1 Exemple 1

1.1 Question de cours

Donner la définition des fonctions partielles d'une fonction définie sur $\mathbb{R}^2$.

1.2 Exercice préparé

On rappelle que si S et T sont deux variables aléatoires réelles de densités respectives f_S et f_T et indépendantes, alors $S+T$ est une variable aléatoire à densité dont une densité est donnée par la formule de convolution :

$$\forall t \in \mathbb{R},\ f_{S+T}(t) = \int_{-\infty}^{+\infty} f_S(s)\, f_T(t-s)\ \mathrm{d}s. \tag{E}$$

1. Soit U une variable aléatoire de loi uniforme sur $]0,1[$ et λ un réel strictement positif.

On note $X = -\dfrac{1}{\lambda}\ln(1-U)$. Vérifier que X suit une loi exponentielle dont on précisera le paramètre.

2. Écrire un programme Python qui simule une loi exponentielle.
3. On considère une suite $(X_n)_{n\in\mathbb{N}^*}$ de variables aléatoires indépendantes, de même loi que X.
   On définit la suite de variables aléatoires $(S_n)_{n\in\mathbb{N}}$ par : $S_0 = 0 \quad \text{et} \quad \forall n \in \mathbb{N}^*, \quad S_n = X_1 + X_2 + \cdots + X_n$.
   (a) À l'aide d'une récurrence, montrer que la fonction f_n définie ensuite est une densité de probabilité.
   $$\forall n \in \mathbb{N}^*, \quad f_n(t) = \begin{cases} 0 & \text{si } t < 0 \\ \dfrac{\lambda e^{-\lambda t}(\lambda t)^{n-1}}{(n-1)!} & \text{si } t \geqslant 0 \end{cases}$$
   (b) En utilisant la formule de convolution (E) et une récurrence, montrer que f_n est une densité de S_n pour tout $n \geqslant 1$.
4. On suppose qu'à un arrêt, les différences entre les horaires de passage successifs d'un bus sont indépendantes et de même loi exponentielle de paramètre λ.
   On définit un instant $S_0 = 0$, puis on note S_1, S_2, etc les horaires de passage successifs des bus.
   On note alors, pour $t > 0$, N_t le nombre de bus passés à l'arrêt, entre l'instant 0 et l'instant t.
   Autrement dit : $\forall n \geqslant 0$, $[N_t = n] = [S_n \leqslant t < S_{n+1}]$.
   (a) Pour $n \geqslant 0$, exprimer (avec soin) l'événement $[N_t \geqslant n]$ à l'aide de S_n.
   (b) Justifier alors que : $\forall n \geqslant 0$, $P(N_t = n) = P(S_n \leqslant t) - P(S_{n+1} \leqslant t)$.
   (c) En déduire que N_t suit une loi de Poisson de paramètre λt.
5. On suppose plus précisément que les horaires de passages successifs d'un bus sont, en moyenne, de 10 minutes.
   Un individu arrive à l'arrêt à l'instant $T = 100$min pour prendre le bus.

```python
import math as m
import random as rd

def autobus():
  a, b, N = 0, 0, 10000
  for k in range(N):
    s = 0
    while s < 100:
      r = s
      s = s - 10*m.log(1-rd.random())
    u, v = s - 100, s-r
    a, b = a + u, b + v
  return a / N, b / N
```

On se pose alors les deux questions suivantes :
- Combien de temps en moyenne va-t-il attendre le prochain bus ?
- Combien de temps en moyenne s'écoule-t-il entre le prochain bus et le bus qui a précédé ?

Pour y répondre, on réalise le programme Python ci-contre :

(a) Expliquer ce que représentent les variables r, s, u et v dans le programme.

(b) Le programme affiche les valeurs suivantes :
10.062252 20.315494
Pourquoi les valeurs affichées sont-elles paradoxales vis à vis de la situation ?

1.3 Exercice non préparé

1. Écrire une fonction d'argument une liste `L` qui teste si la liste `L` vérifie l'hypothèse $\mathcal{H}$ suivante :
   $\mathcal{H}$: les éléments de la liste `L` sont des entiers qui sont compris au sens large entre 0 et $n-1$ avec n la longueur de la liste `L`.
2. Écrire une fonction d'argument une liste `L` vérifiant l'hypothèse $\mathcal{H}$ qui teste si la liste `L` vérifie l'hypothèse $\mathcal{H}'$ suivante :
   $\mathcal{H}'$: la liste `L` contient exactement une fois chaque valeur entre 0 et $n-1$ où n la longueur de la liste `L`.

