Corrections d'exercices sur le chapitre 2 (feuilles de TD 2 et 3)

Exercice 6, TD 2, *Importance du pivotage dans la méthode de Gauss*

Montrer sur l'exemple ci-dessous que la méthode de Gauss sans pivotage donne des résultats catastrophiques si l'on considère une machine hypothétique travaillant avec l' ensemble de flottants $\mathbb{F}(10, 4, ., .)$ et une arithmétique de type IEEE (c-a-d que si u et v sont 2 flottants alors $u \odot v = fl(u \cdot v)$, tant que l'on ne rencontre pas d'*overflow* ou d'*underflow*, $\cdot$ désignant l'une des 4 opérations et $\odot$ l'opération machine correspondante).

$$\left\{ \begin{array}{rcl} 5 \cdot 10^{-5} x + y & = & 1 \\ x - y & = & 0 \end{array} \right.$$

réponse : On remarque que tous les coefficients initiaux sont justes dans le système flottant utilisé. Sur ce système 2×2 la méthode de Gauss ne comporte qu'une seule étape qui consiste à modifier la deuxième équation selon $L_2^{(n)} := L_2 - (1/(5 \cdot 10^{-5}))L_1$. Ces opérations sont effectées dans le système flottant mais on force néanmoins le coefficient qui doit être nul (celui devant x) à bien l'être. La deuxième équation devient donc :

$$(-1 \ominus (1 \oslash 5 \cdot 10^{-5}) \otimes 1)y = 0 \ominus (1 \oslash 5 \cdot 10^{-5}) \otimes 1$$

Pour obtenir les résultats, on rappelle que les calculs en arithmétique flottante consistent à faire comme si on faisait des calculs exacts suivi de l'application de l'opérateur fl, c-a-d arrondi au nombre flottant le plus proche, ce qui nous donne ici :

$$\begin{array}{rcl} 1 \oslash 5 \cdot 10^{-5} & = & fl(1/(5 \cdot 10^{-5})) = fl(2 \cdot 10^4) = 2 \cdot 10^4 \\ (1 \oslash 5 \cdot 10^{-5}) \otimes 1 & = & 2 \cdot 10^4 \\ -1 \ominus ((1 \oslash 5 \cdot 10^{-5}) \otimes 1) & = & -1 \ominus 2 \cdot 10^4 = fl(-20001) = -20000 \end{array}$$

Ainsi on se retrouve avec le système triangulaire suivant :

$$\left\{ \begin{array}{rcl} 5 \cdot 10^{-5} x + y & = & 1 \\ -20000 y & = & -20000 \end{array} \right.$$

La résolution en machine du système triangulaire nous donne donc :

$$y = -20000 \oslash -20000 = fl(-20000/-20000) = fl(1) = 1$$

puis on calcule x par :

$$x = (1 \ominus y) \oslash 5 \cdot 10^{-5} = (1 \ominus 1) \oslash 5 \cdot 10^{-5} = 0$$

Ainsi la solution numérique calculée est $x = 0, y = 1$ alors que la solution exacte est

$$x = y = \frac{1}{1 + 5 \cdot 10^{-5}} = \frac{20000}{20001} \simeq 0.99995000249987500062496875156$$

Dans le système flottant utilisé, la meilleure approximation de la solution serait donc $x = y = 1$. Montrer qu'on obtient bien cette solution par la méthode de Gauss à pivot partiel.

Exercice 6, TD 2, *Factorisation de Cholesky*

Soit une matrice $A \in \mathcal{M}_{n,n}(\mathbb{R})$ symétrique et définie positive.

1. Montrer qu'une telle matrice est nécessairement inversible (aide : montrer que son noyau ne contient que le vecteur nul $KerA = \{0\}$).

   réponse : Le noyau de A est l'ensemble des vecteurs x tels que $Ax = 0$. Supposons qu'il existe un vecteur $y \neq 0$ dans le noyau de A. Alors du fait que $Ay = 0$ on obtient $(y|Ay) = 0$ mais comme $y \neq 0$ et A définie positive on a $(y|Ay) > 0$ ce qui est absurde. Donc le noyau ne contient que le vecteur nul. Comme A est carrée cela implique l'inversibilité de A.$\square$

2. On admettra qu'une telle matrice admet une factorisation dite de Cholesky de la forme $A = CC^\top$ où C est une matrice triangulaire inférieure dont les éléments diagonaux sont tous strictement positifs ($C_{i,i} > 0$). Le but de cette question est d'écrire un algorithme pour calculer C.

