Banque "Agro-Véto"
A-0219

MATHÉMATIQUES

Modélisation mathématique et informatique

Durée : 3 heures 30 minutes

Chaque candidat est responsable de la vérification de son sujet d'épreuve : pagination et impression de chaque page. Ce contrôle doit être fait en début d'épreuve : en cas de doute, il doit alerter au plus tôt le chef de centre qui contrôlera et éventuellement remplacera le sujet.

Si, au cours de l'épreuve, un candidat repère ce qui lui semble être une erreur d'énoncé, il le signale sur sa copie et poursuit sa composition en expliquant les raisons des initiatives qu'il a été amené à prendre.

L'usage d'une calculatrice est autorisé pour cette épreuve. Les questions d'informatique devront être rédigées en langage Python exclusivement.

Les différentes parties de l'épreuve sont indépendantes sauf les parties 4 et 5. Les questions d'informatique sont regroupées dans la partie 2.

Le but de cette épreuve est d'étudier différentes méthodes pour modéliser des phénomènes qui dépendent du temps. Parmi les problématiques souvent étudiées, on peut par exemple citer l'étude de l'évolution de la population d'un pays au cours du temps à des fins de contrôles démographiques, l'analyse de l'évolution de la concentration en ozone en fonction du temps afin d'évaluer le niveau de pollution d'une zone donnée ou encore l'évolution de la température au cours du temps dans un pays afin d'étudier la problématique du réchauffement climatique.

Dans ce sujet, on va s'intéresser à la modélisation des niveaux annuels du lac Huron exprimés en mètres entre 1875 et 1972 représentés dans la figure 1.

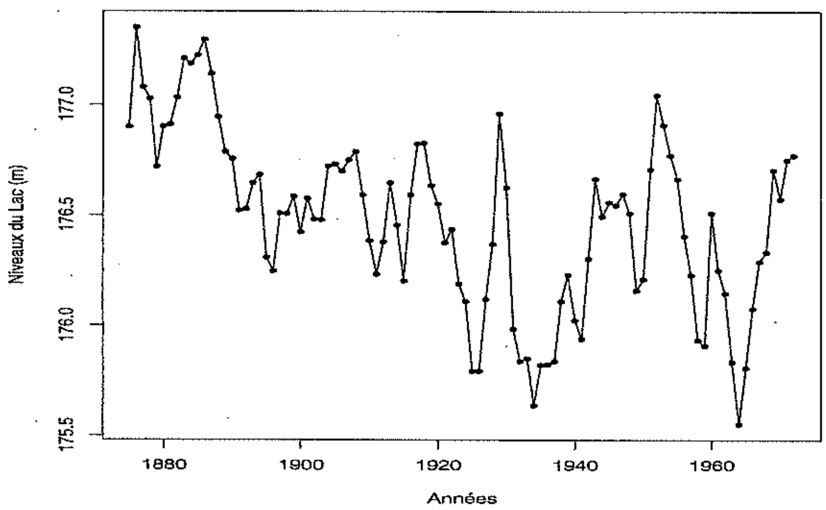

FIGURE 1 – Niveaux annuels du lac Huron exprimés en mètres.

PARTIE 1

Dans cette partie, on note n un entier naturel supérieur ou égal à 3. Pour tout entier i dans $\{1, 2, \ldots, n\}$, on note y_i le niveau du lac Huron de la $i$ème observation à l'instant t_i. Dans la figure 1, les points de coordonnées (t_i, y_i) sont représentés par des points ('•'). Dans la suite, les t_i sont supposés distincts. On s'intéresse à l'ajustement d'un polynôme de degré 2 au nuage de points $((t_1, y_1), (t_2, y_2), \ldots, (t_n, y_n))$. Un tel ajustement est représenté dans la figure 2. Pour réaliser cet ajustement on utilise le critère des moindres carrés défini ci-dessous.

1. On définit la fonction $F : \mathbb{R}^3 \longrightarrow \mathbb{R}$ par :

$$\forall (a,b,c) \in \mathbb{R}^3, \quad F(a,b,c) = \sum_{i=1}^{n} \left(y_i - a - bt_i - ct_i^2\right)^2.$$

   a) Expliquer en quelques lignes ou à l'aide d'un dessin ce que l'on cherche à faire lorsque l'on minimise la fonction F ci-dessus par rapport à a, b et c.

   b) Justifier que les dérivées partielles de F par rapport à a, b et c existent. Calculer les dérivées partielles de F par rapport à a, b et c que l'on notera dans la suite : $\frac{\partial F}{\partial a}(a,b,c)$, $\frac{\partial F}{\partial b}(a,b,c)$ et $\frac{\partial F}{\partial c}(a,b,c)$, respectivement.

