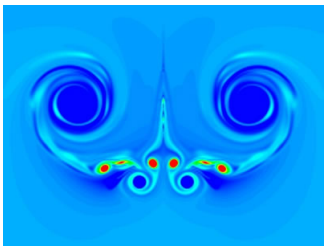
Département de Mathématiques

Eric DELABAERE

ANALYSE NUMÉRIQUE MATRICIELLE MASTER 1 MATH - MFA ET DS



Eric DELABAERE
Faculté de Sciences, Laboratoire de Mathématiques LAREMA, UMR CNRS 6093, Université d'Angers, 2 Boulevard Lavoisier, 49045 Angers Cedex 01, France.
E-mail : eric.delabaere@univ-angers.fr

ANALYSE NUMÉRIQUE MATRICIELLE
MASTER 1 MATH - MFA ET DS

Eric DELABAERE

Résumé. — *Objectif du cours* (40 heures - 30 séances): Complexité d'un algorithme; conditionnement d'une matrice; rayon spectral; systèmes linéaires: méthodes de Gauss, factorisation (P)LU, méthode de Cholesky, méthode QR; applications aux moindres carrés; méthode de Jacobi, méthode de Gauss-Seidel. Décompositions en valeurs propres et en valeurs singulières (SVD), recherche des valeurs propres : méthode de Jacobi, méthode QR, méthode des puissances.
Prérequis: Algèbre linéaire et bilinéaire en dimension finie (licence mathématiques L3); analyse numérique (licence mathématiques L3); langage Python .
Compétences attendues : A l'issue de la formation les étudiants ou stagiaires seront en capacité:
- *Complexité, normes matricielles, rayon spectral, conditionnement* (Estimation : 5 séances de 1h20)
  - d'évaluer la complexité d'un algorithme simple;
  - de manipuler et de calculer des normes de matrices et le rayon spectral;
  - de savoir expliquer l'impact du conditionnement d'une matrices en calcul numériques.
- *Systèmes linéaires et résolution par méthodes directes:* (Estimation : 13 séances de 1h20 dont 4 TP)
  - connaître les conditions d'application des méthodes suivantes de résolution directe de systèmes linéaires, savoir les expliquer et les mettre en oeuvre pour des matrices de petites tailles: méthodes de Gauss, factorisation LU et PLU, méthode de Cholesky, méthode QR;
  - savoir résoudre théoriquement et numériquement les problèmes de moindres carrés.
- *Systèmes linéaires et résolution par méthodes itératives:* (Estimation : 5 séances de 1h20 dont 2 TP)
  - connaître les conditions d'application des méthodes suivantes de résolution itérative de systèmes linéaires, savoir les expliquer et les mettre en oeuvre pour des matrices de petites tailles, savoir analyser leur convergence: méthode de Jacobi, méthode de Gauss-Seidel.
- *Recherche de valeurs propres:* (Estimation : 7 séances de 1h20 dont 2 TP)
  - connaître les conditions d'application des méthodes suivantes de recherche des valeurs propres, savoir les expliquer et les mettre en oeuvre pour des matrices de petites tailles, savoir analyser leur convergence : méthode des puissances, méthode de Jacobi, méthode QR.
- *De manière générale:* savoir expliquer ou construire un script Python des algorithmes précédents, en proposer des améliorations dans certains cadres applicatifs; connaître et savoir utiliser sous Python des bibliothèques de type numpy ou scipy.linalg; Dans des cas pratiques simples, savoir modéliser un problème menant à la résolution de systèmes linéaires, le traiter numériquement sous Python par application des résultats du cours, et être capable d'interpréter les résultats obtenus. (Mini projet)

Une partie de ce document s'appuie sur les notes de cours du Pr. Bernard Landreau, avec son aimable autorisation. Les erreurs éventuelles sont de la seule responsabilité de l'auteur.

TABLE DES MATIÈRES

CHAPITRE I

PRÉPARATIFS THÉORIQUES

I.1. Notions de complexité

Calcul scientifique. D'une manière générale, on peut définir le calcul scientifique comme l'utilisation de l'ordinateur en tant qu'outil de travail dans une discipline scientifique quelconque. Mais il est nécessaire de préciser les termes.

Calcul. En fait, le terme "calcul" peut être défini comme une succession d'opérations automatiques, basées sur des fonctions mathématiques telles que les opérations élémentaires ou la trigonométrie ou les logarithmes et exponentielles. En ce sens, le calcul a toujours existé. Ce que les ordinateurs ont apporté de révolutionnaire est, encore plus que l'automatisation, le changement d'échelle des problèmes à résoudre: l'exemple le plus frappant est sans doute ici la résolution de systèmes linéaires, dont l'ordre était inférieur à 10 dans les années 1950 et atteint aujourd'hui 10^4 lorsque la matrice est pleine, 10^6 lorsqu'elle est suffisamment creuse. La taille de mémoire et la capacité de calcul n'ont pas fini de grandir, la seule limite théorique dans la transmission des données étant la vitesse de la lumière. Il faut donc s'attendre à d'autres changements d'échelle dans les prochaines années, avec l'avénement des ordinateurs quantiques.

Le produit du calcul est le plus souvent une suite de nombres, mais peut également être une proposition mathématique. C'est déjà le cas pour le calcul dit "formel". Ne peut-on pas s'attendre à ce que l'ordinateur énonce un jour lui-même sa conjecture à partir des calculs qu'il aurait reçu l'ordre d'effectuer? L'intelligence artificielle est un sujet en plein essor, nul ne sait où il s'arrêtera.

Scientifique. La première remarque à faire est qu'il existe du calcul non scientifique, tel que la comptabilité d'une entreprise. Le calcul dit "scientifique" doit donc servir une démarche scientifique visant à l'élaboration d'une théorie ou à la confrontation avec la réalité. En ce sens, il sert de catalyseur aux nouvelles façons d'aborder les problèmes: nouveaux concepts, nouveaux types de raisonnement, nouvelles démarches de recherche. Dans une théorie, il produit le plus souvent des résultats quantitatifs mais aussi qualitatifs (par exemple, pour les systèmes dynamiques). Il peut ainsi jouer un rôle dans toutes les étapes d'un problème: élaboration et validation de modèles, choix et interprétation en temps réel d'expériences, réalisation et optimisation de produits, énoncé et vérification de conjectures. Il sert parfois à retrouver numériquement des résultats d'expériences, mais il a parfois de l'avance sur elles: citons pour l'instant la supraconductivité à haute température.

Formel. Le calcul formel (Computer Algebra en anglais) a commencé à se développer dans les années 1970. C'est la discipline dont l'objectif est de mettre au point des algorithmes efficaces pour la manipulation des objets formels utilisés par les mathématiciens : rationnels, nombres algébriques,

fonctions, polynômes, fractions rationnelles, séries... On parle aussi de calcul symbolique, il y a une subtilité entre les deux notions. Le calcul symbolique est quelques chose de plus général. Les objets manipulés sont formels, par exemple le rationnel $\frac{1}{3}$ est stocké tel quel et non pas sous la forme de nombre réel avec virgule flottante $0,3333\ldots$.

Algorithme. Un algorithme[(1)] est un ensemble fini d'instructions et d'opérations élémentaires qui définit complètement un traitement à effectuer sur des données et qui, en un temps fini, produit un résultat.
Les algorithmes que nous verrons dans ce cours seront présentés dans un langage standard aisément transcriptible en langage Python, Scilab, MAPLE ou C par exemple.

Complexité. Pour chaque algorithme étudié, il sera intéressant de se poser le problème de la complexité. En effet, lorsqu'il faudra choisir entre plusieurs algorithmes, il faudra connaître les caractéristiques de chacun à savoir :

- le temps de calcul requis (complexité en temps de calcul),
- la taille mémoire nécessaire (complexité en espace mémoire),
- et pour les calculs effectués sur des réels (nombres à virgule flottante) la précision du résultat.

La complexité d'un algorithme (en temps de calcul) est le nombre d'opérations élémentaires requises par l'exécution de l'algorithme, ce nombre étant exprimé en fonction des données. Les opérations élémentaires sont les opérations arithmétiques de base : additions, multiplications, divisions ainsi que les comparaisons. Il ne faut pas occulter l'aspect accès à la mémoire. Des accès répétés à des zones de mémoire non contigues peuvent augmenter considérablement le temps de calcul. Il faut veiller à la taille de la mémoire cache ainsi qu'à l'ordonnancement des accès mémoire notamment pour les tableaux et matrices volumineuses. Il faut aussi veiller à la taille des objets manipulés (nombres, matrices).

On utilisera souvent une estimation de la complexité à l'aide des notations de Landau.
Rappels: si f et g sont des fonctions définie sur $\mathbb{N}$ avec g positive

- $f = o(g)$ signifie que f/g tend vers 0 à l'infini (f est négligeable par rapport à g),
- $f = O(g)$ signifie qu'il existe $K > 0$ tel que pour tout n on a $|f(n)| \leq Kg(n)$.
- $f \asymp g$ signifie qu'il existe $A, B > 0$ tels que $Ag(n) \leq |f(n)| \leq Bg(n)$

Pour le calcul de la complexité, il n'est pas toujours facile de compter le nombre d'opérations élémentaires qui peut être variable suivant les branchements conditionnels de l'algorithme. On donnera alors parfois plusieurs complexités:

- complexité dans le pire des cas,
- complexité en moyenne (calculée ou établie à l'aide de tests statistiques),
- complexité dans le meilleur des cas.

Notation. On notera $N_{op}(n)$ la complexité (en temps de calcul) qui désigne le nombre d'opérations nécessaires pour un objet de taille n.

Les différentes algorithmes que l'on rencontre habituellement seront de type:

(1) Ce mot vient du mathématicien persan (9ème siècle) **Abu Abdallah Muhammad ibn Musa-al-Khwarizmi**; on lui doit un ouvrage célèbre le "Kitab **al jabr** wal-muqabala" (étude des équations linéaires et quadratiques) qui a donné naissance au mot familier "algèbre".

- logarithmiques $N_{op}(n) \asymp \ln n$. Exemple : algorithme d'Euclide.
- linéaire $N_{op}(n) \asymp n$. Exemples : calcule d'un produit scalaire dans $\mathbb{R}^n$.
- quasi-linéaire $N_{op}(n) \asymp n \ln n$. Exemples : algorithme de tri rapide
- polynomiaux, $N_{op}(n) \asymp n^k$. Exemples : algorithme de tri élémentaires $N_{op}(n) = O(n^2)$, algorithme de Gauss $N_{op}(n) = O(n^3)$,
- exponentiels, $N_{op}(n) \asymp a^n$. Exemple : tour de Hanoï

Les algorithmes logarithmiques sont très rapides, les algorithmes linéaires et quasi-linéaires aussi, les algorithmes polynomiaux sont lents mais encore praticables ($O(n^2)$ ou $O(n^3)$, tout dépend du degré), les algorithmes exponentiels sont inutilisables.