   (a) En utilisant la formule du produit matriciel et en exploitant la forme triangulaire inférieure de C ($C_{i,j} = 0$ pour $j > i$) montrer que :

$$A_{i,j} = \sum_{k=1}^{\min\{i,j\}} C_{i,k} C_{j,k} \tag{1}$$

   réponse : D'après la formule du produit matriciel et de la définition de la transposée d'une matrice :

$$A_{i,j} = \sum_{k=1}^{n} C_{i,k}(C^\top)_{k,j} = \sum_{k=1}^{n} C_{i,k} C_{j,k}$$

   il reste à exploiter le fait que C est triangulaire inférieure c'est à dire qu'on a $C_{i,k} = 0$ dès que $k > i$ et $C_{j,k} = 0$ dès que $k > j$ et donc le produit $C_{i,k}C_{j,k}$ est nul dès que $k > \min\{i,j\}$.$\square$

   (b) Spécifier la formule obtenue pour $i = 1$ et $j = 1$ et en déduire la valeur de $C_{1,1}$.

   réponse : $A_{1,1} = \sum_{k=1}^{1} C_{1,k} C_{1,k} = C_{1,1}^2$, d'où $C_{1,1} = \sqrt{A_{1,1}}$.$\square$

   (c) Montrer qu'une fois $C_{1,1}$ connu on peut en déduire $C_{i,1}$ pour $i = 2, \ldots, n$.

   réponse : On a $A_{i,1} = \sum_{k=1}^{1} C_{i,k} C_{1,k} = C_{i,1} C_{1,1}$ et donc $C_{i,1} = A_{i,1}/C_{1,1}$ pour $i = 2, \ldots, n$.$\square$

   (d) Déduire de (1) que :

$$A_{j,j} = \sum_{k=1}^{j-1} C_{j,k}^2 + C_{j,j}^2, \text{ et pour } i > j, \quad A_{i,j} = \sum_{k=1}^{j-1} C_{i,k} C_{j,k} + C_{i,j} C_{j,j}.$$

   réponse : Il suffit d'écrire la formule (1) en séparant le dernier terme :

$$A_{j,j} = \sum_{k=1}^{j} C_{j,k}^2 = \sum_{k=1}^{j-1} C_{j,k}^2 + C_{j,j}^2$$

   et pour $i > j$:

$$A_{i,j} = \sum_{k=1}^{j} C_{i,k} C_{j,k} = \sum_{k=1}^{j-1} C_{i,k} C_{j,k} + C_{i,j} C_{j,j}.\square$$

   (e) Déduire du résultat précédent qu'on peut obtenir C en utilisant un algorithme qui calcule successivement les colonnes de C (première colonne, puis seconde, etc...) chaque colonne j étant calculée en commençant par le coefficient diagonal $C_{j,j}$ puis les autres. Aide : il suffit de montrer que les autres coefficients de C dont on a besoin pour obtenir $C_{j,j}$ puis $C_{i,j}$ pour $i > j$ ont déjà été calculés.

   réponse : De la première des deux formules précédentes on tire que :

$$C_{j,j} = \sqrt{A_{j,j} - \sum_{k=1}^{j-1} C_{j,k}^2}$$

où l'on remarque que le membre de droite est fonction des coefficients $C_{j,1}$, $C_{j,2}$, ..., $C_{j,j-1}$ donc connus lorsqu'on calcule la colonne j (ayant au préalable calculé les coefficients des colonnes précédentes). De même pour $i = j+1, \dots, n$ la deuxième formule nous donne :

$$C_{i,j} = \frac{A_{i,j} - \sum_{k=1}^{j-1} C_{i,k} C_{j,k}}{C_{j,j}}$$

ce qui est calculable si les colonnes précédentes de C ont déjà été obtenues ainsi que le coefficient diagonal $C_{j,j}$.□

(f) Ecrire l'algorithme correspondant.