2 Exemple 2

2.1 Question de cours

Donner la définition d'un sous-espace vectoriel F d'un espace vectoriel E.

2.2 Exercice préparé

Soit f une fonction de $\mathbb{R}$ dans $\mathbb{R}$. On dit qu'une suite $u = (u_n)_{n\in\mathbb{N}^*}$ est adaptée à f si et seulement si

$$\forall n \in \mathbb{N}^*, \forall x \in \mathbb{R}, \quad u_n f(nx) = \sum_{k=0}^{n-1} f\left(x + \frac{k}{n}\right) \qquad (\star)$$

On admet que si une fonction non identiquement nulle f possède une suite adaptée, alors cette suite est unique. On note E l'ensemble des fonctions de $\mathbb{R}$ dans $\mathbb{R}$ possédant une suite adaptée.

1. Montrer que la suite constante égale à 1 est adaptée à la fonction $x \mapsto x - \frac{1}{2}$.
2. Montrer que si f est une fonction dérivable admettant la suite $(u_n)_{n\in\mathbb{N}^*}$ comme suite adaptée, alors la suite $(nu_n)_{n\in\mathbb{N}^*}$ est adaptée à la fonction f'.
3. On admet dans la suite que l'on peut définir une suite de polynômes $(B_p)_{p\in\mathbb{N}}$ de la façon suivante :

$$\begin{cases} B_0 = 1 \\ \forall p \in \mathbb{N}^*, B_p' = pB_{p-1} \text{ et } \int_0^1 B_p(t)\mathrm{d}t = 0 \end{cases}$$

   L'objectif est alors de montrer que pour tout entier naturel p, B_p appartient à E.

   (a) Écrire une fonction Python qui prend en argument une liste de réels $L = [a_0, a_1, \ldots, a_n]$ et renvoie la valeur de l'intégrale $\displaystyle\int_0^1 \sum_{k=0}^{n} a_k x^k \mathrm{d}x$.

   (b) Calculer B_1 et B_2 et vérifier que B_0 et B_1 appartiennent à E.

   (c) Déterminer, pour tout entier naturel p, le degré et le coefficient dominant de B_p.

   (d) Soit $p \in \mathbb{N}^*$.
   On suppose que B_{p-1} appartient à E et on cherche à montrer que B_p appartient aussi à E.

      i. Montrer que si B_p appartient à E, alors la suite $\left(\frac{1}{n^{p-1}}\right)_{n\in N^*}$ est adaptée à B_p.

      ii. Montrer que la fonction $\varphi : x \mapsto \frac{1}{n^{p-1}} B_p(nx) - \sum\limits_{k=0}^{n-1} B_p\left(x + \frac{k}{n}\right)$ est constante.

      iii. Calculer $\int_0^{1/n} \varphi(x)\mathrm{d}x$.

      iv. Conclure.

2.3 Exercice non préparé

Soit a et b deux entiers naturels avec $a < b$.

1. Écrire une fonction `verif(L,a,b)` qui a pour paramètres une liste `L` d'entiers naturels et 2 entiers `a` et `b` et qui renvoie `True` si tous les éléments de `L` sont dans l'intervalle $\llbracket a, b \rrbracket$ et `False` sinon.
2. Soit `L` une liste dont chaque élément est dans l'intervalle d'entiers $\llbracket a, b \rrbracket$.
   Écrire une fonction `denombre(L,a,b)` qui détermine le nombre d'apparitions de chacune des valeurs possibles (entre a et b) de la liste `L` et qui renvoie le résultat dans une liste dont le premier élément représentera le nombre de a dans `L`, le deuxième le nombre de $a + 1$ dans `L` etc.
   Exemple : pour $a = 0$ et $b = 4$, `denombre([1,3,0,4,1,3,1],0,4)` renverra `[1,3,0,2,1]`.

3 Exemple 3

3.1 Question de cours

Donner la définition d'une densité de probabilité.