Remarque I.1.1.

1. Signalons qu'il est parfois difficile de séparer la complexité d'un algorithme de la machine sur lequel il est implanté.
2. Attention aux valeurs de constantes cachées dans les O.

L'unité de mesure de vitesse d'un ordinateur est le flops (Floating Point Operation Per Second). L'ordre de grandeur pour les ordinateurs classiques récents est le Teraflops (10^{12} flops) voire le Petaflops (10^{15} flops) pour les récents supercalculateurs (comme ceux d'Atos produits à Angers) qui vise le Exaflops (10^{18} flops) avant 2030.

Exemple I.1.2 (Exercice)**.**

Calculer la complexité du calcul d'un déterminant de taille n en utilisant la formule avec permutations: $\det \begin{vmatrix} a_{11} \dots a_{1n} \\ \vdots \\ a_{n1} \dots a_{nn} \end{vmatrix} = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) a_{1\sigma(1)} \dots a_{n\sigma(n)}$.

Exemple I.1.3 (Exercice)**.**

Calculer la complexité:

1. d'un produit scalaire de vecteurs,
2. d'un produit matrice-vecteur,
3. d'un produit matrice-matrice.

I.2. Normes matricielles, rayon spectral

Dans tout le cours, $\mathbb{K}$ désignera le corps $\mathbb{R}$ ou $\mathbb{C}$.

I.2.1. Quelques notation et rappels. —

Définition I.2.1.

Soit $A \in \mathcal{M}_n(\mathbb{K})$ une matrice carrée.

1. La matrice ${}^tA$ est la transposée de A, aussi notée A^T.

2. La matrice A est symétrique si ${}^tA = A$.
3. La matrice A est dite orthogonale ssi ${}^tAA = I_n$ ($K = \mathbb{R}$). Le groupe des matrices orthogonales est noté $O_n(\mathbb{R})$
4. La matrice $\overline{A}$ est la matrice complexe conjuguée de A.
5. La matrice adjointe de A est $A^\star = {}^t\overline{A}$. (Donc $A^\star = {}^tA$ si $A \in \mathcal{M}_n(\mathbb{R})$).
6. La matrice A est dite hermitienne ou auto-adjointe ssi $A^\star = A$.
7. La matrice A est dite unitaire ssi $A^\star A = I_n$ ($\mathbb{K} = \mathbb{C}$). Le groupe des matrices unitaires est noté $\mathcal{U}_n(\mathbb{C})$

Proposition I.2.2.

1. *La matrice $A \in \mathcal{M}_n(\mathbb{R})$ est symétrique (${}^tA = A$) ssi il existe $Q \in \mathcal{O}_n(\mathbb{R})$ orthogonale et $D \in \mathcal{M}_n(\mathbb{R})$ diagonale telles que $A = QD\,{}^tQ$.*
2. *La matrice $A \in \mathcal{M}_n(\mathbb{C})$ est hermitienne ($A^\star = A$) ssi il existe $U \in \mathcal{U}_n(\mathbb{C})$ unitaire et $D \in \mathcal{M}_n(\mathbb{R})$ diagonale telles que $A = UDU^\star$.*

Remarque I.2.3.
Autrement dit:

1. La matrice $A \in \mathcal{M}_n(\mathbb{R})$ est symétrique $\Leftrightarrow$ les valeurs propres de A sont réelles et A se diagonalise dans une base orthonormée (pour le produit scalaire naturel $\langle x, y\rangle = {}^txy = \sum x_iy_i$ de $\mathbb{R}^n$) de vecteurs propres.
2. La matrice $A \in \mathcal{M}_n(\mathbb{C})$ est hermitienne $\Leftrightarrow$ les valeurs propres de A sont réelles et A se diagonalise dans une base orthonormée (pour le produit hermitien naturel $\langle x, y\rangle = {}^t\overline{x}y = \sum \overline{x}_iy_i$ de $\mathbb{C}^n$) de vecteurs propres.

I.2.2. Normes matricielles. — Parmi toutes les normes que l'on peut définir sur les matrices de l'espace vectoriel $\mathcal{M}_n(\mathbb{K})$, on utilisera de préférence les normes dites matricielles.

Définition I.2.4.

Une norme $\|\cdot\|$ sur $\mathcal{M}_n(\mathbb{K})$ est dite **matricielle** si elle vérifie la propriété suivante (dite de sous-multiplicativité) bien commode: $\forall A, B \in \mathcal{M}_n(\mathbb{K}), \quad \|AB\| \leq \|A\| \times \|B\|$.


L'expression "norme matricielle" prête à confusion. Cela ne veut pas simplement dire que c'est une norme sur les matrices, cela inclut une propriété particulière.
Toutes les normes sur les matrices ne sont pas matricielles, par exemple la norme $\|\ \|$ définie comme le maximum des modules des coefficients ne l'est pas (considérer la matrice A composée uniquement de 1, comparer $\|A\|$ et $\|A^2\|$).

Parmi toutes les normes matricielles, on utilisera souvent les normes issues de normes sur les vecteurs.

Proposition I.2.5 (Norme subordonnée à une norme vectorielle)**.**

Soit N une norme sur $\mathbb{K}^n$. L'application qui à $A \in \mathcal{M}_n(\mathbb{K})$, associe

$$\|A\| := \sup_{x \neq 0} \frac{N(Ax)}{N(x)} = \sup_{N(x)=1} N(Ax)$$

est une norme matricielle sur $\mathcal{M}_n(\mathbb{K})$ dite norme subordonnée ou associée à N ou bien encore induite par N

Démonstration. — Cf. TD. □

Remarque I.2.6.

1. Pour une norme matricielle $\|\cdot\|$ induite par une norme vectorielle N, on a :
$$\boxed{\forall x \in \mathbb{K}^n, \quad N(Ax) \leq \|A\|.N(x).}$$
De plus, comme le sup est atteint (exercice: pourquoi ?), c'est un maximum: il existe $x_0 \in \mathbb{K}^n$ tel que $N(Ax_0) = \|A\|.N(x_0)$.
Interpétation : le nombre réel $\|A\|$ est en quelque sorte le **coefficient d'amplification maximale** lorsqu'on fait agir la matrice A sur tous les vecteurs de $\mathbb{K}^n$.
2. Pour une norme matricielle $\|\cdot\|$ induite par une norme vectorielle, on a toujours $\|I_n\| = 1$.
3. Il existe des normes matricielles qui ne sont pas induites par des normes vectorielles. Par exemple la norme de Frobénius $\|\cdot\|_F$ (appelée aussi norme de Schur), définie par : $\|A\|_F^2 = \text{Trace}(A^\star A) = \sum_{i=1}^{n}\sum_{j=1}^{n} |a_{i,j}|^2$, pour laquelle $\|I_n\|_F = \sqrt{n} \neq 1$.

Proposition I.2.7 (Normes induites usuelles)**.**

Les normes matricielles induites par les normes vectorielles usuelles $\|\cdot\|_1$, $\|\cdot\|_2$ et $\|\cdot\|_\infty$ sont respectivement

$$\|A\|_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{i,j}|, \quad \|A\|_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{i,j}|, \quad \|A\|_2 = \sqrt{\rho}.$$

où ρ est la plus grande des valeurs propres de la matrice hermitienne (ou symétrique) ${}^t\overline{A}A$.

Démonstration. — Cf. TD. □

Remarque I.2.8.

Si A est une matrice carrée (et même rectangulaire), la matrice ${}^t\overline{A}A$ est hermitienne ou symétrique positive, donc toutes ses valeurs propres sont réelles positives. En effet ${}^t\overline{A}Ax = \lambda x \Rightarrow {}^t\overline{x}\,{}^t\overline{A}Ax = \lambda {}^t\overline{x}\,x \Rightarrow \|Ax\|_2^2 = \lambda\|x\|_2^2 \Rightarrow \lambda \geq 0$. Les racines carrées des valeurs propres de ${}^t\overline{A}A$ s'appellent les **valeurs singulières de la matrice** A.

Exemple I.2.9.

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

1. $\|A\|_1 = \max\{2+3, 2+1\} = 5$.
2. $\|A\|_\infty = \max\{2+2, 3+1\} = 4$.
3. ${}^t\overline{A}A = \begin{pmatrix} 2 & 3 \\ -2 & 1 \end{pmatrix}\begin{pmatrix} 2 & -2 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 13 & -1 \\ -1 & 5 \end{pmatrix}$. On a $\det(\lambda I_2 - {}^t\overline{A}A) = \lambda^2 - 18\lambda + 64$. L'ensemble spectral de la matrice symétrique ${}^t\overline{A}A$ est donc $\mathrm{Sp}({}^t\overline{A}A) = \{9 - \sqrt{17}, 9 + \sqrt{17}\}$. On en déduit que $\|A\|_2 = \sqrt{9 + \sqrt{17}}$.

I.2.3. Rayon spectral, lien avec les normes. —

Définition I.2.10 (Rayon spectral)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$. Le rayon spectral de la matrice A, noté $\rho(A)$, est le nombre réel défini par:

$$\rho(A) = \max_{\lambda \in \mathrm{Sp}(A)} |\lambda|.$$

où $\mathrm{Sp}(A)$ désigne l'ensemble spectral de M.

Remarque I.2.11.
Le rayon spectral est une mesure de la matrice mais ce n'est pas une norme car il peut être nul sans que la matrice soit nulle. Par exemple, pour les matrices nilpotentes.

On a la proposition suivante:

Proposition I.2.12 (Lien rayon spectral et norme induite)**.**

Soit $\|\cdot\|$ une norme sur $\mathcal{M}_n(\mathbb{K})$ induite par une norme vectorielle. Alors:

1. *pour tout $A \in \mathcal{M}_n(\mathbb{K})$, $\quad \rho(A) \leq \|A\|$.*
2. *pour toute matrice A et tout $\epsilon > 0$, il existe une norme induite par une norme vectorielle $\|\cdot\|$ (qui dépend de A et de ϵ) telle que: $\|A\| \leq \rho(A) + \epsilon$.*

Démonstration. — On suppose que $\|\cdot\|$ est une norme sur $\mathcal{M}_n(\mathbb{K})$ induite par la norme vectorielle N sur K^n: $\|A\| = \sup_{x \neq 0} \dfrac{N(Ax)}{N(x)} = \sup_{N(x)=1} N(Ax)$.

