Analyse numérique matricielle

Licence 3 Mathématiques

David Manceau

Table des matières

Première partie

Rappels et compléments d'algèbre linéaire

Chapitre 1

Applications linéaires

1.1 Généralités

On suppose connues les notions basiques d'espaces vectoriels et de matrices. Dans toute la suite on emploiera les notations données ci-dessous.

Notations 1.1.

1. On note $\mathbb{K} := \mathbb{R}$ ou $\mathbb{C}$.
2. n et p désignent deux entiers naturels non nuls.
3. On désigne par $\mathcal{M}_{n,p}(\mathbb{K})$ le $\mathbb{K}$-espace vectoriel des matrices à n lignes et p colonnes à coefficients dans $\mathbb{K}$. Dans le cas $n = p$, on note simplement $\mathcal{M}_n(\mathbb{K}) := \mathcal{M}_{n,n}(\mathbb{K})$.
4. Un vecteur x de $\mathbb{K}^n$ est un n-uplet $x := (x_1, \ldots, x_n)$, où $x_i \in \mathbb{K}$ pour tout $i = 1, \ldots, n$ et pourra être identifié à un vecteur colonne appartenant à $\mathcal{M}_{n,1}(\mathbb{K})$.
5. Si $A \in \mathcal{M}_{n,p}(\mathbb{K})$, on note $A^T \in \mathcal{M}_{p,n}(\mathbb{K})$ sa transposée.

Définition 1.2. Soient E et F deux $\mathbb{K}$-espaces vectoriels. Une application u de E dans F est dite **linéaire** si

$$\forall\ \lambda \in \mathbb{K},\ \forall\ x, y \in E, \quad u(\lambda\, x + y) = \lambda\, u(x) + u(y).$$

On note $L(E, F)$ l'**ensemble des applications linéaires** de E dans F. Lorsque $E := F$, on note simplement $L(E) := L(E, E)$ et les éléments de $L(E)$ sont appelés des **endomorphismes** de E.

On rappelle le lien entre applications linéaires en dimension finie et matrices. Soient E et F deux $\mathbb{K}$-espaces vectoriels de dimensions finies et de bases respectives $\mathcal{B}_E := \{e_1, \ldots, e_p\}$ et $\mathcal{B}_F := \{f_1, \ldots, f_n\}$. Soit $u \in L(E, F)$. Si $x \in E$, il existe $x_1, \ldots, x_p \in \mathbb{K}$ tels que

$$x := \sum_{j=1}^{p} x_j e_j.$$

Alors, par linéarité, on a

$$u(x) = \sum_{j=1}^{p} x_j u(e_j).$$

Puisque, pour tout $j \in \{1, \dots, p\}$, $u(e_j) \in F$, il existe $a_{ij} \in \mathbb{K}$, $i \in \{1, \dots, n\}$ tels que

$$u(e_j) = \sum_{i=1}^{n} a_{ij} f_i. \tag{1.1.1}$$

On en déduit

$$u(x) = \sum_{i=1}^{n} \sum_{j=1}^{p} a_{ij} x_j f_i. \tag{1.1.2}$$

Définition 1.3. Soient E et F deux $\mathbb{K}$-espaces vectoriels de dimensions finies et de bases respectives $\mathcal{B}_E := \{e_1, \dots, e_p\}$ et $\mathcal{B}_F := \{f_1, \dots, f_n\}$. Soit $u \in L(E, F)$. On appelle **matrice associée** à u dans les bases $\mathcal{B}_E$ et $\mathcal{B}_F$ la matrice notée $\mathrm{Mat}(u \mid \mathcal{B}_E, \mathcal{B}_F) \in \mathcal{M}_{n,p}(\mathbb{K})$ dont les coefficients sont les a_{ij} donnés par (1.1.1). Si $A \in \mathcal{M}_{n,p}(\mathbb{K})$ a pour coefficients a_{ij}, on définit **l'application linéaire associée** u à A dans les bases $\mathcal{B}_E$ et $\mathcal{B}_F$ par (1.1.2).

Rappelons maintenant la formule donnant le produit de deux matrices à partir de la formule (1.1.2). Soient E, F, G trois $\mathbb{K}$-espaces vectoriels de dimension finie et de bases respectives $\mathcal{B}_E := \{e_1, \dots, e_p\}$, $\mathcal{B}_F := \{f_1, \dots, f_n\}$ et $\mathcal{B}_G := \{g_1, \dots, g_q\}$. Soient $u \in L(E, F)$ et $v \in L(G, E)$. Alors, $w := u \circ v \in L(G, F)$. En effet, si $\lambda \in \mathbb{K}$, $x, y \in G$, on a

$$w(\lambda\, x + y) = u\,(v(\lambda\, x + y)) = u\,(\lambda\, v(x) + v(y)) = \lambda\, u\,(\lambda v(x)) + u\,(v(y)) = \lambda\, w(x) + w(y).$$

On pose

$$A := \mathrm{Mat}(u \mid \mathcal{B}_E, \mathcal{B}_F) = (a_{ij})_{\substack{1 \le i \le n \\ 1 \le j \le p}} \quad \text{et} \quad B := \mathrm{Mat}(v \mid \mathcal{B}_G, \mathcal{B}_E) = (b_{ij})_{\substack{1 \le i \le p. \\ 1 \le j \le q}}$$

La **matrice produit** $C := AB$ est définie par $C = \mathrm{Mat}(u \circ v \mid \mathcal{B}_G, \mathcal{B}_F) = (c_{ij})_{\substack{1 \le i \le n. \\ 1 \le j \le q}}$

Déterminons ces coefficients c_{ij}. Soit $x \in G$. On a $x := \sum_{j=1}^{p} x_j g_j$, où $x_1, \dots, x_q \in \mathbb{K}$. Alors,

$$\begin{aligned} w(x) \;&= u\,(v(x)) = u\left(\sum_{k=1}^{p}\sum_{j=1}^{q} b_{kj} x_j e_k\right) = \sum_{k=1}^{p}\sum_{j=1}^{q} b_{kj} x_j u(e_k) \\ &= \sum_{k=1}^{p}\sum_{j=1}^{q} b_{kj} x_j \sum_{i=1}^{n} a_{ik} f_i = \sum_{j=1}^{q}\sum_{i=1}^{n}\left(\sum_{k=1}^{p} a_{ik} b_{kj}\right) x_j f_i. \end{aligned}$$

D'autre part, on a

$$w(x) = \sum_{i=1}^{n}\sum_{j=1}^{q} c_{ij} x_j f_i.$$

Puisque $\{f_1, \dots, f_n\}$ est une base de F, on en déduit

$$c_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj}, \tag{1.1.3}$$

pour tous $i \in \{1, \dots, n\}$ et $j \in \{1, \dots, q\}$.

Définitions 1.4. Soient E, F deux $\mathbb{K}$-espaces vectoriels et $u \in L(E,F)$.

1. Le **noyau** de u est le sous-espace vectoriel $\mathrm{Ker}(u)$ de E défini par
$$\mathrm{Ker}(u) \ := \ \{x \in E \mid u(x) = 0\} = u^{-1}(\{0\}).$$
2. L'**image** de u est le sous-espace vectoriel $\mathrm{Im}(u)$ de F défini par
$$\mathrm{Im}(u) \ := \ \{y \in F \mid \exists\, x \in E,\ y = u(x)\} = \{u(x) \mid x \in E\} = u(E).$$
3. Le **rang** de u est la dimension de $\mathrm{Im}(u)$, noté $\mathrm{rg}(u)$.

Proposition 1.5. *Soient E, F deux $\mathbb{K}$-espaces vectoriels de même dimension finie n et $u \in L(E,F)$. Alors, u est bijective si et seulement si l'une des conditions suivantes est vérifiée :*
i) $\mathrm{Ker}(u) = \{0\}$ *(injectivité),*
ii) $\mathrm{Im}(u) = F$ *(surjectivité),*
iii) $\mathrm{rg}(u) = \dim(F)$ *(surjectivité bis).*
De plus, on a
$$n = \dim(\mathrm{Ker}(u)) + \mathrm{rg}(u). \tag{1.1.4}$$

Les définitions de noyau, image et rang s'adaptent au cas des matrices par l'identification (1.1.2). En particulier, si $A \in \mathcal{M}_{n,p}(\mathbb{K})$, on a
$$\begin{cases} \mathrm{Ker}(A) = \{x \in \mathcal{M}_{p,1}(\mathbb{K}) \mid Ax = 0\} \quad (\text{ici } 0 \in \mathcal{M}_{n,1}(\mathbb{K})), \\ \mathrm{Im}(A) = \{Ax \mid x \in \mathcal{M}_{p,1}(\mathbb{K})\}, \end{cases}$$
et $\mathrm{rg}(A)$ est le nombre de colonnes linéairement indépendantes de A (ou de lignes car $\mathrm{rg}(A)$=$\mathrm{rg}(A^T)$).

Définition 1.6. Une matrice $A \in \mathcal{M}_n(\mathbb{K})$ est dite **inversible** s'il existe une matrice $B \in \mathcal{M}_n(\mathbb{K})$ telle que $AB = BA = I_n$, où I_n est la matrice identité d'ordre n.

Une matrice est inversible si et seulement si toute application linéaire qui lui est associée est bijective, on a alors le résultat suivant :

Proposition 1.7. *Une matrice $A \in \mathcal{M}_n(\mathbb{K})$ est inversible si et seulement si l'une des conditions suivantes est vérifiée :*
i) $\mathrm{Ker}(A) = \{0\}$,
ii) $\mathrm{Im}(A) = \mathcal{M}_{n,1}(\mathbb{K})$,
iii) il existe $B \in \mathcal{M}_n(\mathbb{K})$ telle que $AB = I_n$ ou $BA = I_n$,
iv) $\det(A) \neq 0$.

Remarque 1.8. On rappelle que si $A \in \mathcal{M}_n(\mathbb{K})$ est inversible, alors son inverse est donné par
$$A^{-1} = \frac{1}{\det(A)}\ \tilde{A}^T, \tag{1.1.5}$$
où $\tilde{A}$ est la **matrice des cofacteurs** de A dont les coefficients sont donnés par
$$\tilde{a}_{ij} \ := \ (-1)^{i+j} \det(A^{ij}),$$
avec $i, j \in \{1, \ldots, n\}$ et A^{ij} la matrice obtenue en supprimant la i-ème ligne et la j-ème colonne de A.

1.2 Matrices similaires et changement de bases

Soient E et F deux $\mathbb{K}$-espaces vectoriels de dimensions finies p et n. Soient $\mathcal{B}_E, \mathcal{B}'_E$ deux bases de E et $\mathcal{B}_F, \mathcal{B}'_F$ deux bases de F. On considère

$$u \in L(E,F), \quad A := \mathrm{Mat}(u \mid \mathcal{B}_E, \mathcal{B}_F) \quad \text{et} \quad B := \mathrm{Mat}(u \mid \mathcal{B}'_E, \mathcal{B}'_F).$$

On va étudier le lien existant entre A et B.

Définitions 1.9.

1. Les matrices A et B sont dites **similaires**. Dans le cas où $E = F$, $\mathcal{B}_E = \mathcal{B}_F$ et $\mathcal{B}'_E = \mathcal{B}'_F$, on dit que A et B sont **semblables**.
2. Soit $P \in \mathcal{M}_n(\mathbb{K})$ la matrice associée à l'application identité id_E de E dans les bases $\mathcal{B}'_E$ et $\mathcal{B}_E$, *i.e.*
$$P := \mathrm{Mat}(id_E \mid \mathcal{B}'_E, \mathcal{B}_E). \tag{1.2.1}$$
On dit que P est la **matrice de passage** de la base $\mathcal{B}_E$ à la base $\mathcal{B}'_E$.

Autrement dit, si un vecteur de E a pour coordonnées $x' = (x'_1, \ldots, x'_p)^T$ dans la base $\mathcal{B}'_E$, alors ses coordonnées $x = (x_1, \ldots, x_p)^T$ dans la base $\mathcal{B}_E$ sont données par $x = Px'$.

Proposition 1.10. *Si P est la matrice de passage de la base $\mathcal{B}_E$ à la base $\mathcal{B}'_E$, alors P est inversible et P^{-1} est la matrice de passage de la base $\mathcal{B}'_E$ à la base $\mathcal{B}_E$. De plus, toute matrice inversible est une matrice de changement de base.*

Remarque 1.11. Si $\mathcal{B}_E := \{e_1, \ldots, e_p\}$ et $\mathcal{B}'_E := \{e'_1, \ldots, e'_p\}$, il existe $\alpha_{ij} \in \mathbb{K}$, $1 \leq i, j \leq p$, tels que

$$\forall\, j \in \{1, \ldots, p\}, \quad e'_j = \sum_{i=1}^{p} \alpha_{ij} e_i. \tag{1.2.2}$$

Alors, $P = (\alpha_{ij})_{1 \leq i,j \leq p}$ d'après (1.2.1) et (1.1.1).

Théorème 1.12 (Théorème de changement de base)**.**
Soit P la matrice de passage de $\mathcal{B}_E$ à $\mathcal{B}'_E$ et Q la matrice de passage de $\mathcal{B}_F$ à $\mathcal{B}'_F$. Alors, on a

$$B = Q^{-1}AP.$$

Démonstration. En effet, on a

$$\begin{aligned} Q^{-1}AP &= \mathrm{Mat}(id_F \mid \mathcal{B}_F, \mathcal{B}'_F)\, \mathrm{Mat}(u \mid \mathcal{B}_E, \mathcal{B}_F)\, \mathrm{Mat}(id_E \mid \mathcal{B}'_E, \mathcal{B}_E) \\ &= \mathrm{Mat}(id_F \mid \mathcal{B}_F, \mathcal{B}'_F)\, \mathrm{Mat}(u \mid \mathcal{B}'_E, \mathcal{B}_F) \\ &= \mathrm{Mat}(u \mid \mathcal{B}'_E, \mathcal{B}'_F). \end{aligned}$$

□

Exemple 1.13. Soient $E, F := \mathbb{K}^3$ et $u \in L(E)$ telle que sa matrice dans la base canonique $\mathcal{B}_E$ est

$$A := \begin{pmatrix} 1 & 0 & 4 \\ 2 & 5 & -3 \\ -1 & 3 & 2 \end{pmatrix}.$$

On considère la base de $\mathbb{R}^3$ donnée par $\mathcal{B}'_E := \{e'_1, e'_2, e'_3\}$, où

$$e'_1 := (1,1,1),\ e'_2 := (1,0,-1) \quad \text{et} \quad e'_3 := (1,1,2).$$

Alors, P est donnée par

$$P := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & -1 & 2 \end{pmatrix}.$$

Donc $B = \mathrm{Mat}(u \mid \mathcal{B}'_E, \mathcal{B}'_E)$ est donnée par

$$B = P^{-1}AP = \begin{pmatrix} 6 & -18 & -6 \\ 0 & -7 & 6 \\ -1 & 14 & 9 \end{pmatrix}.$$

Corollaire 1.14. *Deux matrices $A, B \in \mathcal{M}_n(\mathbb{K})$ sont semblables si et seulement s'il existe $P \in \mathcal{M}_n(\mathbb{K})$ inversible telle que $A = PBP^{-1}$.*

Proposition 1.15. *Si $A, B \in \mathcal{M}_n(\mathbb{K})$ sont semblables, alors* $\mathrm{tr}(A) = \mathrm{tr}(B)$.

Démonstration. Voir Td. □

1.3 Réduction des endomorphismes

Dans cette section, pour $A \in \mathcal{M}_n(\mathbb{K})$, on va déterminer dans quel cas il est possible de trouver une matrice semblable à A de forme simple (diagonale ou triangulaire).

1.3.1 Définitions et propriétés

Définition 1.16. Soit $A \in \mathcal{M}_n(\mathbb{K})$. On appelle **polynôme caractéristique** de A, noté P_A, le polynôme de degré n défini par

$$\forall\ X \in \mathbb{K}, \quad P_A(X) \ := \ \det(A - XI_n).$$

Remarque 1.17. Si A et B sont semblables, alors $P_A = P_B$ (voir Td). En particulier, si $u \in L(\mathbb{K}^n)$, on définit le polynôme caractéristique P_u de u par $P_u = P_A$, où A est n'importe quelle matrice associée à u dans une base de $\mathbb{K}^n$.

Définition 1.18. Soit $A \in \mathcal{M}_n(\mathbb{K})$. On appelle **valeurs propres** de A (ou de l'application linéaire u associée) les racines de P_A. Si λ est racine de P_A d'ordre m, *i.e.* $P_A(X) = (\lambda - X)^m Q(X)$, où $Q \in \mathbb{K}[X]$ avec $Q(\lambda) \neq 0$, on dit que λ est valeur propre de A de **multiplicité** m.

Proposition 1.19. *Soient $A \in \mathcal{M}_n(\mathbb{K})$ et λ une valeur propre de A. Alors, il existe $x \in \mathcal{M}_{n,1}(\mathbb{K}) \setminus \{0\}$ tel que $Ax = \lambda x$. Le vecteur x est appelé un* **vecteur propre** *associé à la valeur propre λ.*

Démonstration. On a $\det(A - \lambda I_n) = 0$. Donc $A - \lambda I_n$ n'est pas inversible, ce qui entraîne $\mathrm{Ker}(A - \lambda I_n) \neq \{0\}$. □

Proposition 1.20. *Soit $A \in \mathcal{M}_n(\mathbb{K})$. Alors, A est inversible si et seulement si A n'a pas de valeur propre nulle.*

Démonstration. La matrice A a pour valeur propre 0 si et seulement s'il existe $x \in \mathcal{M}_{n,1}(\mathbb{K}) \setminus \{0\}$ tel que $Ax = 0$. Autrement dit, 0 est valeur propre de A si et seulement si $\mathrm{Ker}(A) \neq \{0\}$. □

Définition 1.21. Soient $A \in \mathcal{M}_n(\mathbb{K})$ et λ une valeur propre de A. On appelle **sous-espace propre** associé à la valeur propre λ le sous-espace vectoriel E_λ engendré par les vecteurs propres associés à λ, *i.e.* $E_\lambda := \mathrm{Ker}(A - \lambda I_n)$.

Proposition 1.22. *Soient $A \in \mathcal{M}_n(\mathbb{K})$ et λ une valeur propre de A. Si $dim(E_\lambda) = r$, alors λ est de multiplicité $m \geq r$.*

En particulier si λ est une valeur propre de A de multiplicité m, alors $1 \leq \dim(E_\lambda) \leq m$.

Démonstration. Supposons que $\dim(E_\lambda) = r$. Soit $u \in L(\mathbb{K}^n)$ tel que $A = Mat(u \mid \mathcal{B}, \mathcal{B})$, où $\mathcal{B}$ est la base canonique de $\mathbb{K}^n$. Il existe une base $\{f_1, \ldots, f_r\}$ de E_λ. D'après le théorème de la base incomplète, il existe $f_{r+1}, \ldots, f_n$ des vecteurs de $\mathbb{K}^n$ tels que $\mathcal{B}' = \{f_1, \ldots, f_n\}$ est une base de $\mathbb{K}^n$. Alors, la matrice $A' = \mathrm{Mat}(u \mid \mathcal{B}', \mathcal{B}')$ est de la forme

$$A' = \left(\begin{array}{cccc|c} \lambda & 0 & \cdots & 0 & \\ 0 & \ddots & & \vdots & \\ \vdots & & \ddots & \vdots & A_1 \\ 0 & \cdots & 0 & \lambda & \\ \hline 0 & \cdots & \cdots & 0 & A_2 \end{array}\right).$$

On en déduit $P_A(X) = P_{A'}(X) = (\lambda - X)^r \det(A_2 - X\, I_{n-r})$. Donc λ est de multiplicité supérieure ou égale à r. □

Remarque 1.23. On en déduit que si λ est une valeur propre simple, alors $\dim(E_\lambda) = 1$.

Proposition 1.24. *Les sous-espaces propres d'une matrice A associés a des valeurs propres distinctes sont en somme directe.*

Démonstration. Soient $A \in \mathcal{M}_n(\mathbb{K})$ de valeurs propres distinctes $\lambda_1 \ldots, \lambda_k$. Montrons par récurrence que les espaces propres $E_{\lambda_1}, \ldots, E_{\lambda_i}$ sont en somme directe pour tout $i \in \{1, \ldots, k\}$.

Pour $i = 1$, le résultat est immédiat. Supposons que les espaces propres $E_{\lambda_1}, \ldots, E_{\lambda_{i-1}}$ sont en somme directe pour un $i \in \{1, \ldots, k\}$. Notons $S_i := \bigoplus_{j=1}^{i-1} E_{\lambda_j}$ et soit $x \in S_i \cap E_{\lambda_i}$.

Montrons que $x = 0$. Il existe $x_1 \in E_{\lambda_1}, \ldots, x_{i-1} \in E_{\lambda_{i-1}}$ tels que $x = x_1 + \cdots + x_{i-1}$. D'autre part, $x \in E_{\lambda_i}$ donc $Ax = \lambda_i\, x$. On en déduit

$$\lambda_i\, x = A(x_1 + \cdots + x_{i-1}) = \lambda_1 x_1 + \cdots + \lambda_{i-1} x_{i-1} \quad \text{et} \quad \lambda_i x = \lambda_i x_1 + \cdots + \lambda_i\, x_{i-1}.$$

En soustrayant les deux on obtient $(\lambda_1 - \lambda_i)x_1 + \cdots + (\lambda_{i-1} - \lambda_i)x_{i-1} = 0$. Puisque $E_{\lambda_1}, \ldots, E_{\lambda_{i-1}}$ sont en somme directe, on en déduit $(\lambda_j - \lambda_i)x_j = 0$ pour tout $j \in \{1, \ldots, i-1\}$. Or les valeurs propres considérés sont distinctes, donc $x_j = 0$, pour tout $j \in \{1, \ldots, i-1\}$. D'où $x = x_1 + \cdots + x_{i-1} = 0$. On en déduit que $E_{\lambda_1}, \ldots, E_{\lambda_i}$ sont en somme directe pour tout $1 \leq i \leq k$. □

1.3.2 Diagonalisation

Définition 1.25. On dit que $A \in \mathcal{M}_n(\mathbb{K})$ est **diagonalisable** si A est semblable à une matrice diagonale, *i.e.* il existe $P \in \mathcal{M}_n(\mathbb{K})$ inversible et $D := \mathrm{diag}(\lambda_1, \ldots, \lambda_n) \in \mathcal{M}_n(\mathbb{K})$ telles que $A = PDP^{-1}$.

Remarque 1.26. Si $A \in \mathcal{M}_n(\mathbb{K})$ est diagonalisable, alors A est semblable à $D := \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, où les λ_i sont les valeurs propres (non nécessairement distinctes) de A. En effet, A et D ont le même polynôme caractéristique et $P_D(X) = \prod_{i=1}^{n}(\lambda_i - X)$.

Proposition 1.27. *Une matrice $A \in \mathcal{M}_n(\mathbb{K})$ est diagonalisable si et seulement s'il existe une base de $\mathbb{K}^n$ formée de vecteurs propres de A.*

Démonstration. Soit u l'endomorphisme de $\mathbb{K}^n$ associé à A dans la base canonique de $\mathbb{K}^n$. Alors, A est diagonalisable si et seulement s'il existe une base $\mathcal{B}$ de $\mathbb{K}^n$ telle que

$$\mathrm{Mat}(u \mid \mathcal{B}) = D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n).$$

On note $\mathcal{B} := \{f_1, \ldots, f_n\}$, alors $\mathrm{Mat}(u \mid \mathcal{B}) = D$ si et seulement si $u(f_i) = \lambda_i f_i$, pour tout $i \in \{1, \ldots, n\}$, ce qui signifie que $\mathcal{B}$ est formée de vecteurs propres de u. □

Remarque 1.28. Soit $A \in \mathcal{M}_n(\mathbb{K})$ de valeurs propres distinctes $\lambda_1, \ldots, \lambda_k$, $k \leq n$.

1. Puisque les sous-espaces propres d'une matrice sont en somme directe, on en déduit que A est diagonalisable si et seulement si $\mathbb{K}^n = \bigoplus_{i=1}^{k} E_i$ ce qui équivaut à $\dim(E_1) + \cdots + \dim(E_k) = n$.
2. Si A n'a que des valeurs propres simples $\lambda_1, \ldots, \lambda_n$, alors, pour tout $i \in \{1, \ldots, n\}$, $\dim(E_{\lambda_i}) = 1$. On en déduit que si A n'a que des valeurs propres simples alors A est diagonalisable.

Définition 1.29. On dit qu'un polynôme de $\mathbb{K}[X]$ est **scindé** dans $\mathbb{K}$ s'il admet toutes ses racines dans $\mathbb{K}$. En particulier, pour $\mathbb{K} := \mathbb{C}$, tout polynôme est scindé dans $\mathbb{K}$.

Théorème 1.30 (Caractérisation)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$ telle que son polynôme caractéristique P_A est scindé dans $\mathbb{K}$, i.e.

$$P_A(X) = \prod_{i=1}^{p} (\lambda_i - X)^{m_i},$$

où $\lambda_1, \ldots, \lambda_p \in \mathbb{K}$ sont les valeurs propres deux à deux distinctes de A de multiplicités respectives $m_1, \ldots, m_p$. Alors, A est diagonalisable si et seulement si, pour tout $i \in \{1, \ldots, p\}$, $\dim(E_{\lambda_i}) = m_i$.

Démonstration. S'il existe $i \in \{1, \ldots, p\}$ tel que $\dim(E_{\lambda_i}) < m_i$, alors

$$\sum_{i=1}^{k} \dim(E_{\lambda_i}) < \sum_{i=1}^{n} m_i = n.$$

Donc A n'est pas diagonalisable, ce qui montre le sens direct.

Réciproquement supposons, pour tout $i \in \{1, \ldots, p\}$, $\dim(E_{\lambda_i}) = m_i$. Alors,

$$\sum_{i=1}^{k} \dim(E_{\lambda_i}) = \sum_{i=1}^{n} m_i = n.$$

D'après la Remarque 1.28., on en déduit que A est diagonalisable. □

Exercice 1.31. Soit $A \in \mathcal{M}_3(\mathbb{R})$ donnée par

$$A := \begin{pmatrix} 1 & 2 & -2 \\ 2 & -3 & 2 \\ -2 & 2 & 1 \end{pmatrix}$$

On montrera en Td que A a trois valeurs propres simples $1, 3$ et -5 donc est diagonalisable, puis on calculera P telle que $A = PDP^{-1}$, où $D = \mathrm{diag}(1, 3, -5)$.

1.3.3 Trigonalisation

On a vu que $A \in \mathcal{M}_n(\mathbb{K})$ est diagonalisable si et seulement si, pour toute valeur propre λ de A de multiplicité m, on a $\dim(E_\lambda) = m$. Si $\dim(E_\lambda) \neq m$, est-il quand même possible de réduire A ?

Définition 1.32.

1. On dit qu'une matrice $T := (t_{ij})_{1 \leq i,j \leq n} \in \mathcal{M}_n(\mathbb{K})$ est **triangulaire supérieure** si tous ses coefficients sous la diagonale sont nuls, *i.e.* $t_{ij} = 0$ si $i > j$.
2. On dit que $A \in \mathcal{M}_n(\mathbb{K})$ est **trigonalisable** s'il existe une matrice triangulaire supérieure semblable à A

Remarque 1.33. Si $T \in \mathcal{M}_n(\mathbb{K})$ est triangulaire supérieure, alors T est de la forme

$$T = \begin{pmatrix} \lambda_1 & \cdots & \cdots & \cdots & t_{1n} \\ 0 & \lambda_2 & \cdots & \cdots & t_{2n} \\ 0 & 0 & \ddots & \cdots & \vdots \\ 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \lambda_n \end{pmatrix}. \tag{1.3.1}$$

Donc le polynôme caractéristique de T est $P_T(X) = (\lambda_1 - X)\dots(\lambda_n - X)$. En particulier, si A est trigonalisable alors A est semblable à T où les coefficients diagonaux de T sont les valeurs propres, non nécessairement distinctes, de A.

Théorème 1.34. *Soit $A \in \mathcal{M}_n(\mathbb{K})$. La matrice A est trigonalisable si et seulement si son polynôme caractéristique est scindé dans $\mathbb{K}$.*

Corollaire 1.35. *Toute matrice de $\mathcal{M}_n(\mathbb{C})$ est trigonalisable dans $\mathbb{C}$.*

Corollaire 1.36. *Soit $A \in \mathcal{M}_n(\mathbb{K})$ de polynôme caractéristique scindé dans $\mathbb{K}$. Alors, on a $\operatorname{tr}(A) = \sum_{i=1}^{n} \lambda_i$, où $\lambda_1, \dots, \lambda_n$ sont les valeurs propres, non nécessairement distinctes, de A.*

Définition 1.37. On appelle **bloc de Jordan** associé à $\lambda \in \mathbb{K}$ de taille $m \geq 1$, la matrice $J_m(\lambda) := \lambda I_m + J_m \in \mathcal{M}_m(\mathbb{K})$, où J_m est la $m \times m$ matrice de coefficients

$$(J_m)_{ij} := \begin{cases} 1 & \text{si } j = i+1, \\ 0 & \text{sinon.} \end{cases}$$

i.e. les coefficients de $J_m(\lambda)$ sont λ sur la diagonale, 1 au dessus de la diagonale et 0 partout ailleurs.

Exemple 1.38. Les matrices suivantes sont des blocs de Jordan de tailles respectives 2, 3 et 4

$$J_2(\lambda) := \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}, \quad J_3(\mu) := \begin{pmatrix} \mu & 1 & 0 \\ 0 & \mu & 1 \\ 0 & 0 & \mu \end{pmatrix} \quad \text{et} \quad J_4(\nu) := \begin{pmatrix} \nu & 1 & 0 & 0 \\ 0 & \nu & 1 & 0 \\ 0 & 0 & \nu & 1 \\ 0 & 0 & 0 & \nu \end{pmatrix}.$$

Théorème 1.39 (Théorème de Jordan)**.**
Soit $A \in \mathcal{M}_n(\mathbb{K})$ trigonalisable de valeurs propres $\lambda_1, \dots, \lambda_p$ avec pour multiplicité $m_1, \dots, m_p$. Alors, A est semblable à la matrice triangulaire supérieure T donnée par

$$T := \begin{pmatrix} J_{\lambda_1} & 0 & \cdots & 0 \\ 0 & J_{\lambda_1} & 0 & \vdots \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & J_{\lambda_p} \end{pmatrix},$$

où, pour tout $i \in \{1, \dots, p\}$, J_{λ_i} est le bloc de Jordan de taille m_i associé à la valeur propre λ_i.

