes cumulées, que l'on ajoute progressivement dans la liste `M` demandée.

```
def somme_cumul(L) :
    n = len(L)
    M = []
    s = 0
    for i in range(n) :
        s += L[i]
        M.append(s)
    return M
```

### 🔦 Commentaire

On peut éviter d'introduire la variable `s` en procédant comme suit :

```
M = [L[0]]
for i in range(1,n) :
    M.append(L[i]+M[i-1])
```

En effet, l'élément en position  $i \in \llbracket 1, n-1 \rrbracket$  de la liste `M` est la somme de la somme cumulée précédente (donc `M[i-1]`) et de l'élément en position `i` dans `L` (à savoir `L[i]`).

**2.(a)** Ce sont les temps cumulés que l'on regarde pour attribuer la résolution d'un problème à un joueur plutôt qu'à l'autre. Les listes des temps de résolution (cumulés) sont :

★ `[3,5,10,11,12,13]` pour le joueur *A* ;

★ et `[4,6,8,12,16,18]` pour le joueur *F*.

On observe donc que le joueur *A* a remporté les épreuves numéros 0, 1, 3, 4 et 5. Quant au joueur *B*, il gagne l'épreuve numéro 2 uniquement. Ainsi :

le couple renvoyé par la fonction est `([0,1,3,4,5], [2])`

**2.(b)** On utilise la fonction `somme_cumul` afin d'obtenir les temps cumulés. En comparant les temps cumulés, on peut ensuite construire les listes `listeA` et `listeF` qui contiennent les problèmes résolus (en premier) par chacun des deux joueurs.

```
def répartition(duréesA, duréesF) :
    durées_cumula = somme_cumul(duréesA)
    durées_cumulF = somme_cumul(duréesF)
    listeA, listeF = [], []
    n = len(duréesA)
    for k in range(n) :
        if durées_cumula[k] < durées_cumulF[k] :
            listeA.append(k)
        if durées_cumula[k] > durées_cumulF[k] :
            listeF.append(k)
    return listeA, listeF
```

### 🚧 Commentaire

Un problème n'est pas attribué si les temps (cumulés) de résolution sont égaux, il ne faut donc pas utiliser l'instruction `else`.

# PLANCHE 3

## Question de cours

Soit  $X$  une variable aléatoire suivant la loi de Poisson de paramètre  $\lambda > 0$ . Expliciter la loi de  $X$ , et donner son espérance et sa variance.

★ ★ ★

## Exercice préparé

Pour les programmes **python**, on pourra utiliser la fonction `linalg.matrix_rank()` du module **numpy**, qui permet de déterminer le rang d'une famille de vecteurs.

Exemple d'utilisation de cette fonction :

```
import numpy as np
V = np.array([[1,2,1] , [2,3,2]])
print(np.linalg.matrix_rank(V))
```

La valeur renvoyée par **python** est alors : 2.

On considère la matrice :

$$A = \begin{pmatrix} -4 & -3 & -3 \\ 0 & 2 & 0 \\ 6 & 3 & 5 \end{pmatrix} \in \mathcal{M}_3(\mathbb{R})$$

et  $f$  l'endomorphisme de  $\mathbb{R}^3$  représenté dans la base canonique par la matrice  $A$ .

1. (a) Écrire une fonction **python** prenant en arguments deux vecteurs de taille 3 et renvoyant un booléen (**True** ou **False**) indiquant s'ils sont colinéaires.  
*On pourra représenter les vecteurs par des listes.*  
(b) Écrire une fonction **python** prenant en argument un vecteur de taille 3 et renvoyant un booléen indiquant s'il est vecteur propre de  $A$ .
2. (a) Vérifier que les vecteurs  $(1, -2, 0)$ ,  $(0, 1, -1)$  et  $(1, 0, -1)$  sont des vecteurs propres de  $f$  et préciser pour chacun la valeur propre associée.  
(b) L'endomorphisme  $f$  est-il diagonalisable ?
3. (a) Écrire un programme **python** permettant de déterminer le nombre de vecteurs propres de  $A$  dont les coefficients sont des entiers compris entre  $-10$  et  $10$  (bornes incluses).  
(b) Pour un entier naturel  $N$  non nul, calculer le nombre de vecteurs propres de  $A$  dont les coefficients sont des entiers compris entre  $-N$  et  $N$  (bornes incluses).

4. Soit  $N$  un entier naturel non nul. Une expérience consiste à choisir au hasard de manière indépendante  $N$  vecteurs à coefficients entiers dans  $\llbracket -N, N \rrbracket^3$ .
- (a) Quelle est la probabilité  $p_N$  d'obtenir au moins un vecteur propre parmi ces  $N$  vecteurs ?
- (b) Quelle est la limite de  $N \ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right)$  quand  $N$  tend vers  $+\infty$  ?
- En déduire la limite de  $p_N$  quand  $N$  tend vers  $+\infty$ .

★ ★ ★

### Exercice non préparé

- Écrire une fonction `somme(f,a,b,N)` qui retourne  $\sum_{k=0}^{N-1} f\left(a + k \frac{b-a}{N}\right)$  où  $f$  est une fonction,  $a$  et  $b$  deux réels ( $a < b$ ), et  $N$  un entier supérieur à 1.
- Écrire une fonction prenant en entrée une fonction `f`, deux réels `a`, `b` et qui retourne une valeur approchée de  $\int_a^b f(t) dt$ .

Utiliser cette fonction pour donner une valeur approchée de  $\int_0^{+\infty} e^{-t^2} dt$  à l'aide de l'égalité :

$$\int_0^{+\infty} f(t) dt = \int_0^1 \frac{1}{(1-t)^2} f\left(\frac{t}{1-t}\right) dt$$

# Un corrigé

## Question de cours

On dit que  $X$  suit la loi de Poisson de paramètre  $\lambda > 0$  si :

★ l'univers image de  $X$  est  $X(\Omega) = \mathbb{N}$ ;

★ pour tout  $k \in \mathbb{N}$ , on a l'égalité  $P(X = k) = e^{-\lambda} \frac{\lambda^k}{k!}$ .