1. Supposons que $\lambda \in \mathbb{C}$ soit valeur propre de A telle que $\rho(A) = |\lambda|$. A λ est associé le vecteur propre $x_\lambda \neq 0$. Alors $N(Ax_\lambda) = |\lambda| N(x_\lambda)$ de sorte que $|\lambda| = \dfrac{N(Ax_\lambda)}{N(x_\lambda)} \leq \|A\|$. Par suite $\rho(A) \leq \|A\|$.
2. Admis.

□

Conséquence : en appliquant la proposition précédente aux normes matricielles usuelles $\|\cdot\|_1$ et $\|\cdot\|_\infty$, on obtient la majoration

$$\rho(A) \leq \min\Big(\max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{i,j}|, \max_{1\leq i\leq n} \sum_{j=1}^{n} |a_{i,j}| \Big).$$

I.2.4. Localisation des valeurs propres. — Un résultat simple et pratique permet de localiser le spectre d'une matrice. C'est le Théorème de Gerschgörin [(2)]-Hadamard[(3)].

Théorème I.2.13 (Gerschgörin-Hadamard)**.**

Soit $A = (a_{i,j})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$ une matrice carrée d'ordre n. Toute valeur propre de A appartient à l'un des disques $D_i \subset \mathbb{C}$ (appelés disques de Gerschgörin), $i = 1, ..., n$, définis par: $D_i = \Big\{ z \in \mathbb{C} \mid |z - a_{i,i}| \leq \sum_{j=1, j\neq i}^{n} |a_{i,j}| \Big\}$. Autrement dit, $\mathrm{Sp}(A) \subset \bigcup_{1\leq i\leq n} D_i$.

Démonstration. — En TD. □

Remarque I.2.14.

Comme une matrice A et sa transposée ${}^tA$ ont le même spectre (exercice: pourquoi ?), le résultat est aussi valide pour les sommes sur les colonnes.

Ce théorème est utile pour les matrices à diagonale strictement dominante.

Définition I.2.15 (diagonale strictement dominante)**.**

Une matrice carrée $A = (a_{i,j})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$ est dite à diagonale strictement dominante (sur les lignes) lorsque : $\forall i = 1, ..., n,\ |a_{i,i}| > \sum_{j=1, j\neq i}^{n} |a_{i,j}|$.

On dispose alors des résultats suivants.

Proposition I.2.16.

Une matrice $A = (a_{i,j})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$ à diagonale strictement dominante (sur les lignes) est inversible. De plus, si $\ell_i = |a_{i,i}| - \sum_{j=1, j\neq i}^{n} |a_{i,j}|$ et si $\ell = \min_{1\leq i\leq n} \ell_i$, alors $\|A^{-1}\|_\infty \leq \frac{1}{\ell}$.

Démonstration. — Cf. projet. □

Remarque I.2.17 (Rappel)**.**

Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice réelle symétrique.

1. A est définie positive si et seulement si $\forall x \in \mathbb{R}^n$, ${}^txAx \geq 0$ avec égalité seulement si $x = 0$.

[(2)] Semyon Aronovich Gerschgörin, 1901-1933, mathématicen biélorusse.
[(3)] Jacques Salomon Hadamard, 1865-1963, mathématicien français.

2. Si A est définie positive alors A est inversible (on a $Ax = 0 \Rightarrow {}^t xAx = 0 \Rightarrow x = 0$).
3. Si A est définie positive, ses valeurs propres sont réelles strictement positives (on a $Ax = \lambda x$ avec $\lambda \in \mathbb{R}$ et $x \neq 0 \Rightarrow {}^t xAx > 0 \Rightarrow \lambda \|x\|_2^2 > 0 \Rightarrow \lambda > 0$.)

Proposition I.2.18.

Une matrice réelle $A \in \mathcal{M}_n(\mathbb{R})$, symétrique, à diagonale strictement dominante, et dont les coefficients diagonaux sont strictement positifs, est définie positive.

Démonstration. — En TD. □

Exemple I.2.19.

1. La matrice $A = \begin{pmatrix} 2 & -1 & 0 \\ 1 & -3 & 1 \\ 1 & -1 & 3 \end{pmatrix}$ est à diagonale strictement dominante sur les lignes, elle est donc inversible.
2. La matrice $B = \begin{pmatrix} 2 & 1 & 1 \\ -1 & -3 & -1 \\ 0 & 1 & 3 \end{pmatrix}$ n'est pas à diagonale strictement dominante sur les lignes, mais ${}^t B = A$ l'est. (Autrement dit, B est à diagonale strictement dominante sur les colonnes.) Donc B est inversible.
3. La matrice $C = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 3 & 1 \\ 0 & 1 & 3 \end{pmatrix}$ est symétrique, à diagonale strictement positive et strictement dominante, elle est donc définie positive.

I.3. Convergence des suites matricielles

Dans les méthodes itératives de résolution de systèmes linéaires (que nous verrons au chapitre III), on utilise des suites vectorielles $(x_n \in \mathbb{K}^m)_n$ vérifiant une relation de récurrence du type $x_{n+1} = Px_n + q$ où P est la matrice d'itération de la méthode. Si cette suite converge vers la solution $x^\star$ alors la suite $(x_n - x^\star)_n$ vérifie $(x_{n+1} - x^\star) = P(x_n - x^\star)$ ce qui conduit à $x_n - x^\star = P^n(x_0 - x^\star)$.

La question qui se pose est donc de savoir à quelle condition, étant donné une matrice P et un vecteur y_0 la suite $(P^n y_0)$ converge vers le vecteur nul. Nous avons dans ce sens la proposition importante suivante.

Proposition I.3.1.

Soit $P \in \mathcal{M}_m(\mathbb{K})$ une matrice. La suite $(x_n = P^n x_0)$ converge vers 0 quel que soit $x_0 \in \mathbb{K}^m$ si et seulement si $\rho(P) < 1$.

Démonstration. — Nous rappelons le résultat suivant:

Remarque I.3.2.
Sur $\mathbb{K}^m$ et sur $\mathcal{M}_m(\mathbb{K})$ toutes les normes sont équivalentes (car ce sont sont des espaces vectoriels normés de dimension finie)

Ainsi, dans la preuve suivante, on pourra utiliser n'importe quelle norme.

1. Il est facile de montrer que la condition $\rho(P) < 1$ est nécessaire. Supposons le contraire, dans ce cas, il existe une valeur propre λ telle que $|\lambda| \geq 1$. Soit x_0 un vecteur propre (donc non nul) associé. Alors $P^n x_0 = \lambda^n x_0$ et $\|P^n x_0\| = |\lambda|^n \|x_0\|$ ne tend pas vers zéro puisque $|\lambda| \geq 1$.
2. Pour montrer que la condition $\rho(P) < 1$ est suffisante, c'est plus difficile.
   (a) D'abord, il faut remarquer que
   $$\forall x_0 \in \mathbb{K}^m, \lim_{n\to+\infty} (P^n x_0) = 0 \Leftrightarrow \lim_{n\to+\infty} \|P^n\| = 0 \Leftrightarrow \lim_{n\to+\infty} P^n = 0.$$
   En effet, si on suppose que $\forall x_0 \in \mathbb{K}^m$, $\lim_{n\to+\infty} (P^n x_0) = 0$, en prenant successivement, $x_0 = e_1, \ldots, e_m$ (les vecteurs de la base canonique), et la norme $\|.\|_1$, on obtient que les normes $\|.\|_1$ des colonnes de P^n tendent chacune vers 0, donc la norme $\|.\|_1$ de la matrice P^n tend aussi vers 0.
   (b) Réciproquement si $\|P^n\| \to 0$, en utilisant pour une norme subordonnée à une norme vectorielle la majoration $\|P^n x_0\| \leq \|P^n\| \times \|x_0\|$, il est clair que $(P^n x_0) \to 0$ pour tout x_0.
   (c) Il reste donc à montrer que $\rho(P) < 1 \Rightarrow \lim_{n\to+\infty} \|P^n\| = 0$.
   Dans le cas diagonalisable, on peut facilement démontrer cela. Dans ce cas, on sait qu'il existe une matrice inversible S et une matrice diagonale D telles que $P = S^{-1}DS$. On a alors $\rho(P) = \rho(D)$ et pour tout $n \geq 0$, $P^n = S^{-1}D^nS$. On a pour toute norme matricielle $\|.\|$, $\|P^n\| \leq \|S^{-1}\| \times \|D^n\| \times \|S\|$, et puisque $D^n = SP^nS^{-1}$: $\|D^n\| \leq \|S\| \times \|P^n\| \times \|S^{-1}\|$. Donc, la suite de matrices P^n tend vers 0 si et seulement si la suite D^n tend vers 0. De plus si $D = \mathrm{Diag}(\lambda_1, \ldots, \lambda_m)$, on a aisément $D^n = \mathrm{Diag}(\lambda_1^n, \ldots, \lambda_m^n)$. Pour les normes $\|.\|_p$, $(p = 1, 2, \infty)$ on a donc: $\|D^n\|_p = \rho(D^n) = \rho(D)^n = \rho(P)^n$. On en déduit que pour ces normes
   $$\lim_{n\to\infty} D^n = 0 \Leftrightarrow \lim_{n\to\infty} P^n = 0 \Leftrightarrow \lim_{n\to\infty} \rho(D)^n = 0 \Leftrightarrow \rho(D) < 1 \Leftrightarrow \rho(P) < 1$$
   Cela démontre directement la condition suffisante (et même l'équivalence) dans le cas diagonalisable.
   Dans le cas général, on utilise la Proposition I.2.12 vue plus haut qui relie le rayon spectral et les normes induites. Avec ce résultat, si $\rho(P) < 1$, on prend $\epsilon > 0$ tel que $\rho(P) + \epsilon < 1$ et la norme donnée par la proposition. Il suffit de dire alors que pour une norme subordonnée à une norme vectorielle $\|P^n\| \leq \|P\|^n$ ce qui assure la convergence de P^n vers 0.
   Variante: on peut écrire $P = S^{-1}(D + N)S$ avec $D = \mathrm{Diag}(\lambda_1, \ldots, \lambda_m)$ et N nilpotente (donc $N^m = 0$) telle que $DN = ND$. Ainsi $P^n = S^{-1}\left(\sum_{k=0}^{m-1} \binom{n}{k} N^k D^{n-k}\right) S$. Pour la même raison que ci-avant, on a $P^n \underset{n\to\infty}{\longrightarrow} 0 \Leftrightarrow \sum_{k=0}^{m-1} \binom{n}{k} N^k D^{n-k} \underset{n\to\infty}{\longrightarrow} 0$. Or

$$\|\sum_{k=0}^{m-1} \binom{n}{k} N^k D^{n-k}\|_p \leq \sum_{k=0}^{m-1} \binom{n}{k} \|N\|_p^k \|D\|_p^{n-k} \leq M \sum_{k=0}^{m-1} \binom{n}{k} \rho(D)^{n-k} \underset{n\to\infty}{\longrightarrow} 0 \text{ si } \rho(D) = \rho(P) < 1,$$

où on a noté $M = \max_{0\leq k\leq m-1} \|N\|_p^k$.