2. On note $\langle u; v\rangle$ le produit scalaire dans $\mathbb{R}^n$ de deux vecteurs $u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}$ et $v = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$ de $\mathbb{R}^n$, défini par :

$$\langle u; v\rangle = \sum_{i=1}^{n} u_i v_i.$$

   On note également :

$$T = \begin{pmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_n & t_n^2 \end{pmatrix}, \; Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \text{ et } \beta = \begin{pmatrix} a \\ b \\ c \end{pmatrix}.$$

   a) Montrer que :

$$Y - T\beta = \begin{pmatrix} y_1 - a - bt_1 - ct_1^2 \\ y_2 - a - bt_2 - ct_2^2 \\ \vdots \\ y_n - a - bt_n - ct_n^2 \end{pmatrix}.$$

   En déduire que :

$$\frac{\partial F}{\partial a}(a,b,c) = -2\left\langle Y - T\beta; \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}\right\rangle.$$

   b) Montrer que :

$$\frac{\partial F}{\partial b}(a,b,c) = -2\left\langle Y - T\beta; \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_n \end{pmatrix}\right\rangle.$$

De la même façon, on pourrait montrer que :

$$\frac{\partial F}{\partial c}(a,b,c) = -2\left\langle Y - T\beta ; \begin{pmatrix} t_1^2 \\ t_2^2 \\ \vdots \\ t_n^2 \end{pmatrix} \right\rangle.$$

3. Montrer à l'aide des questions précédentes que si $\widehat{\beta} = \begin{pmatrix} \widehat{a} \\ \widehat{b} \\ \widehat{c} \end{pmatrix}$ est solution du système suivant :

$$\begin{cases} \frac{\partial F}{\partial a}(a,b,c) &= 0, \\ \frac{\partial F}{\partial b}(a,b,c) &= 0. \\ \frac{\partial F}{\partial c}(a,b,c) &= 0. \end{cases}$$

alors

$$({}^tTT)\widehat{\beta} = {}^tTY,$$

où ${}^tT$ désigne la transposée de T. On admet que $\widehat{\beta}$ ainsi défini minimise la fonction F définie dans la question 1.

4. Soit A une matrice réelle ayant n lignes et p colonnes. On note

$$\mathrm{Ker}(A) = \{u \in \mathbb{R}^p, Au = 0\}.$$

   a) Montrer que $\mathrm{Ker}(A) = \mathrm{Ker}({}^tAA)$.
   b) Que vaut $\dim(\mathrm{Ker}({}^tAA)) + \mathrm{rang}({}^tAA)$? Justifier.
   c) Que vaut $\dim(\mathrm{Ker}(A)) + \mathrm{rang}(A)$? Justifier.
   d) En déduire une condition suffisante sur A pour laquelle ${}^tAA$ est inversible. Cette condition est-elle nécessaire ?

5. Dans le cas où ${}^tTT$ est inversible, donner l'expression de $\widehat{\beta}$.

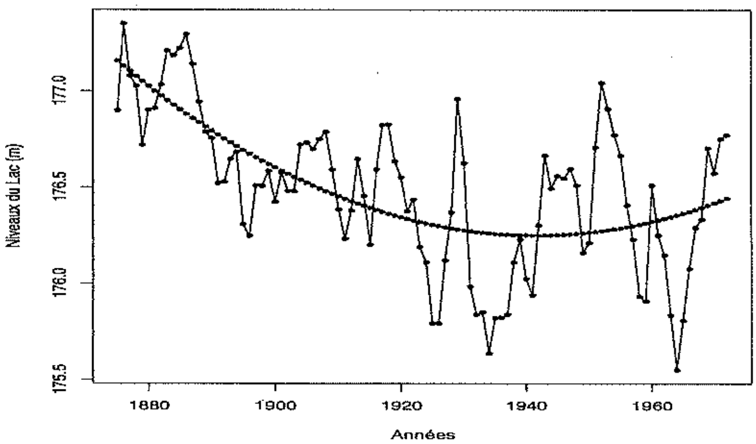

FIGURE 2 – Niveaux annuels du lac Huron et ajustement d'un polynôme de degré 2.

