

MT09 : Chapitre 2

Méthodes directes de résolution des systèmes linéaires

MT09

Vincent.Martin@utc.fr

UTC
Compiègne, France

UTC, A2022

Plan

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

- Pourquoi résoudre des systèmes linéaires ?
 - Pour résoudre des systèmes linéaires $Ax = b$ (A matrice $n \times n$) et non-linéaires $f(x) = 0$ (f fonction $\mathbb{R}^n \rightarrow \mathbb{R}^n$).
 - dans tous les domaines. Ex : éléments finis (NF04)...
 - Très gros systèmes *creux*
 - $n \geq 100\,000$ lignes : typique.
 - très peu de termes non-nuls par ligne.
 - Systèmes pleins de taille modérée

- Pourquoi résoudre des systèmes linéaires ?
 - Pour résoudre des systèmes linéaires $Ax = b$ (A matrice $n \times n$) et non-linéaires $f(x) = 0$ (f fonction $\mathbb{R}^n \rightarrow \mathbb{R}^n$).
 - dans tous les domaines. Ex : éléments finis (NF04)...
 - Très gros systèmes *creux*
 - $n \geq 100\,000$ lignes : typique.
 - très peu de termes non-nuls par ligne.
 - Systèmes pleins de taille modérée
- 2 types de méthodes :

Méthodes directes

⇒ Chap. 2.

Algorithme : on passe par toutes les lignes de A .

$\mathcal{O}(n^p)$ opérations fixe au départ.

Méthodes itératives

⇒ Chap. 4.

Algorithme : suite d'itérés $x^{(k)}$ qui converge vers x .

Nombre infini (?) d'itérations.

Introduction et motivation

- Pourquoi résoudre des systèmes linéaires ?
 - Pour résoudre des systèmes linéaires $Ax = b$ (A matrice $n \times n$) et non-linéaires $f(x) = 0$ (f fonction $\mathbb{R}^n \rightarrow \mathbb{R}^n$).
 - dans tous les domaines. Ex : éléments finis (NF04)...
 - Très gros systèmes *creux*
 - $n \geq 100\,000$ lignes : typique.
 - très peu de termes non-nuls par ligne.
 - Systèmes pleins de taille modérée
- 2 types de méthodes :

Méthodes directes

⇒ Chap. 2.

Algorithme : on passe par toutes les lignes de A .

$\mathcal{O}(n^p)$ opérations fixe au départ.

Méthodes itératives

⇒ Chap. 4.

Algorithme : suite d'itérés $x^{(k)}$ qui converge vers x .

Nombre infini (?) d'itérations.

- Les 2 ont leurs avantages... Voir plus tard (Chap 4).

Résoudre $Ax = b$.

- Algorithme de Gauss : voir Section 2.
 - \implies obtient une factorisation LU : voir Section 3.
- Factorisation $A = LU$:
 - produit de L triangulaire inférieure (“lower”, “tri-inf”) et U triangulaire supérieure (“upper”, “tri-sup”).
 - **Attention** : ce n’est PAS un changement de base !
 - \implies se ramener à des systèmes plus faciles à résoudre : systèmes triangulaires $Lx = b$ et $Ux = b$

$$Ax = b \iff L(\underbrace{Ux}_y) = b \iff \begin{cases} Ly = b \\ Ux = y \end{cases}$$

Résolution des systèmes triangulaires $Lx = b$

Comment résoudre $Lx = b$?

(L inversible $\Leftrightarrow l_{ii} \neq 0 \forall i$).

Note : tous les vecteurs de MT09 sont des vecteurs **colonne** (dans $\mathcal{M}_{n1}(\mathbb{R})$).

$$Lx = b \Leftrightarrow \begin{pmatrix} l_{1,1} & 0 & 0 & \cdots & 0 \\ l_{2,1} & l_{2,2} & 0 & \cdots & 0 \\ l_{3,1} & l_{3,2} & l_{3,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n,1} & l_{n,2} & l_{n,3} & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

$$\Leftrightarrow \begin{cases} l_{1,1}x_1 & = b_1 \\ l_{2,1}x_1 + l_{2,2}x_2 & = b_2 \\ l_{3,1}x_1 + l_{3,2}x_2 + l_{3,3}x_3 & = b_3 \\ \cdots & \\ l_{n,1}x_1 + l_{n,2}x_2 + l_{n,3}x_3 + \cdots + l_{nn}x_n & = b_n \end{cases}$$

$$\Leftrightarrow \forall i = 1, \dots, n \quad \sum_{j=1}^{i-1} l_{i,j}x_j + l_{i,i}x_i = b_i$$

$$\Leftrightarrow \forall i = 1, \dots, n \quad x_i = \frac{1}{l_{i,i}} \left(b_i - \sum_{j=1}^{i-1} l_{i,j}x_j \right), \quad \text{si } l_{i,i} \neq 0.$$

- **Maths :**

$$Lx = b \Leftrightarrow \forall i = 1, \dots, n \quad x_i = \frac{1}{l_{i,i}} \left(b_i - \sum_{j=1}^{i-1} l_{i,j} x_j \right), \quad \text{si } l_{i,i} \neq 0.$$

- **Récurrence :**

- 1 Ligne $i = 1$: $x_1 = \frac{b_1}{l_{1,1}}$ ($\Rightarrow x_1$ connu)
- 2 Ligne $i = 2$: $x_2 = \frac{1}{l_{2,2}} (b_2 - l_{2,1} x_1)$ ($\Rightarrow x_1, x_2$ connus)
- 3 Ligne i : si x_1, x_2, \dots, x_{i-1} connus, alors $x_i = \frac{1}{l_{i,i}} \left(b_i - \sum_{j=1}^{i-1} l_{i,j} x_j \right)$
($\Rightarrow x_1, x_2, \dots, x_i$ connus)

Résolution des systèmes triangulaires $Lx = b$

- **Maths :**

$$Lx = b \Leftrightarrow \forall i = 1, \dots, n \quad x_i = \frac{1}{l_{i,i}} \left(b_i - \sum_{j=1}^{i-1} l_{i,j} x_j \right), \quad \text{si } l_{i,j} \neq 0.$$

Algorithme

```
Pour i=1 à n
  Si L(i,i) = 0
    Sortir, erreur : matrice L non inversible
  Fin Si
  s ← 0
  Pour j=1 à i-1
    s ← s + L(i,j) × x(j)
  Fin Pour
  x(i) ← (b(i) - s) / L(i,i)
Fin Pour
```

Résolution des systèmes triangulaires $Lx = b$

- **Maths :**

$$Lx = b \Leftrightarrow \forall i = 1, \dots, n \quad x_i = \frac{1}{l_{i,i}} \left(b_i - \sum_{j=1}^{i-1} l_{i,j} x_j \right), \quad \text{si } l_{i,j} \neq 0.$$

Algorithme

```
Pour i=1 à n
  Si L(i,i) = 0    (Attention: flottants!)
    Sortir, erreur : matrice L non inversible
  Fin Si
  s ← 0
  Pour j=1 à i-1
    s ← s + L(i,j) × x(j)
  Fin Pour
  x(i) ← (b(i) - s) / L(i,i)
Fin Pour
```

Factorisation et déterminant

Comment calculer $\det(A)$?

Comment calculer $\det(A)$?

- Calcul direct : $\det(A) = \sum_{\sigma} (-1)^{\varepsilon(\sigma)} \prod_{i=1}^n \det A_{|i, \sigma(i)|}$:
 - σ : permutations de $\{1, \dots, n\}$, $A_{|i, \sigma(i)|}$, matrice sans ligne i et colonne $\sigma(i)$
 - \implies coût en $\mathcal{O}(n!)$
 - matrice 100×100 : calculateur à 100 PetaFlops (10^{17} opérations / s) $\implies 10^{141}$ s = $3 \cdot 10^{133}$ ans.
(âge de l'univers $\approx 14 \cdot 10^9$ ans...)
 - calcul analytique : **impossible** en pratique
 - et en plus : **peu précis** (accumulation d'erreurs) !

Factorisation et déterminant

Comment calculer $\det(A)$?

- Calcul direct : $\det(A) = \sum_{\sigma} (-1)^{\varepsilon(\sigma)} \prod_{i=1}^n \det A_{|i, \sigma(i)|}$:
 - σ : permutations de $\{1, \dots, n\}$, $A_{|i, \sigma(i)|}$, matrice sans ligne i et colonne $\sigma(i)$
 - \implies coût en $\mathcal{O}(n!)$
 - matrice 100×100 : calculateur à 100 PetaFlops (10^{17} opérations / s) $\implies 10^{141}$ s = $3 \cdot 10^{133}$ ans.
(âge de l'univers $\approx 14 \cdot 10^9$ ans...)
 - calcul analytique : **impossible** en pratique
 - et en plus : **peu précis** (accumulation d'erreurs) !
- Par la factorisation $A = LU$

$$A = LU = \begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ * & l_{22} & 0 & \cdots & 0 \\ * & * & l_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ * & * & * & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} u_{11} & * & * & \cdots & * \\ 0 & u_{22} & * & \cdots & * \\ 0 & 0 & u_{33} & \cdots & * \\ \vdots & \vdots & \vdots & \ddots & * \\ 0 & 0 & 0 & \cdots & u_{nn} \end{pmatrix}$$

- Calculer la factorisation : \implies coût en $\mathcal{O}(n^3)$ (voir plus loin)
- $\det(A) = \det(L) \times \det(U) = \prod_{i=1}^n l_{ii} u_{ii}$.

- Calcul du déterminant : un enseignement fondamental
 - calcul “analytique” : parfois ne marche pas en pratique !
 - méthodes “naïves” souvent vouées à l’échec
 - \implies besoin de déterminer le **coût** (taille mémoire, nombre d’opérations), la **stabilité** et la **robustesse** d’un algorithme.

- Calcul du déterminant : un enseignement fondamental
 - calcul “analytique” : parfois ne marche pas en pratique !
 - méthodes “naïves” souvent vouées à l’échec
 - \implies besoin de déterminer le **coût** (taille mémoire, nombre d’opérations), la **stabilité et la robustesse** d’un algorithme.

Dicton

En théorie, la pratique et la théorie donnent le même résultat.
En pratique : non.

- 1 Introduction, motivations
- 2 Élimination de Gauss**
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

Élimination de Gauss : objectifs et notations

Hypothèse : A matrice carrée $n \times n$ inversible, $b \in \mathbb{R}^n$.

Objectif

- écrire l'algorithme : maths et informatique.
- voir les conditions de fonctionnement
- construire la factorisation $A = LU$

Notations

- \underline{A}_i : ligne i de A .
- A_j : colonne j de A .
- si $C = AB$, alors $C_j = AB_j$ et $\underline{C}_i = \underline{A}_i B$.

Exemple de produit matriciel

Exemple

$A \in \mathcal{M}_{1,3}$ et $B \in \mathcal{M}_{3,2}$

$$\begin{aligned} AB &= (a_{1,1} \ a_{1,2} \ a_{1,3}) \begin{pmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \\ b_{3,1} & b_{3,2} \end{pmatrix} \\ &= (a_{1,1}b_{1,1} + a_{1,2}b_{2,1} + a_{1,3}b_{3,1} \mid a_{1,1}b_{1,2} + a_{1,2}b_{2,2} + a_{1,3}b_{3,2}) \\ &= a_{1,1} (b_{1,1} \mid b_{1,2}) + a_{1,2} (b_{2,1} \mid b_{2,2}) + a_{1,3} (b_{3,1} \mid b_{3,2}) \\ &= a_{1,1} \underline{B}_1 + a_{1,2} \underline{B}_2 + a_{1,3} \underline{B}_3. \end{aligned}$$

Conclusion :

$$\underline{C}_j = \underline{A}_j B = \sum_{k=1}^n a_{jk} \underline{B}_k$$

[Lien vers LU](#) .

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$

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

$$\begin{cases} \underline{A}_1^{(2)} &= \underline{A}_1^{(1)} \\ \underline{A}_2^{(2)} &= \underline{A}_2^{(1)} - \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_3^{(2)} &= \underline{A}_3^{(1)} - \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \end{cases}$$

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$
$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 5 \end{bmatrix}$$
$$\begin{cases} \underline{A}_1^{(2)} &= \underline{A}_1^{(1)} \\ \underline{A}_2^{(2)} &= \underline{A}_2^{(1)} - \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_3^{(2)} &= \underline{A}_3^{(1)} - \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \end{cases}$$

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$
$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 5 \end{bmatrix}$$
$$\left\{ \begin{array}{l} \underline{A}_1^{(2)} = \underline{A}_1^{(1)} \\ \underline{A}_2^{(2)} = \underline{A}_2^{(1)} - \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_3^{(2)} = \underline{A}_3^{(1)} - \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_1^{(3)} = \underline{A}_1^{(2)} \\ \underline{A}_2^{(3)} = \underline{A}_2^{(2)} \\ \underline{A}_3^{(3)} = \underline{A}_3^{(2)} - \frac{a_{3,2}^{(2)}}{a_{2,2}^{(2)}} \underline{A}_2^{(2)} \end{array} \right.$$

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 5 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & -9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ -9 \end{bmatrix}$$

$$\left\{ \begin{array}{l} \underline{A}_1^{(2)} = \underline{A}_1^{(1)} \\ \underline{A}_2^{(2)} = \underline{A}_2^{(1)} - \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_3^{(2)} = \underline{A}_3^{(1)} - \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_1^{(3)} = \underline{A}_1^{(2)} \\ \underline{A}_2^{(3)} = \underline{A}_2^{(2)} \\ \underline{A}_3^{(3)} = \underline{A}_3^{(2)} - \frac{a_{3,2}^{(2)}}{a_{2,2}^{(2)}} \underline{A}_2^{(2)} \end{array} \right.$$

Élimination de Gauss : exemple 3×3 $Ax = b$

$$Ax = b \Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 11 \\ 3 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 5 \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & -9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ -9 \end{bmatrix}$$

$$\begin{cases} \underline{A}_1^{(2)} = \underline{A}_1^{(1)} \\ \underline{A}_2^{(2)} = \underline{A}_2^{(1)} - \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \\ \underline{A}_3^{(2)} = \underline{A}_3^{(1)} - \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} \underline{A}_1^{(1)} \end{cases}$$

$$\begin{cases} \underline{A}_1^{(3)} = \underline{A}_1^{(2)} \\ \underline{A}_2^{(3)} = \underline{A}_2^{(2)} \\ \underline{A}_3^{(3)} = \underline{A}_3^{(2)} - \frac{a_{3,2}^{(2)}}{a_{2,2}^{(2)}} \underline{A}_2^{(2)} \end{cases}$$

$$\Leftrightarrow A^{(3)}x = b^{(3)}$$

$$A = LU \text{ avec } U = A^{(3)} \text{ et } L = \begin{bmatrix} 1 & 0 & 0 \\ \frac{a_{2,1}^{(1)}}{a_{1,1}^{(1)}} & 1 & 0 \\ \frac{a_{3,1}^{(1)}}{a_{1,1}^{(1)}} & \frac{a_{3,2}^{(2)}}{a_{2,2}^{(2)}} & 1 \end{bmatrix}$$

Élimination de Gauss : cas général

$$A^{(k)} = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & a_{1,3}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & a_{1,k+1}^{(1)} & a_{1,k+2}^{(1)} & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & a_{2,3}^{(2)} & \cdots & a_{2,k-1}^{(2)} & a_{2,k}^{(2)} & a_{2,k+1}^{(2)} & a_{2,k+2}^{(2)} & \cdots & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(3)} & \cdots & a_{3,k-1}^{(3)} & a_{3,k}^{(3)} & a_{3,k+1}^{(3)} & a_{3,k+2}^{(3)} & \cdots & a_{3,n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & a_{k-1,k+1}^{(k-1)} & a_{k-1,k+2}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ 0 & 0 & 0 & \cdots & 0 & a_{k,k}^{(k)} & a_{k,k+1}^{(k)} & a_{k,k+2}^{(k)} & \cdots & a_{k,n}^{(k)} \\ 0 & 0 & 0 & \cdots & 0 & a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & a_{k+1,k+2}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ 0 & 0 & 0 & \cdots & 0 & a_{k+2,k}^{(k)} & a_{k+2,k+1}^{(k)} & a_{k+2,k+2}^{(k)} & \cdots & a_{k+2,n}^{(k)} \\ \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & a_{n,k}^{(k)} & a_{n,k+1}^{(k)} & a_{n,k+2}^{(k)} & \cdots & a_{n,n}^{(k)} \end{bmatrix}$$