3.2 Exercice préparé

Soit $n \in \mathbb{N}^*$. On considère l'espace vectoriel $\mathbb{R}^n$ muni de sa base canonique $\mathcal{B} = (e_1, \dots, e_n)$, de son produit scalaire usuel noté $\langle \cdot, \cdot \rangle$ et de sa norme usuelle notée $\| \cdot \|$.
Soit $p \in [\![1, n]\!]$. Pour toute famille $(u_1, \dots, u_p)$ de vecteurs de $\mathbb{R}^n$, on définit G la matrice de Gram de $(u_1, \dots, u_p)$ par

$$G = \begin{pmatrix} \langle u_1, u_1 \rangle & \langle u_1, u_2 \rangle & \cdots & \langle u_1, u_p \rangle \\ \langle u_2, u_1 \rangle & \langle u_2, u_2 \rangle & \cdots & \langle u_2, u_p \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle u_p, u_1 \rangle & \langle u_p, u_2 \rangle & \cdots & \langle u_p, u_p \rangle \end{pmatrix} \in \mathcal{M}_p(\mathbb{R})$$

1. (a) Écrire une fonction Python `ps` prenant en argument deux vecteurs `u` et `v` sous formes de liste de même taille et renvoyant le produit scalaire $\langle u, v \rangle$.
   (b) Écrire une fonction Python `Gram` prenant en argument une famille de vecteurs $(u_1, \dots, u_p)$ sous forme d'une liste de listes et renvoyant la matrice de Gram de la famille $(u_1, \dots, u_p)$.
   (c) Tester votre fonction avec les vecteurs $u_1 = (1, -1, 0)$, $u_2 = (1, 0, -1)$ et $u_3 = (1, 1, 1)$.
2. Justifier que la matrice de Gram d'une famille de vecteurs de $\mathbb{R}^n$ est diagonalisable.
3. Soit $(u_1, \dots, u_p)$ une famille de $\mathbb{R}^n$ et G sa matrice de Gram. On cherche à montrer que la famille est libre si et seulement si G est inversible.
   (a) On suppose que G est inversible.
   Soient $\alpha_1, \dots, \alpha_p \in \mathbb{R}$ tels que $\displaystyle\sum_{k=1}^{p} \alpha_k u_k = 0$. On pose $X = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{pmatrix} \in \mathcal{M}_{p,1}(\mathbb{R})$.
   Montrer que $GX = 0$, puis en déduire que $(u_1, \dots, u_p)$ est libre.
   (b) On suppose que $(u_1, \dots, u_p)$ est libre.
   Soit $X = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{pmatrix} \in \mathcal{M}_{p,1}(\mathbb{R})$ tel que $GX = 0$.
      i. Montrer que, pour tout $i \in [\![1, p]\!]$, $\left\langle u_i, \displaystyle\sum_{k=1}^{p} \alpha_k u_k \right\rangle = 0$.
      ii. En déduire que $\displaystyle\sum_{k=1}^{p} \alpha_k u_k = 0$.
      iii. Montrer que $X = 0$, puis que G est inversible.
4. Soit $(v_1, \dots, v_n)$ une famille de vecteurs de $\mathbb{R}^n$ telle que

$$\forall i \in [\![1, n]\!],\ \|v_i\| = 1 \quad \text{et} \quad \forall (i, j) \in [\![1, n]\!]^2, i \neq j \Rightarrow \|v_i - v_j\| = 1.$$

   (a) Pour tous a et $b \in \mathbb{R}^n$, montrer que $\|a - b\|^2 = \|a\|^2 + \|b\|^2 - 2 \langle a, b \rangle$.
   (b) En déduire la matrice de Gram de la famille $(v_1, \dots, v_n)$, que l'on notera G.
   (c) On pose $A = 2G$. Exprimer A^2 en fonction de n, A et I_n (la matrice identité de taille n).
   En déduire que A est inversible.
   (d) Montrer que $(v_1, \dots, v_n)$ est une base de $\mathbb{R}^n$.

3.3 Exercice non préparé

1. Écrire une fonction python, d'arguments d'entrée `i` et `N`, qui simule une marche aléatoire sur $\mathbb{Z}$: le marcheur démarre sur l'entier `i` et à chaque pas, il avance de 1 ou recule de 1 avec probabilité 1/2. La marche s'arrête après le N-ième pas et la fonction renvoie l'entier sur lequel le marcheur s'est arrêté.