réponse :

réserver un tableau C de taille $n \times n$ et l'initialiser à 0
pour j de 1 à n
    $C_{j,j} \leftarrow A_{j,j}$
    pour k de 1 à $j-1$
        $C_{j,j} \leftarrow C_{j,j} - C_{j,k}^2$
    $C_{j,j} \leftarrow \sqrt{C_{j,j}}$
    pour i de $j+1$ à n
        $C_{i,j} \leftarrow A_{i,j}$
        pour k de 1 à $j-1$
            $C_{i,j} \leftarrow C_{i,j} - C_{i,k} C_{j,k}$
        $C_{i,j} \leftarrow C_{i,j} / C_{j,j}$

□

(g) Comment résoudre un système linéaire $Ax = b$ lorsqu'on connaît la factorisation de Cholesky de A ?

réponse : $Ax = b$ est donc équivalent à $CC^\top x = b$. On pose $y = C^\top x$ et on résoud successivement les deux systèmes triangulaires :

i. $Cy = b$ (système triangulaire inférieur), on obtient y ;
ii. $C^\top x = y$ (système triangulaire supérieur), on obtient x.

Exercice 3, TD 3, *question 1 (norme matricielle 1) et question 3*

question 1 Montrer que :

$$\|A\|_1 = \max_j \sum_i |a_{i,j}|$$

réponse : On part de :

$$\|A\|_1 := \max_{\|x\|_1=1} \|Ax\|_1$$

Il vient :

$$\begin{aligned}
\|Ax\|_1 &= \sum_i |(Ax)_i| \\
&= \sum_i \left| \sum_j a_{i,j} x_j \right| \\
&\leq \sum_i \sum_j |a_{i,j}||x_j| \\
&\leq \sum_j \sum_i |a_{i,j}||x_j| = \sum_j |x_j| \sum_i |a_{i,j}| \\
&\leq \sum_j |x_j| \max_j \sum_i |a_{i,j}| = \max_j \sum_i |a_{i,j}| \sum_j |x_j| = \max_j \sum_i |a_{i,j}|
\end{aligned}$$

car $\sum_j |x_j| = \|x\|_1 = 1$.

Il reste à montrer que la borne supérieure :

$$B = \max_j \sum_i |a_{i,j}|$$

peut être atteinte par un vecteur particulier (de norme 1 égale à 1). Soit j^* tel que :

$$B = \sum_i |a_{i,j^*}|$$

Prenons x^* avec toutes ses composantes nulles sauf la j^* égale à 1. La norme 1 de ce vecteur est bien égale à 1 et $Ax^* = A^{j^*}$ soit $(Ax^*)_i = a_{i,j^*}$, d'où :

$$\|Ax^*\|_1 = \sum_i |a_{i,j^*}| = B$$

□

question 3 Soit A une matrice réelle non nulle et son codage en machine $\hat{A}$ (supposé sans dépassement de capacité) on pose $\delta A = \hat{A} - A$. Déduire de la question 1 que :

$$\frac{\|\delta A\|_1}{\|A\|_1} \leq \mathbf{u}, \quad \text{et } \frac{\|\delta A\|_\infty}{\|A\|_\infty} \leq \mathbf{u}$$

réponse : comme il n'y a pas de dépassement de capacité, on a $(\hat{A})_{i,j} = a_{i,j}(1+\epsilon_{i,j})$ avec $|\epsilon_{i,j}| \leq \mathbf{u}$. D'où $(\delta A)_{i,j} = a_{i,j}\epsilon_{i,j}$. D'après un résultat de la question 1 :

$$\|\delta A\|_\infty = \max_i \sum_j |a_{i,j}\epsilon_{i,j}|$$

Le max est obtenu pour (au moins) un indice i^*, d'où :

$$\begin{aligned}
\|\delta A\|_\infty &= \sum_j |a_{i^*,j}||\epsilon_{i^*,j}| \\
&\leq \sum_j |a_{i^*,j}|\mathbf{u} = \mathbf{u} \sum_j |a_{i^*,j}| \leq \mathbf{u} \max_i \sum_j |a_{i,j}| = \mathbf{u}\|A\|_\infty
\end{aligned}$$

et donc $\|\delta A\|_\infty / \|A\|_\infty \leq \mathbf{u}$ si A n'est pas la matrice nulle. On procède de manière équivalente pour la norme 1.

□