Élimination de Gauss : cas général

$$A^{(k+1)} = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & a_{1,3}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & a_{1,k+1}^{(1)} & a_{1,k+2}^{(1)} & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & a_{2,3}^{(2)} & \cdots & a_{2,k-1}^{(2)} & a_{2,k}^{(2)} & a_{2,k+1}^{(2)} & a_{2,k+2}^{(2)} & \cdots & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(3)} & \cdots & a_{3,k-1}^{(3)} & a_{3,k}^{(3)} & a_{3,k+1}^{(3)} & a_{3,k+2}^{(3)} & \cdots & a_{3,n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & a_{k-1,k+1}^{(k-1)} & a_{k-1,k+2}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ 0 & 0 & 0 & \cdots & 0 & a_{k,k}^{(k)} & a_{k,k+1}^{(k)} & a_{k,k+2}^{(k)} & \cdots & a_{k,n}^{(k)} \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{k+1,k+1}^{(k+1)} & a_{k+1,k+2}^{(k+1)} & \cdots & a_{k+1,n}^{(k+1)} \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{k+2,k+1}^{(k+1)} & a_{k+2,k+2}^{(k+1)} & \cdots & a_{k+2,n}^{(k+1)} \\ \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{n,k+1}^{(k+1)} & a_{n,k+2}^{(k+1)} & \cdots & a_{n,n}^{(k+1)} \end{bmatrix}$$

Élimination de Gauss : cas général

$$A^{(k+1)} = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & a_{1,3}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & a_{1,k+1}^{(1)} & a_{1,k+2}^{(1)} & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & a_{2,3}^{(2)} & \cdots & a_{2,k-1}^{(2)} & a_{2,k}^{(2)} & a_{2,k+1}^{(2)} & a_{2,k+2}^{(2)} & \cdots & a_{2,n}^{(2)} \\ 0 & 0 & a_{3,3}^{(3)} & \cdots & a_{3,k-1}^{(3)} & a_{3,k}^{(3)} & a_{3,k+1}^{(3)} & a_{3,k+2}^{(3)} & \cdots & a_{3,n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & a_{k-1,k+1}^{(k-1)} & a_{k-1,k+2}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ 0 & 0 & 0 & \cdots & 0 & a_{k,k}^{(k)} & a_{k,k+1}^{(k)} & a_{k,k+2}^{(k)} & \cdots & a_{k,n}^{(k)} \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{k+1,k+1}^{(k+1)} & a_{k+1,k+2}^{(k+1)} & \cdots & a_{k+1,n}^{(k+1)} \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{k+2,k+1}^{(k+1)} & a_{k+2,k+2}^{(k+1)} & \cdots & a_{k+2,n}^{(k+1)} \\ \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & a_{n,k+1}^{(k+1)} & a_{n,k+2}^{(k+1)} & \cdots & a_{n,n}^{(k+1)} \end{bmatrix}$$

$$A^{(k)} \longrightarrow A^{(k+1)} : \begin{cases} \underline{A}_1^{(k+1)} & = \underline{A}_1^{(k)} \\ \underline{A}_2^{(k+1)} & = \underline{A}_2^{(k)} \\ \dots & \\ \underline{A}_{k-1}^{(k+1)} & = \underline{A}_{k-1}^{(k)} \\ \underline{A}_k^{(k+1)} & = \underline{A}_k^{(k)} \end{cases} \quad \left\{ \begin{array}{l} \underline{A}_{k+1}^{(k+1)} = \underline{A}_{k+1}^{(k)} - \frac{a_{k+1,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \underline{A}_{k+2}^{(k+1)} = \underline{A}_{k+2}^{(k)} - \frac{a_{k+2,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \dots \\ \underline{A}_n^{(k+1)} = \underline{A}_n^{(k)} - \frac{a_{n,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \end{array} \right.$$

Élimination de Gauss : méthode

$$A^{(k)} \longrightarrow A^{(k+1)} : \left\{ \begin{array}{l} \underline{A}_1^{(k+1)} = \underline{A}_1^{(k)} \\ \underline{A}_2^{(k+1)} = \underline{A}_2^{(k)} \\ \dots \\ \underline{A}_{k-1}^{(k+1)} = \underline{A}_{k-1}^{(k)} \\ \underline{A}_k^{(k+1)} = \underline{A}_k^{(k)} \end{array} \right. \left\{ \begin{array}{l} \underline{A}_{k+1}^{(k+1)} = \underline{A}_{k+1}^{(k)} - \frac{a_{k+1,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \underline{A}_{k+2}^{(k+1)} = \underline{A}_{k+2}^{(k)} - \frac{a_{k+2,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \dots \\ \underline{A}_n^{(k+1)} = \underline{A}_n^{(k)} - \frac{a_{n,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \end{array} \right.$$

Donc : on pose $A^{(1)} = A$, puis on calcule

$$\forall k = 1, \dots, n-1 : \left\{ \begin{array}{l} \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} \quad \forall i = 1, \dots, k \\ \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} - \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \quad \forall i = k+1, \dots, n \end{array} \right.$$

On obtient $U = A^{(n)}$, triangulaire supérieure. (On fait pareil pour b).

Élimination de Gauss : méthode

$$A^{(k)} \longrightarrow A^{(k+1)} : \begin{cases} \underline{A}_1^{(k+1)} &= \underline{A}_1^{(k)} \\ \underline{A}_2^{(k+1)} &= \underline{A}_2^{(k)} \\ \dots & \\ \underline{A}_{k-1}^{(k+1)} &= \underline{A}_{k-1}^{(k)} \\ \underline{A}_k^{(k+1)} &= \underline{A}_k^{(k)} \end{cases} \quad \begin{cases} \underline{A}_{k+1}^{(k+1)} &= \underline{A}_{k+1}^{(k)} - \frac{a_{k+1,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \underline{A}_{k+2}^{(k+1)} &= \underline{A}_{k+2}^{(k)} - \frac{a_{k+2,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \\ \dots & \\ \underline{A}_n^{(k+1)} &= \underline{A}_n^{(k)} - \frac{a_{n,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} \end{cases}$$

Donc : on pose $A^{(1)} = A$, puis on calcule

$$\forall k = 1, \dots, n-1 : \begin{cases} \underline{A}_i^{(k+1)} &= \underline{A}_i^{(k)} & \forall i = 1, \dots, k \\ \underline{A}_i^{(k+1)} &= \underline{A}_i^{(k)} - \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}} \underline{A}_k^{(k)} & \forall i = k+1, \dots, n \end{cases}$$

On obtient $U = A^{(n)}$, triangulaire supérieure. (On fait pareil pour b).

Notation

On pose $l_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}}$ pour $i = k+1, \dots, n$.

Algorithme d'élimination de Gauss

```
Pour k=1 jusqu'à n-1
  Si  $|A(k,k)| < \text{tol}$ 
    Sortir, erreur : pivot nul
  Sinon
    Pour i=k+1 jusqu'à n
       $c \leftarrow A(i,k) / A(k,k)$ 
       $b(i) \leftarrow b(i) - c \times b(k)$ 
       $A(i,k) \leftarrow 0$ 
      Pour j=k+1 jusqu'à n
         $A(i,j) \leftarrow A(i,j) - c \times A(k,j)$ 
      Fin Pour
    Fin Pour
  Fin Si
Fin Pour
Si  $|A(n,n)| < \text{tol}$ 
  Sortir, erreur : pivot nul
Fin Si
```

Élimination de Gauss : algorithme, coût

Calcul du nombre de multiplications et divisions :

Calcul similaire pour les additions.

Commencer par les boucles les plus internes.

- 2ème boucle interne (en j) : $n - k$ multiplications.
- 1ère boucle interne (en i) : $(n - k)^2 + 2(n - k)$ multiplications et divisions.
- boucle externe (en k) : $\sum_{k=1}^{n-1} (n - k)^2 + 2(n - k)$ multiplications et divisions.

Élimination de Gauss : algorithme, coût

Calcul du nombre de multiplications et divisions :

Calcul similaire pour les additions.

Commencer par les boucles les plus internes.

- 2ème boucle interne (en j) : $n - k$ multiplications.
- 1ère boucle interne (en i) : $(n - k)^2 + 2(n - k)$ multiplications et divisions.
- boucle externe (en k) : $\sum_{k=1}^{n-1} (n - k)^2 + 2(n - k)$ multiplications et divisions.

Coût total :

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

$$(\text{Rappel : } \sum_{p=1}^n p^2 = \frac{n(n+1)(2n+1)}{6} \text{ et } \sum_{p=1}^n p = \frac{n(n+1)}{2}.)$$

Élimination de Gauss : algorithme, coût

Calcul du nombre de multiplications et divisions :

Calcul similaire pour les additions.

Commencer par les boucles les plus internes.

- 2ème boucle interne (en j) : $n - k$ multiplications.
- 1ère boucle interne (en i) : $(n - k)^2 + 2(n - k)$ multiplications et divisions.
- boucle externe (en k) : $\sum_{k=1}^{n-1} (n - k)^2 + 2(n - k)$ multiplications et divisions.

Coût total :

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

$$(\text{Rappel : } \sum_{p=1}^n p^2 = \frac{n(n+1)(2n+1)}{6} \text{ et } \sum_{p=1}^n p = \frac{n(n+1)}{2}.)$$

Théorème (Coût computationnel de Gauss)

Le coût de l'algorithme de Gauss est en $\frac{n^3}{3}$ multiplications.

Élimination de Gauss : factorisation $A = LU$

On pose $A^{(1)} = A$, puis on calcule

$$\forall k = 1, \dots, n-1 : \begin{cases} \underline{A}_i^{(k+1)} &= \underline{A}_i^{(k)} \\ \underline{A}_i^{(k+1)} &= \underline{A}_i^{(k)} - l_{i,k} \underline{A}_k^{(k)} \end{cases} \quad \begin{matrix} \forall i = 1, \dots, k \\ \forall i = k+1, \dots, n \end{matrix} \quad \text{avec } l_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}}$$

On fixe une ligne i et on regarde les modifications :

$$\left\{ \begin{array}{l} \underline{A}_i^{(1)} = \underline{A}_i \\ \underline{A}_i^{(2)} = \underline{A}_i^{(1)} - l_{i,1} \underline{A}_1^{(1)} \\ \underline{A}_i^{(3)} = \underline{A}_i^{(2)} - l_{i,2} \underline{A}_2^{(2)} \\ \dots \\ \underline{A}_i^{(i-1)} = \underline{A}_i^{(i-2)} - l_{i,i-2} \underline{A}_{i-2}^{(i-2)} \\ \underline{A}_i^{(i)} = \underline{A}_i^{(i-1)} - l_{i,i-1} \underline{A}_{i-1}^{(i-1)} \\ \underline{A}_i^{(i+1)} = \underline{A}_i^{(i)} \\ \underline{A}_i^{(i+2)} = \underline{A}_i^{(i+1)} \\ \dots \\ \underline{A}_i^{(n-1)} = \underline{A}_i^{(n-2)} \\ \underline{A}_i^{(n)} = \underline{A}_i^{(n-1)} \end{array} \right.$$

Élimination de Gauss : factorisation $A = LU$

On pose $A^{(1)} = A$, puis on calcule

$$\forall k = 1, \dots, n-1 : \begin{cases} \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} & \forall i = 1, \dots, k \\ \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} - l_{i,k} \underline{A}_k^{(k)} & \forall i = k+1, \dots, n \end{cases} \quad \text{avec } l_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}}$$

On fixe une ligne i et on regarde les modifications :

On somme et on élimine les termes :

$$\left\{ \begin{array}{l} \underline{A}_i^{(1)} = \underline{A}_i \\ \underline{A}_i^{(2)} = \underline{A}_i^{(1)} - l_{i,1} \underline{A}_1^{(1)} \\ \underline{A}_i^{(3)} = \underline{A}_i^{(2)} - l_{i,2} \underline{A}_2^{(2)} \\ \dots \\ \underline{A}_i^{(i-1)} = \underline{A}_i^{(i-2)} - l_{i,i-2} \underline{A}_{i-2}^{(i-2)} \\ \underline{A}_i^{(i)} = \underline{A}_i^{(i-1)} - l_{i,i-1} \underline{A}_{i-1}^{(i-1)} \\ \underline{A}_i^{(i+1)} = \underline{A}_i^{(i)} \\ \underline{A}_i^{(i+2)} = \underline{A}_i^{(i+1)} \\ \dots \\ \underline{A}_i^{(n-1)} = \underline{A}_i^{(n-2)} \\ \underline{A}_i^{(n)} = \underline{A}_i^{(n-1)} \end{array} \right.$$

$$\left\{ \begin{array}{l} \underline{A}_i^{(n)} = \underline{A}_i - l_{i,1} \underline{A}_1^{(1)} - l_{i,2} \underline{A}_2^{(2)} \\ \dots \\ - l_{i,i-2} \underline{A}_{i-2}^{(i-2)} - l_{i,i-1} \underline{A}_{i-1}^{(i-1)} \end{array} \right.$$

Élimination de Gauss : factorisation $A = LU$

On pose $A^{(1)} = A$, puis on calcule

$$\forall k = 1, \dots, n-1 : \begin{cases} \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} & \forall i = 1, \dots, k \\ \underline{A}_i^{(k+1)} = \underline{A}_i^{(k)} - l_{i,k} \underline{A}_k^{(k)} & \forall i = k+1, \dots, n \end{cases} \quad \text{avec } l_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}}$$

On fixe une ligne i et on regarde les modifications :

$$\left\{ \begin{array}{l} \underline{A}_i^{(1)} = \underline{A}_i \\ \underline{A}_i^{(2)} = \underline{A}_i^{(1)} - l_{i,1} \underline{A}_1^{(1)} \\ \underline{A}_i^{(3)} = \underline{A}_i^{(2)} - l_{i,2} \underline{A}_2^{(2)} \\ \dots \\ \underline{A}_i^{(i-1)} = \underline{A}_i^{(i-2)} - l_{i,i-2} \underline{A}_{i-2}^{(i-2)} \\ \underline{A}_i^{(i)} = \underline{A}_i^{(i-1)} - l_{i,i-1} \underline{A}_{i-1}^{(i-1)} \\ \underline{A}_i^{(i+1)} = \underline{A}_i^{(i)} \\ \underline{A}_i^{(i+2)} = \underline{A}_i^{(i+1)} \\ \dots \\ \underline{A}_i^{(n-1)} = \underline{A}_i^{(n-2)} \\ \underline{A}_i^{(n)} = \underline{A}_i^{(n-1)} \end{array} \right.$$

On somme et on élimine les termes :

$$\left\{ \begin{array}{l} \underline{A}_i^{(n)} = \underline{A}_i - l_{i,1} \underline{A}_1^{(1)} - l_{i,2} \underline{A}_2^{(2)} \\ \dots \\ - l_{i,i-2} \underline{A}_{i-2}^{(i-2)} - l_{i,i-1} \underline{A}_{i-1}^{(i-1)} \end{array} \right.$$

ce qui donne

$$\begin{aligned} \underline{A}_i &= \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(j)} \\ &= \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(n)} \end{aligned}$$

$$\text{car } \underline{A}_j^{(j)} = \underline{A}_j^{(j+1)} = \dots = \underline{A}_j^{(n)}.$$

Élimination de Gauss : factorisation $A = LU$

On a pour toute ligne i :

$$\underline{A}_i = \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(n)} = l_{i,1} \underline{A}_1^{(n)} + l_{i,2} \underline{A}_2^{(n)} + \cdots + l_{i,i-1} \underline{A}_{i-1}^{(n)} + \underline{A}_i^{(n)}$$

Élimination de Gauss : factorisation $A = LU$

On a pour toute ligne i :

$$\begin{aligned}\underline{A}_i &= \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(n)} = l_{i,1} \underline{A}_1^{(n)} + l_{i,2} \underline{A}_2^{(n)} + \cdots + l_{i,i-1} \underline{A}_{i-1}^{(n)} + \underline{A}_i^{(n)} \\ &= \begin{bmatrix} l_{i,1} & l_{i,2} & \cdots & l_{i,i-1} & 1 \end{bmatrix} \times \begin{bmatrix} \underline{A}_1^{(n)} \\ \underline{A}_2^{(n)} \\ \cdots \\ \underline{A}_{i-1}^{(n)} \\ \underline{A}_i^{(n)} \end{bmatrix}\end{aligned}$$