L'ajustement $T\widehat{\beta}$ obtenu aux questions précédentes est représenté sur la figure 2 ainsi que les données. On remarque que cette modélisation peut être améliorée, ce sera le but des parties suivantes.

PARTIE 2

Des éléments de syntaxe Python, et en particulier l'usage du module numpy, sont donnés en annexe à la fin de la partie 2. Dans tout ce qui suit, les variables n, p, A, M, i, j et c vérifient les conditions suivantes qui ne seront pas rappelées à chaque question :

— n et p sont des entiers naturels tels que $p \geq n \geq 2$;
— A est une matrice carrée à n lignes inversible ;
— M est une matrice à n lignes et p colonnes telle que la sous-matrice carrée constituée des n premières colonnes de M est inversible ;
— i et j sont des entiers tels que $0 \leq i \leq n-1$ et $0 \leq j \leq n-1$;
— c est un réel non nul.

On note $L_i \leftarrow L_i + cL_j$ l'opération qui ajoute à la ligne i d'une matrice la ligne j multipliée par c.

1. Soit la fonction `initialisation` :

```
def initialisation(A):
    n = np.shape(A)[0]
    mat = np.zeros((n,2*n))
    for i in range(0, n):
        for j in range(0, n):
            mat[i,j] = A[i,j]
    return(mat)
```

   Pour chacune des affirmations suivantes, indiquer si elle est vraie ou fausse, en justifiant. L'appel `initialisation(A)` renvoie :

   (a) une matrice rectangulaire à n lignes et $2n$ colonnes remplie de zéros ;

   (b) une matrice de même taille que A ;

   (c) une erreur au niveau d'un `range` ;

   (d) une matrice rectangulaire telle que les n premières colonnes correspondent aux n colonnes de A, et les autres colonnes sont nulles.

2. Les trois fonctions `multip`, `ajout` et `permut` suivantes ne renvoient rien : elles *modifient* les matrices auxquelles elles s'appliquent.

   (a) Que réalise la fonction `multip` ?

```
def multip(M, i, c):
    p = np.shape(M)[1]
    for k in range(0, p):
        M[i,k] = c*M[i,k]
```

   (b) Compléter la fonction `ajout`, afin qu'elle effectue l'opération $L_i \leftarrow L_i + cL_j$.

```
def ajout(M, i, j, c):
    p = np.shape(M)[1]
    for k in range(0, p):
        _____ ligne(s) à compléter _____
```

   (c) Écrire une fonction `permut` prenant pour argument M, i et j, et qui modifie M en échangeant les valeurs des lignes i et j.

   Dans la suite du sujet, l'expression "opération élémentaire sur les lignes" fera référence à l'utilisation de `permut`, `multip` ou `ajout`.

3. Soit la colonne numéro j dans la matrice M. On cherche le numéro r d'une ligne où est situé le plus grand coefficient (en valeur absolue) de cette colonne parmi les lignes j à $n-1$. Autrement dit, r vérifie :

$$|A[r,j]| = \max \{|A[i,j]| \text{ pour } i \text{ tel que } j \leq i \leq n-1\}.$$

Écrire une fonction `rang_pivot` prenant pour argument M et j, et qui renvoie cette valeur de r. Lorsqu'il y a plusieurs réponses possibles pour r, dire (avec justification) si l'algorithme renvoie le plus petit r, le plus grand r ou un autre choix. *(L'utilisation d'une commande* `max` *déjà programmée dans* `Python` *est bien sûr proscrite.)*

4. Soit la fonction `mystere` :

```
def mystere(M):
    n = np.shape(M)[0]
    for j in range(0, n):
        r = rang_pivot(M, j)
        permut(M, r, j)
        for k in range(j+1, n):
            ajout(M, k, j, -M[k,j]/M[j,j])
        print(M)
```

(a) On considère dans cette question l'algorithme `mystere` appliqué à la matrice $M_1 = \begin{pmatrix} 3 & 2 & 2 \\ -6 & 0 & 12 \\ 1 & 1 & -3 \end{pmatrix}$: indiquer combien de fois la ligne `print(M)` est exécutée ainsi que les différentes valeurs qu'elle affiche.

(b) De façon générale, que réalise cet algorithme ?