□

Remarque I.3.3.
Il est aussi évident, d'après la démonstration du théorème, que la convergence sera d'autant plus rapide que la plus grande valeur absolue des valeurs propres est petite. Tout le problème consistera à trouver une matrice P pour laquelle ceci soit le cas. Cf. chapitre III.

I.4. Conditionnement

Le conditionnement va quantifier la sensibilité d'une matrice A aux perturbations qui peuvent se produire sur les données lors de la résolution d'un système linéaire $Ax = b$.

On dit qu'un problème est **mal conditionné** lorsque de petites variations sur les données entrainent de fortes variations sur le résultat.

Voyons un exemple, la résolution du système linéaire $AX = b$ avec $A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}, b = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$,
qui a pour solution $X = {}^t(1, 1, 1, 1)$. Perturbons un peu le système sur b :

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix} = \begin{pmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{pmatrix}$$

La solution est devenue : $X_1 = {}^t(9.2, -12.6, 4.5, -1.1)$. On observe qu'une erreur relative de l'ordre de $1/200$ entraine une erreur relative de l'ordre de $10/1$.
Perturbons maintenant un peu le système sur A :

$$\begin{pmatrix} 10 & 7 & 8.1 & 7.2 \\ 7.08 & 5.04 & 6 & 5 \\ 8 & 5.98 & 9.89 & 9 \\ 6.99 & 4.99 & 9 & 9.98 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

La solution est devenue : $X_2 = {}^t(-81, 137, -34, 22)$. Là encore, des petites variations sur A entrainent de grandes variations sur le résultat.

Définition I.4.1.
On appelle conditionnement de la matrice inversible $A \in \mathcal{M}_n(\mathbb{K})$ pour la norme $\|.\|$ subordonnée à une norme vectorielle, la quantité:

$$\text{cond}(A) = \|A\| \times \|A^{-1}\|.$$

On note $\text{cond}_p(A)$ pour le conditionnement relatif à la norme $\|.\|_p$ pour $p = 1, 2, \infty$.

Remarque I.4.2.
En anglais, on parle de "condition number" et on le note $\kappa(A)$.

Voici les propriétés principales du conditionnement:

Proposition I.4.3.

Pour toutes matrices inversibles $A, B \in \mathcal{M}_n(\mathbb{K})$, on a:

1. $\text{cond}(A^{-1}) = \text{cond}\, A$;
2. $\text{cond}(\lambda A) = \text{cond}(A)$ *pour tout* $\lambda \in \mathbb{K}^\star$;
3. $\text{cond}(AB) \leq \text{cond}(A)\,\text{cond}(B)$;
4. $\text{cond}(A) \geq 1$;
5. $\text{cond}(A) \geq \dfrac{\max_{\lambda \in \text{Sp}(A)} |\lambda|}{\min_{\lambda \in \text{Sp}(A)} |\lambda|}$,;
6. *si A est orthogonale (resp. unitaire) alors* $\text{cond}_2(A) = 1$;
7. $\text{cond}_2(UA) = \text{cond}_2(AU) = \text{cond}_2(A)$ *pour toute matrice orthogonale (resp. unitaire) U;*
8. $\text{cond}_2(A) = \dfrac{\mu_{max}}{\mu_{min}}$ *où μ_{max} et μ_{min} sont respectivement la plus grande (resp. la plus petite) valeur singulière de A.*

   En particulier, si A est hermitienne, $\text{cond}_2(A) = \dfrac{\max_{\lambda \in \text{Sp}(A)} |\lambda|}{\min_{\lambda \in \text{Sp}(A)} |\lambda|}$.

Démonstration. — Cf. TD. □

Intérêt du conditionnement : On considère un système linéaire $Ax = b$ et on s'intéresse à l'influence sur la solution x de perturbations opérées sur b ou sur A. On se place dans le cadre d'une norme vectorielle et d'une norme matricielle subordonnée. On utilise le conditionnement relatif à ces deux normes. On a les deux propositions suivantes qui font intervenir le conditionnement.

Proposition I.4.4 (Perturbation sur b)**.**

Soit $Ax = b$ un système linéaire $n \times n$ où la matrice A est supposée inversible et $b \neq 0$. Si on note $x + \Delta x$ la solution du système linéaire $A(x + \Delta x) = b + \Delta b$, on a l'inégalité:

$$\frac{||\Delta(x)||}{||x||} \leq \text{cond}(A)\frac{||\Delta(b)||}{||b||}.$$

De plus, la majoration est optimale au sens où pour chaque matrice A, il existe un vecteur b non nul et une perturbation Δb non nulle qui réalisent l'égalité.

Démonstration. — Cf. TD. □

Proposition I.4.5 (Perturbation sur A)**.**

Soit $Ax = b$ un système linéaire $n \times n$ où la matrice A est supposée inversible et $b \neq 0$. Si on note $x + \Delta x$ la solution du système linéaire $(A + \Delta A)(x + \Delta x) = b$, on a l'inégalité:

$$\frac{||\Delta(x)||}{||x + \Delta(x)||} \leq \text{cond}(A)\frac{||\Delta A||}{||A||}.$$

De plus, la majoration est optimale au sens où pour chaque matrice A, il existe un vecteur b non nul et une perturbation ΔA non nulle qui réalisent l'égalité.

Démonstration. — Vue en TD. □

On voit donc que le conditionnement d'une matrice A sert de borne maximale dans l'amplification des erreurs relatives qui peuvent se produire lors de la résolution d'un système linéaire $Ax = b$.
Un exemple de matrices réputées avoir un fort conditionnement : les matrices de Hilbert. On appelle matrice de Hilbert H_n la matrice symétrique

$$H_n := \left(\frac{1}{i+j+1}\right)_{0\leq i,j\leq n-1}.$$

On obtient cette matrice comme matrice de Gram (matrice de produits scalaires) pour la famille de fonctions polynomiales $x \mapsto x^k$, $0 \leq k \leq n-1$, avec le produit scalaire sur $C([0,1],\mathbb{R})$:

$$\langle f, g\rangle = \int_0^1 f(t)g(t)dt.$$

Par exemple $H_4 = \begin{pmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1/2 & 1/3 & 1/4 & 1/5 \\ 1/3 & 1/4 & 1/5 & 1/6 \\ 1/4 & 1/5 & 1/6 & 1/7 \end{pmatrix}$, on trouve déjà que $\text{cond}_2(H_4)$ est de l'ordre de 15000.
Et $\text{cond}(H_8)$, est de l'ordre de 10^{10}!

A retenir : On évitera les systèmes linéaires avec des matrices à mauvais conditionnement, trop sujets à amplification des erreurs. Il faudra alors transformer le système pour arriver à une meillleure situation.

CHAPITRE II

SYSTÈMES LINÉAIRES : RÉSOLUTION PAR DES MÉTHODES DIRECTES

II.1. Introduction

La plupart des problèmes de mathématiques appliquées se ramènent à un problème de résolution de système linéaire. Par exemple, résolution d'un problème d'approximation par la méthode des moindres carrés, résolution d'une équation différentielle par discrétisation... On va donc voir dans ce chapitre les différents moyens pratiques de résoudre un système linéaire. Il y a deux types de méthodes :

- les méthodes directes (dites encore exactes), elles permettent de calculer en un nombre fini d'opérations la valeur exacte[(1)] de la solution;
- les méthodes itératives, elles permettent de calculer une valeur approchée de la solution en itérant un calcul. En augmentant le nombre d'itérations, on augmente en général la précision du résultat.

Le décor. On considère dans ce chapitre des systèmes linéaires carrés de n équations linéaires à n inconnues que l'on écrit sous forme matricielle $Ax = b$, où $A = (a_{ij})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{K})$ est la matrice carrée du système, $\mathbb{K} = \mathbb{R}, \mathbb{C}$ ou $\mathbb{Q}$, $b = {}^t(b_1, \ldots, b_n)$ est le vecteur colonne du second membre, et $x = {}^t(x_1, \ldots, x_n)$ est le vecteur colonne inconnu solution du système.

Notation. On notera pour des raisons de commodité typographique ${}^t(x_1, \ldots, x_n)$ pour désigner le vecteur colonne $\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, le symbole ^{t} signifiant la transposition.

Le point de vue théorique. On sait que lorsque la matrice A est inversible (i.e. lorsque son déterminant est non nul) il existe une solution unique que l'on peut écrire formellement comme $x = A^{-1}b$ mais trouver x sous cette forme pose le problème d'inverser la matrice A, ce qui est, en général, aussi difficile que de résoudre directement le système linéaire.
On dispose également des **formules de Cramer** sous la forme très élégante $x_i = \dfrac{\det A_i}{\det A}$, $i = 1, \ldots, n$, où A_i est la matrice obtenue en remplaçant la i-ème colonne de A par b.
Or, utiliser ces formules exige le calcul de $n + 1$ déterminants et le calcul (rapide) d'un déterminant est un problème aussi difficile que de résoudre directement le système. En pratique, ces formules sont inutilisables dès que $n \geq 4$ (essayer par exemple de calculer un déterminant d'ordre 10).

(1) A condition que les calculs soient menés de manière exacte, sans arrondis.

Prenons un système de taille 25. Il faudra calculer 26 déterminants de taille 25, ce qui représente au moins $26 \times 25!(\textit{permutations}) \times 24(\textit{produits}) \approx 10^{28}$ multiplications. Avec un ordinateur fonctionnant à 1 Gigaflops, i.e. un milliard d'opérations par seconde, cela représente 10^{19} s soit 3.2×10^{11} années! Pour un système de taille 50, cela donne 2×10^{51} années!