Élimination de Gauss : factorisation $A = LU$

On a pour toute ligne i :

$$\begin{aligned}\underline{A}_i &= \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(n)} = l_{i,1} \underline{A}_1^{(n)} + l_{i,2} \underline{A}_2^{(n)} + \cdots + l_{i,i-1} \underline{A}_{i-1}^{(n)} + \underline{A}_i^{(n)} \\ &= \begin{bmatrix} l_{i,1} & l_{i,2} & \cdots & l_{i,i-1} & 1 \end{bmatrix} \times \begin{bmatrix} \underline{A}_1^{(n)} \\ \underline{A}_2^{(n)} \\ \cdots \\ \underline{A}_{i-1}^{(n)} \\ \underline{A}_i^{(n)} \end{bmatrix} \\ &= \begin{bmatrix} l_{i,1} & l_{i,2} & \cdots & l_{i,i-1} & 1 & 0 & \cdots & 0 \end{bmatrix} \times \begin{bmatrix} \underline{A}_1^{(n)} \\ \underline{A}_2^{(n)} \\ \cdots \\ \underline{A}_{i-1}^{(n)} \\ \underline{A}_i^{(n)} \\ \underline{A}_{i+1}^{(n)} \\ \cdots \\ \underline{A}_n^{(n)} \end{bmatrix}\end{aligned}$$

Élimination de Gauss : factorisation $A = LU$

On a pour toute ligne i :

$$\begin{aligned}\underline{A}_i &= \underline{A}_i^{(n)} + \sum_{j=1}^{i-1} l_{i,j} \underline{A}_j^{(n)} = l_{i,1} \underline{A}_1^{(n)} + l_{i,2} \underline{A}_2^{(n)} + \dots + l_{i,i-1} \underline{A}_{i-1}^{(n)} + \underline{A}_i^{(n)} \\ &= \begin{bmatrix} l_{i,1} & l_{i,2} & \dots & l_{i,i-1} & 1 \end{bmatrix} \times \begin{bmatrix} \underline{A}_1^{(n)} \\ \underline{A}_2^{(n)} \\ \dots \\ \underline{A}_{i-1}^{(n)} \\ \underline{A}_i^{(n)} \end{bmatrix} \\ &= \begin{bmatrix} l_{i,1} & l_{i,2} & \dots & l_{i,i-1} & 1 & 0 & \dots & 0 \end{bmatrix} \times \begin{bmatrix} \underline{A}_1^{(n)} \\ \underline{A}_2^{(n)} \\ \dots \\ \underline{A}_{i-1}^{(n)} \\ \underline{A}_i^{(n)} \\ \underline{A}_{i+1}^{(n)} \\ \dots \\ \underline{A}_n^{(n)} \end{bmatrix} \\ &= \underline{L}_i \times \underline{A}^{(n)} \quad \text{et } \underline{L}_i \text{ nulle à droite du terme de la diagonale...}\end{aligned}$$

Élimination de Gauss : factorisation $A = LU$

Conclusion

$$\underline{A}_i = \underline{L}_i \times A^n \quad \forall i = 1, \dots, n \quad \iff \quad A = L \times A^n = LU,$$

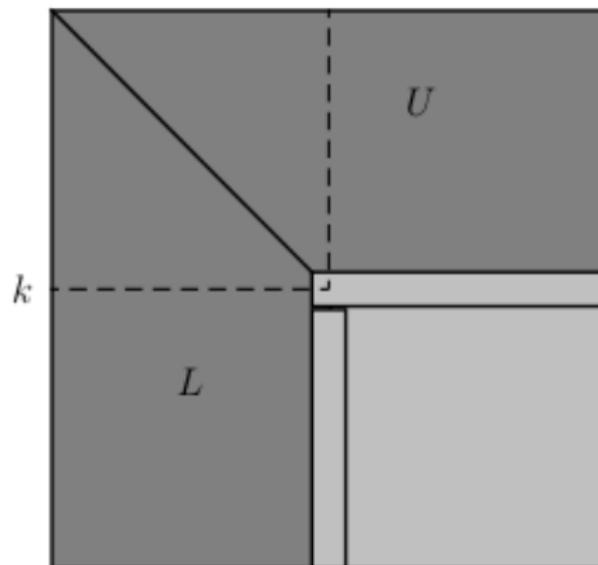
à condition que les pivots $a_{k,k}^{(k)}$ soient tous non nuls.

La matrice L est triangulaire inférieure avec des 1 sur la diagonale.

La matrice U est triangulaire supérieure avec les pivots $a_{k,k}^{(k)}$ sur la diagonale.

Attention : les pivots ne sont pas les termes diagonaux : $a_{k,k}^{(k)} \neq a_{k,k}$.

Élimination de Gauss : aspects pratiques



Note : en pratique, on ne calcule pas les termes qui sont nuls. On ne modifie que la partie carrée en bas à droite de taille $n - k$.

En pratique, on n'a besoin que d'une seule matrice (on utilise l'espace mémoire de A , qui est écrasée). On stocke L dans la partie inférieure stricte (les 1 sur la diagonale ne sont pas stockés) et on stocke U dans la partie supérieure.

Théorème (Existence et unicité de la factorisation LU)

Soit A inversible. Si les pivots sont tous non-nuls pendant l'élimination de Gauss, alors il existe une matrice L triangulaire inférieure et une matrice U triangulaire supérieure inversibles, telles que $A = LU$. De plus, si la diagonale de L est constituée de 1, alors la factorisation est unique.

Démonstration : existence : faite jusqu'ici.

Unicité : supposons que $A = LU = \tilde{L}\tilde{U}$, telles que L et \tilde{L} sont tri-inf avec $\text{diag}(L) = \text{diag}(\tilde{L}) = 1$, et U et \tilde{U} sont tri-sup.

L , \tilde{L} , U et \tilde{U} sont inversibles car $\det(A)$ est non nul.

Donc $LU = \tilde{L}\tilde{U} \iff U = L^{-1}\tilde{L}\tilde{U} \iff U\tilde{U}^{-1} = L^{-1}\tilde{L}$.

(Attention, le produit matriciel n'est pas commutatif!)

$U\tilde{U}^{-1}$ est tri-sup, $L^{-1}\tilde{L}$ est tri-inf, donc $U\tilde{U}^{-1} = L^{-1}\tilde{L}$ est diagonale.

De plus $\text{diag}(L^{-1}) = \text{diag}(\tilde{L}) = 1$, donc $L^{-1}\tilde{L} = I$ et $L = \tilde{L}$. Et donc $U = \tilde{U}$.

Il y a unicité.

Lemme

Soient A et B deux matrices triangulaires supérieures.

Alors le produit $C = AB$ est triangulaire supérieur.

De plus $c_{i,j} = a_{i,j}b_{i,j}$ pour $i = 1, \dots, n$.

Si A est inversible, alors l'inverse $D = A^{-1}$ est triangulaire supérieure.

De plus, $d_{i,i} = \frac{1}{a_{i,i}}$ pour $i = 1, \dots, n$.

La proposition similaire pour les matrices tri-inf est également vraie.

Lemme

Soient A et B deux matrices triangulaires supérieures.

Alors le produit $C = AB$ est triangulaire supérieur.

De plus $c_{i,j} = a_{i,i}b_{i,j}$ pour $i = 1, \dots, n$.

Si A est inversible, alors l'inverse $D = A^{-1}$ est triangulaire supérieure.

De plus, $d_{i,i} = \frac{1}{a_{i,i}}$ pour $i = 1, \dots, n$.

La proposition similaire pour les matrices tri-inf est également vraie.

Voir l'exercice Chapitre 0, TD 11 (Exercice B.1.11 TD0-Exercice11).

Montrer que si A et B sont tri-sup, alors :

$$C_{i,j} = \sum_{k=i}^j A_{i,k} B_{k,j} \quad \forall i, j = 1, \dots, n, \quad (\text{et } C_{i,j} = 0 \text{ si } i > j.)$$

Convention : une somme faite sur un ensemble d'indice vide est nulle.

Trouver la formule similaire pour A et B tri-inf.

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle**
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

Factorisation LU directe : principe

Unicité \implies calcul direct par identification.

Si possible, c'est **LA** factorisation $A = LU$.

[Lien vers prod mat.](#)

$$\begin{aligned} A = LU & \iff \{ \underline{A}_i = \underline{L}_i U \quad \forall i = 1, \dots, n \\ & \iff \{ A_j = L U_j \quad \forall j = 1, \dots, n \\ & \iff \left\{ A_{i,j} = \sum_{k=1}^{\min(i,j)} L_{i,k} U_{k,j} \quad \forall i, j = 1, \dots, n \right. \end{aligned}$$

Factorisation LU directe : principe

Unicité \implies calcul direct par identification.

Si possible, c'est **LA** factorisation $A = LU$.

[Lien vers prod mat.](#)

$$\begin{aligned} A = LU &\iff \left\{ \begin{array}{l} \underline{A}_i = \underline{L}_i U \quad \forall i = 1, \dots, n \\ \underline{A}_j = L U_j \quad \forall j = 1, \dots, n \end{array} \right. \\ &\iff \left\{ \begin{array}{l} A_{i,j} = \sum_{k=1}^{\min(i,j)} L_{i,k} U_{k,j} \quad \forall i, j = 1, \dots, n \end{array} \right. \end{aligned}$$

Donc principe : identifier en alternant les lignes et les colonnes de A .

Calcul matriciel ligne / colonne

Si $A \in \mathcal{M}_{p,n}$, $x \in \mathbb{R}^n$ et $y \in \mathbb{R}^p$, on a :

$$\begin{aligned} Ax &= A_1 x_1 + A_2 x_2 + \dots + A_n x_n \\ y^T A &= y_1 \underline{A}_1 + y_2 \underline{A}_2 + \dots + y_p \underline{A}_p \end{aligned}$$

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

- Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 U = [1 \ 0 \ 0] U = \underline{U}_1 \implies \underline{U}_1 = \underline{A}_1$ connu.

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

- Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 \underline{U} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \underline{U} = \underline{U}_1 \quad \Rightarrow \underline{U}_1 = \underline{A}_1$ connu.
- Colonne $j = 1$: $\underline{A}_1 = \underline{L} \underline{U}_1 = \underline{L} \begin{bmatrix} u_{1,1} \\ 0 \\ 0 \end{bmatrix} = L_1 u_{1,1} \quad \Rightarrow L_1 = \frac{\underline{A}_1}{u_{1,1}}$ connu.

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

• Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 \underline{U} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \underline{U} = \underline{U}_1 \quad \Rightarrow \underline{U}_1 = \underline{A}_1$ connu.

• Colonne $j = 1$: $\underline{A}_1 = \underline{L} \underline{U}_1 = \underline{L} \begin{bmatrix} u_{1,1} \\ 0 \\ 0 \end{bmatrix} = \underline{L}_1 u_{1,1} \quad \Rightarrow \underline{L}_1 = \frac{\underline{A}_1}{u_{1,1}}$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix}$$

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

• Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 \underline{U} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \underline{U} = \underline{U}_1 \implies \underline{U}_1 = \underline{A}_1$ connu.

• Colonne $j = 1$: $\underline{A}_1 = \underline{L} \underline{U}_1 = \underline{L} \begin{bmatrix} u_{1,1} \\ 0 \\ 0 \end{bmatrix} = \underline{L}_1 u_{1,1} \implies \underline{L}_1 = \frac{\underline{A}_1}{u_{1,1}}$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix}$$

• Ligne $i = 2$: $\underline{A}_2 = \underline{L}_2 \underline{U} = \begin{bmatrix} l_{2,1} & 1 & 0 \end{bmatrix} \underline{U} = l_{2,1} \underline{U}_1 + \underline{U}_2 \implies \underline{U}_2 = \underline{A}_2 - l_{2,1} \underline{U}_1$ connu.

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

• Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 \underline{U} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \underline{U} = \underline{U}_1 \implies \underline{U}_1 = \underline{A}_1$ connu.

• Colonne $j = 1$: $\underline{A}_1 = \underline{L} \underline{U}_1 = \underline{L} \begin{bmatrix} u_{1,1} \\ 0 \\ 0 \end{bmatrix} = L_1 u_{1,1} \implies L_1 = \frac{\underline{A}_1}{u_{1,1}}$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix}$$

• Ligne $i = 2$: $\underline{A}_2 = \underline{L}_2 \underline{U} = \begin{bmatrix} l_{2,1} & 1 & 0 \end{bmatrix} \underline{U} = l_{2,1} \underline{U}_1 + \underline{U}_2 \implies \underline{U}_2 = \underline{A}_2 - l_{2,1} \underline{U}_1$ connu.

• Colonne $j = 2$: $\underline{A}_2 = \underline{L} \underline{U}_2 = \underline{L} \begin{bmatrix} u_{1,2} \\ u_{2,2} \\ 0 \end{bmatrix} = L_1 u_{1,2} + L_2 u_{2,2} \implies L_2 = \frac{1}{u_{2,2}} (\underline{A}_2 - L_1 u_{1,2})$ connu.

Factorisation LU directe : exemple 3×3

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \times \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix} = LU???$$

• Ligne $i = 1$: identifier $\underline{A}_1 = \underline{L}_1 \underline{U} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \underline{U} = \underline{U}_1 \implies \underline{U}_1 = \underline{A}_1$ connu.

• Colonne $j = 1$: $\underline{A}_1 = \underline{L} \underline{U}_1 = \underline{L} \begin{bmatrix} u_{1,1} \\ 0 \\ 0 \end{bmatrix} = L_1 u_{1,1} \implies L_1 = \frac{\underline{A}_1}{u_{1,1}}$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & * & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & * & * \\ 0 & 0 & * \end{bmatrix}$$

• Ligne $i = 2$: $\underline{A}_2 = \underline{L}_2 \underline{U} = \begin{bmatrix} l_{2,1} & 1 & 0 \end{bmatrix} \underline{U} = l_{2,1} \underline{U}_1 + \underline{U}_2 \implies \underline{U}_2 = \underline{A}_2 - l_{2,1} \underline{U}_1$ connu.

• Colonne $j = 2$: $\underline{A}_2 = \underline{L} \underline{U}_2 = \underline{L} \begin{bmatrix} u_{1,2} \\ u_{2,2} \\ 0 \end{bmatrix} = L_1 u_{1,2} + L_2 u_{2,2} \implies L_2 = \frac{1}{u_{2,2}} (\underline{A}_2 - L_1 u_{1,2})$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & * \end{bmatrix}$$

Factorisation LU directe : exemple 3×3

- Ligne $i = 3$: $\underline{A}_3 = [l_{3,1} \quad l_{3,2} \quad 1] \underline{U} = l_{3,1}\underline{U}_1 + l_{3,2}\underline{U}_2 + \underline{U}_3$
 $\implies \underline{U}_3 = \underline{A}_3 - l_{3,1}\underline{U}_1 - l_{3,2}\underline{U}_2$ connu.

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & -9 \end{bmatrix}$$

Note : en pratique, on ne calcule pas les termes qui doivent être nuls (parties triangulaires nulles de L et de U) ou égaux à 1 (diagonale de L).

Pour que les calculs soient possibles, il faut que les $u_{i,i}$ soient non nuls. Ce sont les pivots de l'élimination de Gauss.

Factorisation LU directe : cas général

Récurrance :

Supposons que $\underline{U}_1, L_1, \underline{U}_2, L_2, \dots, \underline{U}_{k-1}$ et L_{k-1} sont connus. Alors

- Ligne $i = k$:

$$\begin{aligned}\underline{A}_k &= \underline{L}_k \underline{U} = \begin{bmatrix} l_{k,1} & \dots & l_{k,k-1} & 1 & 0 & \dots & 0 \end{bmatrix} \underline{U} \\ &= l_{k,1} \underline{U}_1 + \dots + l_{k,k-1} \underline{U}_{k-1} + \underline{U}_k \\ \Rightarrow \underline{U}_k &= \underline{A}_k - \sum_{j=1}^{k-1} l_{k,j} \underline{U}_j \quad \text{connu}\end{aligned}$$

On ne modifie que les termes $j = k$ à n (\underline{U} est tri-sup).

Factorisation LU directe : cas général

Récurrance :

Supposons que $\underline{U}_1, L_1, \underline{U}_2, L_2, \dots, \underline{U}_{k-1}$ et L_{k-1} sont connus. Alors

- Ligne $i = k$:

$$\begin{aligned} \underline{A}_k &= \underline{L}_k \underline{U} = \begin{bmatrix} l_{k,1} & \dots & l_{k,k-1} & 1 & 0 & \dots & 0 \end{bmatrix} \underline{U} \\ &= l_{k,1} \underline{U}_1 + \dots + l_{k,k-1} \underline{U}_{k-1} + \underline{U}_k \\ \implies \underline{U}_k &= \underline{A}_k - \sum_{j=1}^{k-1} l_{k,j} \underline{U}_j \quad \text{connu} \end{aligned}$$

On ne modifie que les termes $j = k$ à n (\underline{U} est tri-sup).