Si  $X \hookrightarrow \mathcal{P}(\lambda)$ , alors  $X$  admet une espérance et une variance qui valent :

$$E(X) = \lambda \quad \text{et} \quad V(X) = \lambda$$

★ ★ ★

## Exercice préparé

**1.(a)** On utilise la fonction `linalg.matrix_rank()` du module `numpy` suggérée dans l'énoncé. Deux vecteurs  $u$  et  $v$  de  $\mathbb{R}^3$  sont colinéaires si et seulement si la famille  $(u, v)$  est liée, c'est-à-dire si et seulement si le rang de la famille  $(u, v)$  n'est pas maximal (soit encore  $\text{rg}(u, v) \leq 1$ ).

```
import numpy as np
def colineaires(u,v) :
    M = np.array([u,v])
    r = np.linalg.matrix_rank(M)
    if (r == 2) : #la famille (u,v) est libre
        return False
    return True
```

**1.(b)** Soit  $U \in \mathcal{M}_{3,1}(\mathbb{C})$  tel que  $U \neq 0_{\mathcal{M}_{3,1}(\mathbb{C})}$ . On sait que  $U$  est un vecteur propre de la matrice  $A$  si et seulement s'il existe  $\lambda \in \mathbb{C}$  tel que  $AU = \lambda U$ , ce qui équivaut à la colinéarité des matrices colonnes  $AU$  et  $U$  (que l'on vérifie à l'aide de la fonction `colineaires` précédente). On écrit une fonction `vecteur_propre` qui calcule dans un premier temps le vecteur correspondant au produit matriciel  $AU$ .

```
def vecteur_propre(u) :
    v = [-4*u[0]-3*u[1]-3*u[2], 2*u[1], 6*u[0]+3*u[1]+5*u[2]]
    if (colineaires(u,v) and u != [0,0,0]) :
        return True
    return False
```

## Commentaires

- ▷ Il ne faut pas oublier qu'un vecteur propre est nécessairement *non nul*.  
 ▷ Indiquons également qu'il est possible de calculer le produit matriciel  $AU$  à l'aide de la commande `np.dot`. On remplace alors la deuxième ligne du programme précédent par :

```
A = np.array([[ -4, -3, -3], [0, 2, 0], [6, 3, 5]])
v = np.dot(A,u)
```

**2.(a)** Tout d'abord, les trois vecteurs proposés sont non nuls. De plus :

$$A \begin{pmatrix} 1 \\ -2 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ -4 \\ 0 \end{pmatrix} \quad \text{et} \quad A \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ -2 \end{pmatrix} \quad \text{et} \quad A \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$$

et donc (puisque  $A$  est la matrice de  $f$  exprimée dans la base canonique de  $\mathbb{R}^3$ ) :

$$f(1, -2, 0) = 2(1, -2, 0) \quad \text{et} \quad f(0, 1, -1) = 2(0, 1, -1) \quad \text{et} \quad f(1, 0, -1) = -(1, 0, -1)$$

Ainsi :

les vecteurs  $(1, -2, 0)$  et  $(0, 1, -1)$  sont des vecteurs propres de  $f$  associés à la valeur propre 2 tandis que  $(1, 0, -1)$  est un vecteur propre de  $f$  associé à la valeur propre  $-1$

## Commentaire

Cette question, qui ne pose aucun problème, est surtout l'occasion de contrôler que la fonction établie à la question 1.(b) est juste. Il est d'ailleurs opportun de mettre en œuvre cette fonction ici.

**2.(b)** Posons  $u = (1, -2, 0)$ ,  $v = (0, 1, -1)$  et  $w = (1, 0, -1)$ . En notant respectivement  $E_2(f)$  et  $E_{-1}(f)$  les sous-espaces propres de  $f$  associés aux valeurs propres 2 et  $-1$ , on a les inclusions suivantes (d'après la question 2.(a) et car un sous-espace propre est un espace vectoriel) :

$$\text{Vect}(u, v) \subset E_2(f) \quad \text{et} \quad \text{Vect}(w) \subset E_{-1}(f) \quad (1)$$

Les vecteurs  $u$  et  $v$  ne sont pas colinéaires donc  $\text{rg}(u, v) = 2$  et le vecteur  $w$  est non nul donc  $\text{rg}(w) = 1$ . Les inclusions (1) fournissent donc les inégalités suivantes (en considérant les dimensions) :

$$2 \leq \dim(E_2(f)) \quad \text{et} \quad 1 \leq \dim(E_{-1}(f))$$

Ainsi :

$$3 \leq \dim(E_2(f)) + \dim(E_{-1}(f)) \leq 3$$

car une somme de dimensions des sous-espaces propres de  $f \in \mathcal{L}(\mathbb{R}^3)$  est inférieure ou égale à la dimension de  $\mathbb{R}^3$ . Finalement :

$$\dim(E_2(f)) + \dim(E_{-1}(f)) = 3 = \dim(\mathbb{R}^3)$$

et donc :

l'endomorphisme  $f$  de  $\mathbb{R}^3$  est diagonalisable



### Commentaires

▷ Il n'est pas nécessaire de calculer (par la méthode standard) les sous-espaces propres  $E_2(f)$  et  $E_{-1}(f)$ . On rappelle qu'un endomorphisme  $f$  d'un espace vectoriel  $E$  de dimension  $n \in \mathbb{N}^*$  est diagonalisable si et seulement si la somme des dimensions de ces sous-espaces propres est égale à  $n$ .

▷ Nous pouvons également déduire du raisonnement précédent que le spectre de  $f$  est  $\text{Sp}(f) = \{-1, 2\}$  et que :

$$\dim(E_2(f)) = 2 \quad \text{et} \quad \dim(E_{-1}(f)) = 1$$

**3.(a)** On considère chaque vecteur de  $\mathbb{R}^3$  dont les coordonnées appartiennent à l'intervalle  $\llbracket -10, 10 \rrbracket$ . La fonction `vecteur_propre` permet de tester si le vecteur choisi est un vecteur propre de la matrice  $A$ .

```
nb = 0
for i in range(-10,11) :
    for j in range(-10,11) :
        for k in range(-10,11) :
            if vecteur_propre([i,j,k]) :
                nb = nb+1
print(nb)
```

Cette fonction permet de dénombrer 240 vecteurs propres de  $A$  dans l'ensemble  $\llbracket -10, 10 \rrbracket^3$ .



### Commentaire

Il est inutile de rajouter une condition du type `[i,j,k] != [0,0,0]` car le vecteur nul a déjà été pris en considération dans la fonction `vecteur_propre` de la question 1.(b).