Exemple II.1.1 (Exercice)**.**

Sauriez-vous redémontrer ces formules ? On rappelle:

$$\det A = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) a_{1,\sigma(1)} a_{2,\sigma(2)} \dots a_{n,\sigma(n)}$$

où S_n est le groupe des permutations de l'ensemble à n éléments, $\operatorname{sgn}(\sigma)$ est la signature de la permutation σ.

On verra bientôt qu'avec la méthode du pivot, la résolution d'un tel système ne prendra qu'une fraction de seconde.

Il faut déjà retenir une première leçon:

- un résultat théorique même très explicite peut être tout à fait inutilisable en pratique.
- la façon de calculer est primordiale.

Avant de voir les différentes méthodes de résolution de systèmes linéaires, rappelons un cas très particulier où les calculs sont très rapides.

Cas simples.

- Cas A diagonale. Dans ce cas, le nombre d'opérations à effectuer est très réduit, c'est tout simplement n divisions.
- Cas A triangulaire. En pratique, il est rare que l'on ait d'emblée un système diagonal, ce qui s'en rapproche le plus c'est la forme triangulaire, forme à laquelle on essaiera de se ramener par élimination des variables. On résout le système en cascade par remontée ou descente.
- Cas d'une matrice orthogonale, dans ce cas $A^{-1} = {}^tA$ donc la solution du système est $X = {}^tAb$ qui s'obtient juste par un produit matrice-vecteur.

Dans tout ce qui suit on supposera que l'on a un système de Cramer, c'est à dire que la matrice A est inversible.

II.2. Résolution d'un système triangulaire

Lorsque la matrice A est une matrice triangulaire supérieure (i.e. $a_{ij} = 0$ pour $j < i$), le système a la forme:

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,n-1}x_{n-1} + a_{1,n}x_n = b_1 \\ \qquad\quad a_{2,2}x_2 + \cdots + a_{2,n-1}x_{n-1} + a_{2,n}x_n = b_2 \\ \qquad\qquad\quad \ddots \qquad\qquad\qquad\qquad \vdots \\ \qquad\qquad\qquad\quad a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n = b_{n-1} \\ \qquad\qquad\qquad\qquad\qquad\quad a_{n,n}x_n = b_n \end{cases}$$

et la résolution de ce système est particulièrement simple. En effet, on sait, d'une part que les coefficients diagonaux de la matrice sont tous non nuls puisque la matrice est inversible et que son

déterminant est exactement le produit de ces termes diagonaux. D'autre part, on peut calculer en cascade la valeur des inconnues en commençant par la dernière sous la forme

$$\begin{aligned} x_n &= \frac{b_n}{a_{n,n}} \\ x_{n-1} &= \frac{b_{n-1} - a_{n-1,n}x_n}{a_{n-1,n-1}} \\ &\vdots \\ x_i &= \frac{b_i - \sum_{k=i+1}^{n} a_{i,k}x_k}{a_{i,i}} \\ &\vdots \\ x_1 &= \frac{b_1 - \sum_{k=2}^{n} a_{1,k}x_k}{a_{1,1}} \end{aligned}$$

On peut donner deux algorithmes pour résoudre un tel système.

Algorithme de résolution d'un système linéaire triangulaire (par lignes)
Donne la solution du système linéaire $Ax = b$ lorsque la matrice A est triangulaire supérieure et inversible.
Entrées : la matrice $A = (a_{i,j})_{1\leq i,j\leq n}$ et le vecteur $b = {}^t(b_1, \ldots, b_n)$
Sortie : le vecteur solution $x = {}^t(x_1, \ldots, x_n)$.
Etape 1 [boucle principale] pour i de n à 1 (pas -1) faire

[Calcul du nouveau b_i] $b_i \leftarrow b_i - \sum_{k=i+1}^{n} a_{i,k}b_k$)

[calcul d'une nouvelle coordonnée] $b_i \leftarrow \frac{b_i}{a_{i,i}}$

fin pour
Etape 2 [fin] sortir $b = {}^t(b_1, \ldots, b_n)$ et fin.

Précisions:

- lorsque dans une boucle "pour" l'indice se trouve à commencer à un entier plus grand que celui de la fin, il est convenu que la boucle est vide; idem pour le cas d'une somme,
- si l'on veut faire une boucle avec un indice qui décroit, on précisera dans ce cas que le pas est négatif (pas=-1 par exemple).

Démonstration. — Il est clair que cet algorithme donne bien ce qu'il faut puisqu'il suit pas à pas le calcul des formules précédentes. Le vecteur b va contenir petit à petit, en commençant par la fin, les coordonnées x_i cherchées.
Remarque : pour $i = n$ au début, la boucle intérieure est vide. □

Complexité. Voyons ce qui se passe ici. L'algorithme exige $n(n-1)/2$ multiplications, n divisions et $n(n-1)/2$ additions. Il y a donc un nombre d'opérations $N_{op}(n) \asymp n^2$.

On remarquera que la manière de fonctionner de cet algorithme consiste à retrancher dans chaque ligne, à droite de la diagonale, le produit des coefficients de la matrice par les valeurs déjà calculées de la solution et à calculer ensuite une nouvelle coordonnée de la solution. On travaille

donc ligne par ligne. On remarquera aussi que le calcul se fait sur place (sur le vecteur b) et ne nécessite pas de mémoire supplémentaire.

Comme il est très rapide de résoudre un système triangulaire, les méthodes que l'on va voir par la suite chercheront à se ramener à un système triangulaire.

Remarque II.2.1.
On aurait pu chercher à inverser la matrice puis à exprimer la solution sous la forme $A^{-1}b$.
Le calcul de A^{-1} se fait en prenant successivement comme second membre les n vecteurs colonnes élémentaires du type $e_i = {}^t(0, \dots, 0, 1, 0, \dots, 0)$ (un seul 1 en place i, $1 \leq i \leq n$). En effet, la i-ème colonne de A^{-1} n'est autre que $A^{-1}e_i$ i.e. la solution du système $Ax = e_i$. Cela prendrait déjà $\sum_{j=1}^{n} j^2 \asymp n^3/3$ opérations (on gagne un peu de temps grâce aux zéros) mais il faudrait ensuite faire le produit $A^{-1}b$ soit $2n^2$ opérations donc au total un équivalent de $n^3/3$ opérations.
Ceci est beaucoup plus coûteux que le n^2 de l'algorithme précédent. Ceci est une généralité :

Pour résoudre un système linéaire, on n'a pas du tout intérêt en général à calculer l'inverse de la matrice.

II.3. Méthode du pivot de Gauss (élimination linéaire)

Nous en venons au cadre plus général $Ax = b$ où la matrice carrée A n'est plus triangulaire. La méthode du pivot de Gauss[(2)] que nous allons traiter est l'un des algorithmes les plus importants en algèbre linéaire; il est basé sur le principe bien connu de l'élimination des variables entre équations menée de manière systématique.

II.3.1. Cas où les pivots sont non nuls. —

Etape 1. On suppose que la première équation contient la variable x_1 (i.e. le coefficient de x_1 est non nul). A l'aide de cette première équation, qu'on ne modifie pas, on élimine x_1 de toutes les autres équations. On recommence avec la deuxième variable à partir de la deuxième équation et ainsi de suite. Si tout marche bien, on se retrouve à la fin avec un système triangulaire que l'on sait déjà résoudre. Pour décrire en détail cet algorithme, on va procéder de manière systématique.
On suppose d'abord que le coefficient $a_{1,1}$ de A est non nul. (On verra plus tard comment faire lorsque $a_{1,1} = 0$.) Cet élément sera notre premier **pivot**.
On élimine x_1 dans la seconde équation en ajoutant à celle-ci le produit de la première équation par $-\dfrac{a_{2,1}}{a_{1,1}}$, soit $L_2 \leftarrow L_2 - \dfrac{a_{2,1}}{a_{1,1}}L_1$, dans la troisième, en lui ajoutant la première équation multipliée par $-\dfrac{a_{3,1}}{a_{1,1}}$, soit $L_3 \leftarrow L_3 - \dfrac{a_{3,1}}{a_{1,1}}L_1$, et ainsi de suite.
On peut remarquer que si l'on pose $\boxed{m_{k,1} = -\dfrac{a_{k,1}}{a_{1,1}}}$ pour $k = 2, \dots, n$, éliminer la variable x_1 dans les

(2) Carl Friedrich Gauss, mathématicien, 1777-1855, astronome et physicien allemand

équations suivantes revient à multiplier à gauche le système de départ $Ax = b$ par la matrice

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{2,1} & 1 & 0 & \dots & 0 \\ m_{3,1} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{n,1} & 0 & 0 & \dots & 1 \end{pmatrix}.$$

Exemple II.3.1 (Exercice)**.**

Vérifier cette affirmation. (Cf. TD).

On obtient un nouveau système équivalent au premier car puisque M_1 est inversible $Ax = b$ équivaut à $M_1Ax = M_1b$ soit $A_1x = b^{(1)}$.
L'avantage c'est que maintenant la matrice $A_1 := M_1A$ est de la forme

$$A_1 = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,n} \\ 0 & a'_{2,2} & a'_{2,3} & \dots & a'_{2,n} \\ 0 & a'_{3,2} & a'_{3,3} & \dots & a'_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a'_{n,2} & a'_{n,3} & \dots & a'_{n,n} \end{pmatrix}.$$

D'un point de vue pratique, il est inutile de calculer les coefficients nuls dans cette matrice puisque l'on sait qu'ils doivent être nuls.
On peut remarquer au passage que l'on a $\det(A_1) = \det A$ car $\det(M_1) = 1$.

Etape 2. Pour pouvoir passer à l'étape 2, il faut supposer que les opérations effectuées n'ont pas rendu nul dans cette matrice le coefficient $a'_{2,2}$. Lorsque $a'_{2,2} \neq 0$, on utilise la matrice

$$M_2 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & m_{3,2} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & m_{n,2} & 0 & \dots & 1 \end{pmatrix},$$

avec $\boxed{m_{k,2} = -\dfrac{a'_{k,2}}{a'_{2,2}}}$ pour $k = 3, \dots, n$ et on obtient le système équivalent $A_2x = b^{(2)}$ avec $A_2 = M_2A_1$ et $b^{(2)} = M_2b^{(1)}$. On a toujours $\det(A_2) = \det(A_1) = \det A$. On remarquera que la multiplication à gauche par M_2 conserve les deux premières lignes de la matrice A_1.

On itère le processus jusqu'au rang $n - 1$, ceci est possible en supposant que tous les pivots trouvés sont non nuls.
On a alors au final un système équivalent au premier qui est $Ux = b'$ où $U = M_{n-1} \dots M_1A$ est triangulaire supérieure (Upper triangular) et $b' = M_{n-1} \dots M_1b$.