- Colonne $j = k$: si $u_{k,k} \neq 0$

$$\begin{aligned} \underline{A}_k &= \underline{L} \underline{U}_k = \underline{L} \begin{bmatrix} u_{1,k} & \dots & u_{k-1,k} & u_{k,k} & 0 & \dots & 0 \end{bmatrix}^T \\ &= L_1 u_{1,k} + \dots + L_{k-1} u_{k-1,k} + L_k u_{k,k} \\ \implies L_k &= \frac{1}{u_{k,k}} \left(\underline{A}_k - \sum_{j=1}^{k-1} L_j u_{j,k} \right) \quad \text{connu} \end{aligned}$$

Ne modifier que les termes $i = k + 1$ à n (\underline{L} est tri-inf et $L_{i,i} = 1$).

Factorisation LU directe : cas général

Récurrance :

Supposons que $\underline{U}_1, L_1, \underline{U}_2, L_2, \dots, \underline{U}_{k-1}$ et L_{k-1} sont connus. Alors

- Ligne $i = k$:

$$\begin{aligned} \underline{A}_k &= \underline{L}_k \underline{U} = \begin{bmatrix} l_{k,1} & \dots & l_{k,k-1} & 1 & 0 & \dots & 0 \end{bmatrix} \underline{U} \\ &= l_{k,1} \underline{U}_1 + \dots + l_{k,k-1} \underline{U}_{k-1} + \underline{U}_k \\ \implies \underline{U}_k &= \underline{A}_k - \sum_{j=1}^{k-1} l_{k,j} \underline{U}_j \quad \text{connu} \end{aligned}$$

On ne modifie que les termes $j = k$ à n (\underline{U} est tri-sup).

- Colonne $j = k$: si $u_{k,k} \neq 0$

$$\begin{aligned} \underline{A}_k &= \underline{L} \underline{U}_k = \underline{L} \begin{bmatrix} u_{1,k} & \dots & u_{k-1,k} & u_{k,k} & 0 & \dots & 0 \end{bmatrix}^T \\ &= L_1 u_{1,k} + \dots + L_{k-1} u_{k-1,k} + L_k u_{k,k} \\ \implies L_k &= \frac{1}{u_{k,k}} \left(\underline{A}_k - \sum_{j=1}^{k-1} L_j u_{j,k} \right) \quad \text{connu} \end{aligned}$$

Ne modifier que les termes $i = k + 1$ à n (\underline{L} est tri-inf et $L_{i,i} = 1$).

Factorisation LU : algorithme de Doolittle

Algorithme de Doolittle (sans écraser A)

Initialiser L à I_n et U à 0 dans $\mathcal{M}_{n,n}$.

Pour $k=1$ jusqu'à n

 Pour $j=k$ jusqu'à n

$$U(k, j) \leftarrow A(k, j) - \sum_{p=1}^{k-1} L(k, p) \times U(p, j)$$

 Fin Pour

 Si $|U(k, k)| < \text{tol}$

 Sortir, erreur : pivot nul

 Fin Si

 Pour $i=k+1$ jusqu'à n (sauté pour $i = n$)

$$L(i, k) \leftarrow 1/U(k, k) \times \left(A(i, k) - \sum_{p=1}^{k-1} L(i, p) \times U(p, k) \right)$$

 Fin Pour

Fin Pour

Coût identique à Gauss : $\sim \frac{n^3}{3}$ multiplications.

(À montrer, Chap 2, Exo TD n° 3.).

Plan

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage**
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

Pivot nul : ?

Que faire si un pivot est nul ? Exemple : $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ inversible, mais

$$a_{1,1}^{(1)} = a_{1,1} = 0!$$

→ permuter des lignes de A (ou des colonnes, inconnues x).

Définition

Soit $A \in \mathcal{M}_{n,n}$.

La sous matrice principale d'ordre $k \leq n$ de A notée $[A]_k$, est définie par :

$$[A]_k = (a_{i,j})_{i=1,\dots,k;j=1,\dots,k}$$

Exemple

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 5 & 3 \\ -1 & 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & -9 \end{bmatrix} \quad \begin{cases} \det([A]_1) = 1 \\ \det([A]_2) = 1, \\ \det([A]_3) = -9. \end{cases}$$

Lemma (matrices par blocs)

Soit une matrice carrée, triangulaire supérieure *par blocs*. Alors son déterminant vaut :

$$\det \left(\begin{bmatrix} A & B \\ 0 & C \end{bmatrix} \right) = \det(A) \times \det(C), \text{ et } \det \left(\begin{bmatrix} D & 0 \\ E & F \end{bmatrix} \right) = \det(D) \times \det(F).$$

Démonstration : extension de Chap 0, exo TD n° 12.

Remarque : tri-inf par bloc \neq tri-inf. (idem pour tri-sup). Prenez un exemple !

Lemma (matrices par blocs)

Soit une matrice carrée, triangulaire supérieure *par blocs*. Alors son déterminant vaut :

$$\det \left(\begin{bmatrix} A & B \\ 0 & C \end{bmatrix} \right) = \det(A) \times \det(C), \text{ et } \det \left(\begin{bmatrix} D & 0 \\ E & F \end{bmatrix} \right) = \det(D) \times \det(F).$$

Démonstration : extension de Chap 0, exo TD n° 12.

Remarque : tri-inf par bloc \neq tri-inf. (idem pour tri-sup). Prenez un exemple !

$$\det \left(\begin{bmatrix} A & B \\ C & D \end{bmatrix} \right)$$

Lemma (matrices par blocs)

Soit une matrice carrée, triangulaire supérieure *par blocs*. Alors son déterminant vaut :

$$\det \left(\begin{bmatrix} A & B \\ 0 & C \end{bmatrix} \right) = \det(A) \times \det(C), \text{ et } \det \left(\begin{bmatrix} D & 0 \\ E & F \end{bmatrix} \right) = \det(D) \times \det(F).$$

Démonstration : extension de Chap 0, exo TD n° 12.

Remarque : tri-inf par bloc \neq tri-inf. (idem pour tri-sup). Prenez un exemple !

$$\det \left(\begin{bmatrix} A & B \\ C & D \end{bmatrix} \right) \neq \det(A) \det(D) - \det(C) \det(B)$$

Attention : ARCHI faux ! car B et C sont rectangulaires en général !!

Théorème (2.4.2)

Les trois propositions suivantes sont équivalentes :

- Proposition 1 :

Tous les pivots $a_{kk}^{(k)}$ sont définis et $a_{kk}^{(k)} \neq 0$, $k = 1, \dots, n$.

- Proposition 2 :

A admet une factorisation $A = LU$ avec U inversible .

- Proposition 3 :

$[A]_1, [A]_2, \dots, [A]_n$ inversibles.

Théorème (2.4.2)

Les trois propositions suivantes sont équivalentes :

- Proposition 1 :

Tous les pivots $a_{kk}^{(k)}$ sont définis et $a_{kk}^{(k)} \neq 0$, $k = 1, \dots, n$.

- Proposition 2 :

A admet une factorisation $A = LU$ avec U inversible .

- Proposition 3 :

$[A]_1, [A]_2, \dots, [A]_n$ inversibles.

Prop. 3 : critère théorique

→ Prop. 3 vraie pour certaines matrices : symétrique définie positive (SDP), à diagonale strictement dominante (voir plus tard et Exo5 TD)

→ En pratique : on ne calcule pas les $\det([A]_k)$, on calcule les pivots et on vérifie au fur et à mesure s'ils sont nuls ou pas.

Démonstration :

- On a déjà prouvé Prop. 1 \implies Prop. 2 :

élimination de Gauss faisable jusqu'au bout

\iff les pivots existent et sont tous non nuls

\implies factorisation $A = LU$ faisable (et $\det(U) = \prod_{i=1}^n u_{i,i} = \prod_{i=1}^n a_{i,i}^{(i)} \neq 0$).

Démonstration :

- On a déjà prouvé Prop. 1 \implies Prop. 2 :
élimination de Gauss faisable jusqu'au bout
 \iff les pivots existent et sont tous non nuls
 \implies factorisation $A = LU$ faisable (et $\det(U) = \prod_{i=1}^n u_{i,i} = \prod_{i=1}^n a_{i,i}^{(i)} \neq 0$).
- Prop. 2 \implies Prop. 3 : soit $1 \leq k \leq n$. Décomposition de $A = LU$ en blocs de tailles k et $(n - k)$:

$$A = \begin{bmatrix} [A]_k & B \\ C & D \end{bmatrix} = \begin{bmatrix} [L]_k & 0 \\ E & F \end{bmatrix} \times \begin{bmatrix} [U]_k & G \\ 0 & H \end{bmatrix}$$

Démonstration :

- On a déjà prouvé Prop. 1 \implies Prop. 2 :
élimination de Gauss faisable jusqu'au bout
 \iff les pivots existent et sont tous non nuls
 \implies factorisation $A = LU$ faisable (et $\det(U) = \prod_{i=1}^n u_{i,i} = \prod_{i=1}^n a_{i,i}^{(i)} \neq 0$).
- Prop. 2 \implies Prop. 3 : soit $1 \leq k \leq n$. Décomposition de $A = LU$ en blocs de tailles k et $(n - k)$:

$$\begin{aligned} A = \begin{bmatrix} [A]_k & B \\ C & D \end{bmatrix} &= \begin{bmatrix} [L]_k & 0 \\ E & F \end{bmatrix} \times \begin{bmatrix} [U]_k & G \\ 0 & H \end{bmatrix} \\ &= \begin{bmatrix} [L]_k [U]_k & [L]_k G \\ E [U]_k & EG + FH \end{bmatrix} \end{aligned}$$

En identifiant : $[A]_k = [L]_k [U]_k$. Comme $0 \neq \det(U) = \det([U]_k) \det(H)$ et $1 = \det(L) = \det([L]_k) \det(F)$, on a $\det([A]_k) \neq 0$.

Donc $[A]_k$ est inversible.

On a $\det([A]_k) = 1 \times \det([U]_k) = \prod_{i=1}^k u_{i,i}$ (produit des k premiers pivots.)

Démonstration :

- On a déjà prouvé Prop. 1 \implies Prop. 2 :
élimination de Gauss faisable jusqu'au bout
 \iff les pivots existent et sont tous non nuls
 \implies factorisation $A = LU$ faisable (et $\det(U) = \prod_{i=1}^n u_{i,i} = \prod_{i=1}^n a_{i,i}^{(i)} \neq 0$).
- Prop. 2 \implies Prop. 3 : soit $1 \leq k \leq n$. Décomposition de $A = LU$ en blocs de tailles k et $(n - k)$:

$$\begin{aligned} A = \begin{bmatrix} [A]_k & B \\ C & D \end{bmatrix} &= \begin{bmatrix} [L]_k & 0 \\ E & F \end{bmatrix} \times \begin{bmatrix} [U]_k & G \\ 0 & H \end{bmatrix} \\ &= \begin{bmatrix} [L]_k [U]_k & [L]_k G \\ E [U]_k & EG + FH \end{bmatrix} \end{aligned}$$

En identifiant : $[A]_k = [L]_k [U]_k$. Comme $0 \neq \det(U) = \det([U]_k) \det(H)$ et $1 = \det(L) = \det([L]_k) \det(F)$, on a $\det([A]_k) \neq 0$.

Donc $[A]_k$ est inversible.

On a $\det([A]_k) = 1 \times \det([U]_k) = \prod_{i=1}^k u_{i,i}$ (produit des k premiers pivots.)

- Prop. 3 \implies Prop. 1 : voir le poly (Doc. B.1.1).

Permutation de ligne si pivot nul.

Soit A inversible.

Si $(k-1)$ étapes faisables sans permutations ($a_{i,i}^{(j)} \neq 0, i = 1, \dots, k-1$) :

$$A^{(k)} = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & \cdots & a_{1,k-1}^{(1)} & a_{1,k}^{(1)} & a_{1,k+1}^{(1)} & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & \cdots & a_{2,k-1}^{(2)} & a_{2,k}^{(2)} & a_{2,k+1}^{(2)} & \cdots & a_{2,n}^{(2)} \\ \vdots & \vdots & & \ddots & \ddots & \vdots & & \vdots \\ 0 & 0 & \cdots & a_{k-1,k-1}^{(k-1)} & a_{k-1,k}^{(k-1)} & a_{k-1,k+1}^{(k-1)} & \cdots & a_{k-1,n}^{(k-1)} \\ 0 & 0 & \cdots & 0 & a_{k,k}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{k,n}^{(k)} \\ 0 & 0 & \cdots & 0 & a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & a_{n,k}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{n,n}^{(k)} \end{bmatrix}$$

Écriture par blocs : $A^{(k)} = \begin{bmatrix} [U]_{k-1} & D^{(k)} \\ 0 & S^{(k)} \end{bmatrix}$ avec $S^{(k)} \in \mathcal{M}_{n-k+1, n-k+1}$

Pas de permutation, donc $0 \neq \det(A) = \det(A^{(k)}) = \prod_{i=1}^{k-1} a_{i,i}^{(i)} \times \det(S^{(k)})$.

Donc $S^{(k)}$ inversible, donc colonne $S_1^{(k)}$ non nulle :

\implies il existe une ligne $i_0 \in \{k, k+1, \dots, n\}$ telle que $a_{i_0,k}^{(k)} \neq 0$.

Permutation de lignes si pivot nul.

Après $(k - 1)$ étapes, il existe une ligne $i_0 \in \{k, k + 1, \dots, n\}$ telle que $a_{i_0, k}^{(k)} \neq 0$.

Donc si $a_{k, k}^{(k)} = 0$, on permute les lignes i_0 et k : $A^{(k)} \rightarrow A^{(k)'}$

$\Rightarrow a_{k, k}^{(k)'} = a_{i_0, k}^{(k)} \neq 0$ nouveau pivot

$\Rightarrow \det(A^{(k)'}) = -\det(A^{(k)})$ (car 1 permutation).

Proposition (2.4.1)

Soit A une matrice inversible. À toute étape k de la méthode d'élimination de Gauss, il existe au moins un élément non nul dans la colonne k et situé sur la ligne i avec $i \geq k$.

Théorème (2.4.4)

Soit A une matrice inversible. Alors il existe une matrice triangulaire inférieure L , une matrice triangulaire supérieure U , et une matrice de permutation P telles que

$$PA = LU.$$

- P matrice de permutation : que des 0, un seul 1 par ligne et par colonne.
 $\det(P) = \pm 1$.
- La décomposition $PA = LU$ n'est **pas unique** ! (choix de plusieurs i_0 en général.)
- Démonstration (hors programme) : truc "magique" dans l'échange de lignes.

Permutation de lignes : factorisation $PA = LU$.

Théorème (2.4.4)

Soit A une matrice inversible. Alors il existe une matrice triangulaire inférieure L , une matrice triangulaire supérieure U , et une matrice de permutation P telles que

$$PA = LU.$$

- P matrice de permutation : que des 0, un seul 1 par ligne et par colonne.
 $\det(P) = \pm 1$.
- La décomposition $PA = LU$ n'est **pas unique** ! (choix de plusieurs i_0 en général.)
- Démonstration (hors programme) : truc "magique" dans l'échange de lignes.
- En pratique : on fait des permutations à chaque étape k pour améliorer la précision.
On choisit **le plus grand pivot possible** en valeur absolue.
→ prendre $i_0 \in \{k, k+1, \dots, n\}$ tel que $|a_{i_0, k}^{(k)}| \geq |a_{i, k}^{(k)}|$ pour $i \in \{k, k+1, \dots, n\}$.
- Algorithme de Gauss **avec** permutation : **stable** par rapport aux erreurs d'arrondis (erreurs croissent pas trop vite). (Non démontré).
- Algorithme de Gauss **sans** permutation : **pas stable**.
Voir exemple, TD2-Exercice 6 (Exo C.2.6).

Instabilité de Gauss sans permutation : exemple

Algorithme de Gauss **sans** permutation : **instable**.

Exemple : **sans** permutation avec un pivot "petit" :

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \begin{array}{l} \text{arithm. exacte} \\ \text{arithm. flottante} \end{array} \Rightarrow \begin{array}{l} x = \begin{bmatrix} \frac{1}{1-\varepsilon} \\ \frac{1-2\varepsilon}{1-\varepsilon} \end{bmatrix} \sim \begin{bmatrix} 1+\varepsilon \\ 1-\varepsilon \end{bmatrix} \\ \tilde{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{si } fl(1-\varepsilon) = 1. \end{array}$$

Résultat très mauvais.