**3.(b)** On a vu à la question 2.(b) que les sous-espaces propres de  $f$  sont :

$$E_2(f) = \text{Vect}((1, -2, 0), (0, 1, -1)) \quad \text{et} \quad E_{-1}(f) = \text{Vect}((1, 0, -1))$$

Soit  $N \in \mathbb{N}^*$ . Notons  $V_N$  l'ensemble des vecteurs propres de  $f$  appartenant à  $\llbracket -N, N \rrbracket^3$ . Notons encore  $V_{2,N}$  l'ensemble des vecteurs propres (non nuls donc) de  $f$  associés à la valeur propre 2 dont les coordonnées appartiennent à  $\llbracket -N, N \rrbracket$ , c'est-à-dire :

$$V_{2,N} = E_2(f) \cap (\llbracket -N, N \rrbracket^3 \setminus \{(0, 0, 0)\})$$

et, de la même manière, posons :

$$V_{-1,N} = E_{-1}(f) \cap (\llbracket -N, N \rrbracket^3 \setminus \{(0, 0, 0)\})$$

On a alors l'égalité  $V_N = V_{2,N} \cup V_{-1,N}$ . On sait de plus que  $E_2(f) \cap E_{-1}(f) = \{0_{\mathbb{R}^3}\}$  et  $V_{2,N} \subset E_2(f)$  et  $V_{-1,N} \subset E_{-1}(f)$  donc :

$$V_{2,N} \cap V_{-1,N} = \{0_{\mathbb{R}^3}\}$$

Par conséquent :

$$\text{card}(V_N) = \text{card}(V_{2,N}) + \text{card}(V_{-1,N})$$

Il reste donc à dénombrer les ensembles  $V_{2,N}$  et  $V_{-1,N}$ . Par définition :

$$\begin{aligned} V_{-1,N} &= \{(\alpha, 0, -\alpha) \mid \alpha \in \mathbb{R}\} \cap (\llbracket -N, N \rrbracket^3 \setminus \{(0, 0, 0)\}) \\ &= \{(\alpha, 0, -\alpha) \mid \alpha \in \llbracket -N, N \rrbracket \setminus \{0\}\} \end{aligned}$$

donc (puisque deux vecteurs du type  $(\alpha, 0, -\alpha)$  et  $(\beta, 0, -\beta)$  sont différents si  $\alpha \neq \beta$ ) :

$$\text{card}(V_{-1,N}) = \text{card}(\llbracket -N, N \rrbracket \setminus \{0\}) = (2N + 1) - 1 = 2N \quad (2)$$

Dénombrons maintenant l'ensemble  $V_{2,N}$ . Tout d'abord :

$$E_2(f) = \{(\alpha, -2\alpha + \beta, -\beta) \mid (\alpha, \beta) \in \mathbb{R}^2\}$$

Ainsi, pour tout  $(\alpha, \beta) \in \mathbb{R}^2$ , on a :

$$\begin{aligned} (\alpha, -2\alpha + \beta, -\beta) \in V_{2,N} &\iff \begin{cases} -N \leq \alpha \leq N \\ -N \leq -2\alpha + \beta \leq N \\ -N \leq -\beta \leq N \\ (\alpha, \beta) \neq (0, 0) \end{cases} \\ &\iff \begin{cases} -N \leq \alpha \leq N \\ -N \leq \beta \leq N \\ 2\alpha - N \leq \beta \leq 2\alpha + N \\ (\alpha, \beta) \neq (0, 0) \end{cases} \\ &\iff \begin{cases} -N \leq \alpha \leq N \\ \max(-N, -N + 2\alpha) \leq \beta \leq \min(N, N + 2\alpha) \\ (\alpha, \beta) \neq (0, 0) \end{cases} \end{aligned}$$

Pour tout  $\alpha \in \llbracket -N, N \rrbracket$ , on pose :

$$W_\alpha = \{(\alpha, -2\alpha + \beta, -\beta) \mid \max(-N, -N + 2\alpha) \leq \beta \leq \min(N, N + 2\alpha) \text{ et } (\alpha, \beta) \neq (0, 0)\}$$

et on a l'égalité :

$$V_{2,N} = \bigcup_{\alpha=-N}^N W_\alpha$$

Les ensembles intervenant dans cette réunion sont deux à deux disjoints donc :

$$\text{card}(V_{2,N}) = \sum_{\alpha=-N}^N \text{card}(W_\alpha) \quad (3)$$

On dénombre les ensembles  $W_\alpha$  en distinguant trois cas (remarquons que la condition  $(\alpha, \beta) \neq (0, 0)$  n'est importante que si  $\alpha = 0$ ). Soit  $\alpha \in \llbracket -N, N \rrbracket$ .

★ **Premier cas :  $\alpha < 0$ .**

Alors :

$$\begin{aligned}\text{card}(W_\alpha) &= \text{card}(\llbracket \max(-N, -N + 2\alpha), \min(N, N + 2\alpha) \rrbracket) \\ &= \text{card}(\llbracket -N, N + 2\alpha \rrbracket) \\ &= 2N + 2\alpha + 1\end{aligned}$$

★ **Deuxième cas :  $\alpha = 0$ .**

On a ici :

$$\text{card}(W_0) = \text{card}(\llbracket -N, N \rrbracket \setminus \{0\}) = 2N$$

★ **Troisième cas :  $\alpha > 0$ .**

On a enfin :

$$\begin{aligned}\text{card}(W_\alpha) &= \text{card}(\llbracket \max(-N, -N + 2\alpha), \min(N, N + 2\alpha) \rrbracket) \\ &= \text{card}(\llbracket -N + 2\alpha, N \rrbracket) \\ &= 2N - 2\alpha + 1\end{aligned}$$

D'après (3) et la relation de Chasles, on a :

$$\begin{aligned}\text{card}(V_{2,N}) &= \sum_{\alpha=-N}^{-1} (2N + 2\alpha + 1) + \sum_{\alpha=1}^N (2N - 2\alpha + 1) + 2N \\ &= 2 \sum_{\ell=1}^N (2N - 2\ell + 1) + 2N\end{aligned}$$

en utilisant le changement d'indice  $\ell = -\alpha$  dans la première somme. Le changement d'indice  $k = N - \ell$  et la linéarité de la somme fournissent :

$$\text{card}(V_{2,N}) = 4 \sum_{k=0}^{N-1} k + 2 \sum_{k=0}^{N-1} 1 + 2N = 2N(N-1) + 4N$$

