# Analyse Numérique SV

# Systèmes Linéaires

Simone Deparis

EPFL Lausanne – MATH

Printemps 2025

# Formulation du problème

On appelle système linéaire d'ordre $n$ ($n$ entier positif), une expression de la forme

$$A\mathbf{x} = \mathbf{b},$$

où $A = (a_{ij})$ est une matrice de taille $n \times n$ donnée, $\mathbf{b} = (b_j)$ est un vecteur colonne également donné et $\mathbf{x} = (x_j)$ est le vecteur des inconnues du système. La relation précédente équivaut aux $n$ équations

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

La matrice $A$ est dite *régulière (non singulière)* si $\det(A) \neq 0$. On a l'existence et l'unicité de la solution $\mathbf{x}$ (pour n'importe quel vecteur $\mathbf{b}$ donné) si et seulement si la matrice associée au système linéaire est régulière.

# Résumé

# Algorithme d'élimination de Gauss et factorisation $LU$

Une fois calculée la factorisation $LU$ de $A$ (coût $O(n^3)$), on peut résoudre le système linéaire $A\mathbf{x} = \mathbf{b}$ par les étapes

1. $L\mathbf{y} = \mathbf{b}$
2. $U\mathbf{x} = \mathbf{y}$

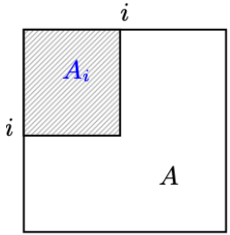
dont le coût est $O(n^2)$. (en fait $\mathbf{b} = L\mathbf{y} = LU\mathbf{x} = A\mathbf{x}$, donc $\mathbf{x}$ est bien la solution recherchée).

Remarque : si on doit résoudre deux systèmes linéaires $A\mathbf{x}_1 = \mathbf{b}_1$ et $A\mathbf{x}_2 = \mathbf{b}_2$ avec la même matrice, on calcule la factorisation $LU$ une seule fois.

# Condition nécessaire et suffisante

Des conditions restrictives sur $A$ sont nécessaires pour assurer que la méthode d'élimination de Gauss (et la factorisation LU) puissent se réaliser sans permutations :

Soit $d_i$ le déterminant de la $i$-ième sous-matrice principale $A_i$ :




Si les mineurs principaux $d_i$ de $A$ sont non nuls pour $i = 1, \ldots, n-1$ alors les pivots correspondants $a_{ii}^{(i)}$ sont également non nuls.

Celle-ci est une **condition nécessaire et suffisante** pour que la méthode de Gauss puisse être appliquée sans permutations.

# Conditions suffisantes

Voici des critères **suffisants** pour que la méthode de Gauss puisse être appliquée sans permutations.

1. Les matrices *à diagonale dominante par ligne*. Une matrice $A$ est dite à diagonale dominante par ligne si

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

2. Les matrices *à diagonale dominante par colonne*. Une matrice $A$ est dite à diagonale dominante par colonne si

$$|a_{jj}| \geq \sum_{i=1,\ldots,n;i\neq j} |a_{ij}|, \quad j = 1, \ldots, n.$$

3. Les matrices *symétriques définies positives*. Une matrice $A$ est symétrique si $A = A^T$ $(a_{ij} = a_{ji} \quad \forall i, j)$ ; elle est définie positive si toutes ses valeurs propres sont positives, c'est-à-dire : $\lambda_i(A) > 0, \quad i = 1, \ldots, n$.

# Factorisation $LU$ avec pivoting

$$LU = QA$$

$Q$ est une matrice orthogonale car les colonnes sont orthonormées. Donc $Q^{-1} = Q^T$. Du coup, pour $P = Q^T$, on a

$$A = PLU.$$

La matrice $Q$ ou $P$ sert à mémoriser les permutations effectuées.

Dans la pratique il est bien d'effectuer le pivoting même si l'élément diagonal n'est pas nul. Le mieux c'est de choisir comme pivot le coefficient le plus grand de la colonne.

# Factorisation $LU$ avec pivoting

**Théorème**

*Toute matrice $A \in \mathbb{R}^{n\times n}$ non singulière admet une factorisation $A = PLU$ où $L$ est une matrice triangulaire inférieure, $U$ une matrice triangulaire supérieure et $P$ une matrice de permutation.*

En Python, l'algorithme de factorisation $LU$ est disponible avec la commande `P,L,U=scipy.linalg.lu(A)` et on obtient $A = PLU$ ou $P^T A = LU$

Une fois calculée la factorisation $LU$ de $A$ avec pivoting (coût $O(n^3)$), on peut résoudre le système linéaire $A\mathbf{x} = \mathbf{b}$ ($\Leftrightarrow P^T A\mathbf{x} = P^T\mathbf{b}$) par les étapes

1. $L\mathbf{y} = P^T\mathbf{b}$
2. $U\mathbf{x} = \mathbf{y}$

dont le coût est $O(n^2)$.