Exemple 1.40. Soit $A \in \mathcal{M}_3(\mathbb{R})$ la matrice donnée par

$$A := \begin{pmatrix} 0 & 2 & 2 \\ 1 & 3 & -1 \\ -1 & 3 & 3 \end{pmatrix}.$$

Alors $P_A(X) = (2-X)^3$ qui est scindé dans $\mathbb{R}$ donc il existe $P \in \mathcal{M}_3(\mathbb{R})$ inversible telle que $A = PTP^{-1}$ où T est donnée par

$$T = \begin{pmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{pmatrix}.$$

Déterminons P. Soit $\mathcal{B} := \{e_1, e_2, e_3\}$ la base canonique de $\mathbb{R}^3$ et $u \in L(\mathbb{R}^3)$ l'endomorphisme tel que $A = \mathrm{Mat}(u \mid \mathcal{B})$. Il suffit de déterminer la base $\mathcal{B}' := \{f_1, f_2, f_3\}$ telle que $T = \mathrm{Mat}(u \mid \mathcal{B}')$. On a

$$u(f_1) = 2f_1, \quad u(f_2) = f_1 + 2f_2 \quad \text{et} \quad u(f_3) = f_2 + 2f_3.$$

Alors, f_1 est un vecteur propre de u. On note $f_1 := x_1e_1 + x_2e_2 + x_3e_3$. On obtient

$$(A - 2I_3) \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = 0,$$

donc

$$\begin{cases} x_1 - x_2 - x_3 &= 0 \\ -x_1 + 3x_2 + x_3 &= 0 \end{cases} \Rightarrow \begin{cases} x_2 &= 0 \\ x_1 &= x_3 \end{cases},$$

d'où $f_1 = x_1(e_1 + e_3)$. En particulier, on peut choisir $f_1 = e_1 + e_3$. Pour f_2, on note $f_2 := y_1e_1 + y_2e_2 + y_3e_3$. On a

$$(A - 2I_3) \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix},$$

d'où

$$\begin{cases} -2y_1 + 2y_2 + 2y_3 &= 1 \\ y_1 + y_2 - y_3 &= 0 \\ -y_1 + 3y_2 + y_3 &= 1 \end{cases} \Rightarrow \begin{cases} y_2 &= \frac{1}{4} \\ y_3 &= y_1 + \frac{1}{4} \end{cases}.$$

Alors, $f_2 = y_1(e_1 + e_3) + \frac{1}{4}(e_2 + e_3)$. En particulier, on peut prendre $f_2 = e_1 + \frac{1}{4}e_2 + \frac{5}{4}e_3$. De même, on obtient $f_3 = e_1 + \frac{3}{8}e_2 + \frac{9}{8}e_3$. Finalement un choix possible pour P est

$$P = \frac{1}{8} \begin{pmatrix} 8 & 8 & 8 \\ 0 & 2 & 3 \\ 8 & 10 & 9 \end{pmatrix}.$$

Chapitre 2

Orthogonalité, matrices hermitiennes

2.1 Produit hermitien

Soit $\mathbb{K} := \mathbb{R}$ ou $\mathbb{C}$.

Définition 2.1. Soit E un $\mathbb{K}$-espace vectoriel. On dit qu'une application $(\cdot,\cdot)$ de $E \times E$ dans $\mathbb{R}$ est un **produit hermitien** si

1. $(\lambda\, x + \mu\, y, z) = \lambda(x,z) + \mu(y,z)$,
2. $(x, \lambda\, y + \mu\, z) = \overline{\lambda}(x,y) + \overline{\mu}(x,z)$,
3. $\overline{(x,y)} = (y,x), \quad \forall\ x,y,z \in E,\ \forall\ \lambda,\mu \in \mathbb{K}$.
4. Pour $x \in E$, $(x,x) \geq 0$ et $(x,x) = 0$ si et seulement si $x = 0$.

Si $\mathbb{K} := \mathbb{R}$, on dit aussi que $(\cdot,\cdot)$ est un **produit scalaire**.

Exemple 2.2. Si $E := \mathbb{K}^n$, le produit hermitien canonique $(\cdot,\cdot)$ est défini par

$$(x,y) \ := \ \sum_{i=1}^{n} x_i\, \overline{y_i}, \quad \forall\ x := (x_1,\ldots,x_n),\ y := (y_1,\ldots,y_n) \in \mathbb{R}^n.$$

On note aussi $(x,y) = x \cdot y$ ou encore $(x,y) = x^T y$ en identifiant $\mathbb{K}^n$ et $\mathcal{M}_{n,1}(\mathbb{K})$. Lorsque $\mathbb{K} := \mathbb{R}$, on l'appelle **produit euclidien**. Dans la suite, on utilisera à plusieurs reprises ce produit scalaire avec cette même notation sans en rappeler la définition.

Remarque 2.3. Soit E un $\mathbb{K}$-espace vectoriel muni d'un produit hermitien $(\cdot,\cdot)$. Alors, E est un espace vectoriel normé muni de la norme $\|\cdot\|$ définie par

$$\forall\ x \in E, \quad \|x\| \ := \ \sqrt{(x,x)}. \tag{2.1.1}$$

Proposition 2.4 (Inégalité de Cauchy-Schwarz)**.**
Soit E un $\mathbb{K}$-espace vectoriel muni d'un produit hermitien $(\cdot,\cdot)$. Alors, on a

$$\forall\ x,y \in E, \quad |(x,y)| \leq \|x\|\,\|y\|,$$

où $\|\cdot\|$ est la norme définie par (2.1.1)*.. L'égalité n'a lieu que si x et y sont liés.*

Définition 2.5. Soit E un $\mathbb{K}$-espace vectoriel muni d'un produit hermitien $(\cdot,\cdot)$.

1. Deux vecteurs x et y sont dits **orthogonaux**, noté $x \perp y$, si $(x,y)=0$.
2. Une famille $\{x_1,\dots,x_n\}$ de E est dite **orthogonale** si
$$\forall\ i,j \in \{1,\dots,n\},\ i \neq j, \quad (x_i,x_j)=0$$
3. Une famille $\{x_1,\dots,x_n\}$ de E est dite **orthonormale** si elle est orthogonale et unitaire, *i.e.*
$$\forall\ i,j \in \{1,\dots,n\}, \quad (x_i,x_j)=\delta_{ij}.$$

Proposition 2.6. *Soient E un $\mathbb{K}$-espace vectoriel muni d'un produit hermitien $(\cdot,\cdot)$ et F un sous-espace vectoriel de E de dimension finie. Alors, l'orthogonal*
$$F^\perp := \{x \in E \mid (x,y)=0, \quad \forall\ y \in F\},$$
de F est un sous-espace vectoriel de E. De plus, si E est de dimension finie, on a
$$E = F \oplus F^\perp \quad \textit{et} \quad \dim(E)=\dim(F)+\dim(F^\perp).$$

Proposition 2.7. *Soient E,F deux $\mathbb{K}$-espaces vectoriels de dimensions finies munis des produits hermitiens $(\cdot,\cdot)_E$ et $(\cdot,\cdot)_F$. Alors, pour toute $u \in L(E,F)$, il existe une unique $u^* \in L(F,E)$ telle que*
$$\forall\ x \in E,\ \forall\ y \in F, \quad (u(x),y)_F = (x,u^*(y))_E.$$
*On dit que u^* est l'***adjointe** *de u.*

Proposition 2.8. *Soient $m,n \in \mathbb{N}$, $u \in L(\mathbb{K}^n,\mathbb{K}^m)$, $\mathcal{B}$ et $\mathcal{B}'$ les bases canoniques de $\mathbb{K}^n$ et $\mathbb{K}^m$. On note $A := \mathrm{Mat}(u \mid \mathcal{B},\mathcal{B}') \in \mathcal{M}_{m,n}(\mathbb{K})$ et $A^* := \mathrm{Mat}(u^* \mid \mathcal{B}',\mathcal{B}) \in \mathcal{M}_{n,m}(\mathbb{K})$. Alors, on a $A^* = \overline{A^T}$.*

Démonstration. On note $\mathcal{B}=\{e_1,\dots,e_n\}$ et $\mathcal{B}'=\{e'_1,\dots,e'_m\}$. Soient $i \in \{1,\dots,m\}$ et $j \in \{1,\dots,n\}$. Par définition de A et A^*, on a
$$u(e_j)=\sum_{k=1}^{m} a_{kj}\,e'_k \quad \text{et} \quad u^*(e'_i)=\sum_{k=1}^{n} a^*_{ki}\,e_k.$$
Puisque $\mathcal{B}$ et $\mathcal{B}'$ sont orthonormales, par linéarité suivant la première variable du produit hermitien, on obtient
$$(u(e_j),e'_i)_{\mathbb{K}^m}=\sum_{k=1}^{m} a_{kj}\,(e'_k,e'_i)_{\mathbb{K}^m}=a_{ij}$$
et, par anti-linéarité suivant la deuxième variable du produit hermitien, on a
$$(e_j,u^*(e'_i))_{\mathbb{K}^n}=\sum_{k=1}^{n} \overline{a^*_{ki}}\,(e_j,e_k)_{\mathbb{K}^n}=\overline{a^*_{ji}}.$$
On en déduit $\overline{a^*_{ji}}=a_{ij}$, soit encore $a^*_{ij}=\overline{a_{ji}}$. Ce qui donne le résultat. □

Définition 2.9. Soit $A \in \mathcal{M}_{m,n}(\mathbb{K})$.

1. On appelle **adjointe** de A la matrice A^* définie par $A^* := \overline{A^T}$.
2. On dit que A est **hermitienne** ou **auto-adjointe** si $A = A^*$. Si $\mathbb{K} := \mathbb{R}$, on parle de matrice **symétrique**.
3. On dit que A est **unitaire** ou **orthogonale** si A est inversible et $A^{-1} = A^*$.
4. Pour $m = n$, on dit que A est **normale** si $AA^* = A^*A$.

Remarque 2.10. Soit $A \in \mathcal{M}_n(\mathbb{K})$ unitaire. Alors, $A^*A = AA^* = I_n$. Notons A suivant ses colonnes : $A = (e_1, \dots, e_n)$, où $\{e_1, \dots, e_n\}$ est une base de $\mathcal{M}_{n,1}(\mathbb{K})$. Alors, $A^*A = I_n$ entraîne

$$e_i \cdot e_j = \delta_{ij}, \quad \forall\ i, j \in \{1, \dots, n\}.$$

Autrement dit, A est la représentation matricielle d'une base orthonormale de $\mathbb{K}^n$ (ou $\mathcal{M}_{n,1}(\mathbb{K})$). En particulier, A est la matrice de passage de deux bases orthonormales. Réciproquement, si P est la matrice de passage de deux bases orthonormales alors P est unitaire.

Proposition 2.11. *Soient $A \in \mathcal{M}_{m,n}(\mathbb{K})$ et $B \in \mathcal{M}_{n,p}(\mathbb{K})$. Alors, on a*

1. $(AB)^* = B^*A^*$,
2. $\mathrm{Ker}(A^*) = \mathrm{Im}(A)^\perp$,
3. $\mathrm{Im}(A^*) = \mathrm{Ker}(A)^\perp$.

Théorème 2.12 (Procédé d'orthonormalisation de Gram-Schmidt)**.**

Tout $\mathbb{K}$-espace vectoriel de dimension finie non réduit à $\{0\}$ muni d'un produit hermitien admet une base orthonormale. Plus précisément, soient E un $\mathbb{K}$-espace vectoriel de dimension finie n muni d'un produit hermitien $(\cdot, \cdot)$ de norme associée $\|\cdot\|$ et $\{f_1, \dots, f_n\}$ une base de E. On pose

$$\begin{cases} e_1 := \dfrac{f_1}{\|f_1\|} \\ \tilde{e_i} := f_i - \displaystyle\sum_{j=1}^{i-1} (f_i, e_j)\, e_j \quad et \quad e_i := \dfrac{\tilde{e_i}}{\|\tilde{e_i}\|}, \quad \forall\ i = 2, \dots, n. \end{cases} \tag{2.1.2}$$

Alors, la famille $\{e_1, \dots, e_n\}$ forme une base orthonormale de E.

Remarque 2.13. En particulier, la définition (2.1.2). de la base $\{e_1, \dots, e_n\}$ signifie que, pour tout $i = 1, \dots, n$, on a $\mathrm{Vect}\{e_1, \dots, e_i\} = \mathrm{Vect}\{f_1, \dots, f_i\}$.

Théorème 2.14 (Factorisation de Schur)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$ de polynôme caractéristique scindé dans $\mathbb{K}$. Alors, il existe une matrice unitaire $U \in \mathcal{M}_n(\mathbb{K})$ telle que U^*AU soit triangulaire supérieure. En particulier, toute matrice carrée de $\mathcal{M}_n(\mathbb{C})$ est trigonalisable dans une base orthonormale.

Démonstration. Soit u l'application linéaire associée à A dans la base canonique de $\mathbb{K}^n$. D'après le Théorème 1.34., il existe une base $\mathcal{B} := \{f_1, \dots, f_n\}$ de $\mathbb{K}^n$ telle que

$T := \mathrm{Mat}(u \mid \mathcal{B}) = (t_{ij})_{1\leq i,j\leq n}$ soit triangulaire supérieure. D'après le procédé d'orthonormalisation de Gram-Schmidt, il existe une base orthonormale $\{g_1, \ldots, g_n\}$ de $\mathbb{K}^n$ telle que

$$\forall\ i = 1, \ldots, n, \quad \mathrm{Vect}\{g_1, \ldots, g_i\} = \mathrm{Vect}\{f_1, \ldots, f_i\}.$$

Puisque T est triangulaire supérieure, on a

$$\forall\ i = 1, \ldots, n, \quad u(f_i) = \sum_{j=1}^{i} t_{ij} f_j \in \mathrm{Vect}\{f_1, \ldots, f_i\},$$

i.e.

$$\forall\ i = 1, \ldots, n, \quad u\left(\mathrm{Vect}\{f_1, \ldots, f_i\}\right) \subset \mathrm{Vect}\{f_1, \ldots, f_i\}.$$

Or, pour tout $i = 1, \ldots, n$, $g_i \in \mathrm{Vect}\{g_1, \ldots, g_i\} = \mathrm{Vect}\{f_1, \ldots, f_i\}$ donc, d'après ce qui précède, $u(g_i) \in \mathrm{Vect}\{g_1, \ldots, g_i\}$, donc il existe $t'_{ij} \in \mathbb{K}$, $j = 1, \ldots, i$, tels que

$$\forall\ i = 1, \ldots, n, \quad u(g_i) = \sum_{j=1}^{i} t'_{ij} g_j.$$

Autrement dit, la matrice associée à u dans la base $\{g_1, \ldots, g_n\}$ est triangulaire supérieure. □

2.2 Diagonalisation des matrices normales

Théorème 2.15. *Soit $A \in \mathcal{M}_n(\mathbb{K})$ de polynôme caractéristique scindé dans $\mathbb{K}$. Alors, A est normale si et seulement si elle est diagonalisable dans une base orthonormale.*

Démonstration. Supposons que A est diagonalisable dans une base orthonormale. Alors, il existe $U \in \mathcal{M}_n(\mathbb{K})$ unitaire et $D \in \mathcal{M}_n(\mathbb{K})$ diagonale telles que $A = UDU^*$. On obtient $A^*A = UD^*U^*UDU^* = UD^*DU^*$. Puisque D et D^* sont diagonales celles-ci commutent et donc $UD^*DU^* = UDD^*U^* = AA^*$, d'où A est normale.

Réciproquement, supposons A normale. D'après le théorème de décomposition de Schur, il existe $U \in \mathcal{M}_n(\mathbb{K})$ unitaire et $T \in \mathcal{M}_n(\mathbb{K})$ triangulaire supérieure telles que $A = UTU^*$. Alors, on obtient $A^*A = UT^*TU^*$ et $AA^* = UTT^*U^*$, d'où $T^*T = TT^*$. Il reste à montrer que T est diagonale. D'abord, puisque $T^* = \overline{T}^T$, on obtient

$$\forall\ i, j = 1, \ldots, n, \quad \sum_{k=1}^{n} \bar{t}_{ki} t_{kj} = \sum_{k=1}^{n} t_{ik} \bar{t}_{jk}.$$

Comme T est triangulaire supérieure, on en déduit

$$\forall\ i, j = 1, \ldots, n, \quad \sum_{k=1}^{\min(i,j)} \bar{t}_{ki} t_{kj} = \sum_{k=\max(i,j)}^{n} t_{ik} \bar{t}_{jk}. \tag{2.2.1}$$

Montrons par récurrence que $t_{ij} = 0$ pour $i \neq j$.

Pour $i = 1$, d'après (2.2.1)., on a

$$|t_{11}|^2 = \sum_{k=1}^{n} |t_{1k}|^2,$$

d'où $t_{1j} = 0$ pour tout $j = 2, \dots, n$.

Supposons $t_{ij} = 0$ pour tout $j \neq i$ avec $i \in \{1, \dots, N-1\}$. D'après (2.2.1)., on a

$$\sum_{k=1}^{i+1} |t_{k(i+1)}|^2 = \sum_{k=i+1}^{n} |t_{(i+1)k}|^2.$$

Or, pour $k \leq i$, on a par hypothèse de récurrence $t_{k(i+1)} = 0$ donc

$$|t_{(i+1)(i+1)}|^2 = \sum_{k=i+2}^{n} |t_{(i+1)k}|^2 + |t_{(i+1)(i+1)}|^2,$$

d'où $t_{(i+1)k} = 0$ pour tout $k \neq i+1$. Ce qui donne le résultat. □

Corollaire 2.16. *Soit $A \in \mathcal{M}_n(\mathbb{K})$. Alors, A est hermitienne si et seulement si elle est diagonalisable dans une base orthonormale et de valeur propres réelles, i.e. il existe $U \in \mathcal{M}_n(\mathbb{K})$ unitaire telle que $A = U\mathrm{diag}(\lambda_1, \dots, \lambda_n)U^*$, avec $\lambda_1, \dots, \lambda_n \in \mathbb{R}$.*

Démonstration. Supposons que $A = U\mathrm{diag}(\lambda_1, \dots, \lambda_n)U^*$, avec $\lambda_1, \dots \lambda_n \in \mathbb{R}$ et $U \in \mathcal{M}_n(\mathbb{K})$ unitaire. Alors, on a $A^* = U\mathrm{diag}(\lambda_1, \dots, \lambda_n)^*U^*$. Or

$$\mathrm{diag}(\lambda_1, \dots, \lambda_n)^* = \mathrm{diag}(\overline{\lambda_1}, \dots, \overline{\lambda_n}) = \mathrm{diag}(\lambda_1, \dots, \lambda_n).$$

Donc $A^* = A$.

Réciproquement, supposons A hermitienne. Alors, le polynôme caractéristique de A est scindé dans $\mathbb{C}$ et A est normale donc, d'après le Théorème 2.15., il existe $U \in \mathcal{M}_n(\mathbb{C})$ unitaire telle que $A = U\mathrm{diag}(\lambda_1, \dots, \lambda_n)U^*$, avec $\lambda_1, \dots \lambda_n \in \mathbb{C}$. Puisque $A^* = A$, on en déduit $\mathrm{diag}(\lambda_1, \dots, \lambda_n) = \mathrm{diag}(\overline{\lambda_1}, \dots, \overline{\lambda_n})$, donc $\lambda_1, \dots, \lambda_n \in \mathbb{R}$. Puisque les valeurs propres de A sont réelles, on en déduit que le polynôme caractéristique de A est scindé dans $\mathbb{K}$. Donc en appliquant de nouveau le Théorème 2.15., on obtient $U \in \mathcal{M}_n(\mathbb{K})$. □

Définition 2.17. Soit $A \in \mathcal{M}_n(\mathbb{K})$ hermitienne. On dit que A est :

1. **positive** si $(Ax, x) = x^*Ax \geq 0$, pour tout $x \in \mathcal{M}_{n,1}(\mathbb{K})$,
2. **définie positive** si $(Ax, x) = x^*Ax > 0$, pour tout $x \in \mathcal{M}_{n,1}(\mathbb{K}) \setminus \{0\}$.

Proposition 2.18. *Soit $A \in \mathcal{M}_n(\mathbb{K})$ hermitienne. Alors, A est positive (resp. définie positive) si et seulement si ses valeur propres sont positives (resp. strictement positives).*

Démonstration. On a $A = U^*DU$, avec $D := \mathrm{diag}(\lambda_1, \dots, \lambda_n)$, $\lambda_1, \dots \lambda_n \in \mathbb{R}$ et $U \in \mathcal{M}_n(\mathbb{K})$ unitaire. Soit $x \in \mathcal{M}_{n,1}(\mathbb{K})$. Alors, on a

$$x^*Ax = x^*U^*DUx = y^*Dy, \quad \text{où } y := Ux.$$

Comme $D := \mathrm{diag}(\lambda_1, \dots, \lambda_n)$, on a $y^*Dy = \sum_{i=1}^{n} \lambda_i |y_i|^2$. Ainsi, si les valeurs propres sont positives, on en déduit $x^*Ax \geq 0$. Réciproquement, si A est définie positive, on prend $i \in \{1, \dots, n\}$ et on pose $x = U^*e_i$, où e_i est le i-ème vecteur de la base canonique de $\mathcal{M}_{n,1}(\mathbb{R})$. On obtient $\lambda_i = x^*Ax \geq 0$. Ce qui donne le résultat. □

2.3 Valeurs singulières d'une matrice

Lemme 2.19. *Soit $A \in \mathcal{M}_{m,n}(\mathbb{K})$. Alors, la matrice A^*A est hermitienne positive. En particulier, les valeurs propres de A^*A sont réelles positives.*

Démonstration. Puisque $A \in \mathcal{M}_{m,n}(\mathbb{K})$ et $A^* \in \mathcal{M}_{n,m}(\mathbb{K})$, on a $A^*A \in \mathcal{M}_n(\mathbb{K})$. De plus, $(A^*A)^* = A^*(A^*)^* = A^*A$. Soit $x \in \mathcal{M}_{n,1}(\mathbb{K})$. Alors, $A^*Ax \cdot x = Ax \cdot Ax = \|Ax\| \geq 0$, où $\|\cdot\|$ est la norme euclidienne sur $\mathcal{M}_{n,1}(\mathbb{K})$. Donc A^*A est positive. □

Définition 2.20. Soit $A \in \mathcal{M}_{m,n}(\mathbb{K})$. On appelle **valeurs singulières** de A les racines carrées des valeurs propres non nulles de A^*A.

Lemme 2.21. *Soit $A \in \mathcal{M}_{m,n}(\mathbb{K})$. Alors, les valeurs propres non nulles de A^*A et AA^* sont les mêmes.*

Démonstration. Voir Td. □

Remarque 2.22.

1. Les valeurs singulières d'une matrice hermitienne sont les modules de ses valeurs propres.
2. Les valeurs singulières permettent d'obtenir un équivalent des valeurs propres pour des matrices rectangulaires. En particulier, on obtient un équivalent de la diagonalisation pour les matrices rectangulaires appelé SVD pour « Singular Value Decomposition » (décomposition en valeurs singulières).

Théorème 2.23 (Décomposition en valeurs singulières)**.**
Soit $A \in \mathcal{M}_{m,n}(\mathbb{K})$ ayant r valeurs singulières. Alors, il existe deux matrices unitaires $U \in \mathcal{M}_n(\mathbb{K})$ et $V \in \mathcal{M}_m(\mathbb{K})$ telles que

$$A = V\tilde{\Sigma}U^* \quad \textit{avec} \quad \tilde{\Sigma} := \begin{pmatrix} \Sigma & \mathbf{0}_{r(n-r)} \\ \mathbf{0}_{(n-r)r} & \mathbf{0}_{(n-r)(n-r)} \end{pmatrix} \in \mathcal{M}_{m,n}(\mathbb{K}),$$

où $\Sigma := diag(\mu_1, \ldots, \mu_r) \in \mathcal{M}_r(\mathbb{K})$ est la matrice diagonale formée des r valeurs singulières rangées par ordre croissant : $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_r > 0$.

Remarque 2.24.

1. Si l'on note u_i, $i \in \{1, \ldots, n\}$, et v_i, $i \in \{1, \ldots, m\}$, les colonnes de U et V, la SVD de A s'écrit plus simplement :

$$A = \sum_{i=1}^{r} \mu_i \, v_i \, u_i^*.$$

2. On a $A^*A = U\tilde{\Sigma}^*V^*V\tilde{\Sigma}U^* = U\tilde{\Sigma}^*\tilde{\Sigma}U^*$. Puisque $\tilde{\Sigma}^*\tilde{\Sigma}$ est diagonale, on en déduit que les colonnes de U sont formées de vecteurs propres de A^*A. De même, les colonnes de V sont formées de vecteurs propres de AA^*.

Démonstration. Dans la suite, on considère le cas $m \geq n$. Le cas $n \geq m$ se traite en considérant la transposée. La matrice $A^*A \in \mathcal{M}_n(\mathbb{K})$ étant hermitienne, il existe une matrice unitaire $U \in \mathcal{M}_n(\mathbb{K})$ telle que $A^*A = UDU^*$, où $D \in \mathcal{M}_n(\mathbb{K})$ est diagonale avec pour éléments diagonaux ses valeurs propres $\mu_1^2 \geq \cdots \geq \mu_r^2 > \mu_{r+1}^2 = \cdots = \mu_n^2 = 0$. Avec ces notations, on a $\tilde{\Sigma} = \mathrm{diag}(\mu_1, \ldots, \mu_n) \in \mathcal{M}_{m,n}(\mathbb{K})$ et $D = \tilde{\Sigma}^T\tilde{\Sigma}$. Soient $u_1, \ldots, u_n \in \mathcal{M}_{n,1}(\mathbb{K})$ les colonnes de U. On a

$$\forall\ i, j = 1, \ldots, n, \quad (Au_i, Au_j) = (A^*Au_i, u_j) = \mu_i^2(u_i, u_j) = \mu_i^2\delta_{ij},$$

d'où $\|Au_i\| = \mu_i$ pour tout $i = 1, \ldots, n$. En particulier, $Au_i = 0$ si $i = r+1, \ldots, n$ et $Au_i \neq 0$ si $i = 1, \ldots, r$. Pour tout $i = 1, \ldots, r$, on pose $v_i := \mu_i^{-1}Au_i \in \mathcal{M}_{m,1}(\mathbb{K})$. On a alors

$$\forall\ i, j = 1, \ldots, r, \quad (v_i, v_j) = \frac{1}{\mu_i\mu_j}(Au_i, Au_j) = \frac{\mu_i^2}{\mu_i\mu_j}\delta_{ij} = \delta_{ij}.$$

D'après le théorème de la base incomplète et le procédé d'orthonormalisation de Gram-Schmidt, il existe $v_{r+1}, \ldots, v_m \in \mathcal{M}_{m,1}(\mathbb{K})$ telle que la famille $\{v_1, \ldots, v_m\}$ est une base orthonormale de $\mathcal{M}_{m,1}(\mathbb{K})$. Soit alors $V \in \mathcal{M}_m(\mathbb{K})$ la matrice de colonnes $v_1, \ldots, v_m$. On obtient

$$\begin{aligned} V\tilde{\Sigma}U^* &= (v_1, \ldots, v_n)\mathrm{diag}(\mu_1, \ldots, \mu_n)U^* \\ &= (\mu_1 v_1, \ldots, \mu_m v_m)U^* \\ &= (Au_1, \ldots, Au_r, 0, \ldots, 0)U^* \\ &= AUU^* = A, \end{aligned}$$

ce qui donne le résultat. □

Remarque 2.25. La SVD permet de définir un équivalent de l'inverse pour des matrices rectangulaires, dite matrice **pseudo inverse de Moore-Penrose** en posant

$$A^\dagger := U\tilde{\Sigma}^\dagger V^* \quad \text{avec} \quad \tilde{\Sigma}^\dagger := \begin{pmatrix} \Sigma^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

ce qui correspond bien à l'inverse pour une matrice carrée inversible.

Chapitre 3

Normes matricielles

3.1 Définitions et propriétés élémentaires

Notation 3.1. On rappelle que dans un espace vectoriel normé de dimension finie, toutes les normes sont équivalentes. En particulier, $\mathbb{K}^n \sim \mathcal{M}_{n,1}(\mathbb{K})$ peut être muni de différentes normes induisant la même topologie. On donne ci-dessous les notations des normes les plus couramment employées dans $\mathbb{K}^n$, où $x := (x_1, \dots, x_n)$ désigne un vecteur de $\mathbb{K}^n$.

1. La **norme euclidienne**, ou norme l^2, est notée $\|\cdot\|_2$ et est définie par

$$\|x\|_2 := \left(\sum_{i=1}^{n} |x_i|^2\right)^{\frac{1}{2}}.$$

2. La norme l^p, où $1 \leq p < \infty$, est notée $\|\cdot\|_p$ et est définie par

$$\|x\|_p := \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}.$$

3. La norme du max, ou norme l^∞, est notée $\|\cdot\|_\infty$ et est définie par

$$\|x\|_\infty := \max_{1\leq i\leq n} |x_i|.$$

Comme rappelé plus haut, ces normes sont équivalentes. De plus, les constantes d'équivalence sont connues (voir Td) :

$$\|x\|_\infty \leq \|x\|_p \leq n^{\frac{1}{p}} \|x\|_\infty, \tag{3.1.1}$$

$$\|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\, \|x\|_2. \tag{3.1.2}$$

Définition 3.2. Pour $A \in \mathcal{M}_{m,n}(\mathbb{K})$ de coefficients a_{ij}, on définit la **norme de Frobenius** par

$$\|A\|_F := \left(\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2\right)^{\frac{1}{2}}.$$

Définition 3.3. On dit qu'une norme $\|\cdot\|$ sur $\mathcal{M}_n(\mathbb{K})$ est une **norme matricielle** si elle vérifie

$$\forall\ A, B \in \mathcal{M}_n(\mathbb{K}), \quad \|AB\| \leq \|A\|\,\|B\|.$$

Exemple 3.4.

1. La norme de Frobenius est une norme matricielle. En effet, soient $A, B \in \mathcal{M}_n(\mathbb{K})$ et c_{ij} les coefficients de AB, alors

$$\|AB\|_F^2 = \sum_{i,j=1}^{n} |c_{ij}|^2 = \sum_{i,j=1}^{n} \left| \sum_{k=1}^{n} a_{ik} b_{kj} \right|^2 .$$

   D'après l'inégalité de Cauchy-Schwarz, on a

$$\left| \sum_{k=1}^{n} a_{ik} b_{kj} \right|^2 \leq \left(\sum_{k=1}^{n} |a_{ik}|^2 \right) \left(\sum_{k=1}^{n} |b_{kj}|^2 \right),$$

   d'où

$$\|AB\|_F^2 \leq \left(\sum_{i,k=1}^{n} |a_{ik}|^2 \right) \left(\sum_{j,k=1}^{n} |b_{kj}|^2 \right) = \|A\|_F^2\,\|B\|_F^2.$$

2. La norme $|||\cdot|||$ définie sur $\mathcal{M}_n(\mathbb{K})$ par $|||A||| := \max\limits_{1\leq i,j\leq n} |a_{ij}|$ n'est pas une norme matricielle. En effet, soit $A \in \mathcal{M}_n(\mathbb{K})$ de coefficients $a_{ij} = 1$ pour tous $1 \leq i, j \leq n$. Alors, on a $|||A||| = 1$ mais A^2 a tous ses coefficients égaux à n, d'où $|||A^2||| = n > |||A|||\,|||A|||$.

Remarque 3.5. Soient $(E, \|\cdot\|)$ un $\mathbb{K}$-espace vectoriel normé et $\mathcal{L}(E)$ l'espace vectoriel des applications linéaires continues de E dans E. On peut définir sur $\mathcal{L}(E)$ une norme $\|\cdot\|_{\mathcal{L}(E)}$ dite subordonnée à $\|\cdot\|$, en posant

$$\forall\ u \in \mathcal{L}(E), \quad \|u\|_{\mathcal{L}(E)} := \sup_{\substack{x\in E \\ \|x\|\leq 1}} \|u(x)\| = \sup_{\substack{x\in E \\ \|x\|=1}} \|u(x)\| = \sup_{x\in E\setminus\{0\}} \frac{\|u(x)\|}{\|x\|}.$$

Si E est de dimension finie, toute application linéaire est continue (*i.e.* $L(E) = \mathcal{L}(E)$) et associée à une matrice. Ainsi, sur $\mathcal{M}_n(\mathbb{K})$, on peut définir une norme dite subordonnée à une norme vectorielle.