5. Soit la fonction `reduire`, qui modifie M :

```
def reduire(M):
    n = np.shape(M)[0]
    mystere(M)
    for i in range(0, n):
        multip(M, i, 1/M[i,i])
    #Les lignes suivantes sont à compléter :
    ---------------------------
```

(a) Compléter la fonction afin que la portion de code manquante effectue les opérations élémentaires suivantes sur les lignes :

```
pour j prenant les valeurs n-1, n-2, ..., 1, faire :
    pour k prenant les valeurs j-1, j-2, ..., 0, faire :
        Lk ← Lk − M[k,j]Lj
```

(b) Indiquer ce que réalise cette fonction.

6. Inversion de A.

(a) Écrire une fonction `augmenter` prenant pour argument A et qui renvoie la matrice de taille $(n, 2n)$ définie ainsi :
- dans la partie gauche (composée des n lignes et n premières colonnes), elle contient les coefficients de A ;
- dans la partie droite (composée des n lignes et n dernières colonnes), elle contient les coefficients de la matrice identité de taille n.

Par exemple,

$$\text{pour } \begin{pmatrix} 1 & 2 \\ -1 & 3 \end{pmatrix} \text{ la fonction renvoie } \begin{pmatrix} 1 & 2 & 1 & 0 \\ -1 & 3 & 0 & 1 \end{pmatrix}.$$

(b) À l'aide des fonctions précédentes, proposer un raisonnement permettant d'inverser A.

(c) Écrire une fonction `inverser` prenant pour argument A et qui renvoie la matrice inverse de A, en suivant le raisonnement décrit en question 6b.

(d) Quelle méthode connue venez-vous d'implémenter ?

Annexe : Rappels Python pour la partie 2

On considère que le module `numpy`, permettant de manipuler des tableaux à deux dimensions, est importé via `import numpy as np`. Pour une matrice M à n lignes et p colonnes, les indices vont de 0 à $n-1$ pour les lignes et de 0 à $p-1$ pour les colonnes.

Python	Interprétation
`abs(x)`	Valeur absolue du nombre x
`M[i,j]`	Coefficient d'indice (i,j) de la matrice M
`np.zeros((n,p))`	Matrice à n lignes et p colonnes remplie de zéros

`T = np.shape(M)`
`T[0]` ou `np.shape(M)[0]`
`T[1]` ou `np.shape(M)[1]`
	Dimensions de la matrice M
	Nombre de lignes
	Nombre de colonnes

`M[a:b,c:d]`	Matrice extraite de M constituée des lignes a à $b-1$ et des colonnes c à $d-1$: si a (resp. c) n'est pas précisé, l'extraction commence à la première ligne (resp. colonne) si b (resp. d) n'est pas précisé, l'extraction finit à la dernière ligne (resp. colonne) incluse

PARTIE 3

Nous allons modéliser ici, par une approche mécanistique, l'évolution dans le temps de la hauteur $x(t)$ de la surface d'un lac dont la géométrie est simplifiée comme sur la figure 3.

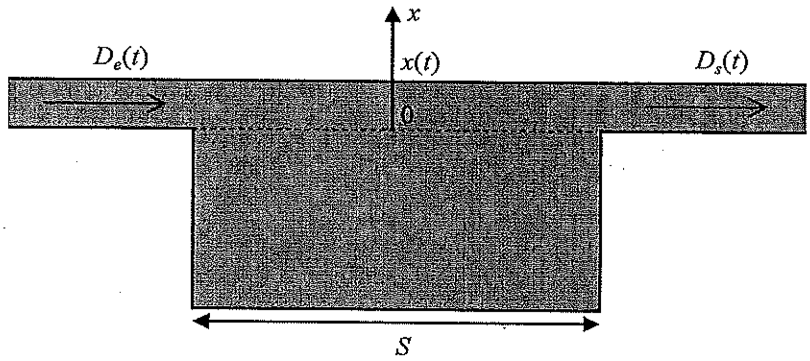

FIGURE 3 – Modélisation d'un lac.

