Concours Blanc n°2

Jeudi 11 Mai
Durée : 4 heures

Dans tout le sujet, concernant les questions d'informatique sous Python, on suppose que les bibliothèques d'usage sont déjà importées sous leurs alias habituels :

```
import numpy as np
import numpy.linalg as al
import numpy.random as rd

import matplotlib.pyplot as plt
```

Exercice 1

Partie I : Réductions simultanées

Dans tout l'exercice, on notera $\mathcal{B} = (e_1, e_2, e_3)$ la base canonique de $\mathbb{R}^3$.

(1) Soit A la matrice de $\mathcal{M}_3(\mathbb{R})$ donnée par : $A = \begin{pmatrix} 2 & 1 & -2 \\ 0 & 3 & 0 \\ 1 & -1 & 5 \end{pmatrix}$.

 (a) Calculer $A^2 - 7A$.
 (b) La matrice A est-elle inversible ?
 (c) On note f l'endomorphisme de $\mathbb{R}^3$ dont A est la matrice dans la base canonique. Justifier que f est un automorphisme.
 (d) On note $u_1 = (1, 1, 0)$. Calculer $f(u_1)$. En déduire que $u_1 \in \mathrm{Ker}(f - 3\mathrm{Id})$.
 (e) Déterminer un vecteur u_2 de $\mathbb{R}^3$ de sorte que (u_1, u_2) forme une base de $\mathrm{Ker}(f - 3\mathrm{Id})$.
 (f) Déterminer un vecteur u_3 de $\mathbb{R}^3$ de sorte que (u_3) forme une base de $\mathrm{Ker}(f - 4\mathrm{Id})$. Que vaut, sans calcul supplémentaire $f(u_3)$?

(2) Soit g l'endomorphisme de $\mathbb{R}^3$ défini par son action sur la base canonique

$$g(e_1) = e_1 - 3e_2 - e_3, \qquad g(e_2) = -g(e_1), \qquad g(e_3) = -e_1 - 3e_2 + e_3.$$

 (a) Expliciter la matrice B de g dans la base canonique $\mathcal{B}$.
 (b) Quel est le rang de g?
 (c) Calculer $g(u_1)$. En déduire sans calcul supplémentaire une base du noyau de g.
 (d) On note $v = e_1 - e_2 - e_3$. Calculer $g(v)$ en fonction de v.

(3) Vérifier que $\mathrm{Ker}(f-3\mathrm{Id}) = \mathrm{Vect}(u_1, v)$.

(4) Montrer que $\mathcal{C} = (v, u_1, u_3)$ forme une base de $\mathbb{R}^3$.

(5) On note P la matrice dont les colonnes sont les coordonnées de v, u_1, u_3 (dans la base canonique). Expliciter P et justifier qu'elle est inversible.

(6) Expliciter les matrices D_1 et D_2 de f et g dans la base $\mathcal{C}$.

(7) Vérifier que $A = PD_1P^{-1}$ et $B = PD_2P^{-1}$.

Partie II : une suite matricielle récurrente linéaire d'ordre 2

On pose $X_0 = \begin{pmatrix} 3 \\ 0 \\ -1 \end{pmatrix}$, $X_1 = \begin{pmatrix} 3 \\ 0 \\ -2 \end{pmatrix}$, et pour tout entier naturel n :

$$X_{n+2} = \frac{1}{6}AX_{n+1} + \frac{1}{6}BX_n.$$

Soit $(Y_n)_{n\in\mathbb{N}}$ la suite matricielle définie par :

$$\forall n \in \mathbb{N}, \quad Y_n = P^{-1}X_n.$$

(8) Démontrer que :

$$\forall n \in \mathbb{N} \qquad Y_{n+2} = \frac{1}{6}D_1Y_{n+1} + \frac{1}{6}D_2Y_n.$$

(9) Pour tout entier naturel n, on note $Y_n = \begin{pmatrix} a_n \\ b_n \\ c_n \end{pmatrix}$.

Déduire de la question précédente que :

$$\forall n \in \mathbb{N} \qquad \begin{cases} a_{n+2} & = \dfrac{1}{2}\, a_{n+1} + \dfrac{1}{2}\, a_n \\[2ex] b_{n+2} & = \dfrac{1}{2}\, b_{n+1} \\[2ex] c_{n+2} & = \dfrac{2}{3}\, c_{n+1} + \dfrac{1}{3}\, c_n \end{cases}$$

(10) Expliciter, à l'aide d'un pivot de Gauss, la matrice P^{-1}.
Calculer ensuite les matrices Y_0 et Y_1.