Définition 3.6. Soit $\|\cdot\|$ une norme sur $\mathbb{K}^n$ (identifié à $\mathcal{M}_{n,1}(\mathbb{K})$). On appelle **norme subordonnée** à $\|\cdot\|$, encore notée $\|\cdot\|$, la norme définie sur $\mathcal{M}_n(\mathbb{K})$ par

$$\forall\ A \in \mathcal{M}_n(\mathbb{K}), \quad \|A\| := \sup_{\substack{x\in \mathbb{K}^n \\ \|x\|\leq 1}} \|Ax\| = \sup_{\substack{x\in \mathbb{K}^n \\ \|x\|=1}} \|Ax\| = \sup_{x\in \mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|}{\|x\|}.$$

Remarque 3.7. Par définition de la norme subordonnée, si $\|\cdot\|$ est une norme sur $\mathbb{K}^n$, on a

$$\forall\ A \in \mathcal{M}_n(\mathbb{K}),\ \forall\ x \in \mathcal{M}_{n,1}(\mathbb{K}), \quad \|Ax\| \leq \|A\|\,\|x\|.$$

Proposition 3.8. *Soit $\|\cdot\|$ une norme sur $\mathbb{K}^n$.*

1. *Pour toute* $A \in \mathcal{M}_n(\mathbb{K})$*, il existe* $x_A \in \mathbb{K}^n \setminus \{0\}$ *tel que*

$$\|A\| = \|Ax_A\| \quad et \quad \|x_A\| \leq 1.$$

2. $\|I_n\| = 1$*, où* $I_n \in \mathcal{M}_n(\mathbb{K})$ *est la matrice identité.*
3. *La norme subordonnée* $\|\cdot\|$ *sur* $\mathcal{M}_n(\mathbb{K})$ *est une norme matricielle.*

Démonstration.
1. L'application $x \in \mathbb{K}^n \mapsto \|Ax\|$ est continue sur la boule unité fermée $\{x \in \mathbb{K}^n \mid \|x\| \leq 1\}$ qui est compacte, donc cette application atteint sa borne supérieure en un point x_A de la boule unité fermée.
2. $\|I_n\| = \sup_{\|x\|=1} \|x\| = 1$.
3. Soient $A, B \in \mathcal{M}_n(\mathbb{K})$ et $x \in \mathbb{K}^n$. Pour tout $y \in \mathbb{K}^n$, on a $\|Ay\| \leq \|A\| \, \|y\|$. On en déduit $\|ABx\| \leq \|A\| \, \|Bx\|$, en passant au sup on obtient $\|AB\| \leq \|A\| \, \|B\|$. □

Remarque 3.9.

1. D'après les inégalités (3.1.1). et (3.1.2)., pour toute $A \in \mathcal{M}_n(\mathbb{K})$, on a (voir Td)

$$n^{-\frac{1}{p}} \|A\|_\infty \leq \|A\|_p \leq n^{\frac{1}{p}} \|A\|_\infty, \tag{3.1.3}$$

$$n^{-\frac{1}{2}} \|A\|_2 \leq \|A\|_1 \leq n^{\frac{1}{2}} \|A\|_2. \tag{3.1.4}$$

2. La norme de Frobenius n'est pas une norme subordonnée. En effet, on a

$$\|I_n\|_F = \left(\sum_{i,j=1}^{n} \delta_{ij} \right)^{\frac{1}{2}} = \sqrt{n} \neq 1.$$

Remarque 3.10. Les normes subordonnées sur l'espace $\mathcal{M}_{m,n}(\mathbb{K})$ des matrices rectangulaires se définissent de manière analogue au cas des matrices carrées. Si on désigne par $\|\cdot\|_k$ une norme sur $\mathbb{K}^k$, alors la norme subordonnée correspondante sur $\mathcal{M}_{m,n}(\mathbb{K})$ est donnée par

$$\forall\ A \in \mathcal{M}_{m,n}(\mathbb{K}), \quad \|A\| := \sup_{x \in \mathbb{K}^n \setminus \{0\}} \frac{\|Ax\|_m}{\|x\|_n}.$$

3.2 Propriétés des normes usuelles

Proposition 3.11. *Les normes subordonnées* $\|\cdot\|_1$ *et* $\|\cdot\|_\infty$ *vérifient*

1. $\forall\ A \in \mathcal{M}_n(\mathbb{K}), \quad \|A\|_1 = \max_{1 \leq j \leq n} \left(\sum_{i=1}^{n} |a_{ij}| \right)$.
2. $\forall\ A \in \mathcal{M}_n(\mathbb{K}), \quad \|A\|_\infty = \max_{1 \leq i \leq n} \left(\sum_{j=1}^{n} |a_{ij}| \right)$.

Démonstration. Voir Td. □

Proposition 3.12. *La norme subordonnée $\|\cdot\|_2$ vérifie*

$$\forall\ A \in \mathcal{M}_n(\mathbb{K}), \quad \|A\|_2 = \|A^*\|_2 = \textit{la plus grande valeur singulière de } A.$$

Démonstration. Rappelons d'abord que la norme $\|\cdot\|_2$ est la norme associée au produit hermitien canonique $(\cdot,\cdot)_2$ de $\mathbb{K}^n$. Soit $A \in \mathcal{M}_n(\mathbb{K})$. On a

$$\|A\|_2^2 = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ax\|_2^2}{\|x\|_2^2} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{(Ax,Ax)_2}{\|x\|_2^2} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{(A^*Ax,x)_2}{\|x\|_2^2}.$$

Puisque A^*A est hermitienne positive, elle est diagonalisable dans une base orthonormale $\{e_1,\dots,e_n\}$ de $\mathbb{K}^n$ formée de vecteurs propres et les valeurs propres correspondantes $\lambda_1,\dots,\lambda_n$ sont positives ou nulles. Pour $x \in \mathbb{K}^n$, on note x_i, $1 \le i \le n$, ses coordonnées dans cette base, *i.e.* $x = x_1e_1 + \cdots + x_ne_n$. Alors, on a

$$\begin{aligned}(A^*Ax,x)_2 &= \left(\sum_{i=1}^n x_iA^*Ae^i, \sum_{j=1}^n x_je_j\right)_2 = \sum_{i,j=1}^n \lambda_i x_i\, x_j (e_i,e_j)_2 = \sum_{i=1}^n \lambda_i |x_i|^2\\ &\le \left(\max_{1\le i\le n}\lambda_i\right) \sum_{i,j=1}^n x_i\, x_j (e_i,e_j)_2\\ &\le \left(\max_{1\le i\le n}\lambda_i\right) \|x\|_2^2.\end{aligned}$$

On en déduit $\|A\|_2 \le \max\limits_{1\le i\le n}\sqrt{\lambda_i}$. Soient $i \in \{1,\dots,n\}$ et $x := e_i$. Alors, $\|x\|_2 = 1$ et on a $\|Ax\|_2 = \sqrt{(A^*Ax,x)_2} = \sqrt{\lambda_i}$. Ainsi, $\|A\|_2 \ge \sqrt{\lambda_i}$, pour tout $i = 1,\dots,n$, donc $\|A\|_2 \ge \max\limits_{1\le i\le n}\sqrt{\lambda_i}$. De plus, pour tout $x \in \mathbb{K}^n \setminus \{0\}$, on a

$$\frac{(Ax,Ax)_2}{\|x\|_2^2} \le \frac{\|A^*Ax\|_2\,\|x\|_2}{\|x\|_2^2} \le \frac{\|A^*Ax\|_2}{\|x\|_2} \le \|A^*A\|_2 \le \|A^*\|_2\,\|A\|_2,$$

d'où $\|A\|_2^2 \le \|A^*\|_2\,\|A\|_2$, ce qui donne $\|A\|_2 \le \|A^*\|_2$. En appliquant ce résultat à A^*, on obtient $\|A^*\|_2 \le \|A\|_2$, d'où $\|A^*\|_2 = \|A\|_2$. □

Définition 3.13. Soit $A \in \mathcal{M}_n(\mathbb{K})$. On appelle **spectre** de A, noté $\sigma(A)$, l'ensemble des valeurs propres de A et **rayon spectral** de A, noté $\rho(A)$, la plus grande valeur propre de A en module, *i.e.*

$$\rho(A) \ :=\ \max\{|\lambda| \mid \lambda \in \sigma(A)\}.$$

Remarque 3.14. Le rayon spectral ne permet pas de définir une norme sur $\mathcal{M}_n(\mathbb{K})$. En effet, si on considère, par exemple, la matrice $A := \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$ celle-ci vérifie $\rho(A) = 0$ mais $A \neq 0$. Néanmoins, le rayon spectral définit une norme sur le sous-espace vectoriel de $\mathcal{M}_n(\mathbb{K})$ formé des matrices normales d'après le résultat ci-dessous.

Théorème 3.15. *Pour toutes matrices $U \in \mathcal{M}_n(\mathbb{K})$ unitaire et $A \in \mathcal{M}_n(\mathbb{K})$, on a*

$$\|AU\|_2 = \|UA\|_2 = \|A\|_2.$$

En particulier, pour toute matrice $A \in \mathcal{M}_n(\mathbb{K})$ normale, on obtient

$$\|A\|_2 = \rho(A).$$

Démonstration. Soient $U \in \mathcal{M}_n(\mathbb{K})$ unitaire et $A \in \mathcal{M}_n(\mathbb{K})$. On a

$$\begin{aligned}\|UA\|_2^2 &= \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|UAx\|_2^2}{\|x\|_2^2} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{(UAx,UAx)_2}{\|x\|_2^2} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{(U^*UAx,Ax)_2}{\|x\|_2^2} \\ &= \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{(Ax,Ax)_2}{\|x\|_2^2} = \|A\|_2^2.\end{aligned}$$

En particulier, si $A = I_n$, on obtient $\|U\|_2 = \|U^*\|_2 = 1$. Soient $x \in \mathbb{K}^n$ et $y = Ux$. On a

$$\|y\|_2 = \|Ux\|_2 \le \|U\|_2\,\|x\|_2 = \|x\|_2 \quad \text{et} \quad \|x\|_2 = \|U^*x\|_2 \le \|U^*\|_2\,\|y\|_2 = \|y\|_2,$$

donc $\|x\|_2 = \|Ux\|_2 = \|U^*x\|_2$. Puisque U est inversible, on en déduit

$$\|AU\|_2^2 = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|AUx\|_2^2}{\|x\|_2^2} = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|AUx\|_2^2}{\|Ux\|_2^2} = \sup_{y\in\mathbb{K}^n\setminus\{0\}} \frac{\|Ay\|_2^2}{\|y\|_2^2} = \|A\|_2^2.$$

Enfin, si A est normale alors il existe $U \in \mathcal{M}_n(\mathbb{K})$ unitaire telle que $A = UDU^*$ avec $D := \mathrm{diag}(\lambda_1,\dots,\lambda_n)$, où $\lambda_1,\dots,\lambda_n$ sont les valeurs propres de A. Donc on obtient

$$\|A\|_2 = \|D\|_2 = \sup_{x\in\mathbb{K}^n\setminus\{0\}} \frac{\|Dx\|_2}{\|x\|_2}.$$

Or $\|Dx\|_2^2 = \sum_{i=1}^n |\lambda_i\, x_i|^2 \le \rho(A)^2\|x\|_2^2$, pour tout $x := (x_1,\dots,x_n) \in \mathbb{K}^n$, d'où $\|D\|_2 \le \rho(A)$. De plus, si $\rho(A) = |\lambda_{i_0}|$, en prenant $x := e^{i_0}$ le i_0-ème vecteur de la base canonique de $\mathbb{K}^n$, on a $\|Dx\|_2 = |\lambda_{i_0}| = \rho(A)$ et $\|x\|_2 = 1$. Donc $\|D\|_2 = \rho(A)$. □

Proposition 3.16 (Propriétés de la norme de Frobenius)**.**
Soit $A \in \mathcal{M}_n(\mathbb{K})$.

1. $\|A\|_F = \sqrt{\mathrm{tr}(A^*A)} = \left(\sum_{i=1}^n \mu_i^2\right)^{\frac12}$, *où les μ_i sont nuls ou sont les valeurs singulières de A.*
2. *Soit $U \in \mathcal{M}_n(\mathbb{K})$ unitaire, alors on a*
$$\|UA\|_F = \|AU\|_F = \|A\|_F.$$
3. *Pour $p = 1$ ou ∞, on a*
$$\|A\|_2 \le \|A\|_F \le \sqrt{n}\,\|A\|_2, \quad \text{et} \quad \frac{1}{\sqrt{n}}\,\|A\|_p \le \|A\|_F \le \sqrt{n}\,\|A\|_p$$

Démonstration.
1. $\|A\|_F^2 = \left(\sum_{i,j=1}^n |a_{ij}|^2\right)^{\frac12}$ et $\mathrm{tr}(A^*A) = \sum_{i=1}^n c_{ii}$, où $c_{ij} = \sum_{k=1}^n \overline{a_{ki}}a_{kj}$. Donc

$$\mathrm{tr}(A^*A) = \sum_{i=1}^n\sum_{k=1}^n |a_{ki}|^2 = \|A\|_F^2.$$

De plus, A^*A est diagonalisable dans $\mathbb{K}$ donc $\mathrm{tr}(A^*A) = \sum_{i=1}^{n} \lambda_i$, où $\lambda_1, \ldots, \lambda_n$ sont les valeurs propres de A^*A, on en déduit le résultat.

2. On a
$$\|UA\|_F^2 = \mathrm{tr}((UA)^*UA) = \mathrm{tr}(A^*U^*UA) = \mathrm{tr}(A^*A) = \|A\|_F^2.$$
De plus, $\|AU\|_F^2 = \mathrm{tr}(U^*A^*AU^*) = \mathrm{tr}(A^*A) = \|A\|_F^2$.

3. $\|A\|_F = \left(\sum_{i=1}^{n} \mu_i^2\right)^{\frac{1}{2}} \leq \sqrt{n} \max_{1\leq i\leq n} \mu_i = \sqrt{n}\,\|A\|_2$. De plus, il existe $i_0 \in \{1, \ldots, n\}$ tel que

$$\|A\|_2 = \mu_{i_0} \leq \left(\sum_{i=1}^{n} \mu_i^2\right)^{\frac{1}{2}} = \|A\|_F.$$

De plus,
$$\|A\|_1 = \max_{1\leq j\leq n} \left(\sum_{i=1}^{n} |a_{ij}|\right).$$

Il existe $j_0 \in \{1, \ldots, n\}$ tel que $\|A\|_1 = \sum_{i=1}^{n} |a_{ij_0}|$. Alors, d'après l'inégalité de Cauchy-Schwarz, on obtient
$$\|A\|_1 \leq \sqrt{n} \left(\sum_{i=1}^{n} |a_{i_0 j}|^2\right)^{\frac{1}{2}} \leq \sqrt{n}\,\|A\|_F.$$

Enfin,
$$\|A\|_F^2 = \sum_{j=1}^{n} \left(\sum_{i=1}^{n} |a_{ij}|^2\right) \leq n \sum_{i=1}^{n} |a_{i_0 j}|^2 \leq n\,\|A\|_1^2.$$

□

Théorème 3.17. *Pour toute norme matricielle $\|\cdot\|$ sur $\mathcal{M}_n(\mathbb{C})$, on a*

$$\forall\ A \in \mathcal{M}_n(\mathbb{C}), \quad \rho(A) \leq \|A\|.$$

Réciproquement, pour tout $\varepsilon > 0$ et pour toute matrice $A \in \mathcal{M}_n(\mathbb{C})$, il existe une norme matricielle subordonnée $\|\cdot\|_A$ telle que

$$\|A\|_A \leq \rho(A) + \varepsilon.$$

Démonstration. Soient $A \in \mathcal{M}_n(\mathbb{C})$ et λ sa valeur propre telle que $\rho(A) = |\lambda|$. Soient $x \in \mathbb{C}^n \setminus \{0\}$ un vecteur propre de A associé à λ et $y \in \mathbb{C}^n$ non nul tel que la matrice $xy^* := (x_i y_j)_{1\leq i,j\leq n}$ soit non nulle. Alors, on a $Axy^* = (Ax)y^* = \lambda xy^*$, d'où

$$|\lambda|\,\|xy^*\| = \|Axy^*\| \leq \|A\|\,\|xy^*\|.$$

On en déduit $|\lambda| \leq \|A\|$.

Réciproquement, soient $\varepsilon > 0$ et $A \in \mathcal{M}_n(\mathbb{C})$. D'après le théorème de Schur, il existe $U \in \mathcal{M}_n(\mathbb{C})$ unitaire et $T \in \mathcal{M}_n(\mathbb{C})$ triangulaire supérieure telles que $A = UTU^*$. Pour tout $\delta > 0$, on pose $D_\delta := \mathrm{diag}(1, \delta, \delta^2, \ldots, \delta^{n-1}) \in \mathcal{M}_n(\mathbb{C})$. Alors, on a

$$\begin{aligned} D_\delta^{-1} T D_\delta &= \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & \delta^{-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \delta^{1-n} \end{pmatrix} \begin{pmatrix} \lambda_1 & t_{12} & \cdots & t_{1n} \\ 0 & \lambda_2 & \ddots & t_{2n} \\ \vdots & \ddots & \ddots & t_{(n-1)n} \\ 0 & \cdots & \cdots & \lambda_n \end{pmatrix} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & \delta & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \delta^{n-1} \end{pmatrix} \\ &= \begin{pmatrix} \lambda_1 & \delta t_{12} & \cdots & \delta^{n-1} t_{1n} \\ 0 & \lambda_2 & \ddots & \delta^{n-2} t_{2n} \\ \vdots & \ddots & \ddots & \delta t_{(n-1)n} \\ 0 & \cdots & \cdots & \lambda_n \end{pmatrix}. \end{aligned}$$

En notant $D_\delta^{-1} T D_\delta = (u_{ij})_{1 \le i,j \le n}$, on obtient

$$\begin{aligned} \|D_\delta^{-1} T D_\delta\|_\infty &= \max_{1 \le i \le n} \sum_{j=1}^{n} |u_{ij}| \\ &= \max_{1 \le i \le n} |\lambda_i| + \max_{1 \le i \le n-1} \left(\sum_{j=i+1}^{n} |\delta^{j-i} t_{ij}| \right) \\ &\le \rho(A) + \max_{1 \le i \le n-1} \left(\sum_{j=i+1}^{n} |\delta^{j-i} t_{ij}| \right). \end{aligned}$$

Le résultat étant vrai pour tout $\delta > 0$, on choisit δ tel que

$$\forall\ i = 1, \ldots, n-1, \quad \sum_{j=i+1}^{n} |\delta^{j-i} t_{ij}| \le \varepsilon.$$

Alors, on a $\|D_\delta^{-1} T D_\delta\|_\infty \le \rho(A) + \varepsilon$. Ainsi, si on pose

$$\|A\|_A \ := \ \|D_\delta^{-1} U^* A U D_\delta\|_\infty = \|(UD_\delta)^{-1} A (UD_\delta)\|_\infty,$$

on obtient le résultat voulu. Il reste cependant à vérifier que la norme $\|\cdot\|_A$ ainsi définie est bien une norme subordonnée. Plus précisément, on vérifie (le faire en exercice), que la norme $\|\cdot\|_A$ est la norme subordonnée à la norme vectorielle $\|\cdot\|_\delta$ définie par

$$\forall\ x \in \mathbb{C}^n, \quad \|x\|_\delta \ := \ \|(UD_\delta)^{-1} x\|_\infty.$$

□

3.3 Suites et séries de matrices carrées

Définition 3.18. On dit qu'une suite $(A_k)_{k \in \mathbb{N}}$ de $\mathcal{M}_n(\mathbb{K})$ converge vers $A \in \mathcal{M}_n(\mathbb{K})$ si, pour une norme matricielle $\|\cdot\|$, on a $\|A - A_k\| \underset{k \to +\infty}{\longrightarrow} 0$.

Lemme 3.19. *Soit $A \in \mathcal{M}_n(\mathbb{C})$. Alors, les conditions suivantes sont équivalentes :*

1. $\lim_{k\to+\infty} A^k = 0$,
2. $\lim_{k\to+\infty} A^k x = 0,\ \forall\ x \in \mathbb{C}^n$,
3. $\rho(A) < 1$,
4. *il existe une norme subordonnée $\|\cdot\|$ telle que $\|A\| < 1$.*

Démonstration.

Supposons que 1. ait lieu. Soit $x \in \mathbb{C}^n$. Alors, on a $\|A^k x\| \leq \|A^k\|\,\|x\| \underset{k\to+\infty}{\longrightarrow} 0$, ce qui donne 2.

Supposons que 2. ait lieu. Soit λ une valeur propre de A telle que $\rho(A) = |\lambda|$. Il existe $x \in \mathbb{C}^n \setminus \{0\}$ tel que $Ax = \lambda\, x$. Donc on a $A^k x = \lambda^k x$, d'où $\rho(A)^k = \dfrac{\|A^k x\|}{\|x\|} \underset{k\to+\infty}{\longrightarrow} 0$. On en déduit $\rho(A) < 1$.

Supposons que 3. ait lieu. Alors, il existe $\varepsilon > 0$ tel que $\rho(A) + \varepsilon < 1$. D'après le Théorème 3.17., il existe une norme subordonnée $\|\cdot\|_A$ telle que $\|A\|_A \leq \rho(A) + \varepsilon < 1$.

Enfin, supposons que 4. ait lieu. Puisque $\|\cdot\|$ est une norme matricielle, on a $\|A^k\| \leq \|A\|^k \underset{k\to+\infty}{\longrightarrow} 0$ car $\|A\| < 1$. □

Remarque 3.20. Puisque $\mathcal{M}_n(\mathbb{K})$ est de dimension finie, il est complet. Autrement dit, une suite $(A_k)_{k\in\mathbb{N}}$ de $\mathcal{M}_n(\mathbb{K})$ converge si et seulement si elle de Cauchy pour la norme $\|\cdot\|$ choisie, *i.e.* pour tout $\varepsilon > 0$, il existe $K \in \mathbb{N}$ tel que, si $k, l \geq K$, $\|A_k - A_l\| < \varepsilon$.

Théorème 3.21. *Soit $\sum_{k\geq 0} a_k z^k$ une série entière de rayon de convergence $R > 0$. Soit $A \in \mathcal{M}_n(\mathbb{C})$. Si $\rho(A) < R$, alors la série entière $\sum_{k\geq 0} a_k\, A^k$ converge et, pour toute norme subordonnée $\|\cdot\|$, on a $\|\sum_{k\geq 0} a_k\, A^k\| < \infty$*

Démonstration. Puisque $\rho(A) < R$, il existe une norme subordonnée $\|\cdot\|$ telle que $\|A\| < R$. Or, pour tous $l, m \in \mathbb{N}$, on a

$$\|\sum_{k=l+1}^{m} a_k\, A^k\| \leq \sum_{k=l+1}^{m} |a_k|\,\|A\|^k.$$

Puisque $\|A\|^k < R$, d'après le critère de Cauchy, le terme de droite tend vers 0, donc le terme de gauche aussi. En appliquant alors le critère de Cauchy sur le terme de gauche, la série $\sum_{k\geq 0} a_k\, A^k$ converge. □

Exemple 3.22.

1. On peut définir $\exp(A)$, $\sin(A)$ et $\cos(A)$ sans restriction sur $\rho(A)$.
2. La fonction $z \mapsto (1-z)^{-1}$ est développable en série entière sur le disque unité (*i.e.* $R = 1$) et $(1-z)^{-1} = \sum_{k\geq 0} z^k$. Ainsi, pour $A \in \mathcal{M}_n(\mathbb{C})$ telle que $\rho(A) < 1$, on peut définir $\sum_{k\geq 0} A^k$.

Proposition 3.23. *Soit $A \in \mathcal{M}_n(\mathbb{C})$ telle que $\rho(A) < 1$. Alors, la matrice $I_n - A$ est inversible et son inverse est donné par $(I_n - A)^{-1} = \sum_{k\geq 0} A^k$.*

Démonstration. On pose $S = \sum_{k\geq 0} A^k$. Il suffit de vérifier que $S(I_n - A) = I_n$. Or on a

$$S(I_n - A) = S - SA = \sum_{k\geq 0} A^k - \sum_{k\geq 0} A^k + 1 = \sum_{k\geq 0} A^k - \sum_{k\geq 1} A^k = A^0 = I_n.$$

□

Deuxième partie

Résolution numérique de systèmes linéaires

Chapitre 4

Systèmes linéaires

L'objet de ce chapitre est d'établir quelques généralités sur la résolution numérique des systèmes linéaires. En particulier, on donne les premiers algorithmes de résolution de systèmes linéaires (remontée et descente).

4.1 Systèmes linéaires carrés

Dans cette première section, on considère les problèmes du type : trouver $x \in \mathcal{M}_{n,1}(\mathbb{K})$ tel que

$$Ax = b, \quad \text{où } A \in \mathcal{M}_n(\mathbb{K}),\ b \in \mathcal{M}_{n,1}(\mathbb{K}). \tag{4.1.1}$$

Trois cas sont alors possibles.

1. A est inversible. Alors (4.1.1). admet une unique solution $x \in \mathcal{M}_{n,1}(\mathbb{K})$.
2. A n'est pas inversible mais $b \in \mathrm{Im}(A)$. Alors, il existe une infinité de solutions. En effet, puisque $b \in \mathrm{Im}(A)$, il existe $x \in \mathcal{M}_{n,1}(\mathbb{K})$ tel que $Ax = b$. Comme A n'est pas inversible, $\mathrm{Ker}(A) \neq \{0\}$, alors pour tout $x' \in \mathrm{Ker}(A) \setminus \{0\}$, on a $A(x + x') = b$.
3. A n'est pas inversible et $b \notin \mathrm{Im}(A)$. Alors, il n'existe pas de solution.

Pour la suite, on ne considère que le cas A inversible. Alors, les coefficients x_i, $1 \leq i \leq n$, de la solution x de (4.1.1). sont donnés par la formule de Cramer

$$x_i = \frac{\det(B_i)}{\det(A)}, \quad \text{où} \quad B_i := \begin{pmatrix} a_{11} & \cdots & a_{1(i-1)} & b_1 & a_{1(i+1)} & \cdots & a_{1n} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & \cdots & a_{n(i-1)} & b_n & a_{n(i+1)} & \cdots & a_{nn} \end{pmatrix},$$

i.e. la matrice B_i s'obtient en remplaçant la i-ème colonne de A par le vecteur b. Cette formule nécessite de l'ordre de $(n+1)!$ multiplications, ce qui est énorme. Dans [1], un exemple est donné pour $n = 50$ avec un ordinateur classique et amène à un calcul de la solution prenant 4.5×10^{49} années. Ainsi, celle-ci n'est absolument jamais employée dans la pratique. Pour pouvoir comparer diverses méthodes numériques de résolution de systèmes linéaires, on compte le nombre d'opérations nécessaires. Le temps de calcul d'une addition étant très faible devant celui d'une multiplication, on ne compte que le nombre de multiplications. Dans toute la suite de ce cours, on appellera **complexité** d'un algorithme le nombre de multiplications qu'il nécessite.

Considérons maintenant le cas de matrices A de forme simple :

1. Si A est diagonale. On a $A = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, alors il est immédiat que x est donné par $x = (\lambda_1^{-1}x_1, \ldots, \lambda_n^{-1}x_n)$. Le calcul de la solution dans ce cas demande d'effectuer n multiplications.

2. Si A est unitaire. On a $x = A^*b$, le calcul de A^* étant négligeable (il s'agit juste d'une transposition et conjugaison), le calcul de x ne nécessite que n^2 multiplications.

3. Si A est triangulaire, on peut construire une algorithme simple de résolution de (4.1.1)..

Ce dernier cas particulier s'avérera très important dans la suite. Ainsi dans la fin de cette section, on ne considère que le cas des matrices triangulaires. Soit A triangulaire inférieure. On note $B := A^{-1} = (b_{ij})_{1\leq i,j\leq n}$. Alors, B est aussi triangulaire inférieure et, puisque $BA = I_n$, on a

$$\forall\ i, j = 1, \ldots, n \quad \sum_{k=1}^{n} b_{ik}a_{kj} = \delta_{ij}.$$

Comme A et B sont triangulaires inférieure, on a $a_{kj} = 0$ si $j > k$ et $b_{ik} = 0$ si $k > i$, d'où

$$\forall\ i, j = 1, \ldots, n \quad \sum_{k=j}^{i} b_{ik}a_{kj} = \delta_{ij}.$$

De plus, A étant inversible et triangulaire, pour tout $i = 1, \ldots, n$, $a_{ii} \neq 0$.

- Pour $j > i$, on a $b_{ij} = 0$.
- Pour $i = j$, on a $b_{ii} = a_{ii}^{-1}$.
- Pour $j < i$, on a $b_{ij}a_{jj} = -\sum_{k=j+1}^{i} b_{ik}a_{kj}$ donc

$$b_{ij} = -\frac{1}{a_{jj}} \sum_{k=j+1}^{i} b_{ik}a_{kj}.$$

Ainsi, on calcule d'abord b_{ii}, puis on a $b_{i(i-1)}$ donné par la formule ci-dessus, on peut alors calculer $b_{i(i-2)}$. Par récurrence, on sait calculer b_{ij} pour tout $j \leq i$. Cela donne l'Algorithme 1.

Comptons le nombre d'opérations nécessaires dans l'Algorithme 1. La boucle sur k effectue $i - j - 1$ multiplications, ensuite on effectue une division, soit $i - j$ opérations dans la boucle j. Puisque j varie de $i - 1$ à 1, cela donne

$$\sum_{j=1}^{i-1}(i - j) = i(i - 1) - \frac{i(i - 1)}{2} = \frac{i(i - 1)}{2}$$

opérations à l'intérieurs de la boucle i. Donc finalement, le nombre total d'opérations est

$$\sum_{i=1}^{n} \frac{i(i - 1)}{2} = \frac{1}{2}\left(\sum_{i=1}^{n} i^2 - \sum_{i=1}^{n} i\right) = \frac{1}{2}\frac{n(n + 1)(2n + 1)}{6} - \frac{n(n + 1)}{4} \sim \frac{n^3}{6}.$$

Autrement dit, le calcul de A^{-1} par cette méthode nécessite de l'ordre de $\frac{n^3}{6}$ opérations. Pour le calcul de x, il faut ensuite effectuer le produit de A^{-1} par x, ce qui nécessite $\frac{n^2}{2}$

Algorithme 1 Inverse d'une matrice triangulaire inférieure

Entrées : A
Sorties : B
  Pour $i = 1 : n$ **faire**
    $b_{ii} = \frac{1}{a_{ii}}$
    Pour $j = i - 1 : 1$ **faire**
      $s = 0$
      Pour $k = j + 1 : i$ **faire**
        $s = s + b_{ik}a_{kj}$
      fin Pour
      $b_{ij} = -s/a_{jj}$
    fin Pour
  fin Pour

Algorithme 2 Algorithme de descente

Entrées : A, b
Sorties : x
  Pour $i = 1 : n$ **faire**
    $s = 0$
    Pour $j = i - 1 : 1$ **faire**
      $s = s + a_{ij}x_j$
    fin Pour
    $x_i = (b_i - s)/a_{ii}$
  fin Pour

opérations (en prenant en compte que la moitié des coefficients de A sont nuls) qui est négligeable devant $\frac{n^3}{6}$. Ainsi finalement le calcul de x par cette méthode demande de l'ordre de $\frac{n^3}{6}$ opérations.