Si S, section horizontale du lac, est supposée constante par rapport à la hauteur, une simple équation de bilan sur les volumes nous permet d'écrire :

$$S\frac{\mathrm{d}x}{\mathrm{d}t}(t) = D_e(t) - D_s(t)$$

où le débit entrant $D_e(t)$ est une fonction supposée connue. On peut modéliser le débit sortant $D_s(t)$ comme étant proportionnel à $x(t)$ soit $D_s(t) = \alpha x(t)$, où α est un réel strictement positif. On obtient donc :

$$\frac{\mathrm{d}x}{\mathrm{d}t}(t) = \frac{D_e(t)}{S} - \frac{\alpha}{S}x(t).$$

Par la suite le modèle d'évolution s'écrira :

$$\frac{\mathrm{d}x}{\mathrm{d}t}(t) = u(t) - kx(t), \tag{1}$$

où $k \in \mathbb{R}_+^*$ et $u(t)$ est une fonction réelle connue, appelée "entrée" du système et dans la suite, cette entrée prendra différentes formes. Dans la suite, on admettra l'existence de solution à l'équation (1).

Soit la fonction $h : \mathbb{R} \longrightarrow \mathbb{R}$ représentée dans la figure 4 et définie comme suit :

$$\begin{cases} h(t) & = 1, \quad \forall t \geq 0 \\ h(t) & = 0, \quad \forall t < 0. \end{cases} \tag{2}$$

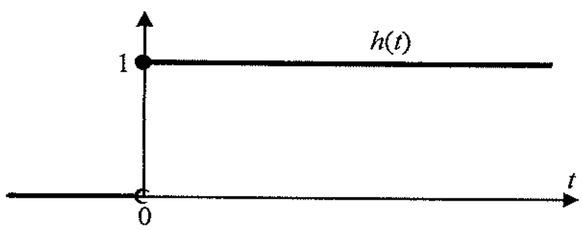

FIGURE 4 – Entrée sous la forme d'une marche.

1. Calculer la solution de (1) pour $t \in [0, +\infty[$ avec $u(t) = h(t)$, où h est définie dans (2) et pour la condition initiale $x(0) = 0$.
2. a) Montrer que l'entrée $\delta(t)$ (Cf. figure 5) peut s'écrire sous la forme $\delta(t) = h(t) - h(t - \Delta)$ où Δ est un réel strictement positif fixé et h est représentée dans la figure 4.

   b) On admet que la solution de (1) pour $t \geq 0$ avec $u(t) = \delta(t)$ et pour condition initiale $x(0) = 0$ est $x(t) = \frac{1}{k}\left(e^{-k(t-\Delta)} - e^{-kt}\right)$ pour $t \geq \Delta$. Exprimer la solution pour $t \in [0, \Delta]$.

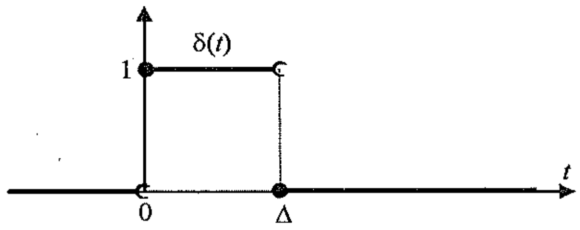

FIGURE 5 – Entrée sous la forme d'un créneau.

Dans la suite on notera $t_i = i\Delta$ où $i \in \mathbb{N}$.

3. Pour $i \in \mathbb{N}^*$, on considère $u(t)$ la fonction créneau définie comme suit : $u(t) = u_i$, lorsque t appartient à $[t_{i-1}, t_i[$ et $u(t) = 0$ sinon (Cf. figure 6). Montrer que :

$$\forall t \geq t_i, \quad x(t) = u_i \frac{1 - e^{-k\Delta}}{k} e^{-k(t-t_i)},$$

si $x(0) = 0$.

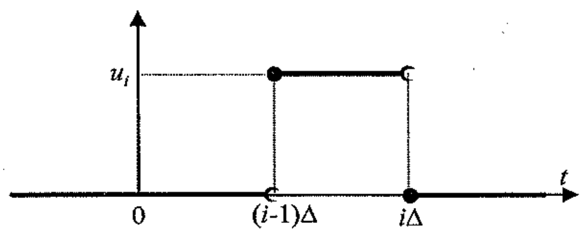

FIGURE 6 – Entrée sous forme d'un créneau entre $[t_{i-1}, t_i[$.

4. Dans cette question on suppose que $u(t)$ est une fonction constante par morceaux définie comme suit : $u(t) = u_i$, lorsque t appartient à $t \in [t_{i-1}, t_i[$, pour tout $i \in \mathbb{N}^*$ (Cf. figure 7)). Montrer que :

$$x(t_i) = \sum_{j=1}^{i} u_j \frac{1 - e^{-k\Delta}}{k} e^{-k(t_i - t_j)},$$

si $x(0) = 0$.