En reprenant le dénombrement (2) de l'ensemble  $V_{-1,N}$ , il vient finalement :

$$\text{card}(V_N) = 2N(N-1) + 6N = 2N(N+2)$$

On peut donc conclure que :

pour tout entier naturel  $N$  non nul, le nombre de vecteurs propres de  $A$  appartenant à l'ensemble  $\llbracket -N, N \rrbracket^3$  est  $2N(N+2)$



## Commentaires

- ▷ Le plus difficile dans cette question est d'écrire *proprement* les choses afin d'utiliser la formule donnant le cardinal d'une réunion disjointe d'ensembles finis.
- ▷ On retrouve bien le résultat obtenu avec l'outil informatique à la question 3.(a) (240 vecteurs propres dont les coordonnées sont comprises entre  $-10$  et  $10$ ).
- ▷ On rappelle que si  $a$  et  $b$  sont des entiers tels que  $a \leq b$ , alors :

$$\text{card}(\llbracket a, b \rrbracket) = \sum_{k=a}^b 1 = b - a + 1$$

**4.(a)** Soit  $N \in \mathbb{N}^*$ . On répète  $N$  fois la même épreuve de Bernoulli (à savoir choisir un vecteur à coefficients entiers dans  $\llbracket -N, N \rrbracket$ ) dont le succès (correspondant au fait d'être un vecteur propre pour  $A$ ) a pour probabilité (par équiprobabilité) :

$$p = \frac{\text{card}(V_N)}{\text{card}(\llbracket -N, N \rrbracket^3)} = \frac{2N(N+2)}{(2N+1)^3}$$

d'après la question 3.(b). Les épreuves sont de plus indépendantes. Si  $X$  désigne la variable aléatoire correspondant au nombre de succès obtenus, on peut conclure que  $X$  suit la loi binomiale de paramètres  $N$  et  $p$ . La probabilité  $p_N$  d'obtenir au moins un vecteur propre parmi les  $N$  vecteurs choisis est alors :

$$p_N = P(X \geq 1) = 1 - P(X = 0) = 1 - \binom{N}{0} p^0 (1-p)^N$$

c'est-à-dire :

$$p_N = 1 - \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right)^N$$

**4.(b)** On sait que :

$$2N(N+2) \underset{N \rightarrow +\infty}{\sim} 2N^2 \quad \text{et} \quad (2N+1)^3 \underset{N \rightarrow +\infty}{\sim} 8N^3$$

donc (par quotient d'équivalents) :

$$\frac{2N(N+2)}{(2N+1)^3} \underset{N \rightarrow +\infty}{\sim} \frac{1}{4N}$$

En particulier :

$$\lim_{N \rightarrow +\infty} \left( -\frac{2N(N+2)}{(2N+1)^3} \right) = \lim_{N \rightarrow +\infty} -\frac{1}{4N} = 0$$

donc :

$$\ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right) \underset{N \rightarrow +\infty}{\sim} -\frac{2N(N+2)}{(2N+1)^3} \underset{N \rightarrow +\infty}{\sim} -\frac{1}{4N}$$

Par produit, il vient :

$$N \ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right) \underset{N \rightarrow +\infty}{\sim} -\frac{1}{4}$$

Finalement :

$$\lim_{N \rightarrow +\infty} N \ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right) = -\frac{1}{4}$$

D'après la question 4.(a), on a :

$$\forall N \in \mathbb{N}^*, \quad p_N = 1 - e^{N \ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right)}$$

et, par composition des limites :

$$\lim_{N \rightarrow +\infty} e^{N \ln \left( 1 - \frac{2N(N+2)}{(2N+1)^3} \right)} = \lim_{X \rightarrow -\frac{1}{4}} e^X = e^{-\frac{1}{4}}$$

On conclut donc que :

$$\lim_{N \rightarrow +\infty} p_N = 1 - e^{-\frac{1}{4}}$$



### Commentaire

Il était ici suggéré de calculer la limite en écrivant  $1 - p_N$  sous forme exponentielle. On est souvent amené à le faire pour des suites avec une expression de la forme  $u_n^{v_n}$  conduisant notamment à la forme indéterminée «  $1^\infty$  ». Il est par exemple très classique de montrer que :

$$\lim_{n \rightarrow +\infty} \left( 1 + \frac{1}{n} \right)^n = e$$

Ajoutons qu'il n'est pas licite de composer les équivalents. On cherche d'abord la limite de  $v_n \ln(u_n)$  (en utilisant l'équivalent usuel du logarithme) puis on compose les *limites*.

★ ★ ★

## Exercice non préparé

**1.** On utilise la structure itérative classique permettant de calculer une somme.

```
def somme(f,a,b,N) :
    s = 0
    for k in range(N) :
        s += f(a+k*(b-a)/N)
    return s
```



## Commentaire

Ce type de question est un grand classique.

**2.** On utilise le théorème sur les sommes de Riemann. D'après ce théorème, une valeur approchée de  $\int_a^b f(t) dt$  est  $\frac{b-a}{N} \sum_{k=0}^{N-1} f\left(a + k \frac{b-a}{N}\right)$  (pour  $N$  suffisamment grand), si la fonction  $f$  est continue sur le segment  $[a, b]$ .

```
def approximation(f,a,b) :  
    N = 1000  
    return (b-a)/N*somme(f,a,b,N)
```

Le changement de variable  $x = \varphi(t)$ , où :

$$\varphi : \begin{cases} [0, 1[ & \longrightarrow & [0, +\infty[ \\ t & \longmapsto & \frac{t}{1-t} = -1 + \frac{1}{1-t} \end{cases},$$

est de classe  $\mathcal{C}^1$  et est strictement croissante (et bijective). Celui-ci fournit l'égalité annoncée puisque :

$$\forall t \in [0, 1[, \quad \varphi'(t) = \frac{1}{(1-t)^2}$$

On en déduit que (la convergence de l'intégrale de Gauss est connue) :

$$\int_0^{+\infty} e^{-t^2} dt = \int_0^1 \frac{1}{(1-t)^2} \exp\left(-\frac{t^2}{(1-t)^2}\right) dt$$