Il est possible de réduire ce temps de calcul en calculant directement x sans calculer l'inverse de A. Pour cela, on remarque que l'on a

$$\forall\ i = 1, \dots, n, \quad b_i = \sum_{j=1}^{n} a_{ij}x_j = \sum_{j=1}^{i} a_{ij}x_j.$$

On en déduit :

$$\forall\ i = 1, \dots, n, \quad x_i = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j}{a_{ii}}.$$

ce qui donne l'**algorithme de descente** (Algorithme 2).

Dans l'Algorithme 2, on effectue i opérations avec i variant de 1 à n, ce qui donne $\frac{n(n+1)}{2}$ opérations. En ne tenant compte que du monôme de plus haut degré, on obtient que l'algorithme de descente nécessite de l'ordre de $\frac{n^2}{2}$ opérations. Ainsi cet algorithme est bien plus rapide que celui d'inversion et donc préférable dans les applications. De manière générale on n'utilise jamais d'algorithme d'inversion car ceux-ci sont trop couteux en temps de calcul.

Algorithme 3 Algorithme de remontée

Entrées : A, b
Sorties : x
  Pour $i = n : 1$ **faire**
    $s = 0$
    Pour $j = i + 1 : n$ **faire**
      $s = s + a_{ij}x_j$
    fin Pour
    $x_i = (b_i - s)/a_{ii}$
  fin Pour

De la même manière, lorsque A est triangulaire supérieure, la résolution de $Ax = b$ se fait en $\frac{n^2}{2}$ opérations par l'**algorithme de remontée** (Algorithme 3).

Pour finir cette section, on donne un exemple concret où la résolution du problème (4.1.1). par un algorithme rapide est nécessaire.

Exemple 4.1. (*Équation de la chaleur stationnaire*)

On considère une tige conductrice de la chaleur (représentée par l'intervalle $[0, 1]$) soumise à une source de chaleur $f \in C^2([0, 1]; \mathbb{R})$ et dont les extrémités sont plongées dans la glace. La source de chaleur étant constante au cours du temps, après un certain laps de temps la température de la tige devient elle-même constante au cours du temps. Soit $u : [0, 1] \to \mathbb{R}$ cette température. Alors, u est solution de l'**équation de la chaleur stationnaire** :

$$\begin{cases} -u''(x) &= f(x), \quad x \in]0, 1[, \\ u(0) = u(1) &= 0. \end{cases} \tag{4.1.2}$$

On suppose dans la suite que $u \in C^4([0, 1]; \mathbb{R})$. Pour résoudre numériquement (4.1.2)., on considère $N \in \mathbb{N}^*$ et on pose

$$\forall\, n \in \{0, 1, \ldots, N + 1\}, \quad x_n \;:=\; \frac{n}{N + 1}.$$

Alors, $x_0 < x_1 < \cdots < x_N < x_{N+1}$ est une partition de $[0, 1]$ telle que $x_0 = 0$ et $x_{N+1} = 1$. Soient $h := \frac{1}{N+1}$ supposé petit et $n \in \{1, \ldots, N\}$. Alors, à partir du développement limité de u autour du point x_i, on obtient (voir Td)

$$u''(x_i) = \frac{u(x_{i+1}) - 2\,u(x_i) + u(x_{i-1})}{h^2} + O(h). \tag{4.1.3}$$

Ainsi l'expression

$$\frac{u(x_{i+1}) - 2\,u(x_i) + u(x_{i-1})}{h^2},$$

est une approximation de $u''(x_i)$ pour h suffisamment petit. Pour $i = 0, \ldots, N + 1$, on note u_i une approximation de $u(x_i)$ avec $u_0 = u_{N+1} = 0$ (prise en compte des conditions aux limites). Alors, l'expression

$$\frac{u_{i+1} - 2\,u_i + u_{i-1}}{h^2}, \tag{4.1.4}$$

est encore une approximation de $u''(x_i)$. Avec ce choix d'approximation, on peut approcher le problème (4.1.2). par le problème discret suivant : trouver $u_1, \dots, u_N \in \mathbb{R}$ tels que

$$\begin{cases} -\dfrac{u_{i+1} - 2\,u_i + u_{i-1}}{h^2} & = f(x_i), \quad \forall\ i = 1, \dots, N, \\ u_0 = u_{N+1} & = 0. \end{cases} \tag{4.1.5}$$

On pose $U_h = (u_1, \dots, u_N)^T$. On obtient alors que (4.1.5). s'écrit (voir Td)

$$A_h U_h = b_h, \tag{4.1.6}$$

où $A_h \in \mathbb{R}^{N\times N}$ et $b_h \in \mathbb{R}^N$ sont donnés par

$$A_h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & \dots & 0 \\ \vdots & \ddots & & \ddots & \vdots \\ \vdots & & \ddots & \ddots & -1 \\ 0 & \dots & \dots & -1 & 2 \end{pmatrix} \quad \text{et} \quad b_h = \begin{pmatrix} f(x_1) \\ \vdots \\ \vdots \\ \vdots \\ f(x_N) \end{pmatrix}. \tag{4.1.7}$$

Puisque l'approximation est d'autant meilleurs que h est petit, dans les applications N peut être très grand et on est donc amené à résoudre un problème linéaire de grande taille (voir très grande si l'on ne se place plus dans $[0,1]$ mais, par exemple, dans un cube tel que $[0,1]^3$).

4.2 Systèmes sur-déterminés et moindres carrés

Dans cette section, on considère toujours le problème $Ax = b$ mais avec $b \in \mathcal{M}_{m,1}(\mathbb{K})$ et $A \in \mathcal{M}_{m,n}(\mathbb{K})$ telle que $m \neq n$ ou, si $m = n$, A non inversible. Deux cas sont alors possibles.

- Si $m < n$, le système est dit **sous-déterminé** : il y a plus d'inconnues que d'équations donc une infinités de solutions. Dans ce cas, il est nécessaire de compléter le système par une condition supplémentaire pour choisir la (ou les) solution(s).
- Si $m > n$, le système est dit **sur-déterminé** : il y a plus d'équations que d'inconnues donc il n'y a pas de solution. Il faut alors déterminer une notion de solution pour ce problème qui correspond à la notion de solution classique pour $m = n$ avec A inversible.

Le cas sous-déterminé ne peut être traité de manière générale : il dépend du problème concret étudié. Pour le cas sur-déterminé, on remarque que si $x \in \mathcal{M}_{n,1}(\mathbb{K})$ vérifie $Ax = b$, alors on a $\|Ax - b\| = 0$ pour toute norme $\|\cdot\|$ de $\mathcal{M}_{m,n}(\mathbb{K})$. Puisque $\|Ay - b\| \geq 0$, pour tout $y \in \mathcal{M}_{n,1}(\mathbb{K})$, on a alors $\|Ax - b\| = \min\limits_{y\in\mathcal{M}_{n,1}(\mathbb{K})} \|Ay - b\|$.

Définition 4.2. Soient $A \in \mathcal{M}_{m,n}(\mathbb{K})$ et $b \in \mathcal{M}_{m,1}(\mathbb{K})$. On dit que $x \in \mathcal{M}_{n,1}(\mathbb{K})$ est **solution aux moindres carrés** de $Ax = b$ si x vérifie

$$\|Ax - b\|_2 = \min_{y\in\mathcal{M}_{n,1}(\mathbb{K})} \|Ay - b\|_2,$$

où $\|\cdot\|_2$ est la norme euclidienne.

La question de la résolution de ce type de problème sera abordée dans le chapitre 6. Pour finir cette section, on donne un exemple de problème nécessitant la notion de solution aux moindres carrés.

Exemple 4.3. (*Régression linéaire*)
Soient n points $(x_1, y_1), \dots, (x_n, y_n)$ de $\mathbb{R}^2$ (ceux-ci peuvent, par exemple, correspondre à des données physiques obtenues expérimentalement). Afin de déterminer une loi expérimentale empirique à partir de ces points, on cherche à déterminer une fonction dont le graphe passe par chacun de ces points. Le plus simple est alors de considérer un polynôme P tel que $y_i = P(x_i)$, pour tout $i = 1, \dots, n$. Supposons ce polynôme de degré k. Alors, si l'on note $P(X) = a_0 + a_1 X + \cdots + a_k X^k$, on obtient $a_0 + a_1 x_i + \cdots + a_k x_i^k = y_i$, pour tout $i = 1, \dots, n$. Ainsi, on est amené à déterminer $a_0, \dots, a_k \in \mathbb{R}$ tels que

$$\begin{pmatrix} 1 & x_1 & \dots & x_1^k \\ 1 & x_2 & \dots & x_2^k \\ \vdots & & & \vdots \\ 1 & x_n & \dots & x_n^k \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix},$$

ce qui s'écrit encore $Ax = b$, avec $A \in \mathcal{M}_{n,k}$, $x \in \mathcal{M}_{k,1}$ et $b \in \mathcal{M}_{n,1}$. Lorsque n est très grand, déterminer un polynôme de degré k avec k grand peut être très couteux numériquement, dans beaucoup d'applications (en statistique notamment) on prend $k = 1$. Dans ce cas, la solution est explicitement connue (voir Td) et est donnée par $P(X) = aX + b$ avec

$$a = \frac{E(XY) - E(X)E(Y)}{E(X^2) - E(X)^2} \quad \text{et} \quad b = E(Y) - aE(X),$$

où

$$E(X) := \frac{1}{n}\sum_{i=1}^{n} x_i, \;\; E(Y) := \frac{1}{n}\sum_{i=1}^{n} y_i, \;\; E(XY) := \frac{1}{n}\sum_{i=1}^{n} x_i y_i \text{ et } E(X^2) := \frac{1}{n}\sum_{i=1}^{n} x_i^2.$$

4.3 Erreurs d'arrondis et conditionnement

On termine ce chapitre avec la notion d'erreurs d'arrondis dans la résolution numérique de systèmes linéaires. On donne simplement une courte présentation de cette notion qui ne sera pas détaillée dans ce cours.

Exemple 4.4. Commençons par une exemple tiré de [1]. On considère le système $Ax = b$ avec

$$A := \begin{pmatrix} 8 & 6 & 4 & 1 \\ 1 & 4 & 5 & 1 \\ 8 & 4 & 1 & 1 \\ 1 & 4 & 3 & 6 \end{pmatrix} \quad \text{et} \quad b := \begin{pmatrix} 19 \\ 11 \\ 14 \\ 14 \end{pmatrix},$$

qui a pour unique solution $x = (1, 1, 1, 1)^T$. Si l'on change le second membre par $b' := (19.01, 11.05, 14.07, 14.05)^T$, alors l'unique solution est $x' = (-2.31, 9.745, -4.85, -1.34)^T$. Autrement dit, une très faible perturbation de b ($b' = b + \varepsilon$) a entraînée une très forte perturbation de la solution. Pour "mesurer" la sensibilité d'une matrice aux perturbations, on introduit la notion de conditionnement.

Soient $A \in \mathcal{M}_n(\mathbb{K})$ inversible et $b \in \mathcal{M}_{n,1}(\mathbb{K})$. Pour $\varepsilon > 0$, on considère la perturbation suivante de b :

$$b_\varepsilon \ := \ b + \varepsilon\, b_1, \quad \text{où} \quad b_1 \in \mathcal{M}_{n,1}(\mathbb{K}).$$

On cherche alors à comparer les solutions $x, x_\varepsilon \in \mathcal{M}_{n,1}(\mathbb{K})$ des systèmes

$$Ax = b \quad \text{et} \quad Ax_\varepsilon = b_\varepsilon,$$

Pour cela, on va estimer l'erreur relative $\frac{\|x-x_\varepsilon\|}{\|x\|}$ pour une norme vectorielle $\|\cdot\|$ fixée. On note de même $\|\cdot\|$ la norme matricielle subordonnée à la norme vectorielle $\|\cdot\|$. On a

$$x_\varepsilon = A^{-1}b_\varepsilon = A^{-1}b + \varepsilon\, A^{-1}b_1 = x + \varepsilon\, A^{-1}b_1,$$

donc

$$\|x - x_\varepsilon\| \leq \varepsilon\, \|A^{-1}\|\, \|b_1\|.$$

De plus, $Ax = b$ entraîne $\|b\| \leq \|A\|\, \|x\|$, d'où $\|x\|^{-1} \leq \frac{\|A\|}{\|b\|}$ et on obtient

$$\frac{\|x - x_\varepsilon\|}{\|x\|} \leq \varepsilon\, \|A\|\|A^{-1}\|\frac{\|b_1\|}{\|b\|} = \|A\|\|A^{-1}\|\frac{\|b - b_\varepsilon\|}{\|b\|}.$$

Ainsi, l'erreur relative sur la solution fait intervenir l'erreur relative commise sur le second membre $\frac{\|b-b_\varepsilon\|}{\|b\|}$ et un terme ne dépendant que de A.

Définition 4.5. Soit $\|\cdot\|$ une norme matricielle subordonnée sur $\mathcal{M}_n(\mathbb{K})$. On appelle **conditionnement** d'une matrice $A \in \mathcal{M}_n(\mathbb{K})$ (relativement à $\|\cdot\|$), le scalaire

$$\kappa(A) \ := \ \|A\|\, \|A^{-1}\|.$$

Si $\|\cdot\| = \|\cdot\|_p$, avec $p \in \{1, 2, \infty\}$, on notera $\kappa = \kappa_p$.

Remarque 4.6.

1. On a $1 = \|I_n\| = \|AA^{-1}\| \leq \kappa(A)$. Une matrice est dite alors bien conditionnée si $\kappa(A)$ est proche de 1. Cela revient à dire que plus une matrice est bien conditionnée moins les erreurs d'arrondis sur b provoqueront des erreurs d'arrondis sur x. Par exemple, dans le cas de l'équation de la chaleur, le second membre b s'obtient en calculant $f(x)$ pour différentes valeurs de x ce qui n'est pas nécessairement un calcul exact et donc peut être entaché d'erreurs.
2. On peut aussi considérer le cas d'une perturbation de la matrice donnée par $A_\varepsilon := A + \varepsilon\, A_1$. En notant x_ε la solution de $A_\varepsilon x_\varepsilon = b_\varepsilon$, on obtient alors (voir Td) l'inégalité suivante :

$$\frac{\|x - x_\varepsilon\|}{\|x\|} \leq \kappa(A)\, \left(\frac{\|b - b_\varepsilon\|}{\|b\|} + \frac{\|A - A_\varepsilon\|}{\|A\|}\right) + O(\varepsilon^2).$$

   Autrement dit, le conditionnement prend aussi en compte les erreurs d'approximation pouvant être commises dans le calcul de A (qui dans certaines applications peut ne pas être un calcul exact).

3. Dans la pratique, on calcule le conditionnement pour l'une des normes vectorielles suivantes : l^1, l^2 ou l^∞, qui sont notés κ_1, κ_2 et κ_∞. Pour faire ces calculs il existe des algorithmes spécifiques (voir [1]).
4. Lorsqu'une matrice A est mal conditionnée il arrive que l'on utilise un préconditionneur, *i.e.* une matrice M telle que $M^{-1}A$ ait un conditionnement plus proche de 1 que celui de A. Puis on résout $M^{-1}Ax = M^{-1}b$. Le choix de M est généralement heuristique et fait de sorte que M soit proche de A et que son inverse soit simple à calculer. Par exemple, on peut prendre M diagonale de coefficients diagonaux ceux de A, *i.e.* $M = \mathrm{diag}(a_{11}, \ldots, a_{nn})$.

Exemple 4.7. D'après le devoir maison, on a

$$\kappa_2(A_h) = \frac{\sin^2\left(\frac{n\pi}{2(n+1)}\right)}{\sin^2\left(\frac{\pi}{2(n+1)}\right)} \underset{h\to 0}{\longrightarrow} 0,$$

où A_h donnée par (4.1.7). est la matrice du laplacien discret. On en déduit que le problème (4.1.6). est très mal conditionné.

On donne ci-dessous quelques propriétés du conditionnement κ_2 qui se déduisent facilement des propriétés de la norme $\|\cdot\|_2$.

Proposition 4.8. *Soit $A \in \mathcal{M}_n(\mathbb{C})$.*

1. *Si A est unitaire, $\kappa_2(A) = 1$.*
2. *Pour toute matrice unitaire $U \in \mathcal{M}_n(\mathbb{C})$, $\kappa_2(UA) = \kappa_2(AU) = \kappa_2(A)$.*
3. *$\kappa_2(A) = \frac{\mu_1(A)}{\mu_r(A)}$, où $\mu_1(A)$ et $\mu_r(A)$ sont la plus grande et la plus petite valeur singulière de A.*
4. *Si A est normale, $\kappa_2(A) = \rho(A)\rho(A^{-1})$.*

Chapitre 5

Méthodes directes de résolution des systèmes linéaires

Dans tout ce chapitre, on étudie la résolution de systèmes linéaires $Ax = b$, où $A \in \mathcal{M}_n(\mathbb{K})$ est inversible et $b \in \mathcal{M}_{n,1}(\mathbb{K})$, par des méthodes directes, *i.e.* on calcule la solution exacte du système.

5.1 Algorithme de Gauss

Soient $A := (a_{ij})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$ inversible, $b := (b_j)_{1\leq j\leq n} \in \mathcal{M}_{n,1}(\mathbb{K})$ et $x =: (x_j)_{1\leq j\leq n} \in \mathcal{M}_{n,1}(\mathbb{K})$ tels que $Ax = b$. Alors, on a

$$\left\{\begin{array}{rcl} a_{11}x_1 + \cdots + a_{1n}x_n & = b_1 & (L_1) \\ a_{21}x_1 + \cdots + a_{2n}x_n & = b_2 & (L_2) \\ \vdots & & \vdots \\ a_{n1}x_1 + \cdots + a_{nn}x_n & = b_n & (L_n) \end{array}\right. \tag{5.1.1}$$

La méthode de Gauss consiste à réduire (5.1.1). sous la forme d'un système triangulaire supérieur en effectuant des opérations sur les lignes (L_i). Puisque A est inversible, chacune de ses colonnes est non nulle. En particulier, il existe $i \in \{1, \ldots, n\}$ tel que $a_{i1} \neq 0$.

Supposons $a_{11} \neq 0$. Alors, on effectue les opérations

$$L_i \to L_i - \frac{a_{i1}}{a_{11}}\, L_1, \quad \forall\, i = 2, \ldots, n.$$

Le système devient alors

$$\left\{\begin{array}{rcccl} a_{11}x_1 & + a_{12}x_2 \quad \ldots & + a_{1n}x_n & = b_1 & (L_1) \\ & a_{22}^{(2)}x_2 \quad + \ldots & + a_{2n}^{(2)}x_n & = b_2 - \frac{a_{21}}{a_{11}}\, b_1 & (L_2) \\ & \vdots & & & \vdots \\ & a_{n2}^{(2)}x_2 \quad + \ldots & + a_{nn}^{(2)}x_n & = b_n - \frac{a_{n1}}{a_{11}}\, b_1 & (L_n) \end{array}\right.$$

qui s'écrit encore $A^{(2)}x = b^{(2)}$ avec

$$A^{(2)} := \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix} \quad \text{et} \quad b^{(2)} := \begin{pmatrix} b_1 \\ b_2 - \frac{a_{21}}{a_{11}} b_1 \\ \vdots \\ b_n - \frac{a_{n1}}{a_{11}} b_1 \end{pmatrix}.$$

Le système ayant une solution unique, $A^{(2)}$ est inversible, d'où

$$0 \neq \det(A^{(2)}) = a_{11} \begin{vmatrix} a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & & \vdots \\ a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{vmatrix}.$$

En particulier, $(a_{22}^{(2)}, \dots, a_{n2}^{(2)})^T \neq 0$ donc il existe $i_0 \in \{2, \dots, n\}$ tel que $a_{i_0 2}^{(2)} \neq 0$. On permute la i_0-ème ligne et la 2-ème ligne, ce qui donne

$$\left\{ \begin{array}{lllll} a_{11}x_1 & + a_{12}x_2 & \dots & +a_{1n}x_n & = b_1 \qquad (L_1) \\ & a_{i_0 2}^{(2)}x_2 & + \dots & +a_{i_0 n}^{(2)}x_n & = b_{i_0}^{(2)} \qquad (L_2^{(2)}) \\ & \vdots & & & \vdots \\ & a_{22}^{(2)}x_2 & + \dots & +a_{2n}^{(2)}x_n & = b_2^{(2)} \qquad (L_{i_0}^{(2)}) \\ & \vdots & & & \vdots \\ & a_{n2}^{(2)}x_2 & + \dots & +a_{nn}^{(2)}x_n & = b_n^{(2)} \qquad (L_n^{(2)}) \end{array} \right. \tag{5.1.2}$$

Ensuite, on effectue les opérations

$$L_i^{(2)} \to L_i^{(2)} - \frac{a_{i2}^{(2)}}{a_{i_0 2}^{(2)}} L_2^{(2)}, \quad \forall\, i = 3, \dots, n.$$

Ce qui donne un système $A^{(3)}x = b^{(3)}$ avec

$$A^{(3)} := \begin{pmatrix} a_{11} & a_{12} & \dots & \dots & a_{1n} \\ 0 & a_{i_0 2}^{(2)} & \dots & \dots & a_{i_0 n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} \\ \vdots & \vdots & & \vdots & \\ 0 & 0 & a_{n3}^{(3)} & \dots & a_{nn}^{(3)} \end{pmatrix}.$$

La méthode présentée ci-dessus se résume alors par

1. On pose $A^{(1)} := A$ et $b^{(1)} := b$.
2. Pour tout $k = 1, \dots, n-1$.
   - Si $a_{kk}^{(k)} \neq 0$. On effectue

$$L_i \to L_i - \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} L_k, \quad \forall\, i = k+1, \dots, n.$$

Le nouveau système s'écrit $A^{(k+1)}x = b^{(k+1)}$.

Algorithme 4 Algorithme de Gauss sans permutation

Entrées : A, b
Sorties : x
  Pour $k = 1 : n-1$ **faire**
    Pour $i = k+1 : n$ **faire**
      $p = a_{ik}/a_{kk}$
      Pour $j = k : n$ **faire**
        $a_{ij} = a_{ij} - p\, a_{kj}$
      fin Pour
      $b_i = b_i - p\, b_k$
    fin Pour
  fin Pour
  $x = \text{Remonte}(A, b)$

- Sinon soit $k_0 \in \{k+1, \dots, n\}$ tel que $a_{k_0 k}^{(k)} \neq 0$. On permute la k_0-ème ligne et la k-ème ligne, puis on effectue

$$L_i \to L_i - \frac{a_{ik}^{(k)}}{a_{k_0 k}^{(k)}}\, L_k, \quad \forall\ i = k+1, \dots, n.$$

  Le nouveau système s'écrit $A^{(k+1)}x = b^{(k+1)}$.

Au final, le système $A^{(n-1)}x = b^{(n-1)}$ est triangulaire supérieure et peut être résolu par l'algorithme de remontée.

Remarque 5.1. À chaque itération, on sépare les cas $a_{kk}^{(k)} = 0$ et $a_{kk}^{(k)} \neq 0$. Ce terme $a_{kk}^{(k)}$ est appelé le **pivot de Gauss** à l'étape k. Dans la pratique, il faut décider d'une méthode sur les lignes à permuter lorsque $a_{kk}^{(k)} = 0$. La plus employée est l'**algorithme de Gauss avec pivot partiel** qui consiste à permuter la k-ème ligne avec la i_0-ème ligne telle que $|a_{i_0 k}^{(k)}| = \max_{k \leq i \leq n} |a_{ik}^{(k)}|$. C'est cet algorithme qui sera traité en Tp.

Dans le cas sans permutation, on aboutit à l'Algorithme 4.

Calculons le nombre d'opérations nécessaires à l'Algorithme 4. Dans la boucle i, on effectue une division, puis $n-k+1$ multiplications ($n-k$ multiplications dans $L_i - c\, L_k$ et 1 multiplication pour $b_i - c\, b_k$). Puisque i varie de $k+1$ à n, cela donne

$$(n-k-1)(n-k+2) = (n-k)^2 + n - k - 2$$

opérations. Comme k varie de 1 à $n-1$, on obtient

$$\sum_{k=1}^{n-1} (n-k)^2 + n - k - 2 = \sum_{k=1}^{n-1} k^2 + k - 2(n-1) = \frac{n(2n-1)(n-1)}{6} + \frac{n(n-1)}{2} - 2(n-1) \sim \frac{n^3}{3},$$

opérations. Autrement dit, la mise sous forme triangulaire du système requiert une complexité de $\frac{n^3}{3}$, la résolution du système par l'algorithme de remontée ayant une complexité de $\frac{n^2}{2}$, on obtient que l'algorithme de Gauss a pour complexité $\frac{n^3}{3}$.

5.2 Décomposition LU

Lorsque l'on est amené à résoudre N systèmes linéaires $Ax = b_i$, $1 \leq i \leq N$, avec la même matrice A l'algorithme de Gauss nécessite alors $N \frac{n^3}{3}$ opérations, ce qui peut devenir très coûteux si N est grand. Or dans ce cas les opérations effectuées pour chacun des systèmes $Ax = b_i$, $1 \leq i \leq N$, sont les mêmes. Il est donc plus avantageux d'effectuer une seule fois ces opérations, les « mémoriser » puis les appliquer à chacun des systèmes. C'est l'objet de la décomposition LU qui consiste à écrire A comme un produit de deux matrices, l'une comportant toutes les opérations de l'algorithme de Gauss et l'autre la forme triangulaire supérieure obtenue à la fin de l'algorithme. Pour cela, on réécrit l'algorithme de Gauss en remplaçant les opérations effectuées par des multiplications de A par des matrices particulières.

Définition 5.2. Soient $k \in \{1, \ldots, n-1\}$ et $l_{(k+1)k}, \ldots, l_{nk} \in \mathbb{R}$. On appelle **matrice élémentaire** $E^{(k)}$ toute matrice sécrivant

$$E^{(k)} := \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -l_{(k+1)k} & 1 & \\ & & \vdots & & \ddots \\ & & -l_{nk} & & 1 \end{pmatrix},$$

les termes n'apparaissant pas étant nuls.

Remarque 5.3. Soient $k \in \{1, \ldots, n-1\}$, $l_{(k+1)k}, \ldots, l_{nk} \in \mathbb{R}$ et $A \in \mathcal{M}_n(\mathbb{K})$.On vérifie aisément les propriétés suivantes :

1. $E^{(k)}A$ est la matrice obtenue à partir de A en effectuant les opérations $L_i \to L_i - l_{ik}\, L_k$ pour tout $i = k+1, \ldots, n$.
2. $E^{(k)}$ est inversible et son inverse est la matrice $E^{(k)}$ où l'on a remplacé l_{ik} par $-l_{ik}$, pour tout $i = k+1, \ldots, n$.

Exemple 5.4. En dimension 3, la matrice $E^{(1)}$ effectuant les opérations $L_2 \to L_2 - 2L_1$ et $L_3 \to L_3 - 3L_1$ est

$$E^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & 0 & 1 \end{pmatrix},$$

celle effectuant $L_3 \to L_3 - \alpha\, L_2$ est

$$E^{(2)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\alpha & 1 \end{pmatrix}.$$

Leurs inverses sont

$$(E^{(1)})^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{pmatrix} \quad \text{et} \quad (E^{(2)})^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \alpha & 1 \end{pmatrix}.$$

On remarque que l'on a

$$E^{(1)}E^{(2)} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & -\alpha & 1 \end{pmatrix},$$

donc $E^{(1)}E^{(2)}$ effectue toutes les opérations. Par contre $E^{(1)}E^{(2)} \neq E^{(2)}E^{(1)}$. Mais on a

$$(E^{(2)}E^{(1)})^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & \alpha & 1 \end{pmatrix},$$

donc $(E^{(2)}E^{(1)})^{-1}$ contient toutes les opérations inverses. De manière plus générale, on obtient le résultat suivant :

Lemme 5.5. *Pour tout $k \in \{1, \ldots, n-1\}$, on pose*

$$E^{(k)} := \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -l_{(k+1)k} & 1 & & \\ & & \vdots & & \ddots & \\ & & -l_{nk} & & & 1 \end{pmatrix},$$

où $l_{(k+1)k}, \ldots, l_{nk} \in \mathbb{R}$. Alors, on a

$$(E^{(k+1)}E^{(k)})^{-1} := \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & l_{(k+1)k} & 1 & & \\ & & \vdots & l_{(k+2)(k+1)} & \ddots & \\ & & \vdots & \vdots & & \\ & & l_{nk} & l_{n(k+1)} & & 1 \end{pmatrix}.$$

En particulier, la matrice $L := (E^{(n-1)} \ldots E^{(2)}\, E^{(1)})^{-1}$ est triangulaire inférieure avec des 1 sur la diagonale et ses coefficients sous-diagonaux sont les l_{ij}.

Définition 5.6. On dit que $P \in \mathcal{M}_n(\mathbb{K})$ est une **matrice de permutations** si la multiplication à gauche (resp. à droite) par P permute des lignes (resp. colonnes).

Remarque 5.7. Toute matrice de permutations s'obtient en permutant les lignes correspondantes dans la matrice identité. Par exemple, la matrice de permutation de la ligne i et de la ligne k s'obtient en permutant la ligne i et la ligne k de la matrice identité.

Avec ces définitions et notations, l'algorithme de Gauss se réécrit sous la forme suivante :

1. $A^{(1)} := A$

2. Pour $k = 1$ à $n-1$,
   - Si $a_{kk}^{(k)} \neq 0$, on pose
   $$A^{(k+1)} = E^{(k)} A^{(k)},$$
   où $l_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$, pour tout $i = k+1, \dots, n$.
   - Si $a_{kk}^{(k)} = 0$, il existe $i_0 \in \{k+1, \dots, n\}$ tel que $a_{i_0 k}^{(k)} \neq 0$. On note $B^{(k)} := P^{(k)} A^{(k)}$, où $P^{(k)}$ est la matrice de permutation des lignes i_0 et k, puis on pose
   $$A^{(k+1)} = E^{(k)} B^{(k)},$$
   où $l_{ik} = \frac{b_{ik}^{(k)}}{b_{kk}^{(k)}}$, pour tout $i = k+1, \dots, n$. De plus, $b_{ik}^{(k)} = a_{ik}^{(k)}$ si $i \neq i_0$ et k, et $b_{i_0 k}^{(k)} = a_{kk}^{(k)}$, $b_{kk}^{(k)} = a_{i_0 k}^{(k)}$.

En particulier, si le pivot de Gauss ne s'annule jamais, on obtient $A^{(n-1)} = E^{(n-2)} \dots E^{(2)} E^{(1)} A$. En posant $U := A^{(n-1)}$ et $L := (E^{(n-2)} \dots E^{(2)} E^{(1)})^{-1}$, on a $A = LU$ avec U triangulaire supérieure et L triangulaire inférieure avec des 1 sur la diagonale.

Définition 5.8. On dit qu'une matrice $A \in \mathcal{M}_n(\mathbb{R})$ admet une **décomposition** LU s'il existe une matrice triangulaire inférieure L (lower) avec des 1 sur la diagonale et une matrice triangulaire supérieure U (upper) telles que $A = LU$.