Exemple II.3.2.

Prenons $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$ et $b = \begin{pmatrix} 4 \\ 4 \\ 4 \end{pmatrix}$. On obtient successivement:

$$\left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ 1 & 2 & 1 & 4 \\ 1 & 1 & 2 & 4 \end{array}\right) \rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ 0 & 3/2 & 1/2 & 2 \\ 0 & 1/2 & 3/2 & 2 \end{array}\right) \begin{array}{l} (L_1) \\ (L_2 \leftarrow L_2 - \frac{1}{2}L_1) \\ (L_3 \leftarrow L_3 - \frac{1}{2}L_1) \end{array}, \quad M_1 = \begin{pmatrix} 1 & 0 & 0 \\ m_{21} = -1/2 & 1 & 0 \\ m_{31} = -1/2 & 0 & 1 \end{pmatrix}$$

$$\rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ 0 & 3/2 & 1/2 & 2 \\ 0 & 0 & 4/3 & 4/3 \end{array}\right) \begin{array}{l} (L_1) \\ (L_2) \\ (L_3 \leftarrow L_3 - \frac{1}{3}L_2) \end{array}, \quad M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & m_{32} = -1/3 & 1 \end{pmatrix}$$

La phase de descente est terminée. On a obtenu en sortie le système $Ux = b'$ avec $U = M_2M_1A = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3/2 & 1/2 \\ 0 & 0 & 4/3 \end{pmatrix}$ et $b' = M_2M_1b = \begin{pmatrix} 4 \\ 2 \\ 4/3 \end{pmatrix}$.

Il sera intéressant (cf. §II.4.3) de conserver en mémoire tous les multiplicateurs $m_{j,k}$, $1 \leq k \leq n-1$, $k+1 \leq j \leq n$ utilisés, c'est à dire l'information essentielle des différentes matrices M_k, $1 \leq k \leq n-1$. Pour des raisons de sauvegarde de mémoire machine, seule la matrice A est utilisée comme on l'explicite sur l'exemple suivant:

Exemple II.3.3.

On reprend l'exemple précédent: $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$ et $b = \begin{pmatrix} 4 \\ 4 \\ 4 \end{pmatrix}$. On obtient successivement:

$$\left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ 1 & 2 & 1 & 4 \\ 1 & 1 & 2 & 4 \end{array}\right) \rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ \boxed{\text{-1/2}} & 3/2 & 1/2 & 2 \\ \boxed{\text{-1/2}} & 1/2 & 3/2 & 2 \end{array}\right) \begin{array}{l} (L_1) \\ (L_2 \leftarrow L_2 - \frac{1}{2}L_1) \\ (L_3 \leftarrow L_3 - \frac{1}{2}L_1) \end{array}$$

$$\rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ \boxed{\text{-1/2}} & 3/2 & 1/2 & 2 \\ \boxed{\text{-1/2}} & \boxed{\text{-1/3}} & 4/3 & 4/3 \end{array}\right) \begin{array}{l} (L_1) \\ (L_2) \\ (L_3 \leftarrow L_3 - \frac{1}{3}L_2) \end{array}$$

La phase de descente est terminée. On a obtenu en sortie le système $Ux = b'$ avec $U = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3/2 & 1/2 \\ 0 & 0 & 4/3 \end{pmatrix}$ et $b' = \begin{pmatrix} 4 \\ 2 \\ 4/3 \end{pmatrix}$.

Il ne reste plus qu'à résoudre le système triangulaire par l'algorithme de remontée déjà vu. Au final, l'algorithme du pivot de Gauss est donc le suivant:

Algorithme du pivot de Gauss
Dans le cas où tous les pivots sont non nuls, donne la forme triangulaire supérieure obtenue par la méthode de Gauss ainsi que la solution du système $Ax = b$.
Entrée : la matrice A et le vecteur b de taille n
Sortie : la matrice triangulaire supérieure U, le vecteur solution x
Etape 0 Rajouter à A le vecteur b : $A \leftarrow (A \mid b)$
Etape 1 [descente]
[boucle principale] pour k de 1 à $n-1$ faire
[boucle sur les lignes] pour i de $k+1$ à n faire
[calcul du coefficient $m_{i,k}$] $a_{i,k} \leftarrow -\dfrac{a_{i,k}}{a_{k,k}}$
[changement de la ligne i à partir de la colonne $k+1$] $L_i \leftarrow L_i + a_{i,k} L_k$
fin pour
fin pour
Etape 2 [remontée, résolution du système triangulaire]
Extraire de A la dernière colonne et la matrice carrée restante triangulaire supérieure : $A, b \leftarrow (A \mid b)$
[boucle principale] pour i de n à 1 (pas -1) faire
[calcul du nouveau b_i] $b_i \leftarrow b_i - \sum_{k=i+1}^{n} a_{i,k} b_k$
[calcul d'une nouvelle coordonnée] $b_i \leftarrow \dfrac{b_i}{a_{i,i}}$,
fin pour
Etape 3 [fin] sortir A et $b = {}^t(b_1, \ldots, b_n)$ et fin.

A l'issue du calcul, le tableau numérique contient sous la diagonale tous les multiplicateurs $m_{j,k}$, $1 \le k \le n-1$, $k+1 \le j \le n$ utilisés. La dernière colonne contient le vecteur solution du système. Il y a une phase de descente qui est la réduction de Gauss à proprement parler et une phase de montée qui est la résolution du système triangulaire obtenue.

Exemple II.3.4.

Prenons $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$ et $b = \begin{pmatrix} 4 \\ 4 \\ 4 \end{pmatrix}$. On obtient successivement:

$$\left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ 1 & 2 & 1 & 4 \\ 1 & 1 & 2 & 4 \end{array}\right) \rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ \boxed{-1/2} & 3/2 & 1/2 & 2 \\ \boxed{-1/2} & 1/2 & 3/2 & 2 \end{array}\right) \rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ \boxed{-1/2} & 3/2 & 1/2 & 2 \\ \boxed{-1/2} & \boxed{-1/3} & 4/3 & 4/3 \end{array}\right)$$

La phase de descente (étape 1) est terminée: on a obtenu le système équivalent $Ux = b'$ avec $U = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3/2 & 1/2 \\ 0 & 0 & 4/3 \end{pmatrix}$ et $b' = \begin{pmatrix} 4 \\ 2 \\ 4/3 \end{pmatrix}$.

On remonte (étape 2):
$$\left(\begin{array}{ccc|c} 2 & 1 & 1 & 4 \\ \boxed{-1/2} & 3/2 & 1/2 & 2 \\ \boxed{-1/2} & \boxed{-1/3} & 4/3 & 4/3 \end{array}\right) \to \left(\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ \boxed{-1/2} & 3/2 & 1/2 & 1 \\ \boxed{-1/2} & \boxed{-1/3} & 4/3 & 1 \end{array}\right)$$ ce qui donne la solution ${}^t(1,1,1)$.

Mémoire. On remarquera qu'aucune mémoire supplémentaire n'est nécessaire puisque tous les calculs se font dans la matrice A.

Complexité.

– Dans la phase de descente, le nombre de multiplications ou divisions à faire est:

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

le nombre d'opérations à faire est donc $N_{op}(n) \asymp \dfrac{2n^3}{3}$.

– La phase de remontée est la résolution d'un système triangulaire et exige $N_{op}(n) \asymp n^2$ opérations.

On retiendra que la complexité de l'algorithme du pivot simple de Gauss est $N_{op}(n) \asymp \dfrac{2n^3}{3}$.

Pour $n = 25$, on pourra comparer $2 \cdot 25^3/3 \approx 10417$ avec le calcul des formules de Cramer qui nécessitent près de 10^{28} multiplications (soit $5 \cdot 10^{-6}$ s à 1 Gflops contre $3,2\ 10^{11}$ années).

Remarque II.3.5.
Un sous-produit de l'algorithme précédent est le déterminant de la matrice que l'on obtient en faisant le produit des termes diagonaux de la nouvelle matrice U obtenue. En effet $U = M_{n-1} \dots M_1 A$ implique:

$$\det U = \det(M_{n-1} \dots M_1) \det A = \det A.$$

Remarque II.3.6 (Matrices sans pivots nuls)**.**
Il y a des matrices dont on sait à l'avance que l'on peut faire la réduction de Gauss à un système triangulaire sans rencontrer de pivots nuls:

– c'est le cas, en particulier, lorsque la matrice A du système est **symétrique définie positive** (on verra pourquoi un peu plus tard);
– un autre cas où les choses se passent de manière analogue concerne les matrices à **diagonale strictement dominante** (cf. definition I.2.15). On verra également un peu plus loin pourquoi c'est le cas.

II.3.2. Cas où un pivot est nul. — Nous supposons ici que la matrice $M_{k-1} \dots M_1 A = A_{k-1} = \left(a''_{i,j}\right)$ obtenue à l'issue de la $(k-1)$-ième étape de la méthode de Gauss est telle que $a''_{k,k} = 0$.
Dans ce cas, comme la matrice A est tout de même non singulière par hypothèse, si le coefficient $a''_{k,k}$ est nul à l'étape k de la méthode de Gauss, il ne peut pas en être de même pour tous les coefficients en-dessous, c'est à dire les $a''_{j,k}$ avec $k+1 \le j \le n$. Sinon en effet on obtiendrait $\det A_{k-1} = 0$, donc $\det A = 0$.

On peut donc supposer donc que $a''_{j,k} \neq 0$ pour un certain j, $k+1 \leq j \leq n$. On va alors permuter les lignes k et j de la matrice à l'aide d'une matrice de permutation particulière, dite de transposition:

Définition II.3.7 (Matrice de transposition)**.**

On note $H_{(k,j)} = \left(h_{i,\ell}\right) \in \mathcal{M}_n(\mathbb{K})$, $1 \leq k < j \leq n$, la matrice dite de **transposition**, déduite de la matrice identité I_n en posant $h_{k,j} = h_{j,k} = 1$ et $h_{j,j} = h_{k,k} = 0$:

$$H_{(k,j)} = \begin{pmatrix} I & & & \\ & 0 & \cdots & 1 \\ & \vdots & I & \vdots \\ & 1 & \cdots & 0 \\ & & & I \end{pmatrix} \begin{matrix} \\ \leftarrow k\text{-ième ligne} \\ \\ \leftarrow j\text{-ième ligne} \\ \\ \end{matrix}$$
$$\begin{matrix} & \uparrow & & \uparrow & \\ & k_{eme} & & j_{eme} & \end{matrix}$$

En pratique pour construire $H_{(k,j)}$, on part de I_n, on déplace le 1 de la place (k,k) à la place (k,j) et de même on déplace le 1 de la place (j,j) à la place (j,k). Cela donne:

Il est facile de voir que pour une telle matrice on a $\det(H_{(k,j)}) = -1$ et aussi que $H^2_{(k,j)} = I_n$, c'est à dire qu'une telle matrice est égale à son inverse.