(11) Pour tout entier naturel n, calculer a_n, b_n et c_n en fonction de n.

(12) En déduire l'expression de X_n en fonction de n, pour tout entier naturel n.
On notera $X_n = \begin{pmatrix} \alpha_n \\ \beta_n \\ \gamma_n \end{pmatrix}$, et on vérifiera que :

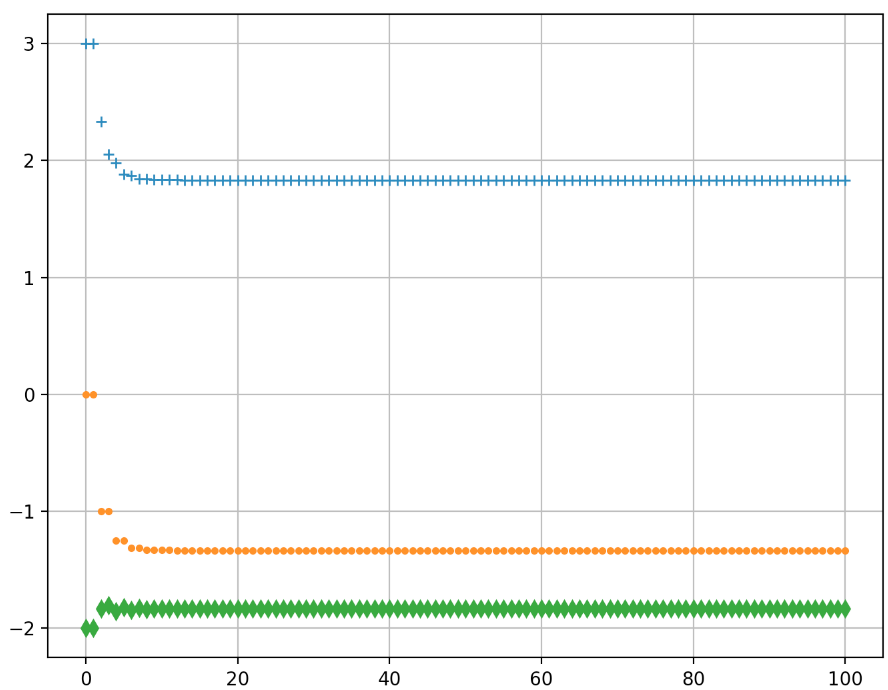
$$\beta_n = \left(\frac{1}{2}\right)^{n-1} - \frac{2}{3}\left(-\frac{1}{2}\right)^n - \frac{4}{3}$$

(13) (a) Compléter la fonction ci-dessous qui prend en argument un entier n supérieur ou égal à 2 et qui renvoie la matrice X_n :

```
def X(n) :
    Xold=np.array([[3],[0],[-1]])
    Xnew=np.array([[3],[0],[-2]])
    A=np.array([[2, 1, -2], [0, 3, 0], [1, -1, 5]])
    B=np.array([[1,-1,-1], [-3, 3, -3], [-1, 1, 1]])
    for i in range(2, n+1) :
        Aux= ...............
        Xold=...............
        Xnew=..............
    return .................
```

(b) La fonction précédente a été utilisée dans un script permettant d'obtenir graphiquement (voir ci-dessous) les valeurs de α_n, β_n et γ_n en fonction de n.
Associer chacune des trois représentations graphiques à chacune des suites $(\alpha_n)_{n\in\mathbb{N}}$, $(\beta_n)_{n\in\mathbb{N}}$, $(\gamma_n)_{n\in\mathbb{N}}$ en justifiant votre réponse.

Affichage Python



Exercice 2

Partie I : Étude de deux suites

Pour tout entier naturel n non nul, on pose :

$$u_n = \sum_{k=1}^{n} \frac{1}{k} - \ln(n), \quad \text{et} \quad v_n = u_n - \frac{1}{n}.$$

(1) Soit f la fonction définie sur $\mathbb{R}_+^*$ par $f(x) = \dfrac{1}{x+1} + \ln(x) - \ln(x+1)$.

(a) Déterminer $\lim\limits_{x\to 0} f(x)$ et $\lim\limits_{x\to +\infty} f(x)$.