Instabilité de Gauss sans permutation : exemple

Algorithme de Gauss **sans** permutation : **instable**.

Exemple : **sans** permutation avec un pivot "petit" :

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \begin{array}{l} \text{arithm. exacte} \\ \text{arithm. flottante} \end{array} \Rightarrow \begin{array}{l} x = \begin{bmatrix} \frac{1}{1-\varepsilon} \\ \frac{1-2\varepsilon}{1-\varepsilon} \end{bmatrix} \sim \begin{bmatrix} 1+\varepsilon \\ 1-\varepsilon \end{bmatrix} \\ \tilde{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{si } fl(1-\varepsilon) = 1. \end{array}$$

Résultat très mauvais.

Avec permutation :

$$\begin{bmatrix} 1 & 1 \\ \varepsilon & 1 \end{bmatrix} \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \text{arithm. flottante} \Rightarrow \tilde{y} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{si } fl(1-\varepsilon) = fl(1-2\varepsilon) = 1.$$

Résultat précis.

Instabilité de Gauss sans permutation : exemple

Algorithme de Gauss **sans** permutation : **instable**.

Exemple : **sans** permutation avec un pivot "petit" :

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \begin{array}{l} \text{arithm. exacte} \\ \text{arithm. flottante} \end{array} \Rightarrow \begin{array}{l} x = \begin{bmatrix} \frac{1}{1-\varepsilon} \\ \frac{1-2\varepsilon}{1-\varepsilon} \end{bmatrix} \sim \begin{bmatrix} 1+\varepsilon \\ 1-\varepsilon \end{bmatrix} \\ \tilde{x} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{si } fl(1-\varepsilon) = 1. \end{array}$$

Résultat très mauvais.

Avec permutation :

$$\begin{bmatrix} 1 & 1 \\ \varepsilon & 1 \end{bmatrix} \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \text{arithm. flottante} \Rightarrow \tilde{y} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{si } fl(1-\varepsilon) = fl(1-2\varepsilon) = 1.$$

Résultat précis.

Conclusion : en pratique, il faut toujours permuter (→ grands pivots !)

Plan

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky**
- 6 Normes matricielles
- 7 Conditionnement

Proposition

Soit A une matrice dont toutes les sous-matrices principales sont régulières. Alors elle admet une unique factorisation sous la forme

$$A = LDV,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière
- V est une matrice triangulaire supérieure à diagonale unité.

Proposition

Soit A une matrice dont toutes les sous-matrices principales sont régulières. Alors elle admet une unique factorisation sous la forme

$$A = LDV,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière
- V est une matrice triangulaire supérieure à diagonale unité.

Schéma de démonstration.

- 1 **Existence** : $A = LU$ existe.
 $D = \text{diag}(U) = \text{diag}((u_{i,i})_{i=1,\dots,n})$ inversible car pivots non nuls.
 $D^{-1} = \text{diag}((\frac{1}{u_{i,i}})_{i=1,\dots,n})$, donc $V = D^{-1}U$ est tri-sup à diagonale unité.

Proposition

Soit A une matrice dont toutes les sous-matrices principales sont régulières. Alors elle admet une unique factorisation sous la forme

$$A = LDV,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière
- V est une matrice triangulaire supérieure à diagonale unité.

Schéma de démonstration.

- 1 **Existence** : $A = LU$ existe.
 $D = \text{diag}(U) = \text{diag}((u_{i,i})_{i=1,\dots,n})$ inversible car pivots non nuls.
 $D^{-1} = \text{diag}((\frac{1}{u_{i,i}})_{i=1,\dots,n})$, donc $V = D^{-1}U$ est tri-sup à diagonale unité.
- 2 **Unicité** : découle de l'unicité de $A = LU$. Prendre $A = LDV = \tilde{L}\tilde{D}\tilde{V}$, et prouver que les matrices sont égales.

Proposition (2.5.1)

Soit A une matrice **symétrique** dont toutes les sous-matrices principales sont régulières.

Alors elle admet une unique factorisation sous la forme

$$A = LDL^T,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière.

Proposition (2.5.1)

Soit A une matrice **symétrique** dont toutes les sous-matrices principales sont régulières.

Alors elle admet une unique factorisation sous la forme

$$A = LDL^T,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière.

Démonstration. Factorisation $A = LDV$ existe et est unique.

A symétrique donc : $A = A^T \iff LDV = (LDV)^T \iff LDV = V^T D^T L^T = V^T D L^T$.

Par unicité $L = V^T$.

Proposition (2.5.1)

Soit A une matrice **symétrique** dont toutes les sous-matrices principales sont régulières.

Alors elle admet une unique factorisation sous la forme

$$A = LDL^T,$$

où

- L est une matrice triangulaire inférieure à diagonale unité,
- D une matrice diagonale régulière.

Démonstration. Factorisation $A = LDV$ existe et est unique.

A symétrique donc : $A = A^T \iff LDV = (LDV)^T \iff LDV = V^T D^T L^T = V^T D L^T$.

Par unicité $L = V^T$.

On ne stocke que la partie utile (tri-inf par exemple).

D : ne contient **pas** les valeurs propres de A !!

$A = LDL^T$ n'est **pas** une diagonalisation de A !!

Définition (matrices symétriques (semi-)définies positives)

Soit $A \in \mathcal{M}_{n,n}$ une matrice symétrique ($A^T = A$). Alors

$$\begin{aligned} A \text{ SDP} &\iff \begin{cases} \forall x \in \mathbb{R}^n : x^T A x \geq 0 \\ \forall x \in \mathbb{R}^n : x^T A x = 0 \implies x = 0 \end{cases} \\ &\iff \{ \forall x \in \mathbb{R}^n \setminus \{0\} : x^T A x > 0 \} \end{aligned}$$

Et :

$$A \text{ semi DP} \iff \{ \forall x \in \mathbb{R}^n : x^T A x \geq 0 \}$$

Définition (matrices symétriques (semi-)définies positives)

Soit $A \in \mathcal{M}_{n,n}$ une matrice symétrique ($A^T = A$). Alors

$$\begin{aligned} A \text{ SDP} &\iff \begin{cases} \forall x \in \mathbb{R}^n : x^T A x \geq 0 \\ \forall x \in \mathbb{R}^n : x^T A x = 0 \implies x = 0 \end{cases} \\ &\iff \{ \forall x \in \mathbb{R}^n \setminus \{0\} : x^T A x > 0 \end{aligned}$$

Et :

$$A \text{ semi DP} \iff \{ \forall x \in \mathbb{R}^n : x^T A x \geq 0 \}$$

Rappels :

- A symétrique \implies A diagonalisable dans une BON. Il existe P orthogonale ($P^T = P^{-1}$, P matrice de changement de BON) et Λ diagonale (contenant les vp réelles de A), telles que $\Lambda = P^T A P = P^{-1} A P$.
- A semi-DP : \implies les vp de A sont ≥ 0 , A SDP : \implies les vp de A sont > 0 .
- Décomposition en carrés de Gauss d'une forme quadratique.
- Voir exemple.

Formes quadratiques : rappels

Rappels :

- soit $A \in \mathcal{M}_{n,p}$, $x \in \mathbb{R}^n$ et $y \in \mathbb{R}^p$. Alors

$$x^T A y = \sum_{i=1}^n \sum_{j=1}^p x_i A_{i,j} y_j = \sum_{j=1}^p \sum_{i=1}^n y_j A_{i,j} x_i = y^T A^T x \quad \in \mathbb{R}.$$

- soit $A \in \mathcal{M}_{n,n}$ **symétrique**, x et $y \in \mathbb{R}^n$. Alors

$$x^T A y = \sum_{i=1}^n \sum_{j=1}^n x_i A_{i,j} y_j = \sum_{i=1}^n \sum_{j=1}^n x_i A_{j,i} y_j = \sum_{k=1}^n \sum_{l=1}^n y_k A_{k,l} x_l = y^T A x \quad \in \mathbb{R}.$$

et on définit une forme quadratique $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}$ par :

$$\Phi(x) = x^T A x = \sum_{i=1}^n \sum_{j=1}^n x_i A_{i,j} x_j = \sum_{i=1}^n A_{i,i} x_i^2 + 2 \sum_{1 \leq i < j \leq n} A_{i,j} x_i x_j.$$

- A semi-DP $\iff \Phi(x) \geq 0 \forall x \in \mathbb{R}^n$.
- A SDP $\iff \Phi(x) > 0 \forall x \in \mathbb{R}^n \setminus \{0\}$.

Formes quadratiques : exemple 3×3

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 9 \end{bmatrix}$$

\implies

$$\begin{aligned} \Phi(x) &= x^\top Ax = \begin{cases} 1x_1^2 + 1x_1x_2 + 2x_1x_3 \\ +1x_2x_1 + 2x_2^2 + 1x_2x_3 \\ +2x_3x_1 + 1x_3x_2 + 9x_3^2 \end{cases} \\ &= 1x_1^2 + 2x_2^2 + 9x_3^2 \\ &\quad + 2(1x_1x_2 + 2x_1x_3 + 1x_2x_3) \end{aligned}$$

Formes quadratiques : exemple 3×3

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 9 \end{bmatrix} \implies \Phi(x) = x^T A x = \begin{cases} 1x_1^2 + 1x_1x_2 + 2x_1x_3 \\ +1x_2x_1 + 2x_2^2 + 1x_2x_3 \\ +2x_3x_1 + 1x_3x_2 + 9x_3^2 \end{cases}$$
$$= 1x_1^2 + 2x_2^2 + 9x_3^2 + 2(1x_1x_2 + 2x_1x_3 + 1x_2x_3)$$

Décomposition de Gauss en carrés :

$$\begin{cases} x_1^2 + 2\alpha x_1 y & = (x_1 + \alpha y)^2 - \alpha^2 y^2 \\ x_1 x_2 + x_1 y_1 + x_2 y_2 & = (x_1 + y_2)(x_2 + y_1) - y_1 y_2 \\ l_1 l_2 & = \frac{1}{4} ((l_1 + l_2)^2 - (l_1 - l_2)^2). \end{cases}$$

Formes quadratiques : exemple 3×3

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 9 \end{bmatrix} \implies \Phi(x) = x^T A x = \begin{cases} 1x_1^2 + 1x_1x_2 + 2x_1x_3 \\ +1x_2x_1 + 2x_2^2 + 1x_2x_3 \\ +2x_3x_1 + 1x_3x_2 + 9x_3^2 \end{cases}$$
$$= 1x_1^2 + 2x_2^2 + 9x_3^2 + 2(1x_1x_2 + 2x_1x_3 + 1x_2x_3)$$

Décomposition de Gauss en carrés :

$$\begin{cases} x_1^2 + 2\alpha x_1 y & = (x_1 + \alpha y)^2 - \alpha^2 y^2 \\ x_1 x_2 + x_1 y_1 + x_2 y_2 & = (x_1 + y_2)(x_2 + y_1) - y_1 y_2 \\ l_1 l_2 & = \frac{1}{4} ((l_1 + l_2)^2 - (l_1 - l_2)^2). \end{cases}$$

On trouve

$$\begin{aligned} \Phi(x) = x^T A x &= 1 \left(\underbrace{1x_1 + 1x_2 + 2x_3}_{y_1} \right)^2 + 1 \left(\underbrace{1x_2 - 1x_3}_{y_2} \right)^2 + 4 \left(\underbrace{1x_3}_{y_3} \right)^2 \\ &= 1y_1^2 + 1y_2^2 + 4y_3^2 \geq 0. \end{aligned}$$

Formes quadratiques : exemple 3×3

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 9 \end{bmatrix} \implies \Phi(x) = x^T A x = \begin{cases} 1x_1^2 + 1x_1x_2 + 2x_1x_3 \\ +1x_2x_1 + 2x_2^2 + 1x_2x_3 \\ +2x_3x_1 + 1x_3x_2 + 9x_3^2 \end{cases}$$
$$= 1x_1^2 + 2x_2^2 + 9x_3^2 + 2(1x_1x_2 + 2x_1x_3 + 1x_2x_3)$$

Décomposition de Gauss en carrés :

$$\begin{cases} x_1^2 + 2\alpha x_1 y & = (x_1 + \alpha y)^2 - \alpha^2 y^2 \\ x_1 x_2 + x_1 y_1 + x_2 y_2 & = (x_1 + y_2)(x_2 + y_1) - y_1 y_2 \\ l_1 l_2 & = \frac{1}{4} ((l_1 + l_2)^2 - (l_1 - l_2)^2). \end{cases}$$

On trouve

$$\begin{aligned} \Phi(x) = x^T A x &= 1 \underbrace{(1x_1 + 1x_2 + 2x_3)}_{y_1}^2 + 1 \underbrace{(1x_2 - 1x_3)}_{y_2}^2 + 4 \underbrace{(1x_3)}_{y_3}^2 \\ &= 1y_1^2 + 1y_2^2 + 4y_3^2 \geq 0. \end{aligned}$$

Donc pour tout $x \in \mathbb{R}^3$, $\Phi(x) \geq 0$ et

$$\begin{aligned} \Phi(x) = 0 &\iff y_3 = y_2 = y_1 = 0 \\ &\iff x_3 = x_2 = x_1 = 0 \iff x = 0 \end{aligned}$$

Donc Φ DP.

Formes quadratiques : exemple 3×3

Décomposition de Gauss en carrés :

$$\begin{aligned}\Phi(x) = x^T Ax &= (\underbrace{1x_1 + 1x_2 + 2x_3}_{y_1})^2 + (\underbrace{1x_2 - 1x_3}_{y_2})^2 + 4(\underbrace{1x_3}_{y_3})^2 \\ &= 1y_1^2 + 1y_2^2 + 4y_3^2 = y^T Dy\end{aligned}$$

avec $y = L^T x$, où $L^T = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$, et $D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix}$. Donc

$$\Phi(x) = x^T Ax = y^T Dy = (L^T x)^T D (L^T x) = x^T LDL^T x$$

donc

$$A = LDL^T.$$

Formes quadratiques : exemple 3×3

Décomposition de Gauss en carrés :

$$\begin{aligned}\Phi(x) = x^T A x &= \underbrace{(1x_1 + 1x_2 + 2x_3)}_{y_1}^2 + \underbrace{(1x_2 - 1x_3)}_{y_2}^2 + 4 \underbrace{(1x_3)}_{y_3}^2 \\ &= 1y_1^2 + 1y_2^2 + 4y_3^2 = y^T D y\end{aligned}$$

avec $y = L^T x$, où $L^T = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}$, et $D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{bmatrix}$. Donc

$$\Phi(x) = x^T A x = y^T D y = (L^T x)^T D (L^T x) = x^T L D L^T x$$

donc

$$A = L D L^T.$$

De plus, ici, $D_{i,i} > 0$, donc il existe G diagonale tq. $D = G^2$ et $G_{i,i} = +\sqrt{D_{i,i}} > 0$.

$$A = L D L^T = (L G)(G L^T) = (L G)(G^T L^T) = (L G)(L G)^T = C C^T$$

et

$$C^T = G^T L^T = \begin{bmatrix} \sqrt{1} & 0 & 0 \\ 0 & \sqrt{1} & 0 \\ 0 & 0 & \sqrt{4} \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & -1 \\ 0 & 0 & 2 \end{bmatrix}.$$

Matrices symétriques définies positives : factorisation de Cholesky

Lemme (2.5.2)

Soit A matrice SDP.

Alors toutes ses sous-matrices principales $[A]_k$ sont régulières.

Démonstration : voir poly.

\Rightarrow donc A SDP admet une factorisation : $A = LDL^T$.

Matrices symétriques définies positives : factorisation de Cholesky

Théorème (2.5.3 : factorisation de Cholesky)

Soit A matrice SDP.

Alors elle admet une unique factorisation de Cholesky sous la forme

$$A = CC^T, \quad \text{où}$$

- C est triangulaire inférieure, et $C_{i,i} > 0, \forall i = 1, \dots, n$.

Matrices symétriques définies positives : factorisation de Cholesky

Théorème (2.5.3 : factorisation de Cholesky)

Soit A matrice SDP.

Alors elle admet une unique factorisation de Cholesky sous la forme

$$A = CC^T, \quad \text{où}$$

- C est triangulaire inférieure, et $C_{i,i} > 0, \forall i = 1, \dots, n$.