Considérons maintenant la fonction :

$$f : \begin{cases} [0, 1] & \longrightarrow & \mathbb{R} \\ t & \longmapsto & \begin{cases} \frac{1}{(1-t)^2} \exp\left(-\frac{t^2}{(1-t)^2}\right) & \text{si } t \in [0, 1[ \\ 0 & \text{si } t = 1 \end{cases} \end{cases}$$

La fonction  $f$  est continue sur  $[0, 1]$ . En effet, elle l'est sur  $[0, 1[$  comme produit et composée de fonctions continues et car :

$$\lim_{t \rightarrow 1^-} f(t) = \lim_{t \rightarrow 1^-} \frac{1}{t^2} \times \frac{t^2}{(1-t)^2} \exp\left(-\frac{t^2}{(1-t)^2}\right) = 0$$

car  $\lim_{t \rightarrow 1^-} \frac{t^2}{(1-t)^2} = +\infty$  donc, d'après le théorème de composition des limites (et par croissances comparées),

$$\lim_{t \rightarrow 1^-} \frac{t^2}{(1-t)^2} \exp\left(-\frac{t^2}{(1-t)^2}\right) = \lim_{x \rightarrow +\infty} x e^{-x} = 0$$

La fonction  $f$  est donc également continue en 1. Le programme suivant permet d'obtenir une approximation de l'intégrale de Gauss.

```

from math import exp

def f(t) :
    if t == 1 :
        return 0
    else :
        return 1/(1-t)**2*exp(-t**2/(1-t)**2)

print(approximation(f,0,1))

```

## ✧ Commentaires

▷ On rappelle le résultat lié aux sommes de Riemann. Soient  $a, b \in \mathbb{R}$  tels que  $a < b$  et  $f : [a, b] \rightarrow \mathbb{R}$  une fonction. Pour tout  $N \in \mathbb{N}^*$ , on pose :

$$R_N(f) = \frac{b-a}{N} \sum_{k=0}^{N-1} f\left(a + k \frac{b-a}{N}\right) \quad (\text{somme de Riemann associée à } f)$$

Si la fonction  $f$  est continue sur  $[a, b]$ , alors la suite  $(R_N(f))_{N \geq 1}$  est convergente de limite  $\int_a^b f(x) dx$ .

▷ Le changement de variable permet d'obtenir l'intégrale d'une fonction continue sur un segment, cadre dans lequel s'inscrit le théorème sur les sommes de Riemann (et le programme écrit à la première question).



# PLANCHE 4

## Question de cours

Énoncer l'inégalité de Bienaymé-Tchebychev.

★ ★ ★

## Exercice préparé

1. Soient  $n \in \mathbb{N}^*$  et  $a_1, \dots, a_n$  des nombres réels non nuls.  
Écrire une fonction **python** qui renvoie **True** si pour tout  $i \in \{1, \dots, n\}$ , on a l'inégalité  $a_i^2 \sum_{k=1}^n \frac{1}{a_k^2} \geq 2$  et qui renvoie **False** sinon. La fonction aura pour seul paramètre une liste contenant les réels  $a_1, \dots, a_n$ .
2. On considère les vecteurs  $\vec{u} = (1, 0, 0)$ ,  $\vec{v} = \sqrt{2}(0, 1, 0)$  et  $\vec{w} = \sqrt{2}(0, 0, 1)$  et on note  $\mathcal{P}$  le plan de  $\mathbb{R}^3$  admettant pour équation dans la base canonique :

$$y - z = 0$$

Déterminer les projetés orthogonaux des vecteurs  $\vec{u}$ ,  $\vec{v}$  et  $\vec{w}$  sur le plan  $\mathcal{P}$  et vérifier que ceux-ci ont la même norme.

3. Soit  $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$  la base canonique de  $\mathbb{R}^3$  et  $a_1, a_2, a_3$  trois réels tous non nuls.  
On suppose qu'il existe un plan  $\mathcal{P}$  tel que les projetés orthogonaux des vecteurs  $a_1 \vec{e}_1$ ,  $a_2 \vec{e}_2$  et  $a_3 \vec{e}_3$  sur ce plan aient tous la même norme que l'on notera  $d$ .  
On considère  $(\vec{\varepsilon}_1, \vec{\varepsilon}_2)$  une base orthonormée de  $\mathcal{P}$  et  $\vec{\varepsilon}_3$  un vecteur normal à  $\mathcal{P}$  de norme 1.  
On note  $p$  la projection orthogonale sur le plan  $\mathcal{P}$ .
  - (a) Donner une expression de  $p(\vec{e}_i)$  pour tout  $i \in \{1, 2, 3\}$  à l'aide des vecteurs  $\vec{\varepsilon}_1$  et  $\vec{\varepsilon}_2$ .
  - (b) Montrer que pour tout  $i \in \{1, 2, 3\}$ , on a :

$$\langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2 = \left( \frac{d}{a_i} \right)^2$$

- (c) Montrer que :

$$\sum_{i=1}^3 \frac{1}{a_i^2} = \frac{2}{d^2}$$

- (d) Pour tout  $i \in \{1, 2, 3\}$ , montrer que  $|a_i| \geq d$  puis que :

$$a_i^2 \sum_{k=1}^3 \frac{1}{a_k^2} \geq 2$$

★ ★ ★

**Exercice non préparé**

Un téléphone a la probabilité  $p \in ]0, 1[$  de tomber en panne chaque jour. On dispose de  $N$  téléphones en état de marche du même modèle le premier jour.

1. Écrire une fonction de paramètres  $N$ ,  $p$  et  $n \in \mathbb{N}$  et qui retourne le nombre de téléphones en état de marche après  $n$  journées.
2. Écrire une fonction retournant le numéro  $J$  du premier jour auquel plus aucun téléphone ne fonctionne.

# Un corrigé

## Question de cours

Soit  $X$  une variable aléatoire admettant une espérance et une variance. Alors :

$$\forall \varepsilon > 0, \quad P(|X - E(X)| \geq \varepsilon) \leq \frac{V(X)}{\varepsilon^2}$$

★ ★ ★

## Exercice préparé

**1.** Si l'une des conditions n'est pas respectée, alors l'instruction `return False` provoque la sortie de la fonction. La première partie du programme permet de calculer la somme

$\sum_{k=1}^n \frac{1}{a_k^2}$  qui apparaît dans chacune des  $n$  conditions.

```
def conditions(L) :  
    S = 0 #on commence par calculer la somme  
    for k in range(len(L)) :  
        S += 1/L[k]**2  
    for i in range(len(L)) :  
        if (L[i]**2*S < 2) :  
            return False  
    return True
```

## 🔦 Commentaire

Soulignons le fait que la structure :

```
if (L[i]**2*S < 2) :  
    return False  
else :  
    return True
```

ne convient pas. En effet, la fonction renverrait alors `True` si (par exemple) la première condition (pour  $i=0$ ) est vérifiée et que l'une des suivantes (au moins) ne l'est pas.