(b) Étudier les variations de la fonction f sur $\mathbb{R}_+^*$ et dresser son tableau de variations.

(c) Démontrer que, pour tout $n \in \mathbb{N}^*$, $u_{n+1} - u_n = f(n)$.

(d) En déduire la monotonie de la suite (u_n).

(e) Écrire une fonction Python d'en tête `def u(n):`, qui prend en argument un entier naturel n non nul et qui renvoie la valeur de u_n.

(2) (a) Montrer que

$$\forall n \in \mathbb{N}^*, \quad v_{n+1} - v_n = \frac{1}{n} - \ln\left(1 + \frac{1}{n}\right).$$

(b) Montrer que pour tout réel x positif, $\ln(1+x) \leq x$.
En déduire que la suite (v_n) est croissante.

(c) (i) Soit $x \in]-1, 1[$. Justifier que la série $\sum\limits_{n\geq 2}(-1)^n x^n$ converge. On note $s(x)$ sa somme.

(ii) Justifier, après l'avoir explicitée, que la fonction $x \mapsto s(x)$ est continue sur $]-1, 1[$.

(iii) On pose alors, pour $x \in]-1, 1[$

$$\varphi(x) = \begin{cases} \dfrac{1}{x^2}\displaystyle\int_0^x s(t)\mathrm{d}t, & \text{si } x \neq 0 \\ \\ 0, & \text{si } x = 0 \end{cases}$$

Montrer que, pour tout $x \in]0, 1[$, $0 \leq \varphi(x) \leq \ln(1+x)$
Montrer de même que, pour tout $x \in]-1, 0[$, $0 \geq \varphi(x) \geq \ln(1+x)$.
En déduire que φ est continue sur $]-1, 1[$.

(iv) Soit $x \in]-1, 1[$ fixé. En remarquant que, pour t entre 0 et x,

$$\frac{1}{1+t} = 1 - t + s(t),$$

et à l'aide de la question précédente, montrer qu'on peut écrire, au voisinage de 0

$$\ln(1+x) = x - \frac{x^2}{2} + o(x^2)$$

(cette dernière relation s'écrit aussi

$$\ln(1+x) = x - \frac{x^2}{2} + x^2\varepsilon(x),$$

avec ε une certaine quantité qui tend vers 0 lorsque $x \to 0$.)
On vient d'obtenir le développement limité à l'ordre 2 en 0 de $\ln(1+x)$.

(d) Utiliser le développement limité obtenu à la question précédente pour obtenir

$$v_{n+1} - v_n = \frac{1}{2n^2} + o\left(\frac{1}{n^2}\right), \qquad n \to +\infty$$

En déduire qu'il existe un entier n_0 tel que, pour tout $n \geq n_0$, on ait

$$0 \leq v_{n+1} - v_n \leq \frac{1}{n^2}.$$

(e) Déterminer la nature de la série de terme général $v_{n+1} - v_n$. On note $\gamma = \sum\limits_{n=1}^{+\infty}(v_{n+1} - v_n)$.

(f) Pour $n \geq 2$, simplifier la somme partielle : $\sum\limits_{k=1}^{n-1}(v_{k+1} - v_k)$.
En déduire que la suite $(v_n)_{n\geq 2}$ converge vers γ.

(3) (a) Déterminer $\lim_{n\to\infty} u_n$.

(b) Montrer que

$$\forall n \in \mathbb{N}^*, \quad v_n \leq \gamma \leq u_n$$

puis que

$$\forall n \in \mathbb{N}^*, \quad |u_n - \gamma| \leq \frac{1}{n}.$$

(c) Écrire une fonction Python d'en-tête `def gamma_app(epsilon):` qui, prenant en argument un réel `epsilon`> 0 renvoie une valeur approchée de γ à `epsilon` près.

Partie II: Étude d'une série

Pour tout entier naturel $n \in \mathbb{N}^*$, on pose $a_n = \dfrac{1}{n(2n-1)}$.

(4) Démontrer, à l'aide d'un argument de comparaison, que la série de terme général a_n converge.