Démonstration : Existence. A SDP : donc $\forall k$ les $[A]_k$ sont SDP donc inversibles. Donc il existe une unique factorisation $A = LDL^T$.

De plus, $\forall x, x^T Ax = x^T LDL^T x = (L^T x)^T D(L^T x) = y^T Dy = \sum_{i=1}^n D_{i,i} y_i^2$, avec $y = L^T x$.

Comme L^T est tri-sup avec diagonale unité, elle est inversible. Donc pour $i = 1, \dots, n$, $y = e_i \iff x = L^{-T} e_i \neq 0$. Donc pour $x = L^{-T} e_i \neq 0$, on a $x^T Ax = (e_i)^T D e_i = D_{i,i} > 0$.

Donc il existe G diagonale tq. $D = G^2$ et $G_{i,i} = +\sqrt{D_{i,i}} > 0$. On pose $C = LG$ et

$$A = LDL^T = (LG)(GL^T) = (LG)(G^T L^T) = (LG)(LG)^T = CC^T.$$

Matrices symétriques définies positives : factorisation de Cholesky

Théorème (2.5.3 : factorisation de Cholesky)

Soit A matrice SDP.

Alors elle admet une unique factorisation de Cholesky sous la forme

$$A = CC^T, \quad \text{où}$$

- C est triangulaire inférieure, et $C_{i,i} > 0, \forall i = 1, \dots, n$.

Démonstration : Existence. A SDP : donc $\forall k$ les $[A]_k$ sont SDP donc inversibles. Donc il existe une unique factorisation $A = LDL^T$.

De plus, $\forall x, x^T Ax = x^T LDL^T x = (L^T x)^T D(L^T x) = y^T Dy = \sum_{i=1}^n D_{i,i} y_i^2$, avec $y = L^T x$.

Comme L^T est tri-sup avec diagonale unité, elle est inversible. Donc pour $i = 1, \dots, n$, $y = e_i \iff x = L^{-T} e_i \neq 0$. Donc pour $x = L^{-T} e_i \neq 0$, on a $x^T Ax = (e_i)^T D e_i = D_{i,i} > 0$.

Donc il existe G diagonale tq. $D = G^2$ et $G_{i,i} = +\sqrt{D_{i,i}} > 0$. On pose $C = LG$ et

$$A = LDL^T = (LG)(GL^T) = (LG)(G^T L^T) = (LG)(LG)^T = CC^T.$$

Démonstration : Unicité. On suppose que $A = CC^T = \tilde{C}\tilde{C}^T$. Comme C est tri-inf et $C_{i,i} > 0$ et \tilde{C}^T tri-sup et $\tilde{C}_{i,i} > 0$, les matrices C et \tilde{C}^T sont inversibles (Voir Chap. 0, exo TD 11). Donc $D = C^T \tilde{C}^{-T} = C^{-1} \tilde{C}$: matrice tri-sup et tri-inf donc diagonale, et $D_{i,i} = C_{i,i} / \tilde{C}_{i,i} = \tilde{C}_{i,i} / C_{i,i}$. Donc $C_{i,i}^2 = \tilde{C}_{i,i}^2 \iff C_{i,i} = \tilde{C}_{i,i}$ car > 0 . Donc $D = I$ et $C = \tilde{C}$. Unicité.

Matrices SDP : principe de la factorisation de Cholesky : $A = CC^T$

$$\begin{bmatrix}
 c_{1,1} & 0 & 0 & \dots & 0 & 0 \\
 c_{2,1} & c_{2,2} & 0 & \dots & 0 & 0 \\
 c_{3,1} & c_{3,2} & c_{3,3} & \dots & 0 & 0 \\
 \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 c_{n-1,1} & c_{n-1,2} & c_{n-1,3} & \dots & c_{n-1,n-1} & 0 \\
 c_{n,1} & c_{n,2} & c_{n,3} & \dots & c_{n,n-1} & c_{n,n}
 \end{bmatrix}
 \begin{bmatrix}
 c_{1,1} & c_{2,1} & c_{3,1} & \dots & c_{n-1,1} & c_{n,1} \\
 0 & c_{2,2} & c_{3,2} & \dots & c_{n-1,2} & c_{n,2} \\
 0 & 0 & c_{3,3} & \dots & c_{n-1,3} & c_{n,3} \\
 \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 0 & 0 & 0 & \dots & c_{n-1,n-1} & c_{n-1,n} \\
 0 & 0 & 0 & \dots & 0 & c_{n,n}
 \end{bmatrix}$$

Unicité de la factorisation : \implies travaille par identification (diagonale /ligne).

Matrices SDP : principe de la factorisation de Cholesky : $A = CC^T$

$$\begin{bmatrix} c_{1,1} & 0 & 0 & \dots & 0 & 0 \\ c_{2,1} & c_{2,2} & 0 & \dots & 0 & 0 \\ c_{3,1} & c_{3,2} & c_{3,3} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{n-1,1} & c_{n-1,2} & c_{n-1,3} & \dots & c_{n-1,n-1} & 0 \\ c_{n,1} & c_{n,2} & c_{n,3} & \dots & c_{n,n-1} & c_{n,n} \end{bmatrix} \begin{bmatrix} c_{1,1} & c_{2,1} & c_{3,1} & \dots & c_{n-1,1} & c_{n,1} \\ 0 & c_{2,2} & c_{3,2} & \dots & c_{n-1,2} & c_{n,2} \\ 0 & 0 & c_{3,3} & \dots & c_{n-1,3} & c_{n,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & c_{n-1,n-1} & c_{n,n} \\ 0 & 0 & 0 & \dots & 0 & c_{n,n} \end{bmatrix}$$

Unicité de la factorisation : \implies travaille par identification (diagonale / ligne).

- \bullet $A_{1,1} = c_{1,1}c_{1,1} \implies c_{1,1} = \sqrt{A_{1,1}}$

Si $i > 1$: $A_{i,1} = c_{i,1}c_{1,1} \implies c_{i,1} = \frac{A_{i,1}}{c_{1,1}}$

Matrices SDP : principe de la factorisation de Cholesky : $A = CC^T$

$$\begin{bmatrix} C_{1,1} & 0 & 0 & \dots & 0 & 0 \\ C_{2,1} & C_{2,2} & 0 & \dots & 0 & 0 \\ C_{3,1} & C_{3,2} & C_{3,3} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ C_{n-1,1} & C_{n-1,2} & C_{n-1,3} & \dots & C_{n-1,n-1} & 0 \\ C_{n,1} & C_{n,2} & C_{n,3} & \dots & C_{n,n-1} & C_{n,n} \end{bmatrix} \begin{bmatrix} C_{1,1} & C_{2,1} & C_{3,1} & \dots & C_{n-1,1} & C_{n,1} \\ 0 & C_{2,2} & C_{3,2} & \dots & C_{n-1,2} & C_{n,2} \\ 0 & 0 & C_{3,3} & \dots & C_{n-1,3} & C_{n,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & C_{n-1,n-1} & C_{n-1,n} \\ 0 & 0 & 0 & \dots & 0 & C_{n,n} \end{bmatrix}$$

Unicité de la factorisation : \implies travaille par identification (diagonale / ligne).

- $A_{1,1} = C_{1,1}C_{1,1} \implies C_{1,1} = \sqrt{A_{1,1}}$

Si $i > 1$: $A_{i,1} = C_{i,1}C_{1,1} \implies C_{i,1} = \frac{A_{i,1}}{C_{1,1}}$

- $A_{2,2} = C_{2,1}C_{2,1} + C_{2,2}C_{2,2} \implies C_{2,2} = \sqrt{A_{2,2} - C_{2,1}^2}$

Si $i > 2$: $A_{i,2} = C_{i,1}C_{2,1} + C_{i,2}C_{2,2} \implies C_{i,2} = \frac{1}{C_{2,2}} (A_{i,2} - C_{i,1}C_{2,1})$

Matrices SDP : principe de la factorisation de Cholesky

Unicité de la factorisation : \implies travaille par identification (diagonale / ligne).
Pour $i, j = 1, \dots, n$, en utilisant le fait que C est tri-inf :

$$A = CC^T \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}(C^T)_{k,j} \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}C_{j,k}$$

Matrices SDP : principe de la factorisation de Cholesky

Unicité de la factorisation : \implies travaille par identification (diagonale / ligne).
Pour $i, j = 1, \dots, n$, en utilisant le fait que C est tri-inf :

$$A = CC^T \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}(C^T)_{k,j} \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}C_{j,k}$$

- $A_{1,1} = C_{1,1}C_{1,1} \implies C_{1,1} = \sqrt{A_{1,1}}$
Si $i > 1$: $A_{i,1} = C_{i,1}C_{1,1} \implies C_{i,1} = \frac{A_{i,1}}{C_{1,1}}$
- $A_{2,2} = C_{2,1}C_{2,1} + C_{2,2}C_{2,2} \implies C_{2,2} = \sqrt{A_{2,2} - C_{2,1}^2}$
Si $i > 2$: $A_{i,2} = C_{i,1}C_{2,1} + C_{i,2}C_{2,2} \implies C_{i,2} = \frac{1}{C_{2,2}}(A_{i,2} - C_{i,1}C_{2,1})$

Matrices SDP : principe de la factorisation de Cholesky

Unicité de la factorisation : \implies travaille par identification (diagonale / ligne).
Pour $i, j = 1, \dots, n$, en utilisant le fait que C est tri-inf :

$$A = CC^T \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}(C^T)_{k,j} \iff A_{i,j} = \sum_{k=1}^{\min(i,j)} C_{i,k}C_{j,k}$$

• $A_{1,1} = C_{1,1}C_{1,1} \implies C_{1,1} = \sqrt{A_{1,1}}$

Si $i > 1$: $A_{i,1} = C_{i,1}C_{1,1} \implies C_{i,1} = \frac{A_{i,1}}{C_{1,1}}$

• $A_{2,2} = C_{2,1}C_{2,1} + C_{2,2}C_{2,2} \implies C_{2,2} = \sqrt{A_{2,2} - C_{2,1}^2}$

Si $i > 2$: $A_{i,2} = C_{i,1}C_{2,1} + C_{i,2}C_{2,2} \implies C_{i,2} = \frac{1}{C_{2,2}}(A_{i,2} - C_{i,1}C_{2,1})$

• Pour $j > 1$ donné :

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

Si $i > j$: $A_{i,j} = C_{i,1}C_{j,1} + C_{i,2}C_{j,2} + \dots + C_{i,j-1}C_{j,j-1} + C_{i,j}C_{j,j}$

$$\implies C_{i,j} = \frac{1}{C_{j,j}} \left(A_{i,j} - \sum_{k=1}^{j-1} C_{i,k}C_{j,k} \right)$$

Algorithme de Cholesky

Pour $jj = 1$ jusqu'à n

$$s \leftarrow A(jj, jj) - \sum_{kk=1}^{jj-1} C(jj, kk)^2$$

Si $s < \text{tol}$

Sortir, erreur : A non SDP

Fin Si

$C(jj, jj) \leftarrow \text{sqrt}(s)$

Pour $ii = jj+1$ jusqu'à n (sauté pour $jj=n$)

$$C(ii, jj) \leftarrow 1 / C(jj, jj) \times (A(ii, jj) - \sum_{kk=1}^{jj-1} C(ii, kk) \times C(jj, kk))$$

Fin Pour

Fin Pour

Exercice

Calculer la factorisation $A = CC^T$ avec l'algorithme de Cholesky pour

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 9 \end{bmatrix}$$

Montrer que l'on retrouve le C obtenu par la décomposition en carrés de Gauss.

Plan

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles**
- 7 Conditionnement

Définition (2.6.1 norme vectorielle)

Soit E un espace vectoriel sur K ($K = \mathbb{R}$ ou \mathbb{C}).

On appelle norme sur E une application de E dans \mathbb{R} notée

$$x \mapsto \|x\|$$

possédant les propriétés suivantes : quel que soit $x \in E, y \in E, \lambda \in K$:

- 1 $\|x\| \geq 0$
- 2 $\|x\| = 0 \iff x = 0$
- 3 $\|\lambda x\| = |\lambda| \|x\|$ (la norme est positivement homogène)
- 4 $\|x + y\| \leq \|x\| + \|y\|$ (inégalité triangulaire)

Pour $a \in \mathbb{C}$ ou \mathbb{R} , $|a|$ = module ou valeur absolue de a ($|a| = \sqrt{a\bar{a}}$).

Exemple (Normes vectorielles usuelles)

1 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_1 = \sum_{i=1}^n |x_i|$,

2 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$, (norme euclidienne),

3 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_\infty = \max_{i=1, \dots, n} |x_i|$.

Exemple (Normes vectorielles usuelles)

1 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_1 = \sum_{i=1}^n |x_i|$,

2 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$, (norme euclidienne),

3 $E = \mathbb{R}^n$ ou \mathbb{C}^n : $\|x\|_\infty = \max_{i=1, \dots, n} |x_i|$.

Théorème (Normes vectorielles équivalentes sur \mathbb{R}^n)

Toutes les normes vectorielles sont **équivalentes** sur \mathbb{R}^n . Soit N une norme sur \mathbb{R}^n , alors

$$\exists \alpha, \beta > 0 : \quad \forall x \in \mathbb{R}^n \quad \alpha N(x) \leq \|x\|_\infty \leq \beta N(x).$$

Pour les normes usuelles, on a :

$$\|x\|_\infty \leq \|x\|_1 \leq \sqrt{n} \|x\|_2 \leq n \|x\|_\infty.$$

seconde inégalité découle de Cauchy-Schwarz.

Exemple (Normes vectorielles sur des espaces de dimension infinie)

- 1 $E = C_0([0, 1])$: $\|f\|_{L^1} = \int_0^1 |f(t)| dt,$
- 2 $E = C_0([0, 1])$: $\|f\|_{L^2} = \sqrt{\int_0^1 f^2(t) dt},$
- 3 $E = C_0([0, 1])$: $\|f\|_{L^\infty} = \max_{t \in [0, 1]} |f(t)|,$
- 4 ...

Théorème (Rappels sur les espaces euclidiens)

Si $E = \mathbb{R}^n$ (résultats similaires adaptés pour \mathbb{C}^n), pour $x, y \in \mathbb{R}^n$:

- ① Norme euclidienne associée à un produit scalaire :

$$\|x\|_2 = \sqrt{\langle x, x \rangle}, \quad \text{avec } \langle x, y \rangle = \sum_{i=1}^n x_i y_i$$

- ② Inégalité de Cauchy-Schwarz (cf. Chap. 2 Exo C.1.8) :

$$|\langle x, y \rangle| \leq \|x\|_2 \|y\|_2 \iff \left| \sum_{i=1}^n x_i y_i \right| \leq \sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}.$$

Égalité dans CS \iff x et y sont liés.

- ③ Théorème de Pythagore :

$$\|x + y\|_2^2 = \|x\|_2^2 + 2\langle x, y \rangle + \|y\|_2^2.$$

- ④ Voir Chap 3...

Définition (2.6.2 norme matricielle)

On appelle **norme matricielle** une application de $\mathcal{M}_{mn}(\mathbb{R})$ dans \mathbb{R} notée

$$A \mapsto \|A\|,$$

vérifiant :

- 1 $\|A\| \geq 0$,
- 2 $\|A\| = 0 \iff A = 0$,
- 3 $\|\lambda A\| = |\lambda| \|A\|, \quad \forall A \in \mathcal{M}_{mn}(\mathbb{R}), \forall \lambda \in \mathbb{R}$,
- 4 $\|A + B\| \leq \|A\| + \|B\|, \quad \forall A \in \mathcal{M}_{mn}(\mathbb{R}), \forall B \in \mathcal{M}_{mn}(\mathbb{R})$,
- 5 $\|AB\| \leq \|A\| \|B\|, \quad \forall A \in \mathcal{M}_{mn}(\mathbb{R}), \forall B \in \mathcal{M}_{np}(\mathbb{R})$.

On déduit :

$$\|B^k\| \leq \|B\|^k, \quad \forall B \in \mathcal{M}_n(\mathbb{R}).$$

Rappels sur sup et max

Soit une fonction $f : X \subset \mathbb{R} \rightarrow \mathbb{R}$.