5. En effectuant un changement d'indice sur la sommation, montrer que le résultat de la question précédente $x(t_i)$ peut s'écrire :

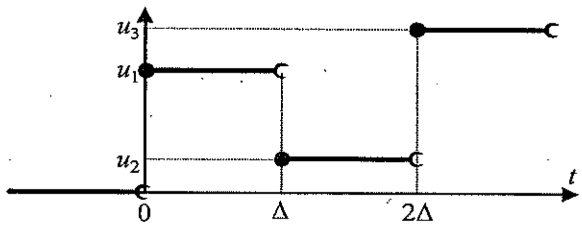

FIGURE 7 – Entrée sous forme d'une fonction en escalier.

$$x(t_i) = \nu \sum_{\ell=0}^{i-1} u_{i-\ell} \Phi^{\ell} \tag{3}$$

où vous expliciterez ν et Φ.

6. Que pouvez vous dire de la suite $(\Phi^{\ell})_{\ell \in \mathbb{N}}$? Interpréter l'équation (3).

Dans la suite, nous allons proposer une modélisation probabiliste de l'évolution dans le temps de la hauteur d'un lac.

PARTIE 4

On rappelle que la covariance entre deux variables aléatoires X et Y est définie par :

$$\mathrm{Cov}(X,Y) = \mathbb{E}\left[(X - \mathbb{E}(X))(Y - \mathbb{E}(Y))\right] = \mathbb{E}(XY) - \mathbb{E}(X)\mathbb{E}(Y). \tag{4}$$

On modélise les observations d'un phénomène qui dépend du temps comme des réalisations de variables aléatoires X_n où n désigne un entier naturel.
On dit que $(X_n)_{n\in\mathbb{N}}$ est un processus stationnaire lorsque :

$$\mathbb{E}(X_n) = \mu,\ \forall n \in \mathbb{N} \text{ et } \mathrm{Cov}(X_n, X_{n+p}) = \gamma_X(p),\ \forall n, p \in \mathbb{N},$$

où μ désigne une constante indépendante de n, Cov est définie dans (4) et γ_X est une fonction indépendante de n.

1. Soient $(Z_n)_{n\in\mathbb{N}}$ des variables aléatoires indépendantes et identiquement distribuées d'espérance m et de variance σ^2.
   a) Calculer $\mathrm{Cov}(Z_n, Z_{n+p})$ lorsque $p = 0$ et lorsque p est un entier naturel non nul.
   b) $(Z_n)_{n\in\mathbb{N}}$ est-il un processus stationnaire ?
2. On dit qu'un processus stationnaire $(W_n)_{n\in\mathbb{N}}$ est un bruit blanc d'espérance nulle et de variance σ^2 que l'on note $(W_n) \sim BB(0,\sigma^2)$ lorsque

$$\mathbb{E}(W_n) = 0,\ \forall n \in \mathbb{N},\ \gamma_W(p) = 0,\ \forall p \in \mathbb{N}^* \text{ et } \gamma_W(0) = \sigma^2.$$

   On considère à présent :

$$X_n = W_n + \theta W_{n-1},\ \forall n \geq 1,$$

   où θ est un nombre réel non nul et $(W_n) \sim BB(0,\sigma^2)$.
   a) Calculer $\mathbb{E}(X_n)$.
   b) Calculer $\mathrm{Cov}(X_n, X_{n+p})$ lorsque $p = 0$ et lorsque p est un entier naturel non nul.
   c) (X_n) est-il un processus stationnaire ?
3. Soit (X_n) le processus stationnaire solution de l'équation suivante :

$$X_n - \phi X_{n-1} = W_n,\ \forall n \geq 1, \tag{5}$$

   où ϕ est un nombre réel non nul tel que : $-1 < \phi < 1$ et $(W_n) \sim BB(0,\sigma^2)$.
   a) Calculer $\mathbb{E}(X_n)$.
   b) En multipliant (5) par X_n et en prenant l'espérance, montrer que

$$\gamma_X(0) - \phi\gamma_X(1) = \mathbb{E}(W_n X_n).$$

   c) On admet pour cette question que $\mathbb{E}(W_n X_q) = 0,\ \forall q < n$.
      (1) Montrer que