Un plateau de jeu est constitué de $n+1$ cases numérotées de 0 à n avec $n > 1$, les cases 0 et n contenant des valeurs a et b, comme ci-dessous où $n = 10$, $a = 1$ et $b = 3$:

1										3
0	1	2	3	4	5	6	7	8	9	10

Pour $i \in \{0, 1, \ldots, n\}$, on considère le jeu J_i suivant: on place un pion dans la case de numéro i et tant que l'on n'est pas dans la case 0 ou dans la case n, on lance une pièce équilibrée: si elle tombe sur face, on se déplace vers la gauche ; sinon, on se déplace vers la droite. On admet que la partie se termine presque sûrement : le gain du joueur est la valeur (a ou b) contenue dans la case finale.

2. Écrire une fonction qui prend en paramètres les valeurs n, a, b et i, qui simule ce jeu et qui renvoie le gain.

4 Exemple 4

4.1 Question de cours

Énoncer le théorème d'intégration par parties pour les intégrales sur un segment.

4.2 Exercice préparé

Soit n un entier naturel non-nul.
On dispose de n jetons et de trois urnes numérotées de 1 à 3.
Pour chaque jeton, on choisit une des trois urnes au hasard et avec équiprobablité et on place le jeton dans l'urne choisie. Le placement de chaque jeton est indépendant du placement de tous les autres jetons.
On note X la variable aléatoire égale au nombre de jetons contenus dans l'urne 1 à la fin de l'expérience, et on note Y le nombre d'urnes restées vides à la fin de l'expérience.

1. Ecrire une fonction Python qui prend en argument un entier naturel n non nul, simule l'expérience aléatoire décrite ci-dessus, et renvoie les valeurs de X et de Y obtenues.
2. Dans cette question, $n = 10$. Utiliser la fonction précédente pour simuler un grande nombre de fois l'expérience et obtenir une valeur approchée de $E(XY)$, $E(X)$ et $E(Y)$.
   Que peut-on conjecturer sur la valeur de la covariance du couple (X, Y) ?
3. Dans cette question, $n = 2$.
   (a) Déterminer $X(\Omega)$ et $Y(\Omega)$, puis donner la loi conjointe du couple (X, Y) sous forme de tableau.
   (b) Donner la loi de X, puis celle de Y.
   (c) Calculer la covariance du couple (X, Y).
4. Dans cette question, on revient au cas général où n est un entier naturel quelconque.
   Pour $i \in \{1, 2, 3\}$, on note Y_i la variable aléatoire qui vaut 1 si l'urne numéro i est vide à la fin de l'expérience, et qui vaut 0 sinon.
   (a) Déterminer la loi de X, et donner la valeur de son espérance.
   (b) En remarquant que $Y = Y_1 + Y_2 + Y_3$, calculer $E(Y)$.
   (c) Démontrer que :
   $$\forall i \in \{2, 3\},\ \forall j \in [\![0, n]\!],\ P(X = j \cap Y_i = 1) = \binom{n}{j} \times \frac{1}{3^n}.$$
   (d) Calculer alors $E(XY_i)$ pour $i \in \{2, 3\}$. Que vaut cette espérance si $i = 1$?
   (e) Calculer la covariance du couple (X, Y).

4.3 Exercice non préparé

On souhaite exploiter le suivi d'une randonnée en effectuant des mesures lors de différents points de passage : latitude, longitude, altitude et temps (date).
Ces données sont contenues dans une liste `coords`, où `coords[i]` désigne la liste des 4 mesures effectuées au point de passage numéro `i`. Ainsi `coords[i][2]` renvoie par exemple l'altitude relevée au point de passage numéro `i`.

1. Écrire une fonction`temps(coords)` qui renvoie la liste des temps relevés lors de la randonnée `coords`.
2. Sans utiliser la fonction `max`, écrire une fonction `plus_haut(coords)` qui renvoie la liste `[lat, long]` du point le plus haut lors de la randonnée `coords`.

5 Exemple 5

5.1 Question de cours

Donner la définition du gradient d'une fonction définie sur $\mathbb{R}^2$.