Definition

Le sup, quand il existe, est **le plus petit des majorants** de f sur X :

$$\forall x \in X : f(x) \leq \sup_{x \in X} f(x), \quad \forall \varepsilon > 0 \exists x_\varepsilon \in X : \sup_{x \in X} f(x) \leq f(x_\varepsilon) + \varepsilon.$$

Le sup est un max quand il est **atteint**, c'est-à-dire si

$$\exists \tilde{x} \in X : f(\tilde{x}) = \sup_{x \in X} f(x) = \max_{x \in X} f(x).$$

Rappels sur sup et max

Soit une fonction $f : X \subset \mathbb{R} \rightarrow \mathbb{R}$.

Definition

Le sup, quand il existe, est **le plus petit des majorants** de f sur X :

$$\forall x \in X : f(x) \leq \sup_{x \in X} f(x), \quad \forall \varepsilon > 0 \exists x_\varepsilon \in X : \sup_{x \in X} f(x) \leq f(x_\varepsilon) + \varepsilon.$$

Le sup est un max quand il est **atteint**, c'est-à-dire si

$$\exists \tilde{x} \in X : f(\tilde{x}) = \sup_{x \in X} f(x) = \max_{x \in X} f(x).$$

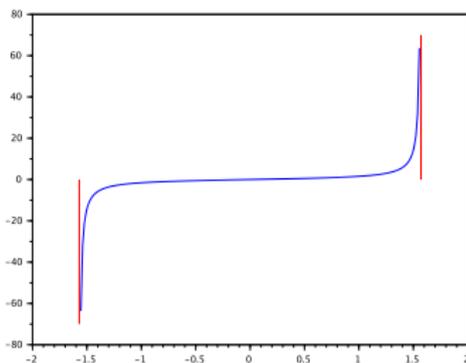
On a

Lemme

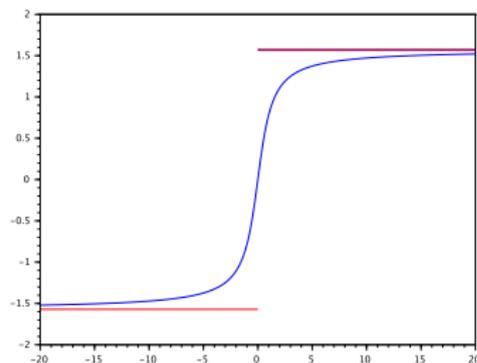
- 1 Si f est majorée ($\exists M \in \mathbb{R}, \forall x \in X : f(x) \leq M$), alors le sup existe dans \mathbb{R} et $\sup_{x \in X} f(x) \leq M$. (propriété de \mathbb{R} , fausse dans \mathbb{Q} .)
- 2 Si $\sup_{x \in X} f(x)$ existe et si $\lambda \geq 0$, alors : $\sup_{x \in X} (\lambda f(x)) = \lambda \sup_{x \in X} f(x)$.
- 3 Si les sup de f et de g existent, alors :
 $\sup_{x \in X} (f(x) + g(x)) \leq \sup_{x \in X} f(x) + \sup_{x \in X} g(x)$.

Exemples de sup et max

$$] - \frac{\pi}{2}, \frac{\pi}{2}[\ni x \rightarrow \tan(x) \in \mathbb{R}$$



$$\mathbb{R} \ni x \rightarrow \arctan(x) \in] - \frac{\pi}{2}, \frac{\pi}{2}[$$



- Sur \mathbb{R} , arctan admet une infinité de majorants : $M = 2$ est un majorant.
- Sur \mathbb{R} , arctan admet un sup (le plus petit des majorants) : $\frac{\pi}{2}$, mais pas de max (le sup n'est pas atteint).
- Sur $] - \frac{\pi}{2}, \frac{\pi}{2}[$, tan n'admet pas de sup.
- Sur $[-\frac{\pi}{4}, \frac{\pi}{4}]$, tan admet un sup (qui vaut 1), et qui est atteint en $\frac{\pi}{4}$: ce sup est un max.
- Sur $] - \frac{\pi}{4}, \frac{\pi}{4}[$, tan admet un sup (qui vaut encore 1), mais qui n'est pas atteint sur l'intervalle : pas de max.

Définition (2.6.3 norme matricielle subordonnée)

Soient deux normes vectorielles sur \mathbb{R}^n et sur \mathbb{R}^m , notées $\|\cdot\|$. On appelle **norme matricielle subordonnée** l'application de $\mathcal{M}_{mn}(\mathbb{R})$ dans \mathbb{R} définie par

$$A \mapsto \|A\| = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|}{\|x\|}.$$

Théorème (norme matricielle subordonnée)

Soit $A \in \mathcal{M}_{mn}(\mathbb{R})$. On a :

- 1 L'application $A \mapsto \|A\|$ est bien définie.
- 2 L'application $A \mapsto \|A\|$ est une norme matricielle.
- 3 Dans la définition, le sup est atteint, c'est un max.
- 4 On a de plus pour toute norme subordonnée :

$$\|Ax\| \leq \|A\| \|x\|, \quad \forall A \in \mathcal{M}_{mn}(\mathbb{R}), \forall x \in \mathbb{R}^n.$$

- 5 Pour $I \in \mathcal{M}_{nn}(\mathbb{R})$, si on prend la même norme vectorielle pour \mathbb{R}^n : $\|I\| = 1$.

Théorème (norme matricielle subordonnée)

Soit $A \in \mathcal{M}_{mn}(\mathbb{R})$. On a :

- 1 L'application $A \mapsto \|A\|$ est bien définie.
- 2 L'application $A \mapsto \|A\|$ est une norme matricielle.
- 3 Dans la définition, le sup est atteint, c'est un max.
- 4 On a de plus pour toute norme subordonnée :

$$\|Ax\| \leq \|A\| \|x\|, \quad \forall A \in \mathcal{M}_{mn}(\mathbb{R}), \forall x \in \mathbb{R}^n.$$

- 5 Pour $I \in \mathcal{M}_{nn}(\mathbb{R})$, si on prend la même norme vectorielle pour \mathbb{R}^n : $\|I\| = 1$.

Démonstrations (partielles) : voir polycopié et notes de cours.

- 1 bien définie : soit $x \in \mathbb{R}^n$ décomposé dans la base canonique de \mathbb{R}^n : $x = \sum_{j=1}^n x_j e_j$. Alors en utilisant l'inégalité triangulaire et l'équivalence avec la norme infinie ($\|x\|_\infty \leq \beta \|x\|$), il vient :
$$\|Ax\| \leq \sum_{j=1}^n |x_j| \|Ae_j\| \leq \|x\|_\infty \sum_{j=1}^n \|Ae_j\| \leq (\beta \sum_{j=1}^n \|Ae_j\|) \|x\|.$$
 \Rightarrow si $x \neq 0$: division par $\|x\|$ puis passage au sup : $\|A\|$ existe et $\|A\| \leq \beta \sum_{j=1}^n \|Ae_j\|$.

Proposition (Normes matricielles subordonnées)

1 avec $\|\cdot\|_1$ pour \mathbb{R}^n et \mathbb{R}^m ($\|A\|_1 = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_1}{\|x\|_1}$), on montre :

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |A_{i,j}| = \max_{1 \leq j \leq n} \|A_j\|_1.$$

2 avec $\|\cdot\|_\infty$ pour \mathbb{R}^n et \mathbb{R}^m ($\|A\|_\infty = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty}$), on montre :

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |A_{i,j}| = \max_{1 \leq j \leq n} \|(A_j)^T\|_1.$$

3 $\|A\|_2$: voir plus loin...

Proposition (Normes matricielles subordonnées)

① avec $\|\cdot\|_1$ pour \mathbb{R}^n et \mathbb{R}^m ($\|A\|_1 = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_1}{\|x\|_1}$), on montre :

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |A_{i,j}| = \max_{1 \leq j \leq n} \|A_j\|_1.$$

② avec $\|\cdot\|_\infty$ pour \mathbb{R}^n et \mathbb{R}^m ($\|A\|_\infty = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty}$), on montre :

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |A_{i,j}| = \max_{1 \leq j \leq n} \|(A_j)^T\|_1.$$

③ $\|A\|_2$: voir plus loin...

Note : il existe des normes matricielles qui ne sont **pas** subordonnées.

Exemple : la norme de Frobenius : $\|A\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n |A_{i,j}|^2} = \sqrt{\text{trace}(A^T A)}$.

Exercice : montrer que c'est une norme matricielle. Puis calculer $\|I\|_F$ et conclure que ce n'est pas une norme subordonnée.

Norme matricielle subordonnée à la norme 1

On pose $N(A) = \max_{1 \leq j \leq n} \sum_{i=1}^n |A_{i,j}| = \max_{j=1, \dots, n} (\|A_j\|_1)$. On montre $\|A\|_1 = N(A)$.

1 **Majoration : on montre que $\|A\|_1 \leq N(A)$.**

Soit $x \neq 0$ dans \mathbb{R}^n . On a $Ax = \sum_{j=1}^n A_j x_j \in \mathbb{R}^m$. On prend la norme et on majore :

$$\|Ax\|_1 \leq \sum_{j=1}^n |x_j| \|A_j\|_1 \leq \max_{j=1, \dots, n} (\|A_j\|_1) \left(\sum_{j=1}^n |x_j| \right) = N(A) \|x\|_1.$$

On divise par $\|x\|_1 > 0$ et on obtient :

$$\forall x \neq 0 \quad \frac{\|Ax\|_1}{\|x\|_1} \leq N(A),$$

donc $N(A)$ est un majorant, le sup existe et $\|A\|_1 \leq N(A)$.

Norme matricielle subordonnée à la norme 1

On pose $N(A) = \max_{1 \leq j \leq n} \sum_{i=1}^n |A_{i,j}| = \max_{j=1, \dots, n} (\|A_j\|_1)$. On montre $\|A\|_1 = N(A)$.

① **Majoration : on montre que $\|A\|_1 \leq N(A)$.**

Soit $x \neq 0$ dans \mathbb{R}^n . On a $Ax = \sum_{j=1}^n A_j x_j \in \mathbb{R}^m$. On prend la norme et on majore :

$$\|Ax\|_1 \leq \sum_{j=1}^n |x_j| \|A_j\|_1 \leq \max_{j=1, \dots, n} (\|A_j\|_1) \left(\sum_{j=1}^n |x_j| \right) = N(A) \|x\|_1.$$

On divise par $\|x\|_1 > 0$ et on obtient :

$$\forall x \neq 0 \quad \frac{\|Ax\|_1}{\|x\|_1} \leq N(A),$$

donc $N(A)$ est un majorant, le sup existe et $\|A\|_1 \leq N(A)$.

② **Minoration : on cherche un \tilde{x} particulier pour lequel $\frac{\|A\tilde{x}\|_1}{\|\tilde{x}\|_1}$ vaut $N(A)$.**

Soit j_0 tel que $N(A) = \sum_{i=1}^n |A_{i,j_0}| = \|A_{j_0}\|_1$ (\max_j atteint en un j_0).

On prend $\tilde{x} = e_{j_0}$, qui vérifie $Ae_{j_0} = A_{j_0}$ et $\|e_{j_0}\|_1 = 1 : \|Ae_{j_0}\|_1 = \|A_{j_0}\|_1 = N(A)$.

$\Rightarrow N(A) = \frac{\|Ae_{j_0}\|_1}{\|e_{j_0}\|_1} \leq \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_1}{\|x\|_1} = \|A\|_1$. Il y a donc égalité.

Théorème (2.6.1)

Si $A \in \mathcal{M}_{nn}(\mathbb{R})$ est symétrique alors

$$\max_{1 \leq k \leq n} \lambda_k = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{x^T A x}{x^T x} = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{\langle x, Ax \rangle}{\langle x, x \rangle},$$

où $(\lambda_k)_{1 \leq k \leq n}$ sont les n valeurs propres réelles, distinctes ou non, de A .

Norme matricielle subordonnée à la norme 2

Démonstration : A symétrique réelle : \implies admet une **base de vecteurs propres orthonormés** $(Y_i)_{1 \leq i \leq n}$ associés aux valeurs propres réelles ordonnées : $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. On a :

$$\langle Y_i, Y_j \rangle = Y_i^T Y_j = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \quad (\text{et donc } \|Y_i\|_2 = 1) \text{ et } AY_i = \lambda_i Y_i.$$

Norme matricielle subordonnée à la norme 2

Démonstration : A symétrique réelle : \implies admet une **base de vecteurs propres orthonormés** $(Y_i)_{1 \leq i \leq n}$ associés aux valeurs propres réelles ordonnées : $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. On a :

$$\langle Y_i, Y_j \rangle = Y_i^T Y_j = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \quad (\text{et donc } \|Y_i\|_2 = 1) \text{ et } AY_i = \lambda_i Y_i.$$

1 **Majoration :** Soit $x \in \mathbb{R}^n$ écrit dans la BON $(Y_i)_i$: $x = \sum_{i=1}^n \xi_i Y_i$. On déduit $Ax = \sum_{i=1}^n \xi_i \lambda_i Y_i$. Il vient en utilisant l'orthonormalité des $(Y_i)_i$:

$$\|x\|_2^2 = \left(\sum_{i=1}^n \xi_i Y_i^T \right) \left(\sum_{j=1}^n \xi_j Y_j \right) = \sum_{i=1}^n \sum_{j=1}^n \xi_i \xi_j Y_i^T Y_j = \sum_{i=1}^n \xi_i^2$$

$$x^T Ax = \left(\sum_{i=1}^n \xi_i Y_i^T \right) \left(\sum_{j=1}^n \xi_j \lambda_j Y_j \right) = \sum_{i=1}^n \sum_{j=1}^n \xi_i \xi_j \lambda_j Y_i^T Y_j = \sum_{i=1}^n \lambda_i \xi_i^2.$$

Donc : $x^T Ax \leq \lambda_1 \sum_{i=1}^n \xi_i^2 = \lambda_1 \|x\|_2^2$. D'où $\forall x \neq 0$ $\frac{x^T Ax}{x^T x} \leq \lambda_1$, donc λ_1 est un

majorant, le sup existe et $\sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{x^T Ax}{x^T x} \leq \lambda_1 = \max_{1 \leq k \leq n} \lambda_k$.

Norme matricielle subordonnée à la norme 2

Démonstration : A symétrique réelle : \implies admet une **base de vecteurs propres orthonormés** $(Y_i)_{1 \leq i \leq n}$ associés aux valeurs propres réelles ordonnées : $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. On a :

$$\langle Y_i, Y_j \rangle = Y_i^T Y_j = \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \quad (\text{et donc } \|Y_i\|_2 = 1) \text{ et } AY_i = \lambda_i Y_i.$$

1 **Majoration :** Soit $x \in \mathbb{R}^n$ écrit dans la BON $(Y_i)_i$: $x = \sum_{i=1}^n \xi_i Y_i$. On déduit $Ax = \sum_{i=1}^n \xi_i \lambda_i Y_i$. Il vient en utilisant l'orthonormalité des $(Y_i)_i$:

$$\|x\|_2^2 = \left(\sum_{i=1}^n \xi_i Y_i^T \right) \left(\sum_{j=1}^n \xi_j Y_j \right) = \sum_{i=1}^n \sum_{j=1}^n \xi_i \xi_j Y_i^T Y_j = \sum_{i=1}^n \xi_i^2$$

$$x^T Ax = \left(\sum_{i=1}^n \xi_i Y_i^T \right) \left(\sum_{j=1}^n \xi_j \lambda_j Y_j \right) = \sum_{i=1}^n \sum_{j=1}^n \xi_i \xi_j \lambda_j Y_i^T Y_j = \sum_{i=1}^n \lambda_i \xi_i^2.$$

Donc : $x^T Ax \leq \lambda_1 \sum_{i=1}^n \xi_i^2 = \lambda_1 \|x\|_2^2$. D'où $\forall x \neq 0$ $\frac{x^T Ax}{x^T x} \leq \lambda_1$, donc λ_1 est un

majorant, le sup existe et $\sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{x^T Ax}{x^T x} \leq \lambda_1 = \max_{1 \leq k \leq n} \lambda_k$.

2 **Minoration :** Pour obtenir l'égalité il suffit de remarquer que $(Y_1)^T AY_1 = \lambda_1 \|Y_1\|_2^2 = \lambda_1$.
Donc

$$\lambda_1 = \frac{(Y_1)^T AY_1}{\|Y_1\|_2^2} \leq \sup_{x \in \mathbb{R}^n \setminus \{0\}} \frac{x^T Ax}{x^T x},$$

donc il y a égalité, le sup est atteint, c'est un max.