# D'autres factorisations

- Factorisation de Choleski
- Factorisation $QR$
- La diagonalisation $PDP^{-1}$
- Factorisation $SVD$

# La factorisation de Cholesky

Dans le cas où $A$ est une matrice $n \times n$ symétrique et définie positive, il existe une unique matrice triangulaire inférieure $H$ avec les éléments diagonaux positifs telle que

$$\boxed{A = H^T H.}$$

Cette factorisation s'appelle la *factorisation de Cholesky*. Dans `scipy.linalg`, on peut utiliser la commande

```
H = cholesky(A)
```

Critère de Sylvester : une matrice symétrique $A \in \mathbb{R}^{n\times n}$ est définie positive si et seulement si les mineurs principaux de $A$ sont tous positifs.

# Considérations sur la précision

Les méthodes qu'on a vu jusqu'à maintenant sont des méthodes qui permettent de trouver la solution d'un système linéaire en un nombre fini d'opérations. C'est pourquoi on les appelle *méthodes directes*. Toutefois, il y a des cas où ces méthodes ne fonctionnent pas de manière satisfaisante.

## Définition

On définit le **conditionnement** d'une matrice $M$ symétrique définie positive comme le rapport entre la valeur maximale et minimale de ses valeurs propres, i.e.

$$\boxed{K(M) = \frac{\lambda_{max}(M)}{\lambda_{min}(M)}}$$

On peut montrer que plus le conditionnement de la matrice est grand, plus la solution du système linéaire obtenue par une méthode directe peut être mauvaise.
Par exemple, considérons un système linéaire $A\mathbf{x} = \mathbf{b}$.
Si on résout ce système avec un ordinateur, à cause des erreurs d'arrondis, on ne trouve pas la solution exacte mais une solution approchée $\hat{\mathbf{x}}$. On peut montrer la relation suivante :

$$\boxed{\frac{\|\mathbf{x}-\hat{\mathbf{x}}\|}{\|\mathbf{x}\|} \leq K(A)\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|}} \qquad (1)$$

où $\mathbf{r}$ est le résidu $\mathbf{r} = \mathbf{b} - A\hat{\mathbf{x}}$ ; on désigne par $\|\mathbf{v}\| = \left(\sum_{k=1}^{n} v_k^2\right)^{1/2}$ la norme euclidienne d'un vecteur $\mathbf{v}$.
On remarque que, si le conditionnement de $A$ est grand, la distance $\|\mathbf{x}-\hat{\mathbf{x}}\|$ entre la solution exacte et celle calculée numériquement peut être très grande même si le résidu est très petit.

# Resoudre avec $LU$ I

On considère le système linéaire $A\mathbf{x} = \mathbf{b}$ où :

$$A = \begin{pmatrix} 3 & 6 & 7 \\ 1 & 1 & 4 \\ 2 & 4 & 8 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix}.$$

a) Calculer la factorisation $LU$ de la matrice $A$ avec Python.

# Resoudre avec $LU$ II

```python
# importing libraries used in this book
import numpy as np
import scipy.linalg as linalg
import pprint

A = np.array([[3, 6, 7],
              [1, 1, 4],
              [2, 4, 8] ])

# LU factorisation with pivoting
P, L, U = linalg.lu(A)

print("A = P L U")
pprint.pprint(P.dot(L.dot(U)) )
```

# Resoudre avec $LU$ III

$$P = I_3 \qquad L = \begin{pmatrix} 1 & 0 & 0 \\ 1/3 & 1 & 0 \\ 2/3 & 0 & 1 \end{pmatrix} \qquad U = \begin{pmatrix} 3 & 6 & 7 \\ 0 & -1 & 5/3 \\ 0 & 0 & 10/3 \end{pmatrix} \qquad \mathbf{b} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix}.$$

b) Résoudre le système linéaire $A\mathbf{x} = \mathbf{b}$ en utilisant la factorisation trouvée au point précédent (Ne plus utiliser Python.)

1. $\mathbf{y} = L^{-1}\mathbf{b}$, $\mathbf{x} = U^{-1}\mathbf{y}$
2. $\mathbf{y} = U^{-1}\mathbf{b}$, $\mathbf{x} = L^{-1}\mathbf{y}$
3. $\mathbf{y} = U\mathbf{b}$, $\mathbf{x} = L\mathbf{y}$
4. $\mathbf{y} = L\mathbf{b}$, $\mathbf{x} = U\mathbf{y}$

# Resoudre avec $LU$ IV

$$\mathbf{x} = \begin{pmatrix} 3 \\ -2 \\ 1 \end{pmatrix}$$

c) Calculer le déterminant de la matrice $A$ en utilisant sa factorisation $LU$.