(5) (a) Justifier que :

$$\forall n \in \mathbb{N}^*, \quad \sum_{k=1}^{n} \frac{1}{2k-1} = \sum_{k=1}^{2n} \frac{1}{k} - \sum_{k=1}^{n} \frac{1}{2k}.$$

(b) Déterminer deux réels α et β tels que :

$$\forall n \in \mathbb{N}^*, \quad a_n = \frac{\alpha}{n} + \frac{\beta}{2n-1}.$$

(c) En déduire que :

$$\forall n \in \mathbb{N}^*, \qquad \sum_{k=1}^{n} a_k = 2 \sum_{k=n+1}^{2n} \frac{1}{k}.$$

(6) (a) Montrer que

$$\forall n \in \mathbb{N}^*, \quad \sum_{k=n+1}^{2n} \frac{1}{k} = u_{2n} - u_n + \ln(2).$$

où (u_n) est la suite définie dans la Partie I.

(b) Calculer alors $\displaystyle\sum_{k=1}^{+\infty} a_k$.

(7) (a) Montrer que

$$\forall n \in \mathbb{N}^*, \quad \sum_{k=n+1}^{2n} \frac{1}{k} = \frac{1}{n} \sum_{k=1}^{n} \frac{1}{1+\frac{k}{n}}.$$

(b) On **admet** le résultat suivant : si h est une fonction continue sur $[a;b]$ alors

$$\lim_{n\to+\infty} \frac{1}{n} \sum_{k=1}^{n} h\left(a + \frac{k}{n}\right) = \int_a^b h(t)\mathrm{d}t.$$

Retrouver alors le résultat de la Question (6b)

Exercice 3

Soit $n \in \mathbb{N}$ un entier supérieur ou égal à 2. Une urne contient n boules numérotées de 1 à n. On prélève au hasard ces n boules une par une et sans remise (afin de vider l'urne).
À la suite de cette expérience, on note, pour tout $i \in [\![1, n]\!]$, u_i le numéro de la boule obtenue au cours du $i-$ème tirage.
Pour tout $i \in [\![2, n]\!]$, on dit qu'il y a un record au $i-$ème tirage si

$$u_i > \max\{u_1, \ldots, u_{i-1}\},$$

autrement dit, si la boule obtenue au $i-$ème tirage porte un numéro strictement supérieurs aux numéros des boules tirées précédemment. D'autre part, on convient qu'il y a systématiquement un record à l'instant 1.

Pour tout $(i, k) \in [\![1, n]\!] \times [\![1, n]\!]$, on introduit les événements :

- R_i : "il y a un record au $i-$ème tirage"
- $B_{i,k}$: "la boule obtenue $i-$ème tirage est numérotée k"
- $A_{i,k}$: "la boule obtenue au $i-$ème tirage porte un numéro strictement inférieur à k"

Par convention, on a donc $P(R_1) = 1$.

Exemple. *Si $n = 8$ et que l'on obtient, dans cet ordre, les boules numérotés ②①③⑤④⑥⑧⑦, alors il y a un record aux tirages 1,3,4,6 et 7. Ainsi les événements*

$$R_1,\ R_3\ , R_4\ , R_6\ , R_7\ , B_{1,2}\ , B_{2,1}\ , B_{3,3}\ , A_{3,6}\ , A_{5,6},$$

notamment (ce ne sont pas les seuls), sont réalisés.

Partie I : Simulation de l'expérience avec Python

Afin de modéliser cette expérience en Python, l'urne est représentée par une liste de numéros de boules et cette liste est donc *actualisée* après chaque tirage d'une boule.

On **admet** que la commande `rd.randint(a, b)` renvoie un nombre entier, choisi au hasard uniformément entre les entiers a et $b - 1$.

(1) Recopier et compléter la fonction suivante qui, prend en argument une liste `L`, choisit le rang k d'un terme de la liste au hasard de manière équiprobable (parmi ceux disponibles) et renvoie le terme correspondant $x = \mathtt{L}[k]$ ainsi que la liste mise à jour.

```
def selection(L):
    n=len(L) # longueur de la liste
    k=rd.randint(0, n)
    x= .....
    .... # on enlève x de la liste L
    return (x, L)
```

(2) Recopier et compléter le programme ci-dessous qui permet de simuler les pioches successives jusqu'à vider l'urne. Le résultat renvoyé par la fonction est donc la liste ordonnée des numéros des boules piochées.

```
def pioche(n):
    U= [..............] # Urne avant la première pioche
    T=[ ]
    for k in range(n):
        x,U=.........
        T.append(x)
  return T
```

(3) Écrire alors une fonction d'en-tête `def X(n):` qui effectue les n pioches successives dans l'urne et renvoie le nombre de records. On utilisera bien entendu la fonction `pioche` ci-dessus.
(*On rappelle que la commande* `np.max(L)` *renvoie la plus grande valeur de la liste* `L`.)