$$\mathbb{E}(W_n X_n) = \sigma^2.$$

      (2) En multipliant (5) par X_{n-1} et en prenant l'espérance, montrer que

$$\gamma_X(1) - \phi\gamma_X(0) = 0.$$

      (3) En déduire $\gamma_X(0)$ et $\gamma_X(1)$ en fonction de ϕ et σ^2.
   e) En utilisant (5) écrire X_n en fonction de $X_{n-(p+1)}$, W_{n-p}, $W_{n-(p-1)}$,...,W_{n-2}, W_{n-1} et W_n où p est un entier naturel quelconque inférieur à $n-1$. Faire le lien entre l'expression ainsi obtenue et celle de l'équation (3).

PARTIE 5

On souhaite à présent utiliser les notions de la partie 4 pour modéliser les niveaux annuels du lac Huron représentés dans la figure 1.
On propose de modéliser ces observations comme suit :

$$X_n - \phi X_{n-1} = W_n, \ \forall n \geq 1, \tag{6}$$

où $(W_n) \sim BB(\mu, \sigma^2)$.

1. Comment interprétez-vous cette modélisation ?

Le modèle (6) peut se réécrire comme suit :

$$X_n = \mu + \phi X_{n-1} + Z_n, \ \forall n \geq 1, \tag{7}$$

où $(Z_n) \sim BB(0, \sigma^2)$. On propose d'estimer μ et ϕ en utilisant un critère des moindres carrés analogue à celui de la partie 1. On notera $\widehat{\mu}$ et $\widehat{\phi}$ les estimateurs de μ et ϕ ainsi obtenus.
Afin de vérifier si le modèle (7) est approprié pour modéliser les niveaux annuels du lac Huron, on considère :

$$U_n = X_n - \widehat{\mu} - \widehat{\phi} X_{n-1}, \ \forall n \geq 1.$$

Si le modèle est adapté aux données, (U_n) doit avoir les mêmes propriétés que (Z_n) c'est-à-dire : $(U_n) \sim BB(0, \sigma^2)$. Dans la suite, on souhaite vérifier cela en supposant que (U_n) est un processus stationnaire.

2. Pour vérifier les propriétés de γ_U, on se propose d'estimer γ_U à partir de U_1, U_2,..., U_n, par :

$$\widehat{\gamma}_U(p) = \frac{1}{n}\sum_{j=1}^{n-p}(U_j - \bar{U})(U_{j+p} - \bar{U}), \ \forall p < n,$$

où $\bar{U} = (\sum_{i=1}^{n} U_i)/n$. On va montrer que lorsque $p = 0$, $\widehat{\gamma}_U(0)$ est un estimateur asymptotiquement sans biais de $\gamma_U(0)$ c'est-à-dire que $\mathbb{E}(\widehat{\gamma}_U(0))$ tend vers $\gamma_U(0)$ lorsque n tend vers l'infini. On admettra que c'est également le cas lorsque $p > 0$.

a) Montrer que

$$\widehat{\gamma}_U(0) = \frac{1}{n}\sum_{j=1}^{n} U_j^2 - \left(\frac{1}{n}\sum_{j=1}^{n} U_j\right)^2.$$

b) Montrer que lorsque (U_n) est un processus stationnaire d'espérance m

$$\mathbb{E}(\widehat{\gamma}_U(0)) = \gamma_U(0) + m^2 - \frac{1}{n^2}\left(n(\gamma_U(0) + m^2) + 2\sum_{p=1}^{n-1}(n-p)(\gamma_U(p) + m^2)\right).$$

c) A partir de l'équation précédente, montrer que :

$$\mathbb{E}(\widehat{\gamma}_U(0)) = \gamma_U(0) - \frac{1}{n^2}\left(n\gamma_U(0) + 2\sum_{p=1}^{n-1}(n-p)\gamma_U(p)\right).$$

d) En supposant que $\sum_{p\geq 1}|\gamma_U(p)| < \infty$, montrer que :

$$\lim_{n\to\infty} \mathbb{E}(\widehat{\gamma}_U(0)) = \gamma_U(0).$$

3. On souhaite prédire les valeurs des niveaux du lac Huron pour les années suivantes. Comment utiliseriez-vous la modélisation proposée dans cette partie pour obtenir une telle prédiction ?

FIN DU SUJET