Définition (Spectre et rayon spectral)

Soit $A \in \mathcal{M}_{nn}(\mathbb{C})$. On appelle spectre de A l'ensemble

$$\text{Sp}(A) = \{\lambda_i(A) \mid 1 \leq i \leq n\} \subset \mathbb{C},$$

où les $\lambda_i(A)$ sont les valeurs propres de A . On appelle rayon spectral de A le réel

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i(A)| \in \mathbb{R}.$$

Proposition (Propriétés du rayon spectral)

On a :

- 1 $\text{Sp}(AB) \setminus \{0\} = \text{Sp}(BA) \setminus \{0\}$ et $\rho(AB) = \rho(BA)$, $\forall A \in \mathcal{M}_{np}(\mathbb{C})$ et $B \in \mathcal{M}_{pn}(\mathbb{C})$.
- 2 $\rho(A) \leq \|A\|$, pour toute norme subordonnée $\|\cdot\|$ et $\forall A \in \mathcal{M}_{nn}(\mathbb{C})$.
- 3 $\rho(A^2) = \rho(A)^2$, $\forall A \in \mathcal{M}_{nn}(\mathbb{C})$.

Démonstration : (1) et (3, cas symétrique) voir Chapitre 2, exercices TD n° 11 et 12.

Théorème (2.6.2 : norme euclidienne subordonnée)

1 Soit $A \in \mathcal{M}_{np}(\mathbb{R})$, alors

$$\begin{aligned}\|A\|_2^2 &= \rho(A^T A) = \rho(AA^T), \\ \|A\|_2^2 &= \left\| \|AA^T\|_2 \right\| = \left\| \|A^T A\|_2 \right\|.\end{aligned}$$

2 Si la matrice C est symétrique, alors

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

Théorème (2.6.2 : norme euclidienne subordonnée)

1 Soit $A \in \mathcal{M}_{np}(\mathbb{R})$, alors

$$\begin{aligned}\|A\|_2^2 &= \rho(A^T A) = \rho(AA^T), \\ \|A\|_2^2 &= \left\| \|AA^T\|_2 \right\|_2 = \left\| \|A^T A\|_2 \right\|_2.\end{aligned}$$

2 Si la matrice C est symétrique, alors

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

Démonstration : (2) voir Chapitre 2, exercices C.1.23 et TD n° 11.

On montre $\|A\|_2^2 = \rho(A^T A)$. ($\rho(AA^T) = \rho(A^T A)$ car $\rho(AB) = \rho(BA)$.)

Théorème (2.6.2 : norme euclidienne subordonnée)

1 Soit $A \in \mathcal{M}_{np}(\mathbb{R})$, alors

$$\begin{aligned}\|A\|_2^2 &= \rho(A^T A) = \rho(AA^T), \\ \|A\|_2^2 &= \left\| \|AA^T\|_2 \right\|_2 = \left\| \|A^T A\|_2 \right\|_2.\end{aligned}$$

2 Si la matrice C est symétrique, alors

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

Démonstration : (2) voir Chapitre 2, exercices C.1.23 et TD n° 11.

On montre $\|A\|_2^2 = \rho(A^T A)$. ($\rho(AA^T) = \rho(A^T A)$ car $\rho(AB) = \rho(BA)$.)

On note : $\|A\|_2^2 = \left(\sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \right)^2 = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2}$, car sup de termes ≥ 0 .

Théorème (2.6.2 : norme euclidienne subordonnée)

1 Soit $A \in \mathcal{M}_{np}(\mathbb{R})$, alors

$$\begin{aligned}\|A\|_2^2 &= \rho(A^T A) = \rho(AA^T), \\ \|A\|_2^2 &= \left\| \|AA^T\|_2 \right\|_2 = \left\| \|A^T A\|_2 \right\|_2.\end{aligned}$$

2 Si la matrice C est symétrique, alors

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

Démonstration : (2) voir Chapitre 2, exercices C.1.23 et TD n° 11.

On montre $\|A\|_2^2 = \rho(A^T A)$. ($\rho(AA^T) = \rho(A^T A)$ car $\rho(AB) = \rho(BA)$.)

On note : $\|A\|_2^2 = \left(\sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \right)^2 = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2}$, car sup de termes ≥ 0 .

On a : $\|Ax\|_2^2 = (Ax)^T (Ax) = x^T A^T Ax = x^T (A^T A)x = x^T Bx$ où l'on a noté $B = A^T A \in \mathcal{M}_{nn}$.
 B est symétrique semi-DP (car $x^T Bx = \|Ax\|_2^2 \geq 0$), donc ses n valeurs propres $\mu_k(B)$ sont ≥ 0 .

Théorème (2.6.2 : norme euclidienne subordonnée)

1 Soit $A \in \mathcal{M}_{np}(\mathbb{R})$, alors

$$\begin{aligned}\|A\|_2^2 &= \rho(A^T A) = \rho(AA^T), \\ \|A\|_2^2 &= \left\| \|AA^T\|_2 \right\|_2 = \left\| \|A^T A\|_2 \right\|_2.\end{aligned}$$

2 Si la matrice C est symétrique, alors

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

Démonstration : (2) voir Chapitre 2, exercices C.1.23 et TD n° 11.

On montre $\|A\|_2^2 = \rho(A^T A)$. ($\rho(AA^T) = \rho(A^T A)$ car $\rho(AB) = \rho(BA)$.)

On note : $\|A\|_2^2 = \left(\sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} \right)^2 = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2}$, car sup de termes ≥ 0 .

On a : $\|Ax\|_2^2 = (Ax)^T (Ax) = x^T A^T Ax = x^T (A^T A)x = x^T Bx$ où l'on a noté $B = A^T A \in \mathcal{M}_{nn}$.
 B est symétrique semi-DP (car $x^T Bx = \|Ax\|_2^2 \geq 0$), donc ses n valeurs propres $\mu_k(B)$ sont ≥ 0 .

\Rightarrow Théorème (2.6.1) : $\rho(A^T A) = \rho(B) = \max_{1 \leq k \leq n} |\mu_k(B)| = \max_{1 \leq k \leq n} (\mu_k(B)) =$
 $\max_{x \in \mathbb{R}^n, x \neq 0} \frac{x^T Bx}{x^T x} = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{x^T A^T Ax}{x^T x} = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_2^2}{\|x\|_2^2} = \|A\|_2^2.$

Exemple (norme euclidienne subordonnée)

$\rho(A) \neq \|A\|_2$ en général.

Soit : $A = \begin{pmatrix} 0 & 2 \\ 0 & 0 \end{pmatrix}$. $\text{Sp}(A) = \{0\}$, donc $\rho(A) = 0$, mais $A \neq 0$, donc $\|A\|_2 \neq 0!!$

$A^T A = \begin{pmatrix} 0 & 0 \\ 0 & 4 \end{pmatrix}$, donc $\text{Sp}(A^T A) = \{0, 4\}$ et $\rho(A^T A) = 4$, donc $\|A\|_2 = 2!!$

Plan

- 1 Introduction, motivations
- 2 Élimination de Gauss
- 3 Factorisation LU directe : Doolittle
- 4 Pivotage
- 5 Matrices symétriques : factorisations LDL^T et Cholesky
- 6 Normes matricielles
- 7 Conditionnement

Définition (Conditionnement)

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$ inversible. On appelle conditionnement de A relatif à la norme $\|\cdot\|$ le réel

$$\chi(A) = \|A\| \|A^{-1}\| \in \mathbb{R}.$$

On a : $1 = \|I\| = \|AA^{-1}\| \leq \|A\| \|A^{-1}\| = \chi(A)$. Donc :

Définition (Conditionnement)

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$ inversible. On appelle conditionnement de A relatif à la norme $\|\cdot\|$ le réel

$$\chi(A) = \|A\| \|A^{-1}\| \in \mathbb{R}.$$

On a : $1 = \|I\| = \|AA^{-1}\| \leq \|A\| \|A^{-1}\| = \chi(A)$. Donc :

Proposition

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$ inversible. Alors $\chi(A) \geq 1$.

Définition (Conditionnement)

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$ inversible. On appelle conditionnement de A relatif à la norme $\|\cdot\|$ le réel

$$\chi(A) = \|A\| \|A^{-1}\| \in \mathbb{R}.$$

On a : $1 = \|I\| = \|AA^{-1}\| \leq \|A\| \|A^{-1}\| = \chi(A)$. Donc :

Proposition

Soit $A \in \mathcal{M}_{nn}(\mathbb{R})$ inversible. Alors $\chi(A) \geq 1$.

Proposition (Conditionnement χ_2 , cas symétrique)

Soit $C \in \mathcal{M}_{nn}(\mathbb{R})$ *symétrique* et inversible. Alors

$$\chi_2(C) = \rho(C)\rho(C^{-1}) = \frac{\max_{i=1,\dots,n} |\lambda_i(C)|}{\min_{i=1,\dots,n} |\lambda_i(C)|}$$

Conditionnement : exemple

Soit le système $Ax = b$ dans \mathbb{R}^4 :

$$A = \begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix}, b = \begin{bmatrix} 32 \\ 23 \\ 33 \\ 31 \end{bmatrix} \Rightarrow \text{sol. } x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{cases} \|b\|_\infty = 33, \\ \|x\|_\infty = 1 \\ \|A\|_\infty = \|(A_3)^T\|_1 = 33 \end{cases}$$

Conditionnement : exemple

Soit le système $Ax = b$ dans \mathbb{R}^4 :

$$A = \begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix}, b = \begin{bmatrix} 32 \\ 23 \\ 33 \\ 31 \end{bmatrix} \implies \text{sol. } x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{cases} \|b\|_\infty = 33, \\ \|x\|_\infty = 1 \\ \|A\|_\infty = \|(A_3)^T\|_1 = 33 \end{cases}$$

On perturbe le second membre : $b \rightarrow b + \delta b$

$$b + \delta b = \begin{bmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{bmatrix} \implies \text{solution } x + \delta x = \begin{bmatrix} 9.2 \\ -12.6 \\ 4.5 \\ 1.1 \end{bmatrix} \left(\begin{array}{l} \|\delta b\|_\infty = 10^{-1}, \\ \|\delta x\|_\infty = 13.6 \end{array} \right).$$

Conditionnement : exemple

Soit le système $Ax = b$ dans \mathbb{R}^4 :

$$A = \begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix}, b = \begin{bmatrix} 32 \\ 23 \\ 33 \\ 31 \end{bmatrix} \implies \text{sol. } x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{cases} \|b\|_\infty = 33, \\ \|x\|_\infty = 1 \\ \|A\|_\infty = \|(A_3)^T\|_1 = 33 \end{cases}$$

On perturbe le second membre : $b \rightarrow b + \delta b$

$$b + \delta b = \begin{bmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{bmatrix} \implies \text{solution } x + \delta x = \begin{bmatrix} 9.2 \\ -12.6 \\ 4.5 \\ 1.1 \end{bmatrix} \left(\begin{array}{l} \|\delta b\|_\infty = 10^{-1}, \\ \|\delta x\|_\infty = 13.6 \end{array} \right).$$

Écarts en norme infinie : $\frac{\|\delta b\|_\infty}{\|b\|_\infty} \approx 3 \cdot 10^{-3}$, $\frac{\|\delta x\|_\infty}{\|x\|_\infty} = 13.6$. Et donc : $\frac{\frac{\|\delta x\|_\infty}{\|x\|_\infty}}{\frac{\|\delta b\|_\infty}{\|b\|_\infty}} \approx 4500$.

On calcule $\|A\|_\infty = 33$ et $\|A^{-1}\|_\infty = \|(A_2^{-1})^T\|_1 = 136$, avec

$$A^{-1} = \begin{bmatrix} 25 & -41 & 10 & -6 \\ -41 & 68 & -17 & 10 \\ 10 & -17 & 5 & -3 \\ -6 & 10 & -3 & 2 \end{bmatrix}, \text{ et donc : } \chi_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty = 4488.$$

$\implies Ax = b$: système mal conditionné.

Théorème (conditionnement et erreur)

Soit A inversible dans $\mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n \setminus \{0\}$. On considère le système $Ax = b$.

- Si on perturbe le second membre :

$$A(x + \delta x) = b + \delta b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x\|} \leq \chi(A) \frac{\|\delta b\|}{\|b\|}.$$

- Si on perturbe la matrice (en supposant que $x + \delta x \neq 0$) :

$$(A + \delta A)(x + \delta x) = b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x + \delta x\|} \leq \chi(A) \frac{\|\delta A\|}{\|A\|}.$$

Théorème (conditionnement et erreur)

Soit A inversible dans $\mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n \setminus \{0\}$. On considère le système $Ax = b$.

- Si on perturbe le second membre :

$$A(x + \delta x) = b + \delta b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x\|} \leq \chi(A) \frac{\|\delta b\|}{\|b\|}.$$

- Si on perturbe la matrice (en supposant que $x + \delta x \neq 0$) :

$$(A + \delta A)(x + \delta x) = b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x + \delta x\|} \leq \chi(A) \frac{\|\delta A\|}{\|A\|}.$$

- Si le conditionnement est grand, une petite perturbation (sur b ou sur A) peut provoquer une grande erreur relative sur la solution : au lieu de trouver x (qu'on cherche), on obtient $x + \delta x$ qui peut être loin de x .

Théorème (conditionnement et erreur)

Soit A inversible dans $\mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n \setminus \{0\}$. On considère le système $Ax = b$.

- Si on perturbe le second membre :

$$A(x + \delta x) = b + \delta b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x\|} \leq \chi(A) \frac{\|\delta b\|}{\|b\|}.$$

- Si on perturbe la matrice (en supposant que $x + \delta x \neq 0$) :

$$(A + \delta A)(x + \delta x) = b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x + \delta x\|} \leq \chi(A) \frac{\|\delta A\|}{\|A\|}.$$

- Si le conditionnement est grand, une petite perturbation (sur b ou sur A) peut provoquer une grande erreur relative sur la solution : au lieu de trouver x (qu'on cherche), on obtient $x + \delta x$ qui peut être loin de x .
- Il y a toujours des erreurs (au moins l'erreur d'arrondi due aux flottants) : $\|\delta b\|$ et $\|\delta A\|$ sont $\geq \varepsilon_{\text{mach},2} \approx 10^{-16}$.

Théorème (conditionnement et erreur)

Soit A inversible dans $\mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n \setminus \{0\}$. On considère le système $Ax = b$.

- Si on perturbe le second membre :

$$A(x + \delta x) = b + \delta b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x\|} \leq \chi(A) \frac{\|\delta b\|}{\|b\|}.$$

- Si on perturbe la matrice (en supposant que $x + \delta x \neq 0$) :

$$(A + \delta A)(x + \delta x) = b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x + \delta x\|} \leq \chi(A) \frac{\|\delta A\|}{\|A\|}.$$

- Si le conditionnement est grand, une petite perturbation (sur b ou sur A) peut provoquer une grande erreur relative sur la solution : au lieu de trouver x (qu'on cherche), on obtient $x + \delta x$ qui peut être loin de x .
- Il y a toujours des erreurs (au moins l'erreur d'arrondi due aux flottants) : $\|\delta b\|$ et $\|\delta A\|$ sont $\geq \varepsilon_{\text{mach},2} \approx 10^{-16}$.
- Ces majorations sont optimales : il y a des cas où ce sont des égalités (choix de b et δb en fonction de A) : le pire peut arriver.

Théorème (conditionnement et erreur)

Soit A inversible dans $\mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n \setminus \{0\}$. On considère le système $Ax = b$.

- Si on perturbe le second membre :

$$A(x + \delta x) = b + \delta b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x\|} \leq \chi(A) \frac{\|\delta b\|}{\|b\|}.$$

- Si on perturbe la matrice (en supposant que $x + \delta x \neq 0$) :

$$(A + \delta A)(x + \delta x) = b, \quad \text{alors} \quad \frac{\|\delta x\|}{\|x + \delta x\|} \leq \chi(A) \frac{\|\delta A\|}{\|A\|}.$$

- Si le conditionnement est grand, une petite perturbation (sur b ou sur A) peut provoquer une grande erreur relative sur la solution : au lieu de trouver x (qu'on cherche), on obtient $x + \delta x$ qui peut être loin de x .
- Il y a toujours des erreurs (au moins l'erreur d'arrondi due aux flottants) : $\|\delta b\|$ et $\|\delta A\|$ sont $\geq \varepsilon_{\text{mach},2} \approx 10^{-16}$.
- Ces majorations sont optimales : il y a des cas où ce sont des égalités (choix de b et δb en fonction de A) : le pire peut arriver.
- Le conditionnement n'est pas lié à la méthode de résolution : c'est intrinsèque au problème. C'est le **système** qui est **mal conditionné**.