TP 4 : Décompositions LU et Cholesky.

On cherche à résoudre le système $Ax = b$ où $A \in \mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n$.

1) Écrire deux fonctions `solveTinf` et `solveTsup` permettant respectivement de résoudre le système $Ax = b$ dans le cas où la matrice A est triangulaire inférieure, et dans le cas où A est triangulaire supérieure.

On pourra commencer par écrire les algorithmes sur le papier avant de les programmer. On vérifiera le bon fonctionnement des méthodes sur des matrices de petite taille.

2) Écrire une fonction `factorLU` qui prend en entrée une matrice carrée A et qui renvoie les matrices L, U de sa factorisation LU. On rappelle le pseudo code de cette décomposition [1] :

pour $i = 1, \dots, n$ **faire**
  pour $k = i, \dots, n$ **faire**
    $U_{ik} \leftarrow A_{ik} - \sum_{j=1}^{i-1} L_{ij} U_{jk}$
  fin pour
  $L_{ii} \leftarrow 1$
  pour $k = i+1, \dots, n$ **faire**
    $L_{ki} \leftarrow \frac{1}{U_{ii}} \left(A_{ki} - \sum_{j=1}^{i-1} L_{kj} U_{ji} \right)$
  fin pour
fin pour

On fera en sorte que la fonction renvoie un message d'erreur si un des pivots U_{ii} est nul [2]. Ici aussi, on vérifiera que la fonction marche bien sur des exemples.

3)

1. Écrire une fonction `solveLU` permettant de résoudre le système $Ax = b$ en passant par la factorisation $A = LU$. La fonction doit s'écrire en trois instructions.
2. Vérifier que cette fonction marche bien sur des exemples (on pourra comparer avec la solution donnée par `A\b`, ou bien mesurer la norme de $Ax - b$.
3. Résoudre le système lorsque $A = \begin{pmatrix} 10^{-17} & 1 \\ 1 & 1 \end{pmatrix}$, $b = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$. Que constate-t-on ? Essayer, en analysant le code de la fonction `solveLU`, d'où vient le problème. On pourra également remplacer A_{11} par 10^{-n} pour d'autres valeurs de n.

4)

1. Une somme réalisée sur un ensemble d'indices vide est égale à zéro.
2. Pour afficher un message d'erreur, et arrêter en même temps l'exécution de la fonction en cours, on utilisera la fonction `error`.

1. Écrire une fonction qui prend en entrée un entier n, et renvoie la matrice du laplacien en dimension n, définie par

$$\frac{1}{h^2}\begin{pmatrix} 2 & -1 & 0 & \cdots & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \cdots & 0 & -1 & 2 & -1 \\ 0 & \cdots & \cdots & 0 & -1 & 2 \end{pmatrix}, \qquad h = \frac{1}{n+1}.$$

   On pourra réutiliser le code d'une fonction du TP1.

2. Considérons le système $Ax = b$ lorsque A est une matrice du laplacien, et b est un vecteur dont tous les coefficients sont 1. Utiliser la fonction `solveLU` pour le résoudre, et mesurer le temps de calcul pour trouver la solution. On utilisera pour cela la fonction `timer`. Comparer avec le temps pris par la fonction `A\b`. Que constate-t-on ?

3. En gardant le contexte de la question précédente, écrire une fonction `plotLUtesterror` qui prend en entrée un entier n, et qui retourne deux quantités : le temps de calcul de la fonction `solveLU`, et l'écart avec la solution du système `A\b` (en norme $\|\cdot\|_\infty$).

4. Utiliser la fonction `plotLUtesterror` pour tracer le temps de calcul de la fonction `solveLU` en fonction de la taille de la matrice (On pourra faire varier n entre 1 et 150).

5. Vérifier expérimentalement que le temps de calcul de la méthode est de l'ordre de $\mathcal{O}(n^3)$. Noter c'est équivalent à vérifier que la racine cubique du temps évolue en $\mathcal{O}(n)$. On pourra utiliser la commande `[a,b]=reglin(X,Y)` qui prend en entrée deux vecteurs ligne X, Y et renvoie les coefficients de la régression linéaire de Y par X, de telle manière que $Y \simeq aX + b$.

5)

1. Écrire une fonction `factorCholesky` qui prend en entrée une matrice symétrique définie positive A et qui renvoie la matrice L de sa factorisation de Cholesky, telle que $A = LL^*$. On rappelle le pseudo code pour l'implémentation de cette décomposition [3] :

   pour $i = 1, \ldots, n$ **faire**
   pour $k = i, \ldots, n$ **faire**
   $L_{ii} \leftarrow \sqrt{A_{ii} - \sum_{j=1}^{i-1} L_{ij}^2}$
   fin pour
   pour $k = i+1, \ldots, n$ **faire**
   $L_{ki} \leftarrow \frac{1}{L_{ii}} \left(A_{ki} - \sum_{j=1}^{i-1} L_{kj} L_{ij} \right)$
   fin pour
   fin pour

   Vérifier que la fonction marche sur des exemples [4].

2. Reprendre les questions 3)1. et 3)2. avec la décomposition de Cholesky.

3. Reprendre les questions de 4)1. à 4)4. avec la décomposition de Cholesky. On comparera LU vs. Cholesky en termes de temps de calcul, et de précision.

3. Ici aussi, une somme réalisée sur un ensemble d'indices vide est égale à zéro.
4. Rappelons que toute matrice $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive s'écrit $A = BB^*$ où $B \in GL_n(\mathbb{R})$.