**2.** Déterminons une famille génératrice du plan  $\mathcal{P}$ . Soit  $(x, y, z) \in \mathbb{R}^3$ . Alors :

$$\begin{aligned}(x, y, z) \in \mathcal{P} &\iff y - z = 0 \iff y = z \\ &\iff (x, y, z) = (x, y, y) = x(1, 0, 0) + y(0, 1, 1) \\ &\iff (x, y, z) \in \text{Vect}((1, 0, 0), (0, 1, 1))\end{aligned}$$

et donc  $\mathcal{P} = \text{Vect}((1, 0, 0), (0, 1, 1))$ .

On note dans la suite  $p(\vec{y})$  le projeté orthogonal d'un vecteur  $\vec{y}$  de  $\mathbb{R}^3$  sur le plan  $\mathcal{P}$ . Comme  $\vec{u}$  appartient au plan  $\mathcal{P}$ , il est son propre projeté orthogonal, c'est-à-dire :

$$p(\vec{u}) = \vec{u}$$

Posons  $p(\vec{v}) = (x, y, z) \in \mathbb{R}^3$ . On sait que ce vecteur est caractérisé par les conditions  $p(\vec{v}) \in \mathcal{P}$  et  $p(\vec{v}) - \vec{v} \perp \mathcal{P}$ . Comme  $p(\vec{v}) - \vec{v} = (x, y - \sqrt{2}, z)$ , il vient :

$$\begin{cases} p(\vec{v}) \in \mathcal{P} \\ \langle p(\vec{v}) - \vec{v}, (1, 0, 0) \rangle = 0 \\ \langle p(\vec{v}) - \vec{v}, (0, 1, 1) \rangle = 0 \end{cases} \iff \begin{cases} y = z \\ x = 0 \\ y - \sqrt{2} + z = 0 \end{cases} \iff \begin{cases} x = 0 \\ y = z = \frac{\sqrt{2}}{2} \end{cases}$$

Ainsi,  $p(\vec{v}) = \left(0, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$ . Les calculs sont identiques pour déterminer  $p(\vec{w})$  et on obtient aussi que  $p(\vec{w}) = \left(0, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$ . On remarque de plus que les vecteurs  $\vec{u}$  et  $\vec{x} = \left(0, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$  sont de norme 1. Finalement :

le projeté orthogonal de  $\vec{u}$  sur le plan  $\mathcal{P}$  est  $\vec{u}$ , les vecteurs  $\vec{v}$  et  $\vec{w}$  ont tous les deux le même projeté orthogonal (le vecteur  $\vec{x}$ ) et les trois projetés orthogonaux sont de norme 1

### ✎ Commentaires

▷ Soit  $F$  un sous-espace vectoriel de  $\mathbb{R}^n$  dont une famille génératrice est  $(u_1, \dots, u_p)$  (où  $p \in \llbracket 1, n \rrbracket$ ) et soit  $u \in \mathbb{R}^n$ . On rappelle que  $u$  est orthogonal à  $F$  (ce qui se note  $u \in F^\perp$ ) si et seulement si  $u$  est orthogonal à chacun des vecteurs  $u_1, \dots, u_p$ .

▷ On peut également utiliser l'expression de la projection orthogonale sur le plan  $\mathcal{P}$  dont on rappelle ici le résultat. Soit  $n \in \mathbb{N}^*$ . Si  $F$  est un sous-espace vectoriel de  $\mathbb{R}^n$  de dimension  $p \in \llbracket 1, n \rrbracket$  et si  $(u_1, \dots, u_p)$  est une **BASE ORTHONORMÉE** de  $F$ , alors la projection orthogonale  $p$  sur  $F$  est déterminée par :

$$\forall x \in \mathbb{R}^n, \quad p(x) = \sum_{i=1}^p \langle x, u_i \rangle u_i$$

Une base de  $\mathcal{P}$  est ici  $((1, 0, 0), (0, 1, 1))$  (puisque'elle est génératrice de  $\mathcal{P}$  et libre car constituée de deux vecteurs non colinéaires). On remarque de plus que les vecteurs constitutifs de cette base sont orthogonaux et le vecteur  $(1, 0, 0)$  est normé. La norme de  $(0, 1, 1)$  vaut  $\sqrt{2}$  donc une base orthonormée du plan  $\mathcal{P}$  est la famille  $(\vec{u}, \vec{x})$  où :

$$\vec{u} = (1, 0, 0) \quad (\text{défini dans l'énoncé}) \quad \text{et} \quad \vec{x} = \frac{1}{\sqrt{2}}(0, 1, 1)$$

Pour tout vecteur  $\vec{y}$  de  $\mathbb{R}^3$ , on a :

$$p(\vec{y}) = \langle \vec{y}, \vec{u} \rangle \vec{u} + \langle \vec{y}, \vec{x} \rangle \vec{x}$$

De plus  $\langle \vec{v}, \vec{u} \rangle = 0$  et  $\langle \vec{w}, \vec{u} \rangle = 0$  donc :

$$p(\vec{v}) = p(\vec{w}) = \sqrt{2} \times \frac{1}{\sqrt{2}} \vec{x} = \vec{x}$$

**3.(a)** Comme  $p$  est la projection orthogonale sur le plan  $\mathcal{P}$  et puisque  $(\vec{\varepsilon}_1, \vec{\varepsilon}_2)$  est une base orthonormée de  $\mathcal{P}$ , on a :

$$\forall i \in \{1, 2, 3\}, \quad p(\vec{e}_i) = \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle \vec{\varepsilon}_1 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle \vec{\varepsilon}_2$$

### ✎ Commentaire

La famille  $(\vec{\varepsilon}_1, \vec{\varepsilon}_2)$  est une base orthonormée *quelconque* de  $\mathcal{P}$ . Il ne s'agit pas nécessairement des vecteurs  $\vec{u}$  et  $\vec{x}$ .