La multiplication à gauche de la matrice A_{k-1} par $H_{(k,j)}$ échange très exactement les lignes j et k. Elle transforme le système en un système équivalent avec cette fois un pivot diagonal non nul. Le déterminant de la nouvelle matrice est l'opposé du précédent car on a effectué une transposition de lignes qui est une permutation impaire.

Remarque II.3.8 (Rappel)**.**
Le déterminant vu comme application sur n vecteurs (colonnes) est multilinéaire alterné i.e.

$$\det(V_{\sigma(1)}, V_{\sigma(2)}, \ldots, V_{\sigma(n)}) = sgn(\sigma)\det(V_1, V_2, \ldots, V_n),$$

pour toute permutation σ du groupe symétrique S_n. C'est la même chose pour les lignes puisque $\det A = \det {}^t A$.

Une fois la transposition effectuée, la suite des calculs est alors celle déjà décrite.

Exemple II.3.9.

Prenons $A = \begin{pmatrix} 2 & 1 & 1 \\ 4 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$ et $b = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$. On obtient successivement:

$$\left(\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ 4 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 \end{array}\right) \rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ \boxed{-2} & 0 & -1 & -1 \\ \boxed{-1/2} & 1/2 & 3/2 & 1/2 \end{array}\right) \begin{matrix} (L_1) \\ (L_2 \leftarrow L_2 - 2L_1) \\ (L_3 \leftarrow L_3 - \frac{1}{2}L_1) \end{matrix}, \quad M_1 = \begin{pmatrix} 1 & 0 & 0 \\ \boxed{-2} & 1 & 0 \\ \boxed{-1/2} & 0 & 1 \end{pmatrix}$$

$$\rightarrow \left(\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ \boxed{-1/2} & 1/2 & 3/2 & 1/2 \\ \boxed{-2} & 0 & -1 & -1 \end{array}\right) \begin{array}{l}(L_1)\\(L_2 \leftarrow L_3)\\(L_3 \leftarrow L_2)\end{array}, \quad H_{(2,3)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

La phase de descente (étape 1) est terminée. On a obtenu en sortie le système $Ux = b'$ avec $U = H_{(2,3)}M_1A = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 1/2 & 3/2 \\ 0 & 0 & -1 \end{pmatrix}$ et $b' = H_{(2,3)}M_1b = \begin{pmatrix} 1 \\ 1/2 \\ -1 \end{pmatrix}$.

On remonte (étape 2): $\left(\begin{array}{ccc|c} 2 & 1 & 1 & 1 \\ \boxed{-1/2} & 1/2 & 3/2 & -2 \\ \boxed{-2} & 0 & -1 & 1 \end{array}\right)$ ce qui donne la solution ${}^t(1, -2, 1)$.

NB: le fait d'avoir fait une transposition $H_{2,3}$ ne change pas les coordonnées de la solution (on n'a fait que permuter des équations).

Finalement, ce que l'on vient de montrer c'est le théorème suivant:

Théorème II.3.10 (Elimination de Gauss)**.**

Soit A une matrice carrée inversible, il existe une matrice inversible M telle que la matrice $U = MA$ soit triangulaire supérieure.

Démonstration. — Fait précédemment, M est un produit de matrices de type M_i et éventuellement de matrices de transposition. □

Exemple II.3.11 (Exercice)**.**

Résoudre par Gauss le système pour $A = \begin{pmatrix} 2 & 4 & -4 & 1 \\ 3 & 6 & 1 & -2 \\ -1 & 1 & 2 & 3 \\ 1 & 1 & -4 & 1 \end{pmatrix}$ et $b = {}^t(0, -7, 4, 2)$.

II.3.3. Cas de plusieurs systèmes simultanés. — Nous avons considéré le cas où le système à résoudre était unique (un seul second membre à calculer). Il arrive souvent que l'on ait à résoudre plusieurs fois le même système avec des seconds membres différents b, b', b'' .... Dans ce cas il y a deux possibilités :

1. si tous les seconds membres sont connus depuis le départ, on peut concaténer la matrice A avec tous les seconds membres (cette remarque est aussi valable quand on n'a qu'un seul système à résoudre) et on fait les opérations que l'on vient de décrire sur cette matrice qui n'est plus carrée. Ceci est possible puisque toutes les opérations décrites sont des multiplications à gauche.
2. Dans le cas où tous les seconds membres ne seraient pas connus au départ, il faut garder en mémoire toutes les opérations faites à gauche autrement dit conserver tous les coefficients $m_{j,k}$ pour pouvoir les appliquer à droite sur le b qui se présente. Cela revient à calculer le produit de toutes les matrices de transformation $M = M_{n-1} \dots M_1$ une fois pour toutes. En fait, dans ce cas, il n'est pas nécessaire d'effectuer ce produit, il est plus avantageux d'utiliser la décomposition LU de la matrice qui découle de la méthode de Gauss. On va y venir.

II.3.4. Problèmes liés aux pivots trop petits. — Même dans le cas où tous les pivots rencontrés sont non nuls, on peut avoir des problèmes de précision comme le montre l'exemple suivant.

Exemple II.3.12.

1. On considère le système $\begin{cases} 10^{-4}x + y &= 1 \\ x + y &= 2 \end{cases}$ et supposons que l'on travaille avec 3 chiffres significatifs. On obtient:

   $$\left(\begin{array}{cc|c} 10^{-4} & 1 & 1 \\ 1 & 1 & 2 \end{array}\right) \to \left(\begin{array}{cc|c} 10^{-4} & 1 & 1 \\ 0 & -9999 & -9998 \end{array}\right) \to \left(\begin{array}{cc|c} 10^{-4} & 1 & 1 \\ 0 & -9999 & 9998/9999 \end{array}\right)$$

   $$\to \approx \left(\begin{array}{cc|c} 10^{-4} & 1 & 0 \\ 0 & -9999 & 1 \end{array}\right)$$ ce qui donne comme solution du système $y = \dfrac{2 - 10^4}{1 - 10^4} = 0,99989998$ qui sera arrondi à 1 puisque l'on travaille avec 3 chiffres significatifs, puis $x = \dfrac{1 - y}{10^{-4}} = 0$, ce qui est clairement faux.
   Ce qui s'est passé, c'est que dans le premier cas, l'erreur commise par arrondi sur y dans l'équation 2, de l'ordre de $\epsilon = 1 \times 10^{-4}$, a été multiplié dans le calcul de x dans l'équation 1 par $\dfrac{1}{10^{-4}}$ ce qui donne un erreur sur x qui vaut $10^4\epsilon$ qui est de l'ordre de grandeur de x lui-même.
2. On considère maintenant le système $\begin{cases} x + y &= 2 \\ 10^{-4}x + y &= 1 \end{cases}$ où on a intervertit les deux lignes du système précédent. On obtient après pivot: $\left(\begin{array}{cc|c} 1 & 1 & 2 \\ 0 & 1 - 10^{-4} & 1 - 2.10^{-4} \end{array}\right)$ et en tenant compte des arrondis $\left(\begin{array}{cc|c} 1 & 1 & 1 \\ 0 & 1 & 1 \end{array}\right)$ ce qui donne comme solution $y = \dfrac{1 - 2.10^{-4}}{1 - 10^{-4}}$ qui sera arrondi à 1 dans l'équation 2 et $x = \dfrac{2 - y}{1} = 1$ dans l'équation 1 ce qui est beaucoup plus raisonnable. Dans ce second cas, l'erreur commise par arrondi sur y a simplement été multipliée par 1.

Cet exemple montre que l'utilisation de pivots qui sont très petits, pose souvent des problèmes. En effet, si un pivot est très petit, multiplier par son inverse va amplifier toutes les erreurs d'arrondis. Il faut donc prendre un pivot le plus grand possible.

On utilise alors la méthode que l'on appelle du **pivot partiel** et qui consiste à faire, à chaque étape $k = 1, 2, \ldots, n-1$ une recherche de pivot de la manière suivante: on détermine la valeur de $j \geq k$ tel que $|a_{j,k}| = \max\limits_{k \leq l \leq n} |a_{l,k}|$ et on fait ensuite un échange entre les lignes j et k pour avoir comme pivot l'élément de plus grande valeur absolue parmi les derniers coefficients de la colonne considérée.

Une variante de cette méthode, que l'on appelle méthode du **pivot total** consiste à déterminer la ligne $i \geq k$ et la colonne $j \geq k$ telle que $|a_{i,j}| = \max\limits_{k \leq l,m \leq n} |a_{l,m}|$, et à faire cette fois un échange de lignes et de colonnes pour ramener à la position (k, k) le terme que l'on va utiliser comme pivot. Ceci dit, si l'on gagne en précision, on va perdre en temps puisque cela va nécessiter des comparaisons à chaque étape.

Remarque II.3.13.
L'échange des lignes k et l dans une matrice se fait par multiplication à gauche par une matrice élémentaire de transposition $H_{(k,l)}$ comme on l'a déjà vu.

Le système obtenu en multipliant le système de départ (ou un système lui étant équivalent) à gauche par une matrice de transposition est encore équivalent au précédent.

II.3.5. Quelques remarques. — Nous terminons cette section par quelques remarques.

- On peut signaler un autre problème que l'on rencontre si l'on effectue la méthode de Gauss sur des rationnels. On s'affranchit du coup de tous les problèmes d'arrondis mais survient alors un nouveau problème, c'est celui de l'explosion très rapide du numérateur ou du dénominateur des rationnels qui interviennent. Il existe des méthodes pour limiter cet effet (méthode du sous-résultant ou encore méthode de Bareiss) mais nous ne les étudierons pas dans ce cours.
- Signalons aussi que la méthode de Gauss vue ici pour des matrices inversibles fonctionne aussi pour des matrices non inversibles voire des matrices non carrées. Des résultats analogues au théorème II.3.10 existent mais nous ne les citons pas ici pour ne pas alourdir.

II.4. Décomposition LU

Une autre manière de résoudre le même système linéaire $Ax = b$ avec des seconds membres différents non connus au départ consiste en ceci. Supposons qu'il soit possible de trouver une décomposition de la matrice A du système sous la forme d'un produit de deux matrices $A = LU$ où L est une matrice triangulaire inférieure et U une matrice triangulaire supérieure (L pour Lower, U pour Upper).