5.2 Exercice préparé

Soit $n \in \mathbb{N}^*$. On note $\mathbb{R}_n[X]$ l'espace vectoriel des fonctions polynomiales de degré inférieur ou égal à n.
On définit sur $\mathbb{R}_n[X]$ l'application D par :

$$\forall P \in \mathbb{R}_n[X], \quad D(P) = P(X+1) - P(X).$$

1. (a) Montrer que D est un endomorphisme de $\mathbb{R}_n[X]$.
   (b) Déterminer $D(1)$ puis $D(X^k)$ pour tout entier naturel k de $[\![1, n]\!]$.
   (c) Donner la matrice de D dans la base canonique de $\mathbb{R}_n[X]$.
   (d) Déterminer le spectre de D. D est-il diagonalisable ?
2. On pose $H_0(X) = 1$ et pour tout $k \in \mathbb{N}^*$, $H_k(X) = \displaystyle\prod_{i=0}^{k-1}(X - i)$.
   (a) Montrer que $\mathcal{B} = (H_0, \dots, H_n)$ est une base de $\mathbb{R}_n[X]$.
   (b) Calculer $D(H_0)$. Montrer que pour tout $k \in [\![1, n]\!]$, $D(H_k) = kH_{k-1}$.
   (c) Déterminer la matrice représentative de D dans la base $\mathcal{B}$.
3. En Python, un polynôme de $\mathbb{R}_n[X]$ est codé en listant ses $n+1$ coefficients par ordre croissant de degré. Par exemple, dans $\mathbb{R}_4[X]$, le polynôme $P = 5X^3 - 2X + 3$ est représenté par la liste `[3,-2,0,5,0]`.
   (a) Programmer une fonction Python qui prend en argument une liste de longueur $n+1$ modélisant un polynôme P de degré inférieur ou égal à $n-1$ dans $\mathbb{R}_n[X]$ et un réel a, et qui renvoie alors la liste modélisant $(X-a)P$.
   (b) Programmer une fonction Python qui prend en argument un entier naturel non nul n et qui renvoie la liste modélisant le polynôme H_n.
4. Soit Y une variable aléatoire suivant une loi de Poisson de paramètre $\lambda > 0$.
   (a) Montrer que $H_2(Y)$ admet une espérance, en déduire que $H_1(Y)$ et $H_0(Y)$ admettent une espérance. Déterminer alors $E(H_0(Y))$, $E(H_1(Y))$ et $E(H_2(Y))$.
   (b) Déterminer les coordonnées de 1, X et X^2 dans la base (H_0, H_1, H_2).
   (c) Retrouver la valeur de la variance de Y.
5. On note $\mathscr{C}$ l'espace vectoriel des fonctions continues de $\mathbb{R}$ dans $\mathbb{R}$.
   À tout élément $f \in E$, on associe la fonction $g = \tilde{D}(f)$ définie par :

   $$\forall x \in \mathbb{R}, \quad g(x) = \tilde{D}(f)(x) = f(x+1) - f(x).$$

   (a) On dit qu'un réel λ est une **valeur propre de** $\tilde{D}$ s'il existe une fonction non nulle f de E telle que $\tilde{D}(f) = \lambda f$. En considérant les fonctions $h_a : x \longmapsto e^{ax}$ et $k_a : x \longmapsto \sin(\pi x)e^{ax}$ où a est un réel, déterminer les valeurs propres de $\tilde{D}$.
   (b) Si F désigne la fonction de répartition d'une variable aléatoire à densité X, montrer que $g = \tilde{D}(F)$ est une densité de probabilité.
   (c) Expliciter g si X suit la loi uniforme sur $[0, 1]$.

5.3 Exercice non préparé

1. Écrire une fonction Python qui simule une série de N lancers d'une pièce équilibrée, et qui renvoie la liste des résultats de ces lancers ("Pile" est codé par 1, et "Face" par 0).
2. Écrire une fonction Python qui simule une série de lancers d'une pièce équilibrée jusqu'à l'obtention de la configuration "Pile, Pile, Face", et renvoie le nombre de lancers nécessaires à l'apparition de cette configuration. A l'aide de cette fonction, évaluer le temps moyen d'attente de cette configuration.