

## Examen

Durée 2h. Documents fournis sur le site du cours et documents sur papier autorisés. Autres documents et accès à internet interdits. La barème est à titre indicatif et peut encore changer. Il faut que vos programmes soient **exécutables** par la commande `python3 nomdufichier.py` et qu'ils **affichent leurs résultats** (notamment, ils ne doivent pas dépendre de l'environnement `Spyder` pour fonctionner correctement).

### Exercice 1. Programmation élémentaire, graphisme (7 points)

Réalisez une fonction `L(n, x)` qui renvoie le  $n$ -ième *polynôme de Laguerre* évalué à  $x$ . Tracez  $L_n(x)$  pour  $n = 1, 2, 3$  sur l'intervalle  $x \in [0, 5]$  (dans un seul graphique).

Les polynômes de Laguerre sont définis par

$$L_0(x) = 1, \quad L_1(x) = 1 - x, \quad L_n(x) = \frac{(2n - 1 - x)L_{n-1}(x) - (n - 1)L_{n-2}(x)}{n} \quad (n \geq 2).$$

Pour cet exercice vous n'avez pas le droit de vous servir de la bibliothèque `scipy`. Vous pouvez utiliser soit une méthode récursive, soit une méthode itérative à votre choix.

### Exercice 2. Recherche des zéros (6 points)

Réalisez un programme qui trouve tous les zéros de la fonction  $f(x) = \sin(x) - \frac{1}{x}$  dans l'intervalle  $x \in ]0, 10]$  avec une précision d'au moins  $10^{-9}$ .

*Indication* : Il peut être utile de réaliser d'abord un programme auxiliaire qui trace  $f(x)$  afin de repérer les valeurs approximatives des zéros, mais vous n'êtes pas obligés.

### Exercice 3. Algèbre linéaire et ajustement (7 points)

On modélise l'ensoleillement mensuel moyen  $E(m)$  à Montpellier par une fonction qui prend en compte la position du soleil (par temps clair, un jour d'été sera plus ensoleillé qu'un jour d'hiver) ainsi que la couverture nuageuse (il y a normalement peu de nuages en juillet-août). Ainsi, pour les mois  $m \in \{1, 2, \dots, 12\}$

$$E(m) = \beta_0 + \beta_1 \cos\left(\frac{2\pi(m - 6)}{12}\right) + \beta_2 \chi(m), \quad \chi(m) = \begin{cases} 0, & m = 7 \text{ ou } m = 8 \\ 1, & \text{autrement} \end{cases}$$

avec  $\beta_0$ ,  $\beta_1$  et  $\beta_2$  des paramètres.

- Réalisez un programme qui ajuste les paramètres  $\beta_{0,1,2}$  par une régression linéaire. Vous importerez les données du fichier `ens.dat` qui est disponible sur Moodle.

*Indication* : Pour définir la fonction  $\chi(m)$ , il peut être utile d'employer `numpy.vectorize`.

La *décomposition de Cholesky* d'une matrice réelle, symétrique et définie positive  $B$  est donnée par  $B = U^T U$  avec  $U$  une matrice triangulaire supérieure.

- Rendez-vous compte que, si  $U$  est connue, le système linéaire  $B\vec{x} = \vec{b}$  peut être efficacement résolu pour tout  $\vec{b}$ . Par un commentaire dans votre programme, décrivez brièvement ce qu'un programme fera pour calculer  $\vec{x}$  étant donné  $U$  et  $\vec{b}$ . Vous pouvez faire référence aux résultats du cours.
- On obtient  $U$  avec les commandes `import scipy.linalg as la; U = la.cholesky(B)`. Compte tenu du fait que la matrice  $A^T A$  dans les équations normales de la régression linéaire est symétrique et définie positive, refaites le calcul de 1. avec la décomposition de Cholesky.  
*Indication* : Il convient de se servir de la fonction `resoudre(P, L, U, b)` du cours.