**3.(b)** Soit  $i \in \{1, 2, 3\}$ . Les vecteurs  $\langle \vec{e}_i, \vec{\varepsilon}_1 \rangle \vec{\varepsilon}_1$  et  $\langle \vec{e}_i, \vec{\varepsilon}_2 \rangle \vec{\varepsilon}_2$  sont orthogonaux (puisque les vecteurs  $\vec{\varepsilon}_1$  et  $\vec{\varepsilon}_2$  le sont) donc, d'après le théorème de Pythagore (et par homogénéité de la norme), on a :

$$\|p(\vec{e}_i)\|^2 = \|\langle \vec{e}_i, \vec{\varepsilon}_1 \rangle \vec{\varepsilon}_1\|^2 + \|\langle \vec{e}_i, \vec{\varepsilon}_2 \rangle \vec{\varepsilon}_2\|^2 = \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 \|\vec{\varepsilon}_1\|^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2 \|\vec{\varepsilon}_2\|^2$$

Or les vecteurs  $\vec{\varepsilon}_1$  et  $\vec{\varepsilon}_2$  sont unitaires donc :

$$\|p(\vec{e}_i)\|^2 = \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2 \quad (1)$$

On sait de plus que le projeté orthogonal du vecteur  $a\vec{e}_i$  est de norme  $d$ , c'est-à-dire  $\|p(a\vec{e}_i)\| = d$ . Par linéarité de  $p$  et par homogénéité de la norme, on a :

$$\|p(a_i \vec{e}_i)\| = \|a_i p(\vec{e}_i)\| = |a_i| \|p(\vec{e}_i)\|$$

En élevant au carré, il vient :

$$d^2 = \|p(a_i \vec{e}_i)\|^2 = |a_i|^2 \|p(\vec{e}_i)\|^2 = a_i^2 \|p(\vec{e}_i)\|^2$$

et donc  $\|p(\vec{e}_i)\|^2 = \frac{d^2}{a_i^2}$  car  $a_i \neq 0$ . En reportant cette égalité dans (1), on obtient bien :

$$\forall i \in \{1, 2, 3\}, \quad \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2 = \left( \frac{d}{a_i} \right)^2$$

### ✎ Commentaire

Soient  $n$  et  $p$  deux entiers naturels non nuls tels que  $p \leq n$  et  $(u_1, \dots, u_p)$  une famille de vecteurs deux à deux orthogonaux de  $\mathbb{R}^n$ . D'après le théorème de Pythagore :

$$\left\| \sum_{i=1}^p u_i \right\|^2 = \sum_{i=1}^p \|u_i\|^2$$

**3.(c)** En sommant les trois égalités obtenues à la question 3.(b), il vient :

$$\sum_{i=1}^3 (\langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2) = \sum_{i=1}^3 \left( \frac{d}{a_i} \right)^2$$

et, par linéarité de la somme :

$$\sum_{i=1}^3 (\langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2) = \sum_{i=1}^3 \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 + \sum_{i=1}^3 \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2$$

Comme  $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$  est la base canonique de  $\mathbb{R}^3$ , les produits scalaires  $\langle \vec{e}_1, \vec{\varepsilon}_1 \rangle$ ,  $\langle \vec{e}_2, \vec{\varepsilon}_1 \rangle$  et  $\langle \vec{e}_3, \vec{\varepsilon}_1 \rangle$  sont les coordonnées du vecteur  $\vec{\varepsilon}_1$  dans la base canonique. Par conséquent (par définition de la norme d'un vecteur) :

$$\sum_{i=1}^3 \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2 = \|\vec{\varepsilon}_1\|^2 = 1$$

et, de la même manière :

$$\sum_{i=1}^3 \langle \vec{e}_i, \vec{\varepsilon}_2 \rangle^2 = \|\vec{\varepsilon}_2\|^2 = 1$$

Ainsi :

$$2 = \sum_{i=1}^3 \left( \frac{d}{a_i} \right)^2 = d^2 \sum_{i=1}^3 \frac{1}{a_i^2}$$

par linéarité de la somme. Remarquons qu'une conséquence de cette égalité est que  $d$  est non nul, d'où :

$$\sum_{i=1}^3 \frac{1}{a_i^2} = \frac{2}{d^2}$$

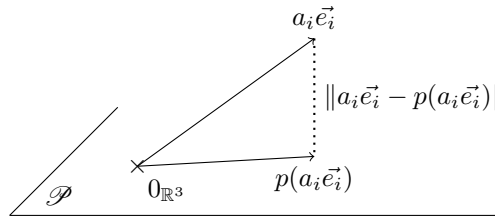
### ✶ Commentaire

Comme  $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$  est une base orthonormée de  $\mathbb{R}^3$ , on peut aussi interpréter la somme  $\sum_{i=1}^3 \langle \vec{e}_i, \vec{\varepsilon}_1 \rangle^2$  comme le carré de la norme de la projection orthogonale du vecteur  $\vec{\varepsilon}_1$  sur  $\mathbb{R}^3$  (cette projection étant  $\vec{\varepsilon}_1$ ).

**3.(d)** Soit  $i \in \{1, 2, 3\}$ . Par définition de  $d$ , on a  $\|p(a_i \vec{e}_i)\| = d$ . Or on sait que :

$$a_i \vec{e}_i = p(a_i \vec{e}_i) + (a_i \vec{e}_i - p(a_i \vec{e}_i)) \quad \text{avec} \quad p(a_i \vec{e}_i) \perp a_i \vec{e}_i - p(a_i \vec{e}_i)$$

par définition de la projection orthogonale  $p$ .



En utilisant le théorème de Pythagore, il vient :

$$\|a_i \vec{e}_i\|^2 = \|p(a_i \vec{e}_i)\|^2 + \|a_i \vec{e}_i - p(a_i \vec{e}_i)\|^2 \geq \underbrace{\|p(a_i \vec{e}_i)\|^2}_{=d^2}$$

car  $\|a_i \vec{e}_i - p(a_i \vec{e}_i)\|^2 \geq 0$ . Or, par homogénéité de la norme :

$$\|a_i \vec{e}_i\|^2 = |a_i|^2 \|\vec{e}_i\|^2 = a_i^2$$

puisque les vecteurs constitutifs de la base canonique de  $\mathbb{R}^3$  sont de norme 1. Par croissance de la fonction racine carrée sur  $\mathbb{R}_+$ , il vient bien (puisque  $d$  est positif) :

$$\boxed{\forall i \in \{1, 2, 3\}, \quad |a_i| \geq d}$$

Soit  $i \in \{1, 2, 3\}$ . Par croissance de la fonction carrée sur  $\mathbb{R}_+$ , on a  $a_i^2 = |a_i|^2 \geq d^2$ .