Théorème 5.9. *Soit $A \in \mathcal{M}_n(\mathbb{R})$ inversible. Alors, il existe une matrice de permutations P telle que PA admet une unique décomposition LU.*

Démonstration. Pour tout $k = 1, \dots, n-1$, on a $A^{(k+1)} = E^{(k)} P^{(k)} A^{(k)}$, où $P^{(k)}$ permute la ligne k et une ligne $i \geq k$ (si $i = k$, P est la matrice identité). On a

$$A^{(n-1)} = E^{(n-2)} P^{(n-2)} E^{(n-3)} \dots E^{(2)} P^{(2)} E^{(1)} P^{(1)} A.$$

On étudie les termes de la forme $P^{(k+1)} E^{(k)} P^{(k)}$. Puisque $P^{(k+1)} P^{(k+1)} = I_n$, on obtient

$$P^{(k+1)} E^{(k)} P^{(k)} = P^{(k+1)} E^{(k)} P^{(k+1)} P^{(k+1)} P^{(k)} = P^{(k+1)} E^{(k)} P^{(k+1)} Q^{(k)},$$

où $Q^{(k)} := P^{(k+1)} P^{(k)}$ est une matrice de permutations. De plus, si $P^{(k+1)}$ permute la ligne $k+1$ et la ligne i_0, on a

$$P^{(k+1)} E^{(k)} P^{(k+1)} = P^{(k+1)} \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & -l_{(k+1)k} & 1 & & & \\ & & \vdots & & \ddots & & \\ & & -l_{i_0 k} & & & 1 & \\ & & \vdots & & & & \ddots \\ & & -l_{nk} & & & & 1 \end{pmatrix} P^{(k+1)}$$

$$P^{(k+1)}E^{(k)}P^{(k+1)} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & -l_{i_0k} & & 1 & & \\ & & \vdots & & & \ddots & \\ & & -l_{(k+1)k} & 1 & & & \\ & & \vdots & & & & \ddots \\ & & -l_{nk} & & & & 1 \end{pmatrix} P^{(k+1)}$$

$$= \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & -l_{i_0k} & 1 & & & \\ & & \vdots & & \ddots & & \\ & & -l_{(k+1)k} & & 1 & & \\ & & \vdots & & & \ddots & \\ & & -l_{nk} & & & & 1 \end{pmatrix} = F^{(k)},$$

où $F^{(k)}$ est une matrice élémentaire. Par récurrence, on obtient qu'il existe une matrice de permutations P telle que

$$A^{(n-1)} = F^{(n-2)}F^{(n-3)} \dots F^{(2)}F^{(1)}PA.$$

Or les $F^{(k)}$ sont des matrices élémentaires, donc $PA = LU$ avec $U := A^{(n-1)}$ et $L := (F^{(n-2)}F^{(n-3)} \dots F^{(2)}F^{(1)})^{-1}$ qui est bien triangulaire inférieure avec des 1 sur la diagonale. Ce qui montre l'existence. Pour montrer l'unicité, supposons que PA admet deux décompositions LU : $PA = LU = L'U'$. Alors, A, L et L' sont inversibles donc U et U' aussi, d'où $(L')^{-1}L = U'U^{-1}$. Or $L(L')^{-1}$ est triangulaire inférieure comme produit de matrices triangulaires inférieures, et, de même, $U'U^{-1}$ est triangulaire supérieure. Donc $(L')^{-1}L = U'U^{-1}$ est diagonale. Puisque L et L' ont des 1 sur la diagonale et sont triangulaires, $(L')^{-1}L$ aussi. Autrement dit, $(L')^{-1}L$ est diagonale avec des 1 sur la diagonale, *i.e.* $(L')^{-1}L = I_n$ donc $L = L'$ et $U'U^{-1} = I_n$, d'où $U = U'$. □

Il reste à déterminer dans quel cas il est possible d'obtenir une décomposition LU sans permutation.

Définition 5.10. Soit $A := (a_{ij})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$. On appelle **sous-matrice principale** de A d'ordre $k \in \{1, \dots, n\}$, la matrice $\Delta^k \in \mathcal{M}_k(\mathbb{K})$ donnée par

$$\Delta^k := \begin{pmatrix} a_{11} & \dots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \dots & a_{kk} \end{pmatrix}.$$

Théorème 5.11. *Soit $A \in \mathcal{M}_n(\mathbb{R})$. Si toutes les sous-matrices principales Δ^k, $k = 1, \dots, n-1$, de A sont inversibles, alors A admet une décomposition LU. De plus, si A est inversible cette décomposition est unique.*

Démonstration. L'unicité se montre de la même manière que dans le cas avec permutation. Pour l'existence, il suffit de prouver qu'à chaque étape de l'algorithme de Gauss le pivot de Gauss est non nul, *i.e.*, $a_{kk}^{(k)} \neq 0$, pour tout $k = 1, \dots, n-1$. On raisonne par récurrence.

Pour $k = 1$, $a_{11}^{(1)} = \Delta^1 \neq 0$.

Supposons $a_{11}^{(1)}, \dots, a_{kk}^{(k)} \neq 0$. Alors, on peut effectuer l'algorithme de Gauss sans permutation, jusqu'à l'itération $k+1$:

$$A^{(k+1)} = E^{(k)} E^{(k-1)} \dots E^{(1)} A.$$

On pose $L := (E^{(k)} E^{(k-1)} \dots E^{(1)})^{-1} = (l_{ij})_{1 \leq i,j \leq n}$. Alors, L est triangulaire inférieure avec des 1 sur la diagonale donc inversible et $LA^{(k+1)} = A$. Ce qui s'écrit en forme par blocs :

$$\begin{pmatrix} L^{(k+1)} & \mathbf{0} \\ \cdot & \cdot \end{pmatrix} \begin{pmatrix} U^{(k+1)} & \cdot \\ \cdot & \cdot \end{pmatrix} = \begin{pmatrix} \Delta^{k+1} & \cdot \\ \cdot & \cdot \end{pmatrix},$$

où la matrice $L^{(k+1)} := (l_{ij})_{1 \leq i,j \leq k+1}$ est triangulaire inférieure avec des 1 sur la diagonale et la matrice $U^{(k+1)} := (a_{ij}^{(k+1)})_{1 \leq i,j \leq k+1}$ est triangulaire supérieure. On en déduit $L^{(k+1)} U^{(k+1)} = \Delta^{k+1}$ donc

$$\det(U^{(k+1)}) = \det(L^{(k+1)} U^{(k+1)}) = \det(\Delta^{k+1}) \neq 0.$$

Or on a $\det(U^{(k+1)}) = \prod_{i=1}^{k+1} a_{ii}^{(i)}$, d'où $a_{(k+1)(k+1)}^{(k+1)} \neq 0$. □

Remarque 5.12. D'après la démonstration précédente, si une matrice $A \in \mathcal{M}_n(\mathbb{R})$ admet une décomposition LU, alors ses sous-matrices principales Δ^k, $k = 1, \dots, n$, vérifient

$$\det(\Delta^k) = \prod_{i=1}^{k} u_{ii}, \quad \forall\ k = 1, \dots, n,$$

où $U = (u_{ij})_{1 \leq i,j \leq n}$.

On écrit ci-dessous l'algorithme de décomposition LU dans le cas où A admet une décomposition LU (le cas avec permutation sera fait en Tp). On note

$$L = I_n + \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ l_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ l_{n1} & \dots & l_{n(n-1)} & 0 \end{pmatrix} \quad \text{et} \quad U = \begin{pmatrix} u_{11} & \cdots & \cdots & u_{1n} \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & u_{nn} \end{pmatrix}$$

et on pose

$$Y := \begin{pmatrix} u_{11} & \cdots & \cdots & u_{1n} \\ l_{21} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ l_{n1} & \dots & l_{n(n-1)} & u_{nn} \end{pmatrix}.$$

On écrit alors un algorithme qui renvoie la matrice $Y = (y_{ij})_{1 \leq i,j \leq n}$ (Algorithme 5).

Algorithme 5 Algorithme de décomposition LU

Entrées : A
Sorties : Y
  $Y = A$
  Pour $k = 1 : n-1$ **faire**
    Pour $i = k+1 : n$ **faire**
      $y_{ik} = y_{ik}/y_{kk}$
      Pour $j = k+1 : n$ **faire**
        $y_{ij} = y_{ij} - y_{ik}\, y_{kj}$
      fin Pour
    fin Pour
  fin Pour

L'algorithme de décomposition LU étant juste une réécriture de l'algorithme de Gauss, celui-ci a une complexité de l'ordre de $\frac{n^3}{3}$. Revenons à la résolution de N systèmes linéaires $Ax = b_i$, $1 \leq i \leq N$, avec la même matrice A. On calcule P, L et U tels que $PA = LU$ (l'algorithme correspondant sera traité en Tp). Alors, on a $PAx = Pb_i$, pour tout $1 \leq i \leq N$. Ensuite, pour tout $1 \leq i \leq N$, on résout $Ly = Pb_i$ par l'algorithme de descente (donc $\frac{n^2}{2}$ opérations) puis $Ux = y$ par l'algorithme de remontée. Ainsi, par l'utilisation de la décomposition LU, on obtient une complexité totale de l'ordre de $\frac{n^3}{3} + N\, n^2$, ce qui est beaucoup plus intéressant que la complexité $N\, \frac{n^3}{3}$ obtenue en résolvant chaque système par l'algorithme de Gauss.

5.3 Méthode de Cholesky

Lemme 5.13. *Toute matrice symétrique définie positive admet une unique décomposition LU.*

Démonstration. Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive. Pour tout $k = 1, \ldots, n$, soit Δ^k la sous-matrice principale d'ordre k de A. On fixe $k \in \{1, \ldots, n\}$ et on suppose que Δ^k n'est pas inversible. Alors, il existe $y := (y_1, \ldots, y_k)^T \in \mathcal{M}_{k,1}(\mathbb{R}) \setminus \{0\}$ tel que $\Delta^k y = 0$. On pose $x := (y_1, \ldots, y_k, 0, \ldots, 0) \in \mathcal{M}_{n,1}(\mathbb{R})$. Alors, on a

$$Ax \cdot x = \sum_{i,j=1}^{n} a_{ij} x_i x_j = \sum_{i,j=1}^{k} a_{ij} y_i y_j = \Delta^k y \cdot y = 0.$$

Ce qui est impossible puisque $x \neq 0$ et A est définie positive. On en déduit que toutes les sous-matrices principales de A sont inversibles. Donc, d'après le Théorème 5.11., A admet une unique décomposition LU. □

Théorème 5.14. *Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive. Alors, il existe une unique matrice $B \in \mathcal{M}_n(\mathbb{R})$ triangulaire inférieure de coefficients diagonaux strictement positifs telle que $A = BB^T$. Cette décomposition est appelée la* **décomposition de Cholesky** *de A.*

Remarque 5.15. La matrice triangulaire inférieure de la décomposition de Cholesky est notée B pour ne pas la confondre avec la matrice triangulaire inférieure L de la décomposition LU car on a, en général, $L \neq B$. L'avantage de la décomposition de Cholesky par rapport à la décomposition LU est qu'il n'y a qu'une seule matrice à calculer et donc le temps de calcul est moins important (cela sera justifié plus loin).

Démonstration. D'après le Lemme 5.13., A admet une décomposition LU. On note $U := (u_{ij})_{1\leq i,j\leq n}$ et, pour tout $k = 1, \ldots, n$, Δ^k la sous-matrice principale d'ordre k de A. Soit $k \in \{1, \ldots, n\}$ fixé. D'après la Remarque 5.12., on a $\prod_{i=1}^{k} u_{ii} = \det(\Delta^k)$. Or on a $\det(\Delta^k) > 0$ (voir Td). On en déduit $u_{ii} > 0$, pour tout $i = 1, \ldots, n$. On pose

$$D := \operatorname{diag}(\sqrt{u_{11}}, \ldots, \sqrt{u_{nn}}), \quad B := LD \quad \text{et} \quad C := D^{-1}U.$$

Alors, B est triangulaire inférieure de coefficients diagonaux $\sqrt{u_{11}}, \ldots, \sqrt{u_{nn}}$ qui sont strictement positifs, en particulier B est inversible. De plus, $A = LU = BC$. Montrons que $C = B^T$. Puisque A est symétrique, on a

$$BC = A = A^T = C^T B^T,$$

donc $C(B^T)^{-1} = B^{-1}C^T$. Or C et B^T sont triangulaires supérieures alors que B^{-1} et C^T sont triangulaires inférieures. On en déduit que $C(B^T)^{-1} = B^{-1}C^T$ est diagonale. Comme B et C ont pour coefficients diagonaux $\sqrt{u_{11}}, \ldots, \sqrt{u_{nn}}$, $B^{-1}C^T$ a tous ses coefficients diagonaux égaux à 1, d'où $C(B^T)^{-1} = B^{-1}C^T = I_n$, ce qui donne $C = B^T$.

Il reste à montrer l'unicité. Supposons que A admet deux décompositions de Cholesky $A = B_1B_1^T = B_2B_2^T$. Alors, on a $B_2^{-1}B_1 = B_2^T(B_1^T)^{-1}$. Comme précédemment, on en déduit que $B_2^{-1}B_1 = B_2^T(B_1^T)^{-1}$ est diagonale égale à $D := \operatorname{diag}(d_1, \ldots, d_n)$. On a $B_1 = B_2D$ donc $A = B_2DD^TB_2^T = B_2B_2^T$. Puisque B_2 est inversible, on en déduit $D^2 = DD^T = I_n$. Donc $d_i^2 = 1$, pour tout $i = 1, \ldots, n$. Or B_1 et B_2 étant triangulaires de coefficients diagonaux strictement positifs, on a $d_i > 0$, pour tout $i = 1, \ldots, n$. Il s'en suit $D = I_n$, d'où $B_1 = B_2D = B_2$. □

Remarque 5.16. Contrairement au cas de la décomposition LU, la preuve de l'existence de la décomposition de Cholesky ne donne pas de méthode de calcul de celle-ci qui soit utilisable en pratique. En effet, la méthode présente dans la démonstration nécessite de calculer d'abord la décomposition LU puis de calculer B en effectuant le produit LD ce qui serait finalement plus couteux que le seul calcul de la décomposition LU.

Soit $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive de décomposition de Cholesky $A = BB^T$. Puisque B est triangulaire inférieure, il n'est nécessaire de ne calculer que la moitié des coefficients de B (ceux non nuls), de plus chaque colonne de B contient un coefficient non nul de moins que la précédente. Prenant cela en compte, on va déterminer chaque colonne de B en connaissant la précédente. Puisque $b_{ij} = 0$ si $j > i$, de l'égalité $A = BB^T$, on obtient

$$\forall\, i, j = 1, \ldots, n, \quad a_{ij} = \sum_{k=1}^{n} b_{ik}b_{jk} = \sum_{k=1}^{\min(i,j)} b_{ik}b_{jk}.$$

Procédons par récurrence.

Pour la première colonne $j = 1$, on a

$$\forall\ i = 1, \dots, n, \quad a_{i1} = b_{i1} b_{11},$$

d'où

$$b_{11} = \sqrt{a_{11}} \quad \text{et} \quad b_{i1} = \frac{a_{i1}}{b_{11}}, \quad \forall\ i = 2, \dots, n.$$

On suppose que l'on connait les $j-1$ première colonnes de B et l'on veut déterminer la j-ème colonne de B. Soit $i \in \{j, \dots, n\}$.

- Si $i = j$. On obtient

$$a_{jj} = b_{jj}^2 + \sum_{k=1}^{j-1} b_{jk}^2.$$

  Les termes b_{jk} pour $k = 1, \dots, j-1$ appartiennent aux $j-1$ première colonnes de B donc sont connus. On en déduit

$$b_{jj} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} b_{jk}^2}.$$

- Si $i > j$, on a

$$a_{ij} = \sum_{k=1}^{j} b_{ik} b_{jk} = \sum_{k=1}^{j-1} b_{ik} b_{jk} + b_{ij} b_{jj}.$$

  Les termes de la somme et b_{jj} sont connus donc on a

$$b_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} b_{ik} b_{jk}}{b_{jj}}.$$

Cette procédure de détermination de B donne l'Algorithme 6. À partir de cet algorithme, il est alors possible de déterminer la complexité du calcul de la décomposition de Cholesky. Le calcul des termes diagonaux nécessite $(j-1)$ produits (et une extraction de racine). Celui des termes non diagonaux nécessite $(n-j-1)j$ multiplications et divisions. Ainsi, dans la boucle j, il y a $(n-j-1)j+j-1$ opérations. On en déduit que la complexité du calcul de la décomposition de Cholesky est de l'ordre de

$$\sum_{j=1}^{n}((n-j-1)j+j-1) = \frac{n^2(n+1)}{2} - \frac{n(2n+1)(n-1)}{6} - n \sim \frac{n^3}{2} - \frac{n^3}{3} = \frac{n^3}{6}.$$

Finalement, le calcul de la décomposition de Cholesky est deux fois plus rapide que celui de de la décomposition LU. Ainsi, pour une matrice symétrique définie positive, il est préférable d'utiliser la décomposition de Cholesky.

5.4 Décomposition QR

On finit ce chapitre avec la décomposition QR pour laquelle on énonce seulement le résultat, celui-ci étant de peu d'intérêt dans le cas de matrices carrées (par rapport

Algorithme 6 Algorithme de décomposition de Cholesky

Entrées : A
Sorties : B
  $B = 0$
  Pour $j = 1 : n$ **faire**
    $s = 0$
    Pour $k = 1 : j - 1$ **faire**
      $s = s + b_{jk}^2$
    fin Pour
    $b_{jj} = \sqrt{a_{jj} - s}$
    Pour $i = j + 1 : n$ **faire**
      $s = 0$
      Pour $k = 1 : j - 1$ **faire**
        $s = s + b_{ik} b_{jk}$
      fin Pour
      $b_{ij} = (a_{ij} - s)/b_{jj}$.
    fin Pour
  fin Pour

aux décompositions précédentes). Néanmoins, dans le chapitre suivant, on étudiera la généralisation de ce résultat au cas des matrices rectangulaires où cette méthode présente un grand intérêt.

Théorème 5.17. *Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice inversible. Alors, il existe un unique couple de matrices (Q, R), où $Q \in \mathcal{M}_n(\mathbb{R})$ est unitaire et $R \in \mathcal{M}_n(\mathbb{R})$ est triangulaire supérieure de coefficients diagonaux strictement positifs tel que $A = QR$. Cette décomposition est appelée la* **décomposition** *QR de A.*

Schéma de preuve. Soient $c_1, \ldots, c_n$ les colonnes de A. Alors $\{c_1, \ldots, c_n\}$ est une base de $\mathbb{R}^n$. On applique le procédé d'orthonormalisation de Gramm-Schmidt : on pose

$$\forall\ i = 1, \ldots, n, \quad \tilde{e}_i := c_i - \sum_{j=1}^{i-1} (c_i, e_j)\, e_j \quad \text{et} \quad e_i := \frac{\tilde{e}_i}{\|\tilde{e}_i\|_2}.$$

On définit Q comme étant la matrice de colonnes $e_1, \ldots, e_n$ et R comme étant la matrice de coefficients r_{ij} tels que $r_{ij} = 0$ si $i > j$ $r_{ii} := \|\tilde{e}_i\|_2$ et $r_{ij} = (e_i, c_j)$ si $i < j$. On vérifie alors que $A = QR$. □

Exercice 5.18. Écrire l'algorithme de décomposition QR correspondant à la démonstration décrite ci-dessus. En déduire que ce calcul de la décomposition QR a une complexité de l'ordre de n^3.

Remarque 5.19. Le calcul de la factorisation QR ayant une complexité de l'ordre de n^3, ce qui est trois fois supérieur à la décomposition LU, celle-ci n'est pas compétitive. De plus, le procédé d'orthonormalisation de Gramm-Schmidt est très sensible aux erreurs d'arrondis du fait de la division par $\|\tilde{e}_i\|_2$ qui peut être très petit. Cette méthode est donc inutilisable en pratique.

Chapitre 6

Problème des moindres carrés

Ce chapitre est consacré à la résolution des problèmes $Ax = b$ avec $A \in \mathcal{M}_{m,n}(\mathbb{R})$ et $b \in \mathcal{M}_{m,1}(\mathbb{R})$, avec $m > n$.

6.1 Équation normale et propriétés

On rappelle (voir chapitre 4) que, pour $A \in \mathcal{M}_{m,n}(\mathbb{R})$ et $b \in \mathcal{M}_{m,1}(\mathbb{R})$, on dit que $x \in \mathcal{M}_{n,1}(\mathbb{R})$ est solution aux moindres carrés de $Ax = b$ s'il vérifie

$$\|Ax - b\|_2 = \min_{y \in \mathcal{M}_{n,1}(\mathbb{R})} \|Ay - b\|_2. \tag{6.1.1}$$

Lemme 6.1. *Soient $A \in \mathcal{M}_{m,n}(\mathbb{R})$ et $b \in \mathcal{M}_{m,1}(\mathbb{R})$. Alors, $x \in \mathcal{M}_{n,1}(\mathbb{R})$ est solution de* (6.1.1). *si et seulement si x satisfait l'***équation normale**

$$A^*Ax = A^*b. \tag{6.1.2}$$

Démonstration. Supposons que $x \in \mathcal{M}_{n,1}(\mathbb{R})$ est solution de (6.1.1).. Alors,

$$\forall\, y \in \mathcal{M}_{n,1}(\mathbb{R}), \quad \|Ax - b\|_2 \leq \|Ay - b\|_2.$$

Soient $z \in \mathcal{M}_{n,1}(\mathbb{R})$ et $t \in \mathbb{R}$. En posant $y := x - t\,z$, on obtient

$$\begin{aligned} \|Ay - b\|_2^2 &= (Ay - b, Ay - b)_2 = (Ax - t\,Az - b, Ax - t\,Az - b)_2 \\ &= \|Ax - b\|_2^2 + t^2\,\|Az\|_2^2 - 2t\,(Az, Ax - b). \end{aligned}$$

On en déduit $t^2\,\|Az\|_2^2 - 2t\,(Az, Ax - b) \geq 0$. En prenant $t > 0$ et en divisant par t on a $t\,\|Az\|_2^2 - 2(Az, Ax - b) \leq 0$. Alors, en faisant tendre t vers 0, on obtient $(Az, Ax - b) \leq 0$. Le même raisonnement avec $t < 0$ entraîne $(Az, Ax - b)_2 = 0$, d'où $(A^*Ax - A^*b, z)_2 = 0$, pour tout $z \in \mathcal{M}_{n,1}(\mathbb{R})$. Donc $A^*Ax - A^*b = 0$. Réciproquement, soit $x \in \mathcal{M}_{n,1}(\mathbb{R})$ solution de (6.1.2).. Alors, pour tout $z \in \mathcal{M}_{n,1}(\mathbb{R})$, on a $(A^*Ax - A^*b, z)_2 = 0$. Soient $y \in \mathcal{M}_{n,1}(\mathbb{R})$ et $t \in \mathbb{R}$. Il existe $z \in \mathcal{M}_{n,1}(\mathbb{R})$ tel que $y := x - t\,z$, on obtient

$$\begin{aligned} \|Ay - b\|_2^2 &= \|Ax - b\|_2^2 + t^2\,\|Az\|_2^2 - 2t\,(Az, Ax - b) = \|Ax - b\|_2^2 + t^2\,\|Az\|_2^2 - 2t\,(z, A^*Ax - A^*b) \\ &= \|Ax - b\|_2^2 + t^2\,\|Az\|_2^2 \geq \|Ax - b\|_2^2. \end{aligned}$$

On en déduit que x est bien solution de (6.1.1).. □

Théorème 6.2. *Soient $A \in \mathcal{M}_{m,n}(\mathbb{R})$ et $b \in \mathcal{M}_{m,1}(\mathbb{R})$. Alors, il existe au moins une solution $x \in \mathcal{M}_{n,1}(\mathbb{R})$ de* (6.1.1).. *De plus, cette solution est unique si et seulement si* $\mathrm{Ker}(A) = \{0\}$.

Démonstration. On a $\mathrm{Ker}(A) = \mathrm{Ker}(A^*A)$. En effet, si $x \in \mathrm{Ker}(A)$, alors $Ax = 0$ et donc $A^*Ax = 0$. De plus, si $x \in \mathrm{Ker}(A^*A)$, alors $\|Ax\|_2^2 = (Ax, Ax)_2 = (A^*Ax, x)_2 = 0$. Le problème (6.1.1). a une solution si et seulement si l'équation normale a une solution. Or l'équation normale a une solution si et seulement si $A^*b \in \mathrm{Im}(A^*A)$. Autrement dit, puisque $A^*b \in \mathrm{Im}(A^*)$, il suffit de montrer que $\mathrm{Im}(A^*) \subset \mathrm{Im}(A^*A)$. D'une part, on a $\mathrm{Im}(A^*) = \mathrm{Ker}(A)^\perp$ donc

$$\mathcal{M}_{m,1}(\mathbb{R}) = \mathrm{Ker}(A) \oplus \mathrm{Im}(A^*).$$

D'autre part, puisque A^*A est hermitienne, on a

$$\mathcal{M}_{m,1}(\mathbb{R}) = \mathrm{Ker}(A^*A) \oplus \mathrm{Im}(A^*A).$$

Puisque $\mathrm{Ker}(A) = \mathrm{Ker}(A^*A)$ et $\mathrm{Im}(A^*A) \subset \mathrm{Im}(A^*)$, on a $\mathrm{Im}(A^*) = \mathrm{Im}(A^*A)$. Ce qui donne l'existence d'une solution. On a unicité de la solution si et seulement si A^*A est inversible, d'où $\mathrm{Ker}(A) = \mathrm{Ker}(A^*A) = \{0\}$. □

Dans la pratique, pour résoudre le problème (6.1.1)., on résout l'équation (6.1.2).. Pour cela, il faut d'abord calculer A^*A. Puisque A^*A est symétrique, il suffit de calculer sa partie triangulaire supérieure, soit $\frac{mn(n+1)}{2}$ opérations. Puis il faut calculer A^*b ce qui nécessite mn opérations. Ensuite, l'application de l'algorithme de Gauss demande $\frac{n^3}{3}$ opérations, soit un total de $\frac{mn(n+1)}{2} + \frac{n^3}{3}$ opérations.

De plus, si $\mathrm{Ker}(A) = \{0\}$, alors A^*A est symétrique définie positive (voir Td). On peut alors utiliser la décomposition de Cholesky de A^*A, ce qui nécessite $\frac{mn(n+1)}{2} + \frac{n^3}{6}$ opérations. La méthode QR va nous permettre de réduire ce nombre d'opérations en ne calculant pas le produit A^*A.

6.2 Méthode QR, algorithme de Householder

Supposons que $A \in \mathcal{M}_{m,n}(\mathbb{R})$ admet une décomposition QR, *i.e.* il existe $Q \in \mathcal{M}_m(\mathbb{R})$ unitaire et $R \in \mathcal{M}_{m,n}(\mathbb{R})$ triangulaire supérieure telles que $A = QR$. Alors, on a

$$A^*A = R^*Q^*QR = R^*R.$$

Donc $A^*Ax = A^*b$ équivaut à $R^*Rx = R^*Q^*b$. Ainsi, la résolution de $A^*Ax = A^*b$ se ramène à celle de $Rx = Q^*b$, où R est triangulaire supérieure et Q unitaire. En particulier, il n'est plus nécessaire de calculer A^*A ce qui était l'étape la plus couteuse dans la résolution de l'équation normale. Il reste à déterminer si une telle décomposition est possible et, si oui, quelle est sa complexité.

On donne d'abord l'idée de l'algorithme de Householder : on pose

$$A^{(1)} := A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ a_{22} & & a_{2n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}.$$

Afin de transformer A en une matrice triangulaire supérieure, on multiplie $A^{(1)}$ par une matrice unitaire $H^{(1)} \in \mathcal{M}_m(\mathbb{R})$ telle que la première colonne $(a_{11}, a_{21}, \ldots, a_{m1})^T$ de A est changée en $(c, 0, \ldots, 0)^T$, où $c \in \mathbb{R}$. On obtient

$$A^{(2)} := H^{(1)}A^{(1)} = \begin{pmatrix} c & a_{12}^{(2)} & \ldots & a_{1n}^{(2)} \\ 0 & a_{22}^{(2)} & & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(2)} & \ldots & a_{mn} \end{pmatrix}.$$

Ensuite on ré-effectue la première étape sur la sous matrice

$$\begin{pmatrix} a_{22}^{(2)} & \ldots & a_{2n}^{(2)} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ a_{m2}^{(2)} & \ldots & a_{mn}^{(2)} \end{pmatrix}$$

On obtient alors au bout de $(n+1)$ itérations une matrice triangulaire supérieure R, la matrice unitaire Q associée s'obtient par multiplication des matrices unitaires $H^{(k)}$. La particularité de cet algorithme réside dans le choix des ces matrices $H^{(k)}$.

Définition 6.3. Soit $v \in \mathbb{R}^n \backslash \{0\}$. La **matrice de Householder** $H(v) \in \mathcal{M}_n(\mathbb{R})$ associée à v est la matrice définie par

$$H(v) \ := \ I_n - 2\,\frac{v \otimes v}{\|v\|_2^2},$$

i.e. $H(v)$ a pour coefficients $h_{ij}(v) := \delta_{ij} - 2\,\frac{v_i v_j}{\|v\|_2^2}$, pour tous $i, j = 1, \ldots, n$. Pour $v := 0$, on pose $H(v) = I_n$.

Lemme 6.4. *Soient $v \in \mathbb{R}^n \backslash \{0\}$ et $H(v)$ sa matrice de Householder associée.*

1. *$H(v)$ est symétrique et unitaire.*
2. *Soit $e \in \mathbb{R}^n$ avec $\|e\|_2 = 1$. Si $v \neq \pm\|v\|_2\, e$ (v non colinéaire avec e), alors on a*

$$H(v + \|v\|_2\, e)v = -\|v\|_2\, e \quad \textit{et} \quad H(v - \|v\|_2\, e)v = \|v\|_2\, e$$

Démonstration. Le point 1 sera montré en Td. Dans la suite, on notera $\cdot$ le produit scalaire de $\mathbb{R}^n$ au lieu de $(\cdot, \cdot)_2$. Pour le point 2, on remarque d'abord que, pour tous $u, w \in \mathbb{R}^n$, on a

$$\forall\ i = 1, \ldots, n, \quad ((u \otimes w)v)_i = \sum_{k=1}^{n} (u \otimes w)_{ik} v_k = \sum_{k=1}^{n} u_i w_k v_k = u_i\,(w \cdot v).$$

En particulier, $(u \otimes u)v = (u \cdot v)\,u$. On en déduit

$$(v+\|v\|_2\, e)\otimes(v+\|v\|_2\, e)v = ((v + \|v\|_2\, e) \cdot v)\ (v+\|v\|_2\, e) = \left(\|v\|_2^2 + \|v\|_2\, e \cdot v\right)\ (v+\|v\|_2\, e).$$

De plus, on a

$$\|v + \|v\|_2\, e\|_2^2 = (v + \|v\|_2\, e)\cdot(v + \|v\|_2\, e) = \|v\|_2^2 + 2\|v\|_2\, e\cdot v + \|v\|_2^2 = 2(\|v\|_2^2 + \|v\|_2\, e\cdot v).$$

Donc on obtient

$$H(v + \|v\|_2\, e)v = v - (v + \|v\|_2\, e) = -\|v\|_2\, e.$$

L'autre cas se traite de la même manière. □

Théorème 6.5. *Soit $A \in \mathcal{M}_{m,n}(\mathbb{R})$ avec $m \geq n$. Alors, il existe une matrice unitaire $Q \in \mathcal{M}_m(\mathbb{R})$ et une matrice triangulaire supérieure $R \in \mathcal{M}_{m,n}(\mathbb{R})$ telles que $A = QR$.*

Démonstration. La démonstration qui suit est ce que l'on appelle l'**algorithme de Householder**. Le principe de cet algorithme est de construire une famille $H^{(k)} \in \mathcal{M}_n(\mathbb{R})$ de matrices de Householder et une famille $A^{(k)} \in \mathcal{M}_{m,n}(\mathbb{R})$, $k = 1, \ldots, n+1$, telles que

$$A^{(1)} := A, \quad A^{(k+1)} = H^{(k)}A^{(k)} \quad \text{et} \quad A^{(n+1)} = R,$$

où R est triangulaire supérieure. On obtient alors $R = H^{(n)}H^{(n-1)}\ldots H^{(1)}A$. Puisque toute matrice de Householder est unitaire, $Q := (H^{(n)}H^{(n-1)}\ldots H^{(1)})^T$ est unitaire et $A = QR$. L'ingrédient essentiel dans le choix de ces matrices de Householder est le Lemme 6.4..