Partie II : Un record au $i-$ème tirage

On modélise l'expérience par l'ensemble Ω des $n-$uplets de numéros de boules piochées, dont les composantes sont donc deux à deux distinctes et on considère P l'équiprobabilité sur $(\Omega, \mathcal{P}(\Omega))$.

(4) Quel est le cardinal de Ω?

(5) Combien y a-t-il de tirages de n boules successivement sans remise dont la dernière boule est celle numérotée n?
En déduire que $P(R_n) = \dfrac{1}{n}$.

(6) Soit $i \in [\![2, n]\!]$.
  (a) Que vaut $P(R_i \cap B_{i,k})$ lorsque $k \in [\![1, i-1]\!]$?
  (b) Justifier que, pour tout $k \in [\![1, n]\!]$, $R_i \cap B_{i,k} = \left(\bigcap_{j=1}^{i-1} A_{j,k} \right) \cap B_{i,k}$.
  (c) Soit $j \in [\![1, i-2]\!]$. Sachant que l'on a déjà pioché j boules avec un numéro strictement inférieur à k, combien en reste-t-il ? Et combien reste-t-il de boules au total ?

  En déduire que
$$P(R_i \cap B_{i,k}) = \frac{\frac{(k-1)!}{(k-i)!}}{\frac{n!}{(n-i)!}}.$$
  (d) Montrer alors que
$$P(R_i) = \frac{1}{i} \sum_{k=i}^{n} \frac{\binom{k-1}{i-1}}{\binom{n}{i}}.$$
  (e) Justifier que
$$\sum_{k=i+1}^{n} \left(\binom{k}{i} - \binom{k-1}{i} \right) = \binom{n}{i} - 1.$$
  (f) En déduire enfin que $P(R_i) = \dfrac{1}{i}$.

On vient donc de montrer que, pour tout $i \in [\![1, n]\!]$, $P(R_i) = \dfrac{1}{i}$: il y a une chance sur i qu'il y ait un record au $i-$ème tirage.

Partie III : Une variable aléatoire

Dans la suite, nous allons nous intéresser à la *variable aléatoire* X_n qui compte le nombre de records, c'est à dire que X_n est une application :
$$X_n : \Omega \to \mathbb{R}.$$
L'évènement $[X_n = k]$ est réalisé si et seulement si il y a exactement k records.

(7) Quel est l'ensemble des valeurs que prend X_n, noté $X_n(\Omega)$, et appelé *univers image*?

(8) Justifier que $P(X_n = 1) = \dfrac{1}{n}$.

(9) (a) Écrire l'événement $[X_n = n]$ en fonction des événements de la famille $(B_{i,k})_{1\leq i,k\leq n}$.
(b) En déduire $P(X_n = n)$.

(10) Pour tout $i \in [\![2, n]\!]$, notons E_i l'événement "les $i-1$ premiers tirages amènent des boules dont le numéro est strictement inférieur à n et la première boule tirée porte le plus grand numéro d'entre eux".

(a) Justifier à l'aide d'arguments combinatoires que

$$P(E_i) = \frac{\binom{n-1}{i-1} \times (i-2)!}{\frac{n!}{(n-i+1)!}}$$

et en déduire que $P(E_i) = \dfrac{n-i+1}{n(i-1)}$.

(b) Justifier alors que

$$[X_n = 2] = \bigcup_{i=2}^{n} [E_i \cap B_{i,n}].$$

(c) En déduire que

$$P(X_n = 2) = \frac{1}{n} \sum_{k=1}^{n-1} \frac{1}{k}.$$

(d) Écrire une fonction en Python d'en-tête `def proba_X(n):` qui prend en argument n et qui calcule et renvoie $P(X_n = 2)$ à l'aide de la formule ci-dessus.

En général, on essaie de déterminer complètement la loi d'une variable aléatoire, c'est à dire qu'il faudrait déterminer, pour tout $k \in X_n(\Omega)$, la probabilité $P(X_n = k)$. Ici, on l'a fait pour trois valeurs. Le sujet n'en demande pas plus.