Comme  $\sum_{k=1}^3 \frac{1}{a_k^2} \geq 0$ , on obtient (par produit) :

$$a_i^2 \sum_{k=1}^3 \frac{1}{a_k^2} \geq d^2 \sum_{k=1}^3 \frac{1}{a_k^2}$$

La somme étant égale à  $\frac{2}{d^2}$  (question 3.(c)), on peut conclure que :

$$\boxed{\forall i \in \{1, 2, 3\}, \quad a_i^2 \sum_{k=1}^3 \frac{1}{a_k^2} \geq 2}$$

★ ★ ★

## Exercice non préparé

**1.** On dispose initialement de  $\mathbf{nb} = N$  téléphones en état de marche. Pour chacun des  $n$  jours, chaque téléphone tombe en panne avec une probabilité égale à  $p$ . Si l'un des  $N$  téléphones tombe en panne, on décrémente la valeur de  $\mathbf{nb}$ .

```

import random as rd

def nb_telephones(N,p,n) :
    nb = N
    for jour in range(1,n+1) :
        nb_bis = nb
        for telephone in range(1,nb_bis+1) :
            if rd.random() <= p :
                nb -= 1
    return nb

```

### 🔦 Commentaire

La variable `nb_bis` est ici importante. On ne peut pas utiliser :

```

for jour in range(1,n+1) :
    for telephone in range(1,nb+1) :
        if rd.random() <= p :
            nb -= 1

```

En effet, on fait ici évoluer le nombre de tours de boucles (déterminé par la variable `nb`) en éliminant des téléphones qui n'ont pas encore été étudiés.

- 2.** Le nombre `nb` de téléphones en état de marche est initialisé à `N` (au jour `J = 0`). On utilise la fonction `nb_telephones` écrit à la question précédente pour faire évoluer le nombre `nb` en conséquence, jour après jour (on choisit donc `n = 1`).

```

def premier_jour(N,p) :
    J = 0
    nb = N
    while nb > 0 :
        J += 1
        nb -= nb_telephones(nb,N,1)
    return J

```

### 🔦 Commentaire

Un script tel que :

```

n = 1
while nb_telephone(N,p,n) > 0 :
    n += 1
return n

```

ne peut pas convenir. En effet, il faut faire évoluer le lot de téléphones jour après jour (en enlevant progressivement les téléphones en panne).

# PLANCHE 5

## Question de cours

Donner la définition d'une application  $f : E \longrightarrow F$  injective.

★ ★ ★

## Exercice préparé

Soit  $(u_n)_{n \in \mathbb{N}}$  une suite définie par la donnée de ses trois premiers termes  $u_0$ ,  $u_1$  et  $u_2$  et par la relation de récurrence :

$$\forall n \in \mathbb{N}, \quad u_{n+3} = \frac{1}{3}(u_n + u_{n+1} + u_{n+2})$$

On pose  $A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \in \mathcal{M}_3(\mathbb{R})$  et  $X_n = \begin{pmatrix} u_n \\ u_{n+1} \\ u_{n+2} \end{pmatrix} \in \mathcal{M}_{3,1}(\mathbb{R})$  pour tout entier naturel  $n$ .

1. Écrire une fonction en **python** prenant en argument les trois premiers termes de la suite  $(u_n)_{n \in \mathbb{N}}$  et renvoyant la liste de ses 100 premiers termes. Utiliser cette fonction pour étudier le comportement asymptotique de la suite sur quelques exemples.
2. Démontrer que 0 n'est pas valeur propre de  $A$ .
3. Démontrer que, pour tout nombre complexe  $\lambda$ , ce nombre complexe est valeur propre de  $A$  si et seulement si  $\lambda$  est solution de l'équation :

$$x^3 - \frac{x^2}{3} - \frac{x}{3} - \frac{1}{3} = 0$$

4. Démontrer qu'il existe une matrice inversible  $P \in \mathcal{M}_3(\mathbb{C})$  et un nombre complexe  $z$  de module strictement inférieur à 1 tels que :

$$A = P \begin{pmatrix} 1 & 0 & 0 \\ 0 & z & 0 \\ 0 & 0 & \bar{z} \end{pmatrix} P^{-1}$$

5. Pour tout entier naturel  $n$ , exprimer  $X_{n+1}$  en fonction de  $A$  et  $X_n$ , et en déduire une expression de  $X_n$  en fonction de  $n$ ,  $A$  et  $X_0$ .
6. Démontrer alors qu'il existe trois nombres complexes  $a$ ,  $b$  et  $c$  (que l'on ne demande pas d'explicitier) tels que :

$$\forall n \in \mathbb{N}, \quad u_n = a + bz^n + c\bar{z}^n$$

7. Démontrer que  $\lim_{n \rightarrow +\infty} |bz^n + c\bar{z}^n| = 0$ .

Que peut-on alors dire de  $\lim_{n \rightarrow +\infty} \operatorname{Re}(bz^n + c\bar{z}^n)$  et de  $\lim_{n \rightarrow +\infty} \operatorname{Im}(bz^n + c\bar{z}^n)$  ?

8. En déduire que  $a \in \mathbb{R}$  et que  $\lim_{n \rightarrow +\infty} u_n = a$ .

★ ★ ★

### Exercice non préparé

Soit  $n \in \mathbb{N}^*$ . Une urne contient deux lots de jetons numérotés de 1 à  $n$ . À chaque tirage, on prélève deux jetons successivement et sans remise. Si les jetons portent le même numéro, on les retire de l'urne, sinon on remet les deux jetons dans l'urne. On effectue les tirages jusqu'à vider l'urne et on note  $T_n$  la variable aléatoire correspondant au nombre de tirages nécessaires pour vider l'urne.

1. Écrire une fonction qui prend en entrée un entier naturel  $n$  non nul et qui renvoie une valeur simulée de  $T_n$ .
2. Émettre une conjecture quant à la valeur de l'espérance de  $T_n$  en utilisant l'outil informatique.