Étape 1.

On pose $A^{(1)} := A$. On note a_1 la première colonne de A et $\{e_1, \ldots, e_m\}$ la base canonique de $\mathcal{M}_m(\mathbb{R})$.

- Si $a_1 = (a_{11}, 0, \cdots, 0)^T$, *i.e.* $a_1 = \pm\|a_1\|\, e_1$, on pose $H^{(1)} := I_m$.
- Si $a_1 \neq \pm\|a_1\|\, e_1$, on pose $H^{(1)} := H(a_1 + \|a_1\|\, e_1) \in \mathcal{M}_m(\mathbb{R})$. Alors, d'après le Lemme 6.4., on a $H^{(1)}a_1 = -\|a_1\|\, e_1 = (-\|a_1\|, 0, \cdots, 0)^T$. Ensuite, on pose

$$A^{(2)} := H^{(1)}A^{(1)} = \begin{pmatrix} -\|a_1\| & \cdots & \cdots & \cdots \\ 0 & & & \\ \vdots & & & \\ 0 & \cdots & \cdots & \cdots \end{pmatrix}.$$

Étape k. $(2 \leq k \leq n)$

On suppose avoir construit $A^{(k)}$ telle que les $(k-1)$ premières colonnes de $A^{(k)}$ ont des zéros sous la diagonale. Montrons que l'on peut construire $A^{(k+1)}$ telle que les k première colonnes de $A^{(k+1)}$ ont des zéros sous la diagonale. On note

$$A^{(k)} = \begin{pmatrix} a_{11}^{(k)} & & & & & \\ 0 & a_{22}^{(k)} & & & & \\ \vdots & 0 & \ddots & & & \\ \vdots & & 0 & a_{kk}^{(k)} & & \\ \vdots & & & \vdots & & \\ 0 & \cdots & 0 & a_{mk}^{(k)} & \cdots & a_{mn}^{(k)} \end{pmatrix} = \begin{pmatrix} S^{(k)} & . \\ 0 & B^{(k)} \end{pmatrix},$$

où $S^{(k)} \in \mathcal{M}_{k-1}(\mathbb{R})$ est triangulaire supérieure et $B^{(k)} \in \mathcal{M}_{(m-k+1)(n-k+1)}(\mathbb{R})$. On va utiliser la même méthode qu'à l'étape 1 par blocs en considérant la matrice $B^{(k)}$. Si on note $a_k^{(k)}$ la k-ème colonne de $A^{(k)}$ et $b^{(k)}$ la 1ère colonne de $B^{(k)}$, on a

$$a_k^{(k)} = \begin{pmatrix} a_{1k}^{(k)} \\ \vdots \\ \vdots \\ a_{mk}^{(k)} \end{pmatrix} = \begin{pmatrix} a_{1k}^{(k)} \\ \vdots \\ a_{(k-1)k}^{(k)} \\ b^{(k)} \end{pmatrix}.$$

- Si $b^{(k)} = (a_{kk}^{(k)}, 0, \cdots, 0)^T$, il n'y a rien à faire et donc on pose $H^{(k)} := I_m$.
- Sinon on pose $H := H(b^{(k)} + \|b^{(k)}\|_2\, e_1^{(k)}) \in \mathcal{M}_{m-k+1}(\mathbb{R})$ puis

$$H^{(k)} := \begin{pmatrix} I_{k-1} & \mathbf{0} \\ \mathbf{0} & H \end{pmatrix} \quad \text{et} \quad A^{(k+1)} = H^{(k)} A^{(k)},$$

où $e_1^{(k)}$ est le premier vecteur de la base canonique de $\mathcal{M}_{m-k+1,1}(\mathbb{R})$. Alors, on a

$$A^{(k+1)} = \begin{pmatrix} I_{k-1} & \mathbf{0} \\ \mathbf{0} & H \end{pmatrix} \begin{pmatrix} S^{(k)} & \cdot \\ 0 & B^{(k)} \end{pmatrix} = \begin{pmatrix} S^{(k)} & \cdot \\ 0 & HB^{(k)} \end{pmatrix}.$$

Or la première colonne $c^{(k)}$ de $HB^{(k)}$ est $H(b^{(k)} + \|b^{(k)}\|_2\, e_1^{(k)})b^{(k)} = -\|b^{(k)}\|_2\, e_1^{(k)}$, donc la k-ème colonne de $A^{(k+1)}$ est

$$a_k^{(k+1)} = \begin{pmatrix} a_{1k}^{(k)} \\ \vdots \\ a_{(k-1)(k-1)}^{(k)} \\ -\|b^{(k)}\|_2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Ainsi, les k premières colonnes de $A^{(k+1)}$ sont nulles sous la diagonale. Lorsque $k = n$, on obtient donc que $R := A^{(n+1)}$ est triangulaire supérieure. □

L'algorithme de Householder présenté dans la démonstration peut s'écrire en langage Scilab :

```
[m,n]=size(A); //m>=n
for k=1:n
        v=A(k:m,k); //v est de taille m-k+1
        e=eye(m-k+1,1); // 1er vecteur de base en dimension m-k+1
        H=H(v+norm(v)*e);
        A(k:m,k:n)=H*A(k:m,k:n);
end
```

Algorithme 7 Algorithme de Householder de décomposition QR

Entrées : A
Sorties : R
$R = A$;
Pour $k = 1 : n$ **faire**
$v = (r_{kk}, \ldots, r_{mk})^T$;
$e = (1, 0, \ldots, 0)^T \in \mathcal{M}_{m-k+1,1}(\mathbb{R})$;
$\alpha = -\|v\|_2$ si $v_1 \geq 0$ ou $\|v\|_2$ si $v_1 \leq 0$;
$\beta := \alpha^2 - \alpha\, v_1$;
$v_1 = v_1 - \alpha$;
Pour $j = k : n$ **faire**
$\gamma = \frac{1}{\beta}\, v^T (r_{kj}, \ldots, r_{mj})^T$;
$(r_{kj}, \ldots, r_{mj})^T = (r_{kj}, \ldots, r_{mj})^T - \gamma\, v$;
fin Pour
fin Pour

Cet algorithme peut être amélioré en tenant compte de la forme particulière des matrices H. On remarque que, pour tous vecteurs $b, w \in \mathcal{M}_{k,1}(\mathbb{R})$, on a

$$H(w)b = b - \frac{2}{\|w\|^2}(w \cdot b)w.$$

Ainsi pour calculer les produits $H(w)B$, où B est une matrice, il suffit de calculer w et $\|w\|$ puis d'effectuer les produits $H(w)b$, pour chaque colonne b de B, en utilisant l'égalité précédente. De plus, pour $w := v + \|v\|\, e_1$, on a

$$\|w\|^2 = (w, w) = 2(\|v\|^2 + \|v\|(v, e_1)) = 2(\|v\|^2 + \|v\|\, v_1).$$

Une méthode de calcul possible est alors :

— $\alpha := -\|v\|_2$ si $v_1 \geq 0$, $\alpha := +\|v\|_2$ si $v_1 \leq 0$,
— $\beta := \alpha^2 - \alpha\, v_1$,
— $w_1 = v_1 - \alpha$, $w_i = v_i$, si $2 \leq i \leq m - k + 1$.
— Pour chaque colonne b de B, $H(w)b = b - \frac{1}{\beta}(w \cdot b)w$.

On obtient l'Algorithme 7 (l'algorithme complet renvoyant aussi la matrice Q sera effectué en Tp). Remarquons que le choix du signe de α est fait de sorte que β soit le plus grand possible pour diminuer les erreurs d'arrondis. On peut alors calculer la complexité de l'Algorithme de Householder. Dans la boucle j, on effectue de l'ordre de $2(m - k + 1)$ multiplications. Dans la boucle k, les calculs précédents la boucle j nécessitent de l'ordre

de $m-k+1$ multiplications. La complexité de l'algorithme est alors de l'ordre de

$$\begin{aligned}\sum_{k=1}^{n} 2(n-k)(m-k+1)+2(m-k+1) &= 2\sum_{k=1}^{n}(n-k+1)(m-k+1)\\ &= 2n^2m-2(n+m)\sum_{k=1}^{n-1}k+2\sum_{k=1}^{n-1}k^2\\ &= 2n^2m-(n+m)n(n-1)+\frac{n(n-1)(2n-1)}{3}\\ &\sim 2n^2m-n^2m-n^3+\frac{n^3}{3}\\ &\sim mn^2-\frac{n^3}{3}.\end{aligned}$$

En particulier, la méthode est moins couteuse que le calcul de A^*A suivi de la méthode de Cholesky. Par contre lorsque $m=n$, on obtient une complexité de $\frac{2n^3}{3}$ ce qui est deux fois plus lent que l'algorithme de Gauss.

Remarque 6.6. Les matrices de Householder servent aussi dans le calcul numérique de valeurs propres. De plus, il existe une autre méthode de calcul de la décomposition QR où les matrices de Householder sont remplacées par des matrices de rotation, il s'agit de la méthode de Givens.

Chapitre 7

Méthodes itératives de base

Dans tout ce chapitre, on considère le système linéaire

$$Ax = b, \tag{7.0.1}$$

où $A \in \mathcal{M}_n(\mathbb{R})$ est inversible et $b \in \mathcal{M}_{n,1}(\mathbb{R})$. Une **méthode itérative** de résolution de (7.0.1). est une méthode numérique de calcul d'une suite $(x_k)_{k\in\mathbb{N}}$ convergeant, lorsque $k \to +\infty$, vers la solution x de (7.0.1).. Dans ce cours, on ne considéra que le cas de suites définies par une récurrence simple *i.e.* x_k est construit à partir de x_{k-1} (et non de $x_{k-2}, x_{k-3}, \dots$).

7.1 Présentation des méthodes

Soit $x \in \mathcal{M}_{n,1}(\mathbb{R})$ la solution de (7.0.1).. On suppose la diagonale de A inversible, *i.e.* $a_{ii} \neq 0$, pour tout $i = 1, \dots, n$. On a

$$a_{ii}x_i + \sum_{j\neq i} a_{ij}x_j = b_i, \quad \forall\ i = 1, \dots, n,$$

donc, puisque la diagonale de A est inversible,

$$x_i = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i} a_{ij}x_j\right), \quad \forall\ i = 1, \dots, n.$$

La **méthode de Jacobi** consiste à construire la suite $(x^k)_{k\in\mathbb{N}}$ de $\mathcal{M}_{n,1}(\mathbb{R})$ en posant

$$\begin{cases} x^0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ x_i^{k+1} = \dfrac{1}{a_{ii}}\left(b_i - \displaystyle\sum_{j\neq i} a_{ij}x_j^k\right), \quad \forall\ i = 1, \dots, n. \end{cases}$$

Il est clair que si la suite $(x^k)_{k\in\mathbb{N}}$ converge alors celle-ci converge vers la solution x de (7.0.1).. Si l'on note D la diagonale de A, la méthode de Jacobi s'écrit $x^{k+1} = D^{-1}(b - (A - D)x^k)$, soit encore

$$\begin{cases} x^0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ Dx^{k+1} = (D - A)x^k + b. \end{cases}$$

Supposons avoir construit le vecteur x^k par la méthode de Jacobi. Le calcul de x^{k+1} se fait alors coordonnée par coordonnée en utilisant seulement le vecteur x^k. Il serait plus intéressant de calculer chaque coordonnée x_i^{k+1}, $1 \leq i \leq n$, en faisant appel au vecteur x^k et aux coordonnées x_j^{k+1}, $1 \leq j \leq i$, déjà calculées. Partant de la formule

$$x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j<i} a_{ij} x_j - \sum_{j>i} a_{ij} x_j \right),$$

on obtient la **méthode de Gauss-Seidel** en posant

$$\begin{cases} x^0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ x_i^{k+1} = \dfrac{1}{a_{ii}} \left(b_i - \displaystyle\sum_{j<i} a_{ij} x_j^{k+1} - \sum_{j>i} a_{ij} x_j^k \right), \quad \forall\ i = 1, \ldots, n. \end{cases}$$

Si l'on note $-E$ (resp. $-F$) la partie triangulaire inférieure (resp. supérieure) stricte de A, de sorte que $A = D - E - F$, la méthode de Gauss-Seidel s'écrit $Dx^{k+1} = b + Ex^{k+1} + Fx^k$, soit encore

$$\begin{cases} x^0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ (D - E)x^{k+1} = Fx^k + b. \end{cases}$$

En particulier, on remarque que les deux méthodes s'écrivent sous la forme

$$\begin{cases} x^0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ Mx^{k+1} = Nx^k + b, \end{cases}$$

où $A = M - N$. Dans la section suivante, on va étudier l'ensemble des méthodes itératives s'écrivant sous cette forme générale. On en déduira des résultats de convergence des méthodes Jacobi et de Gauss-Seidel.

7.2 Cadre général

Définition 7.1. Soit $A \in \mathcal{M}_n(\mathbb{R})$ inversible. On appelle **décomposition régulière** (ou **splitting**) de A un couple $M, N \in \mathcal{M}_n(\mathbb{R})$, avec M inversible, tel que

$$A = M - N. \tag{7.2.1}$$

La méthode itérative basée sur la décomposition (7.2.1). est définie par

$$\begin{cases} x_0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ Mx_{k+1} = Nx_k + b, \quad \forall\ k \in \mathbb{N}. \end{cases} \tag{7.2.2}$$

Remarque 7.2. La méthode (7.2.2). permet bien de résoudre (7.0.1).. En effet, si la suite $(x_k)_{k\in\mathbb{N}}$ converge vers x dans $\mathcal{M}_{n,1}(\mathbb{R})$, alors on a $Mx = Nx + b$, donc $Ax = Mx - Nx = b$.

Définition 7.3. Soient $A \in \mathcal{M}_n(\mathbb{R})$ inversible de décomposition régulière $A = M - N$ et $(x_k)_{k\in\mathbb{N}}$ la suite définie par (7.2.2)..

1. On dit que la méthode itérative est **convergente** si, pour tout $x_0 \in \mathcal{M}_{n,1}(\mathbb{R})$, la suite $(x_k)_{k\in\mathbb{N}}$ converge vers la solution exacte x de (7.0.1)..
2. On appelle **erreur** à l'itération $k \in \mathbb{N}$, le vecteur

$$e_k \ := \ x - x_k. \tag{7.2.3}$$

En particulier, une méthode itérative converge si et seulement si l'erreur converge vers 0.

Soient $A \in \mathcal{M}_n(\mathbb{R})$ inversible de décomposition régulière $A = M - N$ et $(x_k)_{k\in\mathbb{N}}$ la suite définie par (7.2.2).. Si on pose $\mathcal{L} := M^{-1}N$, on a

$$\forall\ k \in \mathbb{N}, \quad x_{k+1} = \mathcal{L}x_k + M^{-1}b.$$

On dit que $\mathcal{L}$ est la **matrice d'itération** de la méthode itérative. Cette matrice permet de caractériser la convergence de la méthode.

Théorème 7.4. *Une méthode itérative converge si et seulement si sa matrice d'itération $\mathcal{L}$ vérifie $\rho(\mathcal{L}) < 1$.*

Remarque 7.5. Dans la pratique le calcul du rayon spectral peut être particulièrement compliqué puiqu'il nécessite de calculer les valeurs propres. Ainsi, il est préférable d'utiliser le résultat du Lemme 3.19. qui affirme que $\rho(\mathcal{L}) < 1$ si et seulement s'il existe une norme subordonnée $\|\cdot\|$ telle que $\|\mathcal{L}\| < 1$

Démonstration. Soient $A \in \mathcal{M}_n(\mathbb{R})$ inversible de décomposition régulière $A = M - N$ et $(x_k)_{k\in\mathbb{N}}$ définie par (7.2.2).. Soit $k \in \mathbb{N}$. L'erreur à l'itération $k+1$ est donnée par

$$e_{k+1} = x - x_{k+1} = x - M^{-1}Nx_k - M^{-1}b.$$

On a $Ax = b$ donc $Mx = b + Nx$ et $x = M^{-1}b + M^{-1}Nx$, d'où

$$e_{k+1} = x - M^{-1}Nx_k - x + M^{-1}Nx = M^{-1}N(x - x_k) = M^{-1}Ne_k.$$

On en déduit

$$\forall\ k \in \mathbb{N}, \quad e_k = (M^{-1}N)^k e_0. \tag{7.2.4}$$

Alors, d'après le Lemme 3.19., la suite $(e_k)_{k\in\mathbb{N}}$ converge vers 0 si et seulement $\rho(M^{-1}N) < 1$. □

Lorsque A est hermitienne définie positive, on obtient une caractérisation plus simple.

Théorème 7.6. *Soient $A \in \mathcal{M}_n(\mathbb{R})$ hermitienne définie positive de décomposition régulière $A = M - N$. Alors, la matrice $M^* + N$ est hermitienne. De plus, si $M^* + N$ est définie positive alors la méthode itérative associée à la décomposition $A = M - N$ converge.*

Démonstration. On a

$$(M^* + N)^* = M + N^* = A + N + N^* = A^* + N + N^* = M^* - N^* + N + N^* = M^* + N.$$

Donc $M^* + N$ est hermitienne. Puisque A est hermitienne définie positive, l'application $(x, y) \mapsto (Ax, y)_2$ est un produit scalaire sur $\mathbb{R}^n$ de norme associée

$$\forall\ x \in \mathbb{R}^n, \quad |x|_A\ :=\ \sqrt{(Ax, x)_2}.$$

On note $\|\cdot\|_A$ la norme subordonnée sur $\mathcal{M}_n(\mathbb{R})$ associée à la norme vectorielle $|\cdot|_A$. On va montrer que $\|M^{-1}N\|_A < 1$, ce qui donne le résultat. Soit $x \in \mathbb{R}^n$ avec $|x|_A = 1$ tel que $\|M^{-1}N\|_A = |M^{-1}Nx|_A$. Puisque $N = M - A$, en posant $y := M^{-1}Ax$ (ou $Ax = My$), on obtient

$$\begin{aligned}
|M^{-1}Nx|_A^2 &= (AM^{-1}Nx, M^{-1}Nx)_2 = (AM^{-1}(M-A)x, M^{-1}(M-A)x)_2 \\
&= (Ax - AM^{-1}Ax, x - M^{-1}Ax)_2 \\
&= (Ax, x)_2 - (Ax, M^{-1}Ax)_2 - (AM^{-1}Ax, x)_2 + (AM^{-1}Ax, M^{-1}Ax)_2 \\
&= |x|_A^2 - (Ax, M^{-1}Ax)_2 - (M^{-1}Ax, Ax)_2 + (M^{-1}Ax, AM^{-1}Ax)_2 \\
&= |x|_A^2 - (My, y)_2 - (y, My)_2 + (y, Ay)_2 = 1 - (My, y)_2 - (y, Ny)_2 \\
&= 1 - (y, (M^* + N)y)_2.
\end{aligned}$$

Lorsque $M^* + N$ est définie positive on a $(y, (M^* + N)y)_2 > 0$ pour tout $y \neq 0$, d'où $|M^{-1}Nx|_A^2 < 1$. De plus, $y \neq 0$ puisque $y = M^{-1}Ax$ avec $x \neq 0$ et $M^{-1}A$ inversible. □

7.3 Application aux méthodes de Jacobi et Gauss-Seidel

Soit $A := (a_{ij})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{R})$ inversible. On note $D \in \mathcal{M}_n(\mathbb{R})$ sa diagonale, *i.e.* $D = (a_{ii}\delta_{ij})_{1\leq i,j\leq n}$, et $-E$ (resp. $-F$) sa partie triangulaire inférieure (resp. supérieure) stricte, de sorte que $A = D - E - F$. On rappelle ci-dessous les définitions des méthodes de Jacobi et Gauss-Seidel vues dans la première section.

Définition 7.7. Supposons D inversible.

1. La **méthode de Jacobi** est la méthode itérative associée à la décomposition régulière $A = M - N$ avec $M := D$ et $N := D - A$. Sa matrice d'itération est notée $\mathcal{J}$:
$$\mathcal{J}\ :=\ M^{-1}N = I_n - D^{-1}A.$$
2. La **méthode de Gauss-Seidel** est la méthode itérative associée à la décomposition régulière $A = M - N$ avec $M := D - E$ et $N := F$. Sa matrice d'itération est notée $\mathcal{L}_{GS}$:
$$\mathcal{L}_{GS}\ :=\ M^{-1}N = I_n - (D - E)^{-1}A.$$

Proposition 7.8. *On suppose A à diagonale strictement dominante, i.e.*

$$\forall\ i = 1, \ldots, n, \quad |a_{ii}| > \sum_{j\neq i} |a_{ij}|.$$

Alors, A et D sont inversibles et les méthode de Jacobi et Gauss-Seidel convergent.

Démonstration. Le fait que A est inversible a été vu en Td. La matrice D étant diagonale, elle est inversible si et seulement si $a_{ii} \neq 0$, pour tout $i \in \{1, \dots, n\}$. Or on a

$$\forall\ i = 1, \dots, n, \quad |a_{ii}| > \sum_{j \neq i} |a_{ij}| \geq 0.$$

Donc D est inversible.

1. Pour la convergence de la méthode de Jacobi, on va montrer que $\|\mathcal{J}\|_\infty < 1$, ce qui donne le résultat. On note $\mathcal{J} := (b_{ij})_{1 \leq i,j \leq n}$. Pour tous $i, j \in \{1, \dots, n\}$, on a $b_{ij} = \delta_{ij} - \dfrac{a_{ij}}{a_{ii}}$ donc, pour tout $i \in \{1, \dots, n\}$, on obtient

$$\sum_{j=1}^{n} |b_{ij}| = \sum_{j=1}^{n} \left| \delta_{ij} - \frac{a_{ij}}{a_{ii}} \right| = \sum_{j \neq i} \left| \frac{a_{ij}}{a_{ii}} \right| = |a_{ii}|^{-1} \sum_{j \neq i} |a_{ij}| < 1,$$

car A est à diagonale strictement dominante. On en déduit

$$\|\mathcal{J}\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |b_{ij}| < 1,$$

d'où le résultat.

2. Pour la méthode de Gauss-Seidel, on montre aussi que $\|\mathcal{L}_{GS}\|_\infty < 1$. Soit $x \in \mathcal{M}_{n,1}(\mathbb{R}) \setminus \{0\}$ et $y := \mathcal{L}_{GS} x$. Alors, $y = M^{-1} N x$ donc $My = Nx$, d'où $(D - E)y = Fx$, ce qui donne

$$\forall\ i = 1, \dots, n, \quad \sum_{j \leq i} a_{ij} y_j = -\sum_{j > i} a_{ij} x_j.$$

On en déduit

$$\forall\ i = 1, \dots, n, \quad a_{ii} y_i = \sum_{j \leq i} a_{ij} y_j - \sum_{j < i} a_{ij} y_j = -\left(\sum_{j > i} a_{ij} x_j + \sum_{j < i} a_{ij} y_j \right).$$

Soit $i_0 \in \{1, \dots, n\}$ tel que $\|y\|_\infty = |y_{i_0}|$. Alors, on a

$$|a_{i_0 i_0}|\, |y_{i_0}| \leq \sum_{j > i_0} |a_{i_0 j}|\, |x_j| + \sum_{j < i_0} |a_{i_0 j}|\, |y_j| \leq \|x\|_\infty \sum_{j > i_0} |a_{i_0 j}| + \|y\|_\infty \sum_{j < i_0} |a_{i_0 j}|$$

d'où

$$\left(|a_{i_0 i_0}| - \sum_{j < i_0} |a_{i_0 j}| \right) \|y\|_\infty \leq \|x\|_\infty \sum_{j > i_0} |a_{i_0 j}|.$$

Puisque A est à diagonale strictement dominante, on a

$$c \ := \ \sum_{j > i_0} |a_{i_0 j}| \left(|a_{i_0 i_0}| - \sum_{j < i_0} |a_{i_0 j}| \right)^{-1} < 1.$$

Ainsi, on obtient

$$\|\mathcal{L}_{GS} x\|_\infty = \|y\|_\infty \leq c \|x\|_\infty < \|x\|_\infty.$$

On en déduit $\|\mathcal{L}_{GS}\|_\infty < 1$. □

Proposition 7.9. *On suppose A symétrique définie positive. Alors, la méthode de Gauss-Seidel converge.*

Démonstration. Il suffit de montrer que $M^* + N$ est définie positive. Puisque A est symétrique, on a $E^* = F$. De plus, A étant définie positive on a

$$\forall\ i = 1, \dots, n, \quad a_{ii} = (Ae_i, e_i)_2 > 0,$$

où $\{e_1, \dots, e_n\}$ est la base canonique de $\mathbb{R}^n$. Ainsi, D est définie positive (donc inversible). on obtient

$$M^* + N = (D - E)^* + F = D^* - E^* + F = D,$$

d'où $M^* + N$ est définie positive. □

7.4 Implémentation

Dans toute la suite, on considère $A \in \mathcal{M}_n(\mathbb{R})$ inversible de décomposition régulière $A := M - N$ et on note $(x_k)_{k\in\mathbb{N}}$ la suite définie par la méthode itérative associée à cette décomposition. Pour la programmation des méthodes itératives, il est nécessaire de définir un critère d'arrêt, *i.e.* stopper le calcul de l'approximation x_k lorsque celle-ci est suffisamment correcte.

Le choix immédiat serait évidemment d'arrêter le calcul lorsque $\|x - x_k\|_2 < \varepsilon$, où $\varepsilon > 0$ est un petit paramètre fixé. La solution x étant l'inconnue à déterminer cela n'est pas possible. Or puisque x doit vérifier $Ax = b$, donc $\|Ax - b\|_2 = 0$, on peut considérer le critère d'arrêt $\|Ax_k - b\|_2 < \varepsilon$.

Définition 7.10. On appelle **résidu** d'ordre k, $k \in \mathbb{N}$, le vecteur $r_k \in \mathcal{M}_{n,1}(\mathbb{R})$ donné par

$$\forall\ k = 1, \dots, n, \quad r_k := b - Ax_k.$$

Soit $k \in \mathbb{N}$. La suite $(x_k)_{k\in\mathbb{N}}$ vérifie

$$Mx_{k+1} = Nx_k + b = Nx_k + r_k + Ax_k = Mx_k + r_k,$$

d'où

$$x_{k+1} = x_k + M^{-1}r_k.$$

C'est cette dernière égalité qui sera utilisée pour la programmation de la méthode itérative. Pour éviter le calcul de M^{-1}, on pose $y := M^{-1}r_k$ de sorte que y est solution de $My = r_k$. Ainsi, il est nécessaire que la matrice M soit telle que la résolution de ce système soit simple. On obtient alors l'Algorithme 8.

Dans la pratique, on ajoute une condition d'arrêt sur le nombre d'itérations pour éviter un trop grand nombre de calculs (voir les deux algorithmes suivant). L'Algorithme 8 appliqué aux cas des méthodes de Jacobi et Gauss-Seidel (avec itération maximale) donne les algorithmes 9 et 10.

On termine ce chapitre sur la question de la complexité de ces méthodes.

- Le calcul de $\|r\|_2$ nécessite n opérations.

Algorithme 8 Algorithme général des méthodes itératives

Entrées : A, b, x_0, ε
Sorties : x
  $x = x_0$
  $r = b - Ax$
  Tant que $\|r\|_2 \geq \varepsilon$ **faire**
    Calculer y solution de $My = r$
    $x = x + y$
    $r = b - Ax$
  fin Tant que

Algorithme 9 Algorithme de la méthode de Jacobi

Entrées : $A, b, x_0, \varepsilon, k_{\max}$
Sorties : x
  $k = 1$
  $x = x_0$
  $r = b - Ax$
  Tant que $(\|r\|_2 \geq \varepsilon)$ & $(k < k_{\max})$ **faire**
    Pour $i = 1 : n$ **faire**
      $x_i = x_i + \frac{r_i}{a_{ii}}$
    fin Pour
    $r = b - Ax$
    $k = k + 1$
  fin Tant que

- Le calcul de y nécessite
  — n opérations pour la méthode de Jacobi,
  — $\frac{n^2}{2}$ opérations pour la méthode de Gauss-Seidel.
- Le calcul de Ax demande n^2 opérations.

On obtient ainsi une complexité maximale de l'ordre de $\frac{3n^2}{2}$ par itération. Ainsi, si le nombre d'itérations est petit devant n les méthodes itératives sont moins couteuses que les méthodes directes.

Algorithme 10 Algorithme de la méthode de Gauss-Seidel

Entrées : $A, b, x_0, \varepsilon, k_{\max}$
Sorties : x
  $k = 1$
  $x = x_0$
  $r = b - Ax$
  $T = tril(A)$
  Tant que $(\|r\|_2 \geq \varepsilon)$ & $(k < k_{\max})$ **faire**
    $y = Descente(T, r)$
    $x = x + y$
    $r = b - Ax$
    $k = k + 1$
  fin Tant que

Chapitre 8

Introduction à l'optimisation et algorithme du gradient

Les problèmes d'optimisation sont les problèmes de la forme suivante :

$$\begin{cases} \text{Trouver } x \in F \text{ tel que} \\ J(x) = \inf\limits_{y \in F} J(y), \end{cases} \tag{8.0.1}$$

où E est un espace vectoriel, F un sous-espace vectoriel de E et $J : E \to \mathbb{R}$ est une fonction donnée.

Lorsque $F \neq E$, on parle d'**optimisation sous contraintes** (le fait que x_0 soit dans F entraîne que x_0 vérifie des contraintes particulières). Lorsque $E = F$, on parle d'**optimisation sans contrainte**.

Pour une matrice $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive et $b \in \mathcal{M}_{n,1}(\mathbb{R})$, on montrera que le problème

$$\begin{cases} \text{Trouver } x \in \mathcal{M}_{n,1}(\mathbb{R}) \text{ tel que} \\ Ax = b, \end{cases} \tag{8.0.2}$$

est équivalent au problème sans contrainte (8.0.1). avec $E = F := \mathcal{M}_{n,1}(\mathbb{R})$ et $J : \mathcal{M}_{n,1}(\mathbb{R}) \to \mathbb{R}$ définie par

$$J(x) := \frac{1}{2} Ax \cdot x - b \cdot x.$$

Ainsi, les méthodes de résolution numérique de (8.0.1). donnent des méthodes de résolution de (8.0.2).. C'est le cas de la méthode du gradient.

8.1 Introduction à l'optimisation

Dans toute cette section, on cherche à montrer l'existence et l'unicité du problème d'optimisation sans contrainte

$$\begin{cases} \text{Trouver } x \in \mathbb{R}^n \text{ tel que} \\ J(x) = \min\limits_{y \in \mathbb{R}^n} J(y), \end{cases} \tag{8.1.1}$$

où $J : \mathbb{R}^n \to \mathbb{R}$ est continue.