# Mineurs principaux I

**Les mineurs principaux** d'une matrice $A \in \mathbb{R}^{n\times n}$ sont les déterminants des matrices $A_p = (a_{i,j})_{1\leq i,j\leq p}$, $p = 1, ..., n$.
**Critère de Sylvester :** une matrice symétrique $A \in \mathbb{R}^{n\times n}$ est définie positive si et seulement si les mineurs principaux de $A$ sont tous positifs.

On considère le système linéaire $A\mathbf{x} = \mathbf{b}$ où

$$A = \begin{pmatrix} \varepsilon & 1 & 2 \\ 1 & 3 & 1 \\ 2 & 1 & 3 \end{pmatrix}.$$

Déterminer pour quelles valeurs du paramètre réel $\varepsilon \in \mathbb{R}$, la matrice $A$ est symétrique définie positive

- A $\varepsilon > 0$
- B $\varepsilon > \frac{1}{3}$
- C $\varepsilon > \frac{11}{8}$
- D $\frac{1}{3} < \varepsilon < \frac{11}{8}$
- E jamais

# Mineurs principaux II

En appliquant le critère de Sylvester, il suffit d'imposer

$$\begin{cases} \varepsilon > 0, \\ \det \begin{pmatrix} \varepsilon & 1 \\ 1 & 3 \end{pmatrix} = 3\varepsilon - 1 > 0, \\ \det A = 8\varepsilon - 11 > 0, \end{cases} \qquad \Rightarrow \qquad \varepsilon > \frac{11}{8}.$$

# Mineurs principaux III

Soit maintenant $\varepsilon = 0$. On veut résoudre le système $A\mathbf{x} = \mathbf{b}$ par une méthode directe ; quelle factorisation de la matrice $A$ envisageriez-vous ? Justifier votre réponse.

- A Elimination de Gauss
- B Factorisation LU sans permutation
- C Factorisation LU avec permutation
- D Factorisation de Cholevski
- E Cramer

# Mineurs principaux IV

Si $\varepsilon = 0$ la matrice $A$ est symétrique, mais elle n'est pas définie positive ; donc on ne peut pas calculer la factorisation de Cholesky. On utilisera la méthode d'élimination de Gauss avec changement de pivot, puisque $a_{11} = 0$ ; par exemple, on peut considérer la matrice de permutation $P$ par lignes :

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

On peut voir que $A = PLU$ avec

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & -5 & 1 \end{pmatrix} \qquad \text{et} \qquad U = \begin{pmatrix} 1 & 3 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 11 \end{pmatrix}.$$

# Mineurs principaux V

En considérant $\varepsilon = 2$, vérifier que dans ce cas la matrice $A$ est définie positive et calculer sa factorisation de Cholesky $A = H^T H$.

- A Elimination de Gauss
- B Factorisation LU sans permutation
- C Factorisation LU avec permutation
- D Factorisation de Cholevski
- E Cramer

En supposant que $\mathbf{b} = (1, 1, 1)^T$, résoudre le système linéaire $A\mathbf{x} = \mathbf{b}$ en utilisant la factorisation de Cholesky calculée au point c).

# Matrice de Hilbert et précision I

On considère le système $A\mathbf{x} = \mathbf{b}$, où $A$ est la matrice de Hilbert d'ordre 3

$$A = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix},$$

et $\mathbf{b}$ est choisi tel que $\mathbf{b} = A\mathbf{x}$, pour $\mathbf{x} = (1, 1, 1)^T$.

1. Calculer le conditionnement de la matrice $A$, en sachant que son spectre est $\{0.0027,\ 0.1223,\ 1.4083\}$.
2. Estimer l'erreur $\delta\mathbf{x}$ sur la solution $\mathbf{x}$ pour une perturbation $\delta\mathbf{b} = 10^{-10}(1, 1, 1)^T$, en observant que

$$\frac{||\delta\mathbf{x}||_2}{||\mathbf{x}||_2} = \frac{||(\mathbf{x} + \delta\mathbf{x}) - \mathbf{x}||_2}{||\mathbf{x}||_2} \leq K(A)\frac{||\delta\mathbf{b}||_2}{||\mathbf{b}||_2}.$$

# Matrice de Hilbert et précision I

# Matrice de Hilbert et précision II

# Matrice de Hilbert et précision III