Définition 8.1. On dit qu'une fonction $J : \mathbb{R}^n \to \mathbb{R}$ est **strictement convexe** si elle vérifie :

$$\forall\ x, y \in \mathbb{R}^n,\ x \neq y,\ \forall\ \theta \in]0,1[\quad J(\theta x + (1-\theta)y) < \theta J(x) + (1-\theta)J(y).$$

Théorème 8.2. *Soit $J \in C(\mathbb{R}^n;\mathbb{R})$ telle que*

$$\lim_{\|x\|_2 \to +\infty} J(x) = +\infty. \tag{8.1.2}$$

1. *Alors, il existe $x \in \mathbb{R}^n$ solution de* (8.1.1)..
2. *De plus, si J est strictement convexe alors la solution est unique.*

Démonstration.
1. D'après (8.1.2)., pour tout $M > 0$, il existe $\eta > 0$ tel que si $\|x\|_2 > \eta$ alors $J(x) \geq M$. On pose $M := J(0)$ donc il existe $\eta > 0$ tel que $J(x) \geq J(0)$ pour tout $x \in \mathbb{R}^n \setminus B_\eta$, où $B_\eta := \{x \in \mathbb{R}^n \mid \|x\|_2 \leq \eta\}$. Puisque $0 \in B_\eta$, on en déduit

$$\inf_{y\in\mathbb{R}^n} J(y) = \min\{\inf_{y\in B_\eta} J(y); J(0)\} = \inf_{y\in B_\eta} J(y).$$

Or puisque J est continue et que B_η est compact, il existe $x \in B_\eta$ tel que $J(x) = \inf_{y\in B_\eta} J(y)$.
2. Soient $x, \overline{x} \in \mathbb{R}^n$ deux solutions de (8.1.1). telles que $x \neq \overline{x}$. Puisque J strictement convexe, on a

$$J(\theta x + (1-\theta)\overline{x}) < \theta J(x) + (1-\theta)J(\overline{x}),$$

pour tout $\theta \in]0,1[$. En particulier,on en prenant $\theta = 1/2$, on obtient

$$J((x+\overline{x})/2) < \min_{y\in\mathbb{R}^n} J(y),$$

ce qui est absurde. □

Remarque 8.3. Le résultat reste vrai en remplaçant $\mathbb{R}^n$ par un espace vectoriel de dimension finie mais devient faux en dimension infinie. Cela est dû au théorème de Riesz qui affirme que, dans un espace vectoriel normé, les boules fermées sont compactes si et seulement si l'espace est de dimension finie.

Définition 8.4. Soit $J \in C^1(\mathbb{R}^n;\mathbb{R})$. On appelle **gradient** de J, noté ∇J, le vecteur des dérivées partielles de J, *i.e.*

$$\nabla J := \left(\frac{\partial J}{\partial x_1}, \ldots, \frac{\partial J}{\partial x_n}\right)^T.$$

Lemme 8.5. *Soit $J \in C^1(\mathbb{R}^n;\mathbb{R})$. Alors, J est strictement convexe si et seulement si*

$$\forall\ x, y \in \mathbb{R}^n,\ x \neq y, \quad J(y) > J(x) + \nabla J(x)\cdot(y-x).$$

Démonstration. Supposons J strictement convexe. Soient $x, y \in \mathbb{R}^n$, $x \neq y$ et $\theta \in]0,1[$. On pose $z := \frac{1}{2}(y-x)$. On a

$$J(\theta(x+z)+(1-\theta)x) \leq \theta J(x+z) + (1-\theta)J(x),$$

(il n'y a pas d'inégalité stricte car on se sait pas si $x \neq x+z$) donc

$$J(x+\theta z) - J(x) \leq \theta(J(x+z) - J(x)).$$

Or puisque J est de classe C^1, pour tout θ suffisamment petit, on a

$$J(x+\theta(z)) = J(x) + \theta \nabla J(x) \cdot z + \theta\varepsilon(\theta),$$

où $\varepsilon(\theta) \underset{\theta\to 0}{\longrightarrow} 0$. Donc

$$\theta\varepsilon(\theta) + \theta\nabla J(x)\cdot z \leq \theta(J(x+z) - J(x)),$$

d'où

$$\varepsilon(\theta) + \nabla J(x)\cdot z \leq J(x+z) - J(x).$$

En passant à la limite quand θ tend vers 0, on obtient

$$\nabla J(x)\cdot z \leq J(x+z) - J(x).$$

Puisque J est strictement convexe et $x \neq y$, on a

$$J(x+z) = J\left(\frac{x+y}{2}\right) < \frac{1}{2}(J(x) + J(y),$$

d'où

$$\frac{1}{2}\nabla J(x)\cdot(y-x) < \frac{1}{2}(J(y) - J(x)),$$

ce qui donne le résultat.

Réciproquement, supposons

$$\forall\ x, y \in \mathbb{R}^n,\ x \neq y, \quad J(y) > J(x) + \nabla J(x)\cdot(x-y).$$

Soient $x, y, z \in \mathbb{R}^n$ non égaux et $\theta \in]0,1[$. On a

$$J(y) > J(z) + \nabla J(z)\cdot(y-z),$$

et

$$J(x) > J(z) + \nabla J(z)\cdot(x-z).$$

Donc

$$(1-\theta)J(y) + \theta J(x) > J(z) + \nabla J(z)\cdot\left((1-\theta)(y-z) + \theta(x-z)\right).$$

On prend $z := \theta x + (1-\theta)y$. Alors, on obtient

$$\begin{aligned}(1-\theta)J(y) + \theta J(x) &> J(\theta x + (1-\theta)y) \\ &\quad + \nabla J(\theta x + (1-\theta)y)\cdot((1-\theta)\theta(y-x) + \theta(1-\theta)(x-y)) \\ &= J(\theta x + (1-\theta)y),\end{aligned}$$

ce qui donne le résultat. □

Théorème 8.6. *Soit $J \in C^1(\mathbb{R}^n;\mathbb{R})$ strictement convexe vérifiant* (8.1.2).. *Alors, x est l'unique solution de* (8.1.1). *si et seulement si $\nabla J(x) = 0$.*

Remarque 8.7. L'implication est vraie même si J n'est pas convexe tandis que la réciproque nécessite la convexité. La stricte convexité et l'hypothèse (8.1.2). ne servent qu'à obtenir l'existence et unicité de x.

Démonstration. Supposons que x est solution de (8.1.1).. Soit $y \in \mathbb{R}^n$. Pour tout t suffisamment petit, on a

$$J(x+ty) - J(x) = t\nabla J(x)\cdot y + t\varepsilon(t),$$

où $\varepsilon(t) \underset{t\to 0}{\longrightarrow} 0$. Alors, on obtient

$$t\nabla J(x)\cdot y + t\varepsilon(t) \geq 0.$$

Pour $t > 0$, en divisant par t et en passant à la limite on obtient $\nabla J(x)\cdot y \geq 0$. De même, en prenant $t < 0$, on obtient $\nabla J(x)\cdot y \leq 0$. Donc $\nabla J(x)\cdot y = 0$ pour tout $y \in \mathbb{R}^n$. Donc $\nabla J(x) = 0$.

Supposons que $\nabla J(x) = 0$. Soit $y \in \mathbb{R}^n$. Puisque J est convexe, d'après le Lemme 8.5., on a

$$J(y) \geq J(x) + \nabla J(x)\cdot(y-x) = J(x),$$

ce qui donne le résultat. □

Définition 8.8. Soit $J \in C^2(\mathbb{R}^n;\mathbb{R})$. On appelle matrice **hessienne** de J, l'application $\nabla^2 J$ de $\mathbb{R}^n$ dans $\mathcal{M}_n(\mathbb{R})$ donnée par

$$\nabla^2 J \;:=\; \left(\frac{\partial^2 J}{\partial x_i \partial x_j}\right)_{1\leq i,j\leq n}.$$

D'après le théorème de Schwarz, la matrice hessienne est symétrique.

Théorème 8.9. *Soit $J \in C^2(\mathbb{R}^n;\mathbb{R})$. Si, pour tout $x \in \mathbb{R}^n$, $\nabla^2 J(x)$ est définie positive, alors J est strictement convexe.*

Démonstration. Soient $x, y \in \mathbb{R}^n$ avec $x \neq y$. Pour tout $\theta \in]0,1[$, on pose

$$\varphi(\theta) \;:=\; J(x+\theta(y-x)).$$

Alors, $\varphi \in C^2(\mathbb{R};\mathbb{R})$. On a

$$J(y) - J(x) = \varphi(1) - \varphi(0) = \int_0^1 \varphi'(\theta)\, d\theta.$$

Par intégration par parties, on obtient

$$J(y) - J(x) = [\varphi'(\theta)(\theta-1)]_0^1 - \int_0^1 \varphi''(\theta)(\theta-1)\, d\theta = \varphi'(0) + \int_0^1 \varphi''(\theta)(1-\theta)\, d\theta.$$

Or on a

$$\varphi'(\theta) = \nabla J(x+\theta(y-x))\cdot(y-x)$$

et

$$\varphi''(\theta) = \nabla^2 J(x+\theta(y-x))(y-x)\cdot(y-x) > 0,$$

car $x \neq y$ et $\nabla^2 J(x+\theta(y-x))$ est définie positive. Donc

$$J(y) - J(x) > \nabla J(x)\cdot(y-x),$$

d'où le résultat d'après le Lemme 8.5.. □

Proposition 8.10. *Soient $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive et $b \in \mathcal{M}_{n,1}(\mathbb{R})$. On pose*

$$J(x) := \frac{1}{2}Ax\cdot x - b\cdot x, \quad \forall\, x \in \mathcal{M}_{n,1}(\mathbb{R}). \tag{8.1.3}$$

Alors, il existe une unique solution x_0 au problème de minimisation (8.1.1). *et x_0 est l'unique solution de $Ax = b$.*

Remarque 8.11. On dit que la fonction J définie dans la Proposition précédente est une **fonctionnelle quadratique**.

Démonstration. Il est immédiat que J est un polynôme de degré 2 donc, en particulier, J est de classe C^2. De plus, puisque A est définie positive ses valeurs propres $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ sont strictement positives et il existe $U \in \mathcal{M}_n(\mathbb{R})$ unitaire telle que

$$A = U^* D U \quad \text{avec} \quad D := \operatorname{diag}(\lambda_1, \ldots, \lambda_n).$$

Ainsi, on a

$$Ax\cdot x = DUx\cdot Ux = \sum_{i=1}^n \lambda_i |(Ux)_i|^2 \geq \lambda_1 \|Ux\|_2^2 = \lambda_1 \|x\|_2^2.$$

Enfin, d'après l'inégalité de Cauchy-Schwarz, on a $b\cdot x \leq \|b\|_2\|x\|_2$. Donc obtient

$$J(x) \geq \frac{\lambda_1}{2}\|x\|_2^2 - \|b\|_2\|x\|_2 \underset{\|x\|_2\to+\infty}{\longrightarrow} +\infty.$$

On en déduit que le problème de minimisation (8.1.1). admet une solution x_0. Pour déterminer ∇J, on réécrit J sous la forme

$$J(x) = \frac{1}{2}\sum_{i,j=1}^n a_{ij}x_i x_j - \sum_{i=1}^n b_i x_i, \quad \forall\, x \in \mathbb{R}^n.$$

Alors, pour tout $k \in \mathbb{N}$ et tout $x \in \mathbb{R}^n$, on a

$$\begin{aligned}\frac{\partial J}{\partial x_k}(x) &= \frac{1}{2}\sum_{i,j=1}^n a_{ij}\delta_{ik}x_j + \frac{1}{2}\sum_{i,j=1}^n a_{ij}x_i\delta_{kj} - \sum_{i=1}^n b_i\delta_{ik} \\ &= \frac{1}{2}\sum_{j=1}^n a_{kj}x_j + \frac{1}{2}\sum_{i=1}^n a_{ik}x_i - b_k = \frac{1}{2}\Big((Ax)_k + (A^Tx)_k\Big) - b_k,\end{aligned}$$

d'où $\nabla J(x) = \frac{1}{2}(A + A^T)x - b = Ax - b$ car A est symétrique. De même, on obtient $\nabla^2 J(x) = A$ pour tout $x \in \mathbb{R}^n$. On en déduit que J est strictement convexe. Donc il existe une unique solution x_0 de (8.1.1).. De plus, $Ax_0 - b = \nabla J(x_0) = 0$, donc x_0 est la solution de $Ax = b$. □

8.2 Algorithme du gradient

Soit $J \in C(\mathbb{R}^n;\mathbb{R})$ une fonctionnelle à minimiser. Une méthode intuitivement simple de recherche du minimum consiste à choisir $x_0 \in \mathbb{R}^n$ puis à construire les itérés x_{k+1}, $k \in \mathbb{N}$, en posant

$$x_{k+1} = x_k + \alpha_k\, d_k,$$

où α_k et d_k vérifient $J(x_k+\alpha_k\, d_k) \leq J(x_k)$. On appelle ce type de méthode des **méthodes de descente**. Les algorithmes du gradient sont des méthodes de descentes particulières où la direction de la descente est celle du gradient.

Définition 8.12. Soit $J \in C(\mathbb{R}^n;\mathbb{R})$.

1. On dit que $d \in \mathbb{R}^n\setminus\{0\}$ est une **direction de descente** en $x \in \mathbb{R}^n$ s'il existe $\alpha_0 > 0$ tel que
$$\forall\ \alpha \in [0,\alpha_0], \quad J(x+\alpha\, d) \leq J(x). \tag{8.2.1}$$
2. Lorsque l'inégalité (8.2.1). est stricte, on parle de **direction de descente stricte**.

Proposition 8.13. *Soient $J \in C^1(\mathbb{R}^n;\mathbb{R})$ et $x \in \mathbb{R}^n$.*

1. *Si $d \in \mathbb{R}^n \setminus \{0\}$ est une direction de descente en x, alors $d \cdot \nabla J(x) \leq 0$.*
2. *Si $\nabla J(x) \neq 0$, alors $-\nabla J(x)$ est une direction de descente stricte en x.*

Démonstration.
1. Il existe $\alpha_0 > 0$ tel que

$$\forall\ \alpha \in [0,\alpha_0], \quad J(x+\alpha\, d) \leq J(x).$$

On pose

$$\forall\ \alpha \in [0,\alpha_0], \quad \varphi(\alpha)\ :=\ J(x+\alpha\, d). \tag{8.2.2}$$

Alors, $\varphi \in C^1(\mathbb{R};\mathbb{R})$ et $\varphi'(\alpha) = \nabla J(x+\alpha\, d)\cdot d$. De plus, on a

$$\forall\ \alpha \in [0,\alpha_0], \quad \varphi(\alpha) \leq \varphi(0),$$

donc

$$\forall\ \alpha \in [0,\alpha_0], \quad \frac{\varphi(\alpha)-\varphi(0)}{\alpha} \leq 0,$$

d'où $\varphi'(0) \leq 0$, ce qui donne le résultat.
2. On pose $d := -\nabla J(x) \neq 0$. Soit φ définie par (8.2.2).. On a

$$\varphi'(0) = \nabla J(x)\cdot d = -\|d\|_2^2 < 0.$$

Puisque φ' est continue et $\varphi'(0) < 0$, il existe $\alpha_0 > 0$ tel que, pour tout $\alpha \in [0,\alpha_0]$, $\varphi'(\alpha) < 0$. Ainsi, pour tout $\alpha \in [0,\alpha_0]$, on obtient

$$\varphi(\alpha) - \varphi(0) = \int_0^\alpha \varphi'(s)\ ds < 0,$$

donc $J(x+\alpha d) < J(x)$. □

Algorithme 11 Algorithme du gradient à pas fixe

Entrées : A, b, ε
Sorties : x
Choisir $0 < \alpha < \frac{2}{\lambda_n}$.
Prendre $x \in \mathcal{M}_{n,1}(\mathbb{R})$.
Calculer $r = b - Ax$.
Tant que $\|r\|_2 \geq \varepsilon$ **faire**
$x = x + \alpha\, r$
$r = b - Ax$
fin Tant que

Le résultat précédent nous amène à prendre comme direction de descente $-\nabla J(x_k)$ à chaque itération $k \in \mathbb{N}$, on obtient alors l'**algorithme du gradient** donné par :

$$\begin{cases} x_0 \in \mathbb{R}^n, \\ x_{k+1} = x_k - \alpha \nabla J(x_k), \quad \forall\ k \in \mathbb{N}, \end{cases}$$

où $\alpha > 0$ reste à déterminer. Il est alors possible d'étudier la convergence de la suite ainsi construite suivant la fonctionnelle J considérée. Dans la suite, on considère seulement le cas d'une fonctionnelle quadratique.

Soient $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive, $b \in \mathcal{M}_{n,1}(\mathbb{R})$ et J la fonctionnelle quadratique donnée par (8.1.3).. Alors, l'algorithme du gradient s'écrit

$$\begin{cases} x_0 \in \mathcal{M}_{n,1}(\mathbb{R}), \\ r_k = b - Ax_k, \\ x_{k+1} = x_k + \alpha\, r_k, \quad \forall\ k \in \mathbb{N}, \end{cases} \tag{8.2.3}$$

où $\alpha > 0$. On obtient alors l'algorithme 11.

Remarque 8.14. La suite $(x_k)_{k\in\mathbb{N}}$ peut encore s'écrire

$$\alpha^{-1} x_{k+1} = \alpha^{-1} x_k + b - Ax_k = (\alpha^{-1} I_n - A)x_k + b,$$

soit $Mx_{k+1} = Nx_k + b$, où $M := \alpha^{-1} I_n$ et $N := \alpha^{-1} I_n - A$. Autrement dit, l'algorithme (8.2.3). est la méthode itérative correspondant à la décomposition régulière $A = M - N$. Sous cette forme, il est clair qu'il n'est pas nécessaire de supposer que A soit symétrique définie positive pour définir la méthode itérative.

Définition 8.15. Soient $A \in \mathcal{M}_n(\mathbb{R})$ inversible et $b \in \mathcal{M}_{n,1}(\mathbb{R})$. Pour tout $\alpha > 0$, on appelle **méthode du gradient** (ou **méthode de Richardson**) la méthode itérative associée à la décomposition régulière

$$A = M - N, \quad \text{où} \quad M := \alpha^{-1} I_n \quad \text{et} \quad N := \alpha^{-1} I_n - A.$$

La matrice d'amplification est alors $\mathcal{L}_G := I_n - \alpha\, A$.

Proposition 8.16. *Soient $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive, $b \in \mathcal{M}_{n,1}(\mathbb{R})$ et $0 < \lambda_1 \leq \cdots \leq \lambda_n$ les valeur propres de A. Alors, la méthode du gradient converge si et seulement si $0 < \alpha < \frac{2}{\lambda_n}$. De plus, $\rho(\mathcal{L}_G)$ admet un minimum par rapport à α atteint en α_{opt} donné par*

$$\rho(\mathcal{L}_G)(\alpha_{\text{opt}}) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} \quad et \quad \alpha_{\text{opt}} = \frac{2}{\lambda_1 + \lambda_n}. \tag{8.2.4}$$

Remarque 8.17. Le résultat de convergence reste vrai même si A n'est pas symétrique définie positive mais diagonalisable (voir [1]).

Démonstration. Il suffit de calculer $\rho(\mathcal{L}_G)$ et de l'étudier comme fonction de α. On a

$$\rho(\mathcal{L}_G) = \rho(I_n - \alpha\, A) = \max(|1 - \alpha\lambda_i|;\ i = 1, \ldots, n).$$

Alors $\rho(\mathcal{L}_G) < 1$ si et seulement si, pour tout $i = 1, \ldots, n$, $-1 < 1 - \alpha\lambda_i < 1$ donc $0 < \alpha\lambda_i < 2$, ce qui donne le premier résultat. La fonction $\lambda \mapsto |1 - \alpha\lambda|$, pour $\alpha > 0$ fixé, est décroissante sur $]-\infty, 1/a]$ et croissante sur $[1/a, +\infty[$ donc sa valeur maximale est atteinte soit pour la plus petite valeur de λ, soit pour la plus grande valeur de λ, d'où

$$\rho(\mathcal{L}_G) = \max\{|1 - \alpha\lambda_1|, |1 - \alpha\lambda_n|\}$$

On cherche à minimiser $\rho(\mathcal{L}_G)$ suivant α, d'après la figure 8.1. ce minimum est atteint lorsque $\alpha\lambda_n - 1 = 1 - \alpha\lambda_1$, *i.e.* $\alpha = \frac{2}{\lambda_n + \lambda_1}$. Pour cette valeur de α, on obtient alors $\rho(\mathcal{L}_G) = 1 - \alpha\lambda_1 = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}$. □

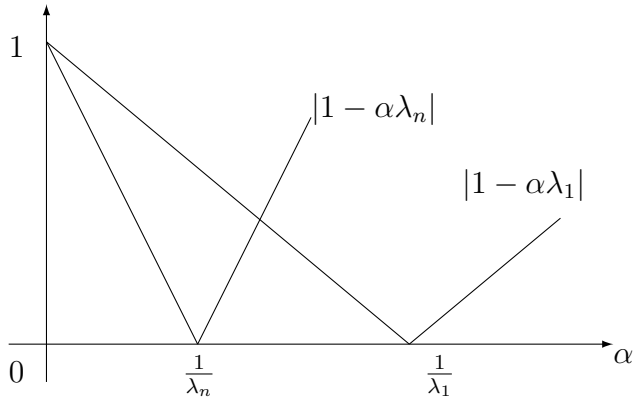

FIGURE 8.1 – Graphes de $|1 - \alpha\lambda_1|$ et $|1 - \alpha\lambda_n|$

Corollaire 8.18. *Soient $A \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive et $b \in \mathcal{M}_{n,1}(\mathbb{R})$. En prenant $\alpha := \alpha_{\text{opt}}$ défini par (8.2.4)., l'erreur dans la méthode du gradient vérifie*

$$\|e_k\|_2 \leq \left(\frac{\kappa_2(A) - 1}{\kappa_2(A) + 1}\right)^k \|e_0\|_2, \quad \forall\ k \in \mathbb{N}. \tag{8.2.5}$$

Autrement dit, l'erreur tend vers 0 d'autant plus vite que la matrice A est bien conditionnée (i.e. proche de 1).

Algorithme 12 Algorithme du gradient à pas optimal

Entrées : A, b, ε
Sorties : x
Prendre $x \in \mathcal{M}_{n,1}(\mathbb{R})$.
Calculer $r = b - Ax$.
Calculer $\alpha = \dfrac{\|r\|_2^2}{Ar \cdot r}$.
Tant que $\|r\|_2 \geq \varepsilon$ **faire**
$x = x + \alpha\, r$
$r = b - Ax$
$\alpha = \dfrac{\|r\|_2^2}{Ar \cdot r}$
fin Tant que

Démonstration. D'après (7.2.4)., on a

$$\|e_k\|_2 \leq \rho(\mathcal{L}_G)^k \|e_0\|_2, \quad \forall\, k \in \mathbb{N}.$$

Alors d'après (8.2.4). et la Proposition 4.8., on obtient (8.2.5).. □

On peut améliorer cet algorithme en choisissant, à chaque itération k, un pas optimal α_k, *i.e.* qui minimise $J(x_k - \alpha \nabla J(x_k))$. On cherche donc à déterminer α_0 tel que

$$J\left(x - \alpha_0 \nabla J(x)\right) = \min\left\{ J\left(x - \alpha \nabla J(x)\right) \mid 0 < \alpha < \frac{2}{\lambda_n} \right\}.$$

Pour tout $\alpha > 0$, on pose $\varphi(\alpha) := J\left(x - \alpha \nabla J(x)\right)$. Alors, $\varphi \in C^1(\mathbb{R}; \mathbb{R})$ et vérifie (8.1.2)., donc α_0 existe et est tel que $\varphi'(\alpha_0) = 0$. Or on a

$$\begin{aligned}
\varphi'(\alpha) &= -\nabla J\left(x - \alpha \nabla J(x)\right) \cdot \nabla J(x) \\
&= -\left(A\left(x - \alpha \nabla J(x)\right) - b\right) \cdot (Ax - b) \\
&= -\|Ax - b\|_2^2 + \alpha\, A \nabla J(x) \cdot (Ax - b) \\
&= -\|Ax - b\|_2^2 + \alpha\, A(Ax - b) \cdot (Ax - b),
\end{aligned}$$

d'où

$$\alpha_0 = \frac{\|Ax - b\|_2^2}{A(Ax - b) \cdot (Ax - b)}.$$

Remarque 8.19.

1. Le choix d'un pas optimal à chaque itération augmente le nombre de calculs à effectuer et, en général, n'améliore pas ou peu la convergence. Ainsi, en pratique, on préférera généralement la méthode à pas constant.
2. Le principal défaut de l'algorithme du gradient réside dans sa vitesse de convergence dite linéaire. Néanmoins, la convergence de l'algorithme s'améliore avec le conditionnement de A.

3. Dans l'Algorithme 11, on a $x_{k+1} = x_k + \alpha\, r_k$ avec $r_k = b - Ax_k$, pour tout $k \in \mathbb{N}$. On obtient donc
$$x_k - x_0 = \sum_{i=0}^{k-1}(x_{i+1} - x_i) = \sum_{i=0}^{k-1} \alpha r_k.$$
Or on a aussi $r_{k+1} - r_k = \alpha A r_k$ donc
$$r_k - r_0 = \sum_{i=0}^{k-1}(r_{i+1} - r_i) = \alpha \sum_{i=0}^{k-1} A r_i = \alpha \sum_{i=0}^{k-1} A^i r_0.$$
On en déduit que, pour tout $k \in \mathbb{N}$, $x_k - x_0 \in \mathcal{K}_k := \mathrm{Vect}\{r_0, Ar_0, \ldots, A^k r_0\}$. La plupart des méthodes itératives modernes consistent à construire une base particulière de ces espaces $\mathcal{K}_k$ appelés **espaces de Krylov** (méthode du gradient conjugué ou biconjugué, GMRES, etc.).

Chapitre 9

Algorithme du gradient conjugué

L'algorithme du gradient conjugué a pour objectif d'améliorer l'algorithme du gradient vu dans la section précédente. Dans tout ce chapitre, on note $A \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive, $b \in \mathcal{M}_{n,1}(\mathbb{R})$ et $x \in \mathcal{M}_{n,1}(\mathbb{R})$ la solution de

$$Ax = b. \tag{9.0.1}$$

9.1 Définition

L'algorithme du gradient étant une méthode de descente on considère de manière générale une méthode de descente

$$\begin{cases} x_0 \in \mathbb{R}^n, \\ x_{k+1} = x_k + \alpha_k\, p_k, \quad \forall\ k \in \mathbb{N}, \end{cases} \tag{9.1.1}$$

où $\alpha_k > 0$ et $p_k \in \mathcal{M}_{n,1}(\mathbb{R})$. De plus, on note r_k, $k \in \mathbb{N}$, le résidu associé, *i.e.*

$$r_k \ := \ b - Ax_k.$$

Comme dans le cas particulier de l'algorithme du gradient, on obtient que le pas optimal α_k à chaque itération $k \in \mathbb{N}$ est donné par

$$\alpha_k = \frac{r_k \cdot p_k}{Ap_k \cdot p_k}. \tag{9.1.2}$$

En conséquence immédiate, à chaque itération $k \in \mathbb{N}$, on obtient

$$r_{k+1} \cdot p_k = r_k \cdot p_k - \alpha_k\, Ap_k \cdot p_k = 0.$$

Afin d'améliorer la convergence de l'algorithme il parait naturel de prendre une direction p_k telle que le reste r_{k+1} soit orthogonal à toutes les directions p_j, $1 \leq j \leq k$, précédentes, *i.e.*

$$\forall\ j = 0, \ldots, k, \quad r_{k+1} \cdot p_j = 0. \tag{9.1.3}$$

Il reste donc à déterminer les directions p_k de sorte que (9.1.3). ait lieu. On a vu que le cas $j = k$ a lieu sans condition sur p_k. Pour $0 \leq j < k$, on a

$$0 = r_{k+1} \cdot p_j = r_k \cdot p_j - \alpha_j\, Ap_j \cdot p_k = -\alpha_j\, Ap_j \cdot p_k.$$

On en déduit que les directions p_k vérifient

$$\forall\ j = 1, \dots, k-1, \quad Ap_k \cdot p_j = 0.$$

Définition 9.1. Soit $k \in \mathbb{N}$. Une famille $\{p_1, \dots, p_k\}$ de $\mathcal{M}_{n,1}(\mathbb{R})$ est dite **A-conjuguée** si elle vérifie

$$\forall\ i, j \in \{1, \dots, k\},\ i \neq j, \quad Ap_i \cdot p_j = 0.$$

D'après ce qui précède, il est immédiat que les directions de descente p_k sont A-conjuguées si et seulement si (9.1.3). a lieu.

Remarque 9.2. Puisque A est symétrique définie positive, l'application $(\cdot, \cdot)_A$ définie par

$$\forall\ x, y \in \mathcal{M}_{n,1}(\mathbb{R}), \quad (x, y)_A\ :=\ Ax \cdot y,$$

est un produit scalaire sur $\mathcal{M}_{n,1}(\mathbb{R})$. En particulier, une famille de $\mathcal{M}_{n,1}(\mathbb{R})$ est A-conjuguée si elle est orthogonale pour le produit scalaire $(\cdot, \cdot)_A$. On en déduit que toute famille A-conjuguée est libre. En particulier, toute famille A-conjuguée $\{p_0, p_1, \dots, p_{n-1}\}$ de $\mathcal{M}_{n,1}(\mathbb{R})$ est une base de $\mathcal{M}_{n,1}(\mathbb{R})$.

Proposition 9.3. *Pour tout $k \in \mathbb{N}$, soit x_k calculé par la méthode de descente* (9.1.1). *et r_k le résidu associé. Si les directions de descente p_k sont A-conjuguées, alors*

1. $\forall\ k \in \mathbb{N}^*,\ \forall\ j \in \{0, \dots, k-1\}, \quad r_k \cdot p_j = 0,$
2. $x_n = x.$

Autrement dit, la méthode converge en n itérations.

Démonstration. Le premier point a déjà été montré. La famille $\{p_0, \dots, p_{n-1}\}$ est une base de $\mathcal{M}_{n,1}(\mathbb{R})$ donc $r_n = a_0 p_0 + \cdots + a_{n-1} p_{n-1}$, où $a_0, \dots, a_{n-1} \in \mathbb{R}$. On en déduit

$$\|r_n\|^2 = \sum_{k=0}^{n-1} a_k p_k \cdot r_n = 0,$$

d'après (9.1.3).. Donc $r_n = 0$, d'où $Ax_n = b$. □

Il reste à déterminer le choix des descentes p_k de sorte que celles-ci forment une famille A-conjuguée. Dans l'algorithme du gradient les descentes étaient prises égales à r_k, l'idée ici consiste, à partir de la famille $\{r_0, \dots, r_{n-1}\}$, à construire une famille de descentes A-conjuguée par orthogonalisation suivant le produit scalaire $(\cdot, \cdot)_A$. Pour cela on considère tout d'abord une famille quelconque $\{d_0, \dots, d_k\}$, $k \in \mathbb{N}$, de $\mathbb{R}^n$ et on pose

$$\forall\ i = 0, \dots, k, \quad p_i\ :=\ d_i - \sum_{j=0}^{i-1} c_{ij} p_j,$$

où les coefficients $c_{ij} \in \mathbb{R}$ sont à déterminer (avec la convention $\sum_{j=0}^{-1} = 0$). On a $f_0 = d_0$ et $f_1 = d_1 - c_{10} d_0$. Puisque l'on doit avoir $(f_1, f_0)_A = 0$, on obtient

$$c_{10} = \frac{(d_1, p_0)_A}{(d_0, p_0)_A} = \frac{Ad_1 \cdot p_0}{Ap_0 \cdot p_0}.$$

Supposons les coefficients c_{ij} calculés de sorte que la famille $\{p_0, \dots, p_l\}$ est A-conjuguée et déterminons les coefficients $c_{(l+1)j}$ de sorte que la famille $\{p_0, \dots, p_{l+1}\}$ est A-conjuguée. Soit $i \in \{0, \dots, l\}$. On a

$$(p_{l+1}, p_i)_A = (d_{l+1}, p_i)_A - \sum_{j=0}^{l} c_{(l+1)j}(p_i, p_j)_A.$$

Puisque la famille $\{p_0, \dots, p_l\}$ est A-conjuguée, on a

$$\sum_{j=0}^{l} c_{(l+1)j}(p_i, p_j)_A = c_{(l+1)i}(p_i, p_i)_A,$$

d'où $(p_{l+1}, p_i)_A = 0$ si et seulement si

$$c_{(l+1)i} = \frac{(d_{l+1}, p_i)_A}{(p_i, p_i)_A} = \frac{Ad_{l+1} \cdot p_i}{Ap_i \cdot p_i}.$$

On en déduit le résultat suivant :

Lemme 9.4. *Soient $k \in \mathbb{N}$ et $\{d_0, \dots, d_k\}$ une famille de $\mathbb{R}^n$. On pose $p_0 = d_0$ et*

$$\forall\ j = 0, \dots, k-1, \quad p_{j+1} \ := \ d_{j+1} - \sum_{i=0}^{j} \frac{Ad_{j+1} \cdot p_i}{Ap_i \cdot p_i}\ p_i.$$

Alors la famille $\{p_0, \dots, p_k\}$ est A-conjuguée et $\mathrm{Vect}\{d_0, \dots, d_k\} = \mathrm{Vect}\{p_0, \dots, p_k\}$.

Démonstration. Il reste à montrer que $\mathrm{Vect}\{d_0, \dots, d_k\} = \mathrm{Vect}\{p_0, \dots, p_k\}$. Pour tout $j \in \{0, \dots, k\}$, on a $p_j \in \mathrm{Vect}\{d_0, \dots, d_j\}$, d'où $\mathrm{Vect}\{p_0, \dots, p_j\} \subset \mathrm{Vect}\{d_0, \dots, d_j\}$. Puisque la famille $\{p_0, \dots, p_k\}$ est A-conjuguée, sa dimension est égale au nombre de vecteurs p_j, $j \in \{1, \dots, k\}$, non nuls. Supposons qu'il existe $j \in \{1, \dots, k\}$ tel que $p_j = 0$. Alors on a

$$d_j = \sum_{i=0}^{j-1} \frac{Ad_{j+1} \cdot p_i}{Ap_i \cdot p_i}\ p_i \in \mathrm{Vect}\{p_0, \dots, p_{j-1}\} \subset \mathrm{Vect}\{d_0, \dots, d_{j-1}\},$$

d'où $\mathrm{Vect}\{d_0, \dots, d_j\} = \mathrm{Vect}\{d_0, \dots, d_{j-1}\}$. On en déduit que la dimension de $\mathrm{Vect}\{d_0, \dots, d_j\}$ est inférieure ou égale à celle de $\mathrm{Vect}\{p_0, \dots, p_j\}$. Or $\mathrm{Vect}\{p_0, \dots, p_k\} \subset \mathrm{Vect}\{d_0, \dots, d_k\}$ donc $\mathrm{Vect}\{d_0, \dots, d_k\} = \mathrm{Vect}\{p_0, \dots, p_k\}$. □

Lemme 9.5. *Soit $x_0 \in \mathbb{R}^n$. On pose $r_0 = p_0 = b - Ax_0$ et, pour tout $k = 0, \dots, n$,*

$$\left\{ \begin{array}{ll} \alpha_k & = \frac{r_k \cdot p_k}{Ap_k \cdot p_k}, \\ x_{k+1} & = x_k + \alpha_k\, p_k, \\ r_{k+1} & = b - Ax_{k+1} = r_k - \alpha_k Ap_k, \\ & \\ p_{k+1} & = r_{k+1} - \displaystyle\sum_{i=0}^{k} \frac{Ar_{k+1} \cdot p_i}{Ap_i \cdot p_i}\ p_i. \end{array} \right. \tag{9.1.4}$$

Alors, la famille $\{p_0, \dots, p_n\}$ est A-conjuguée et on a

$$\mathrm{Vect}\{r_0, Ar_0, \dots, A^k r_0\} = \mathrm{Vect}\{r_0, r_1, \dots, r_k\} = \mathrm{Vect}\{p_0, p_1, \dots, p_k\}, \quad \forall\ k \in \{0, \dots, n\}.$$

Démonstration. D'après le Lemme 9.5. appliqué à la famille $\{r_0, r_1, \dots, r_k\}$, on a

$$\text{Vect}\{r_0, r_1, \dots, r_k\} = \text{Vect}\{p_0, p_1, \dots, p_k\},$$

et la famille $\{p_0, p_1, \dots, p_n\}$ est A-conjuguée. Pour la suite, on raisonne par récurrence. Pour $k = 1$, on a

$$r_1 = r_0 - \alpha_0 A p_0 = r_0 - \alpha_0 A r_0 \in \text{Vect}\{r_0, Ar_0\}.$$

Donc $\text{Vect}\{r_0, r_1\} \subset \text{Vect}\{r_0, Ar_0\}$. De plus, $r_0 \cdot r_1 = 0$ donc $\dim(\text{Vect}\{r_0, r_1\}) = 2 \geq \dim(\text{Vect}\{r_0, Ar_0\})$, d'où $\text{Vect}\{r_0, r_1\} = \text{Vect}\{r_0, Ar_0\}$.

Supposons $\text{Vect}\{r_0, r_1, \dots, r_k\} = \text{Vect}\{r_0, Ar_0, \dots, A^k r_0\}$. On a $r_{k+1} = r_k - \alpha_k A p_k$, or

$$p_k \in \text{Vect}\{p_0, p_1, \dots, p_k\} = \text{Vect}\{r_0, r_1, \dots, r_k\} = \text{Vect}\{r_0, Ar_0, \dots, A^k r_0\}$$

donc $Ap_k \in \text{Vect}\{Ar_0, A^2 r_0, \dots, A^{k+1} r_0\}$, d'où $r_{k+1} \in \text{Vect}\{r_0, Ar_0, \dots, A^{k+1} r_0\}$. On en déduit $\text{Vect}\{r_0, r_1, \dots, r_{k+1}\} \subset \text{Vect}\{r_0, Ar_0, \dots, A^{k+1} r_0\}$. De plus, on obtient qu'il existe $a_0, \dots, a_{k+1}$ tels que

$$r_{k+1} = a_{k+1} A^{k+1} r_0 + \sum_{i=0}^{k} a_i A^i r_0.$$

D'après la Proposition 9.3., on a $r_{k+1} \cdot p_i = 0$ pour tout $i = 0, \dots, k$. Donc $r_{k+1} \notin \text{Vect}\{p_0, p_1, \dots, p_k\}$, d'où $a_{k+1} \neq 0$. Alors, on obtient

$$A^{k+1} r_0 = \frac{1}{a_{k+1}} \left(r_{k+1} - \sum_{i=0}^{k} a_i A^i r_0 \right) ..$$

Or $\text{Vect}\{r_0, Ar_0, \dots, A^k r_0\} = \text{Vect}\{r_0, r_1, \dots, r_k\}$, d'où $A^{k+1} r_0 \in \text{Vect}\{r_0, r_1, \dots, r_k, r_{k+1}\}$. Ce qui donne le résultat □

Lemme 9.6. *Soient $x_0 \in \mathbb{R}^n$, $r_0 = p_0 = b - Ax_0$ et, pour tout $k \in \{1, \dots, n\}$, x_k, p_k, r_k définis par* (9.1.4).. *Alors, la famille $\{r_0, \dots, r_n\}$ est orthogonale et on a*

$$\forall\ k \in \{1, \dots, n\}, \quad p_{k+1} = r_{k+1} - \beta_k p_k, \quad \text{où} \quad \beta_k := \frac{Ap_k \cdot r_{k+1}}{Ap_k \cdot p_k}.$$

Démonstration. On raisonne par récurrence. Puisque $p_0 = r_0$, on a

$$r_1 \cdot r_0 = (r_0 - \alpha_0 A p_0) \cdot r_0 = p_0 \cdot p_0 - \alpha_0 A p_0 \cdot p_0 = 0,$$

d'après la définition de α_0. De plus,

$$p_1 = r_1 - \frac{Ar_1 \cdot p_0}{Ap_0 \cdot p_0}\ p_0.$$

Donc le résultat est vrai pour $n = 1$.

On suppose que, pour $k \in \{1, \dots, n\}$, on a $r_k \cdot r_j = 0$ pour tout $j \in \{0, \dots, k-1\}$ et $p_k = r_k - \beta_{k-1} p_{k-1}$. Soit $j \in \{0, \dots, k\}$. Alors,

$$r_{k+1} \cdot r_j = (r_k - \alpha_k\, Ap_k) \cdot r_j = r_k \cdot r_j - \alpha_k A p_k \cdot r_j.$$

Soit $j \leq k-1$. On a $r_j \in \text{Vect}\{r_0, r_1, \ldots, r_j\} = \text{Vect}\{p_0, p_1, \ldots, p_j\}$ et la famille $\{p_0, \ldots, p_n\}$ est A-conjuguée donc $Ap_k \cdot r_j = 0$. De plus, par hypothèse de récurrence, $r_k \cdot r_j = 0$. Donc $r_{k+1} \cdot r_j = 0$. Pour $j = k$, puisque $p_k = r_k - \beta_{k-1}p_{k-1}$, on a $r_k = p_k + \beta_{k-1}p_{k-1}$. Donc

$$r_{k+1} \cdot r_k = r_k \cdot r_k - \alpha_k Ap_k \cdot r_k = r_k \cdot r_k - \alpha_k Ap_k \cdot p_k - \alpha_k \beta_{k-1} Ap_k \cdot p_{k-1} = r_k \cdot r_k - \alpha_k Ap_k \cdot p_k = 0.$$

Il reste à montrer que $p_{k+1} = r_{k+1} - \beta_k p_k$. Pour tout $i = 0, \ldots, k-1$, $p_i = \alpha_i^{-1}(x_{i+1} - x_i)$ donc

$$Ar_{k+1} \cdot p_i = \alpha_i^{-1} r_{k+1} \cdot A(x_{i+1} - x_i) = \alpha_i^{-1} r_{k+1} \cdot (r_i - r_{i+1}) = -\alpha_i^{-1} r_{k+1} \cdot r_{i+1}.$$

Alors, $Ar_{k+1} \cdot p_i = 0$ si $i \leq k$ d'où

$$p_{k+1} = r_{k+1} - \frac{Ar_{k+1} \cdot p_k}{Ap_k \cdot p_k}\, p_k = r_{k+1} + \frac{\alpha_k^{-1} r_{k+1} \cdot r_{k+1}}{Ap_k \cdot p_k}\, p_k = r_{k+1} + \frac{r_{k+1} \cdot r_{k+1}}{r_k \cdot p_k}\, p_k.$$

Ce qui donne le résultat. □

Lemme 9.7. *Soient $x_0 \in \mathbb{R}^n$, $r_0 = p_0 = b - Ax_0$ et, pour tout $k \in \{1, \ldots, n\}$, x_k, p_k, r_k définis par* (9.1.4)*.. Alors, on a*

$$\forall\ k \in \{1, \ldots, n\}, \quad \alpha_k = \frac{r_k \cdot r_k}{Ap_k \cdot p_k} \quad \textit{et} \quad \beta_k = -\frac{r_{k+1} \cdot r_{k+1}}{r_k \cdot r_k}.$$

Démonstration. Soit $k \in \{1, \ldots, n\}$. On a

$$\alpha_k = \frac{r_k \cdot p_k}{Ap_k \cdot p_k}.$$

Or $r_k \cdot p_k = r_k \cdot (r_k - \beta_{k-1}p_{k-1}) = r_k \cdot r_k - \beta_{k-1} r_k \cdot p_{k-1})$. D'après la Proposition 9.3., puisque la famille $\{p_0, \ldots, p_{n-1}\}$ étant conjuguée, on a $r_k \cdot p_{k-1} = 0$, ce qui donne le résultat. De plus, $Ap_k = \alpha_k^{-1}(r_k - r_{k+1})$, d'où

$$Ap_k \cdot r_{k+1} = \alpha_k^{-1} r_k \cdot r_k - \alpha_k^{-1} r_{k+1} \cdot r_{k+1}.$$

L'expression de α_k précédemment obtenue et le fait que la famille $\{r_0, \ldots, r_{n-1}\}$ est orthogonale donne la formule cherchée pour β_k. □

On en déduit la définition de l'algorithme conjugué.

Définition 9.8. On appelle **algorithme du gradient conjugué** (algorithme 13) la construction de la suite $(x_k)_{k \in \mathbb{N}}$ définie par $x_0 \in \mathcal{M}_{n,1}(\mathbb{R})$, $p_0 = r_0 = b - Ax_0$ et, pour $k \in \mathbb{N}$,

$$\begin{cases} \alpha_k & = \dfrac{r_k \cdot r_k}{Ap_k \cdot p_k}, \\ x_{k+1} & = x_k + \alpha_k\, p_k, \\ r_{k+1} & = b - Ax_{k+1} = r_k - \alpha_k Ap_k, \\ \beta_k & = \dfrac{r_{k+1} \cdot r_{k+1}}{r_k \cdot r_k} \\ p_{k+1} & = r_{k+1} + \beta_k\, p_k. \end{cases} \tag{9.1.5}$$

Algorithme 13 Algorithme du gradient conjugué

Entrées : A, b, ε
Sorties : x
Prendre $x \in \mathcal{M}_{n,1}(\mathbb{R})$.
Calculer $r = b - Ax$.
$p = r$.
Tant que $\|r\|_2 \geq \varepsilon$ **faire**
$c = r \cdot r$
$y = Ap$
$\alpha = \frac{c}{y \cdot p}$
$x = x + \alpha\, p$
$r = r - \alpha y$
$\beta = \frac{r \cdot r}{c}$
$p = r + \beta p$
fin Tant que

En terme de complexité, chaque itération de l'algorithme du gradient conjugué nécessite de l'ordre de n^2 multiplications. ainsi, la méthode du gradient conjuguée en tant que méthode directe nécessite n^3 multiplications et donc est 6 fois plus lente que la méthode de Cholesky.

Remarque 9.9. L'étape la plus couteuse de l'algorithme est le calcul matrice vecteur Ap. Dans le cas de matrice creuses, ce calcul peut être grandement réduit. C'est notamment le cas de la matrice du laplacien vu dans le Chapitre 5. En effet, dans ce cas, on a $(Ap)_i = -p_{i-1} - p_{i+1} + 2p_i$, pour $2 \leq i \leq n-1$, $(Ap)_1 = -p_2 + 2p_1$ et $(Ap)_n = -p_{n-1} + 2p_n$. Ainsi le calcul de Ap ne nécessite que n multiplications au lieu des n^2 pour des matrices pleines. L'algorithme du gradient conjugué peut donc être avantageux pour des matrices creuses en tant que méthode itérative.

Puisque l'algorithme du gradient conjugué n'est intéressant que en tant que méthode itérative, il est important d'en déterminer la vitesse de convergence et de pouvoir accélérer celle-ci. C'est l'objet de la section suivante.

9.2 Vitesse de convergence

Définition 9.10. Soient $r_0 \in \mathcal{M}_{n,1}(\mathbb{R})$ et $k \in \mathbb{N}^*$. On appelle **espace de Krilov** d'ordre k associé à r_0 (et A) l'espace

$$\mathcal{K}_k \;:=\; \mathrm{Vect}\{r_0, Ar_0, \ldots, A^k r_0\}.$$

Remarque 9.11. On a vu dans le Lemme 9.5., que les suites $(p_k)_{k\in\mathbb{N}}$ et $(r_k)_{k\in\mathbb{N}}$ définies par l'algorithme du gradient conjugué vérifient

$$\mathcal{K}_k = \mathrm{Vect}\{p_0, p_1, \ldots, p_k\} = \mathrm{Vect}\{r_0, r_1, \ldots, r_k\}.$$

De plus, l'approximation x_{k+1} calculée par l'algorithme du gradient conjugué est donnée par $x_{k+1} = x_k + \alpha_k\, p_k$ donc

$$x_{k+1} - x_0 = \sum_{i=0}^{k} x_{i+1} - x_i = \sum_{i=0}^{k} \alpha_i\, p_i \in \mathcal{K}_k.$$

Notation 9.12. Pour tout $x_0 \in \mathcal{M}_{n,1}(\mathbb{R})$, on note $[x_0 + \mathcal{K}_k]$, l'ensemble des $x \in \mathcal{M}_{n,1}(\mathbb{K})$ tels que $x - x_0 \in \mathcal{K}_k$.

Afin de déterminer la vitesse de convergence de l'algorithme du gradient conjugué, on va montrer que à l'itération $k+1$, l'approximation x_{k+1} calculée par l'algorithme du gradient conjugué est la solution du problème

$$\begin{cases} \text{Trouver } x_{k+1} \in [x_0 + \mathcal{K}_k] \text{ tel que} \\ f(x_{k+1}) = \min\limits_{x \in [x_0 + \mathcal{K}_k]} f(x), \quad \text{où} \quad f(x) := \dfrac{1}{2}\, Ax \cdot x - b \cdot x. \end{cases} \tag{9.2.1}$$

Pour cela, on aura besoin du résultat auxiliaire donné ci-après

Proposition 9.13. *Soient f la fonction donnée dans* (9.2.1). *et F un sous-espace vectoriel de $\mathbb{R}^n$. Alors, il existe un unique $x_0 \in F$ tel que*

$$f(x_0) = \min_{x \in F} f(x),$$

et, de plus, x_0 est l'unique vecteur de F tel que

$$(Ax_0 - b) \cdot y = 0, \quad \forall\ y \in F.$$

Démonstration. Soit P la projection orthogonale de $\mathbb{R}^n$ sur F. Alors P est surjective et donc

$$\min_{x \in F} f(x) = \min_{x \in \mathbb{R}^n} f(Px).$$

De plus,

$$g(x) := f(Px) = \frac{1}{2}\, P^*APx \cdot x - P^*b \cdot x.$$

La matrice P^*AP est symétrique positive. Si P^*AP est définie positive, d'après la Proposition 8.10., il existe un unique $y_0 \in \mathbb{R}^n$ tel que

$$g(y_0) = \min_{y \in \mathbb{R}^n} g(y) = \min_{x \in F} f(x)$$

et donc $x_0 := Py_0$ répond à la question. De plus, on a $P^*APy_0 = P^*b$, d'où $P^*(Ax_0 - b) = 0$, *i.e.* $(Ax_0 - b) \cdot Px = 0$ pour tout $x \in \mathbb{R}^n$, ce qui donne le résultat.

Si P^*AP n'est pas définie positive, alors g est convexe (et non strictement convexe). De plus,

$$g(y) = f(Py) \underset{\|y\|_2 \to +\infty}{\longrightarrow} +\infty.$$

Donc g admet au moins un minimum y_1 sur $\mathbb{R}^n$ vérifiant $P^*APy_1 = P^*b$, ce qui donne $(APy_1 - b) \cdot y = 0$, pour tout $y \in F$. De plus, si y_2 est une autre solution du problème de minimisation, on a de même $P^*APy_2 = P^*b$. Alors, $P^*AP(y_2 - y_1) = 0$ d'où $AP(y_2 - y_1) \cdot P(y_2 - y_1) = 0$. Or A est définie positive donc $Py_1 = Py_2$. Finalement, $x_0 = Py_0$ est donc l'unique solution du problème. □

Proposition 9.14. *Soit $k \in \mathbb{N}$. Alors, il existe une unique solution $x_{k+1} \in [x_0 + \mathcal{K}_k]$ de (9.2.1). et x_{k+1} est l'unique élément de $[x_0 + \mathcal{K}_k]$ tel que le résidu associé $r_{k+1} := b - Ax_{k+1}$ est orthogonal à $\mathcal{K}_k$.*

Démonstration. Soit $x_{k+1} \in [x_0 + \mathcal{K}_k]$. Alors, il existe $c_0, \ldots, c_{k-1} \in \mathbb{R}$ tels que

$$x_{k+1} = x_0 + \sum_{i=0}^{k} c_i \, A^i r_0.$$

Alors, $r_{k+1} := b - Ax_{k+1}$ vérifie

$$r_{k+1} = r_0 + \sum_{i=0}^{k} c_i \, A^{i+1} r_0 \in \mathcal{K}_{k+1}.$$

Si $x \in [x_0 + K_k]$ alors, $x = x_0 + y$ où $y \in \mathcal{K}_k$ d'où

$$\min_{x \in [x_0 + \mathcal{K}_k]} f(x) = \min_{y \in \mathcal{K}_k} f(x_0 + y),$$

Or

$$f(x_0+y) = \frac{1}{2} A(x_0+y)\cdot(x_0+y) - b\cdot(x_0+y) = \frac{1}{2} Ay\cdot y + \frac{1}{2} Ax_0\cdot x_0 + Ay\cdot x_0 - b\cdot x_0 - b\cdot y.$$

Donc

$$\min_{x \in [x_0 + \mathcal{K}_k]} f(x) = \min_{y \in \mathcal{K}_k} g(y),$$

où

$$g(y) \; := \; \frac{1}{2} Ay \cdot y - (b - Ax_0) \cdot y.$$

Alors, d'après la Proposition 9.13., le problème (9.2.1). admet une unique solution $x_{k+1} = x_0 + y_{k+1}$ où $y_{k+1} \in \mathcal{K}_{k+1}$ est la solution de

$$(Ay_{k+1} - (b - Ax_0)) \cdot z = 0, \quad \forall \; z \in \mathcal{K}_{k+1},$$

i.e.

$$(Ax_{k+1} - b) \cdot z = 0, \quad \forall \; z \in \mathcal{K}_{k+1},$$

ce qui donne le résultat. □

Corollaire 9.15. *Pour tout $k \in \mathbb{N}$, l'approximation x_{k+1} calculée par l'algorithme du gradient conjugué est l'unique solution du problème (9.2.1). et l'unique élément de $[x_0 + \mathcal{K}_k]$ tel que le résidu associé $r_{k+1} := b - Ax_{k+1}$ est orthogonal à $\mathcal{K}_k$.*

Démonstration. Il suffit de montrer que le résidu r_{k+1} est orthogonal à $\mathcal{K}_k$. Or d'après le Lemme 9.6., r_{k+1} est orthogonal à $r_1, \ldots, r_k$ et, d'autre part, $\mathcal{K}_k = \text{Vect}\{r_0, \ldots, r_k\}$ ce qui donne le résultat. □

Théorème 9.16. *Soient x la solution de $Ax = b$ et $(x_k)_{k \in \mathbb{N}}$ la suite des approximations calculées par l'algorithme du gradient conjugué. Alors, on a*

$$\forall \; k \in \mathbb{N}, \quad \|x_k - x\|_2 \leq 2\sqrt{\kappa_2(A)} \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k \|x_0 - x\|_2. \tag{9.2.2}$$

Remarque 9.17. La vitesse de convergence de l'algorithme du gradient conjugué dépend donc de la racine carrée du conditionnement de A, alors que celle de l'algorithme du gradient dépend du conditionnement lui-même. On en déduit que l'algorithme du gradient conjuguée converge plus vite que celui du gradient.

Démonstration. On a montré que x_k est la solution du problème de minimisation de la fonction f sur $[x_0 + \mathcal{K}_{k-1}]$. En notant $\|\cdot\|_A$ la norme associée au produit scalaire $(\cdot,\cdot)_A$, on a pour tout $z \in \mathcal{M}_{n,1}(\mathbb{R})$

$$f(z) = \frac{1}{2} Az\cdot z - b\cdot z = \frac{1}{2} Az\cdot z - Ax\cdot z = \frac{1}{2} A(z-x)\cdot(z-x) - Ax\cdot x = \frac{1}{2}\|x-z\|_A^2 - \frac{1}{2} Ax\cdot x.$$

Autrement dit minimiser f sur $[x_0+\mathcal{K}_{k-1}]$, revient à minimiser $z \mapsto \|x-z\|_A^2$ sur $[x_0+\mathcal{K}_{k-1}]$. On pose $e_k := x_k - x$. Alors, on a

$$\|e_k\|_A = \min_{z\in[x_0+\mathcal{K}_{k-1}]} \|x - z\|_A.$$

□

9.3 Préconditionnement

D'après l'estimation d'erreur (9.2.2)., le préconditionnement peut grandement améliorer la vitesse de convergence de l'algorithme du gradient conjugué.

On rappelle (voir Chapitre ?) que le préconditionnement consiste à choisir une matrice M appelée préconditionneur tel que $M^{-1}A$ a un conditionnement plus petit que A, on résout alors le problème $M^{-1}Ax = M^{-1}b$. pour l'algorithme du gradient conjugué il faut que $M^{-1}A$ soit symétrique définie positive or même si M est symétrique définie positive, $M^{-1}A$ peut ne pas être ne serait-ce que symétrique. Néanmoins, on a le résultat suivant :

Lemme 9.18. *Soit $M \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive. Alors, il existe $E \in \mathcal{M}_n(\mathbb{R})$ inversible telle que $M = EE^T$. De plus, $M^{-1}A$ et $E^{-1}AE^{-T}$ ont les mêmes valeurs propres.*

Remarque 9.19. D'une part, puisque A et M sont symétriques définies positives, on a

$$(M^{-1}A)^T M^{-1}A = A^T M^{-T} M^{-1} A = I_n = M^{-1}A(M^{-1}A)^T,$$

autrement dit $M^{-1}A$ est normale. D'autre part, $E^{-1}AE^{-T}$ est symétrique. D'après **??** et le Lemme, on en déduit que $\kappa_2(M^{-1}A) = \kappa_2(E^{-1}AE^{-T})$.

Démonstration. L'existence de E est assurée, par exemple, par le théorème de décomposition de Cholesky. De plus, si λ est une valeur propre de $M^{-1}A$ et x un vecteur propre associée, en posant $y := E^T x$, on a

$$E^{-1}AE^{-T}y = E^{-1}Ax = E^{-1}MM^{-1}Ax = \lambda E^{-1}EE^T x = \lambda E^T x = \lambda y$$

, donc λ est une valeur propre de $E^{-1}AE^{-T}$. On montre de même que toute valeur propre de $E^{-1}AE^{-T}$ est une valeur propre de $M^{-1}A$. □

Algorithme 14 Gradient conjugué préconditionné 1

Entrées : A, b, ε, E
Sorties : x
Prendre $\tilde{x} \in \mathcal{M}_{n,1}(\mathbb{R})$.
Calculer $\tilde{r} = E^{-1}b - E^{-1}AE^{-T}\tilde{x}$.
$\tilde{p} = \tilde{r}$.
Tant que $\|\tilde{r}\|_2 \geq \varepsilon$ **faire**
$\tilde{c} = \tilde{r} \cdot \tilde{r}$
$\tilde{y} = E^{-1}AE^{-T}\tilde{p}$
$\alpha = \frac{\tilde{c}}{\tilde{y}\cdot\tilde{p}}$
$\tilde{x} = \tilde{x} + \alpha\,\tilde{p}$
$\tilde{r} = \tilde{r} - \alpha\tilde{y}$
$\beta = \frac{\tilde{r}\cdot\tilde{r}}{\tilde{c}}$
$\tilde{p} = \tilde{r} + \beta\tilde{p}$
fin Tant que
$x = E^{-T}\tilde{x}$

Soit $M \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive avec $M = EE^T$, où E est inversible. Alors, d'après ce qui précède $E^{-1}AE^{-T}$ a le même conditionnement que $M^{-1}A$ mais est symétrique définie positive, on peut donc lui appliquer l'algorithme du gradient conjugué. La méthode consiste alors à résoudre

$$E^{-1}AE^{-T}\tilde{x} = E^{-1}b \quad \text{puis} \quad x = E^{-T}\tilde{x}.$$

De cette manière (pour un bon choix de M), on est amené à un problème mieux conditionné auquel on peut appliquer l'algorithme du gradient conjugué.

Le défaut de cette méthode est qu'elle entraîne des calculs supplémentaires : le calcul de E puis des produits matrices-vecteurs $E^{-1}b$, $E^{-1}AE^{-T}x$ dont on ne sait pas *a priori* s'ils sont simples à calculer. On peut changer l'algorithme précédent de sorte à ne plus faire intervenir la matrice E mais la matrice M. Pour cela, on pose

$$E\tilde{r} = r, \quad E^{-T}\tilde{p} = p.$$

Puisque $x = E^{-T}\tilde{x}$ et $M = EE^T$, on obtient l'algorithme

On remarque que M uniquement dans la résolution de système du type $Mq = r$, ainsi il faut pouvoir résoudre simplement ces systèmes. Autrement dit, M doit être une matrice proche de A (pour que $\kappa_2(M^{-1}A) \leq \kappa_2(A)$) qui soit simples à inverser.

Parmi les différents choix possibles de M, on citera les plus connus :

1. $M = \text{diag}(a_{11}, \ldots, a_{nn})$, *i.e* M est la diagonale de A. Le coût de résolution de $Mq = r$ est n.
2. M est la **décomposition de Cholesky incomplète** de A. On entend par décomposition de Cholesky incomplète, une modification de la décomposition de Cholesky BB^T de A telle que $b_{ij} = 0$ si $a_{ij} = 0$. La décomposition ainsi obtenue a l'avantage de conserver la même structure que la matrice. Ainsi si A est une matrice bande, il en va de même pour B.

Algorithme 15 Gradient conjugué préconditionné 2

Entrées : A, b, ε, M
Sorties : x
Prendre $x \in \mathcal{M}_{n,1}(\mathbb{R})$.
Calculer $r = b - Ax$.
Résoudre $Mq = r$.
$p = q$.
Tant que $\|r\|_2 \geq \varepsilon$ **faire**
$c = q \cdot r$
$\alpha = \frac{c}{Ap \cdot p}$
$x = x + \alpha\, p$
$r = r - \alpha Ap$
Résoudre $Mq = r$
$\beta = \frac{r' \cdot r}{c}$
$p = q + \beta p$
fin Tant que

Bibliographie

[1] G. Allaire & S.M. Kaber, *Algèbre linéaire numérique. Cours et exercices*. Édition ellipses, Paris, 2002.

[2] P.G. Ciarlet, *Introduction à l'analyse numérique matricielle et à l'optimisation*, Dunod, Paris, 1998.

[3] P. Lascaux & R. Théodor, *Analyse numérique matricielle appliquée à l'art de l'ingénieur*, Dunod, Paris, 2004.

[4] A. Quarteroni, R. Sacco & F. Saleri, *Méthodes numériques : Algorithmes, analyse et applications*, Springer-Verlag Italia, Milano, 2007.

[5] M. Schatzman, *Analyse numérique : une approche mathématique : cours et exercices*, Dunod, Paris, 2001.

[6] J.H. Wilkinson, *The Algebraic Eigenvalue Problem*. Clarendon Press, Oxford, 1965.

Index