On peut résoudre alors le système $Ax = LUx = b$ en deux étapes: on pose $y = Ux$, on calcule d'abord la solution du système $Ly = b$. Comme ce système est triangulaire inférieur, la solution est aussi simple que celle utilisée pour les systèmes triangulaires supérieurs, la seule différence étant que l'on doit trouver les coordonnées de la solution dans le sens inverse à celui déjà décrit pour les systèmes triangulaires supérieurs.
Comme maintenant le vecteur y est connu, il reste le système $Ux = y$ à résoudre et on obtient le x comme désiré. On remarquera que si les deux matrices L et U sont connues, le nombre d'opérations nécessaires pour résoudre le système est équivalent à $2n^2$ ($2 \times$ nombre d'opérations pour résoudre un système triangulaire).

Cette méthode est particulièrement efficace si l'on doit résoudre plusieurs systèmes $Ax = b^{(i)}$, $1 \leq i \leq k$, mais pas en même temps. On calcule et on stocke une fois pour toutes L et U, on n'a alors à chaque fois qu'à résoudre deux systèmes triangulaires.
Si on utilise la méthode de Gauss, on peut bien sûr stocker la matrice triangulaire T mais il faut aussi mémoriser les opérations faites sur le second membre autrement dit toutes les matrices M_k. Cette information est en fait contenue dans le L la décomposition LU.


Trouver les matrices L et U telles que $A = LU$ n'est pas toujours possible, comme le montre l'exemple de l'exercice suivant mais on dispose d'un critère suffisant pour assurer l'existence d'une telle décomposition.

Exemple II.4.1 (Exercice).

Vérifier que $\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$ n'admet pas de décomposition LU.

Définition II.4.2 (Sous-matrices principales).

Soit $A \in \mathcal{M}_n(\mathbb{K})$ une matrice carrée. Sa **sous-matrice principale** d'ordre $k \in [1, n-1]$ est la matrice carrée $A_k \in \mathcal{M}_k(\mathbb{K})$ formée à l'aide de ses k premières lignes et colonnes.

Théorème II.4.3 (Décomposition LU).

Soient $A \in \mathcal{M}_n(\mathbb{K})$ une matrice inversible d'ordre n. Les 2 propriétés suivantes sont équivalentes.

1. *Il existe une matrice triangulaire inférieure unique $L = (l_{i,j})$ avec $l_{i,i} = 1$, $i = 1, \dots, n$ et une matrice triangulaire supérieure unique $U = (u_{i,j})$ telles que $A = LU$.*
2. *Les sous-matrices principales A_k de A sont inversibles:* $\det A_k \neq 0$, $k \in [1, n-1]$.

Remarque II.4.4.

Si A n'est pas inversible mais $\det A_k \neq 0$ pour $k = 1, 2, \dots, n-1$, la décomposition est encore possible mais est non unique.

Démonstration. —

1. $\boxed{\Leftarrow}$ La preuve se fait par récurrence sur n.
   Soit $\mathscr{P}(n)$ la propriété :
   (A d'ordre n et $\det A_k \neq 0$, $k = 1, 2, \dots, n$) $\Rightarrow$ A est décomposable en $A = LU$ de manière unique.
   - Si $n = 1$, on a $A = (a_{1,1})$ et elle est déjà triangulaire supérieure. On prend $U = A$ et $L = (1)$. De plus, on a forcément $L = (1)$ d'où l'unicité. La propriété $\mathscr{P}(1)$ est donc vraie.
   - Supposons $\mathscr{P}(k-1)$ vraie. Soit A une matrice inversible d'ordre k satisfaisant les hypothèses du théorème.

     On peut écrire la matrice A sous la forme $A = \begin{pmatrix} A_{k-1} & b \\ {}^t c & a_{k,k} \end{pmatrix}$ où ${}^t c$ est un vecteur ligne et b un vecteur colonne de longueur $k-1$. Cherchons à écrire A sous la forme $A = LU$. Les matrices L et U, si elles existent, peuvent s'écrire sous la forme

     $$L = \begin{pmatrix} L_{k-1} & 0 \\ {}^t \ell & 1 \end{pmatrix} \quad \text{et} \quad U = \begin{pmatrix} U_{k-1} & u \\ 0 & u_{k,k} \end{pmatrix},$$

     où L_{k-1} et U_{k-l} sont respectivement triangulaire inférieure (avec une diagonale de 1) et triangulaire supérieure, ℓ et u sont des vecteurs colonne inconnus ainsi que la constante $u_{k,k}$ qu'il faut déterminer.
     En utilisant la multiplication par blocs et en identifiant on obtient:

     $$A_{k-1} = L_{k-1}U_{k-1}, \quad L_{k-1}u = b, \quad {}^t \ell U_{k-1} = {}^t c, \quad \text{et} \quad {}^t \ell u + u_{k,k} = a_{k,k}.$$

     La matrice A_{k-1} satisfait aux hypothèses du théorème donc, grâce à l'hypothèse de récurrence, peut s'écrire de manière unique sous la forme: $A_{k-1} = L_{k-1}U_{k-1}$, L_{k-1}

et U_{k-1} étant respectivement triangulaire inférieure (avec une diagonale de 1) et triangulaire supérieure. Les matrices L_{k-1} et U_{k-l} existent donc et sont uniques avec $\det L_{k-1}.\det U_{k-1} = \det A_{k-1} \neq 0$.
Le système $L_{k-1}u = b$ est donc de Cramer et possède en conséquence une solution u unique. De même le système ${}^t\ell U_{k-1} = {}^t c$ qui équivaut à ${}^t U_{k-1}\ell = c$ est de Cramer et possède en conséquence lui aussi une solution ℓ unique. Finalement le nombre $u_{k,k} = a_{k,k} - {}^t\ell u$ existe et est unique. On a ainsi prouvé l'existence d'une décomposition unique $A = LU$.
La propriété $\mathscr{P}(k)$ est vraie.

Par le principe de récurrence, la propriété $\mathscr{P}(n)$ est vraie pour tout $n \geq 1$.

2. $\boxed{\Rightarrow}$ Réciproquement, supposons qu'il existe une décomposition $A = LU$, on écrit alors pour tout $1 \leq k \leq n-1$ en séparant les k premières lignes et colonnes

$$A = \left(\begin{array}{c|c} A_k & * \\ \hline * & * \end{array}\right) \qquad L = \left(\begin{array}{c|c} L_k & 0 \\ \hline * & * \end{array}\right) \qquad U = \left(\begin{array}{c|c} U_k & * \\ \hline 0 & * \end{array}\right)$$

On obtient de $A = LU$, l'égalité $A_k = L_k U_k$. Or $\det L_k = 1$ et $\det U_k = \prod_{i=1}^{k} u_{i,i} \neq 0$ sinon $\det U = 0$ donc $\det A_k \neq 0$ et ceci est valable pour tout k de 1 à n.
La condition est donc bien nécessaire.

□

Corollaire II.4.5.

1. *Les matrices symétriques définies positives (ou définies négatives) admettent une décomposition LU.*
2. *Les matrices à diagonale strictement dominante admettent une décomposition LU.*

Démonstration. — 1. Si la matrice $A \in \mathcal{M}_n(\mathbb{R})$ est symétriques définies positives, alors les différentes sous-matrices principales A_k correspondent aux matrices des restrictions à des sous-espaces de $\mathbb{R}^n$ de la forme quadratique représentée par A dans la base canonique de $\mathbb{R}^n$. Or toute restriction d'une forme quadratique définie positive est encore définie positive, donc tous les $\det A_k$ sont non nuls (une forme quadratique définie positive est non dégénérée i.e. son déterminant est non nul). Une matrice symétrique définie positive admet donc une décomposition LU.

2. Si la matrice $A \in \mathcal{M}_n(\mathbb{K})$ est à diagonale strictement dominante (cf. definition I.2.15), alors elle est inversible (cf. proposition I.2.16). De plus, chacune de ses sous-matrices principales A_k a encore la propriété de diagonale strictement dominante, donc est inversible. Une matrice à diagonale strictement dominante admet donc une décomposition LU.

□

II.4.1. Lien avec l'existence de pivot nul dans la méthode de Gauss. — Dans la méthode du pivot simple de Gauss, si on tombe au rang k sur un pivot nul, cela implique que $\det A_k = 0$ (produit des éléments diagonaux de 1 à k). Donc la matrice n'aura pas de décomposition LU.

Réciproquement, si la matrice a une décomposition LU, tous les $\det A_k$ sont non nuls, cela implique qu'on ne rencontre jamais de pivot nul. On a donc:

Une matrice inversible admet une décomposition LU $\Leftrightarrow$ on ne rencontre jamais de pivot nul dans la méthode du pivot simple de Gauss.

Ceci explique que dans les deux exemples classiques du corollaire II.4.5 qui admettent une décomposition LU, on ne tombe jamais sur un pivot nul.

On retiendra que les matrices symétriques définies positives et les matrices à diagonale strictement dominante admettent une décomposition LU et ne donnent jamais lieu à un pivot nul dans l'algorithme de Gauss.

Complément. La description de la méthode du pivot de Gauss montre que toute matrice inversible peut, au moyen de changements de lignes convenables, être transformée en une matrice ayant une décomposition LU. En effet, si à un moment donné, une matrice A_k est non inversible, on trouve alors un pivot nul mais en échangent 2 lignes, on retombe sur une matrice inversible.

On introduit alors la définition suivante utile pour établir un résultat général.

Définition II.4.6 (Matrice de permutation)**.**

Soit $\sigma \in S_n$, une permutation de $(1,\dots,n)$. On appelle matrice de permutation $P_\sigma = (p_{i,j})_{1\le i,j\le n}$ associée à σ la matrice vérifiant pour $i = 1,\dots,n$ $p_{i,j} = 1$ si $j = \sigma(i)$ et $p_{i,j} = 0$ sinon ou encore avec le symbole de Kronecker $p_{i,j} = \delta_{j,\sigma(i)}$.

Exemple II.4.7.

Si $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 4 & 1 \end{pmatrix} = (134)$ alors $P_\sigma = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$.

Exemple II.4.8 (Exercice)**.**

Vérifier que l'on a pour tous $\sigma, \tau \in S_n$ $P_{\sigma\circ\tau} = P_\sigma P_\tau$.
Conséquence: comme toute permutation est un produit de transposition, toute matrice de permutation est un produit de matrices de transposition.

On obtient alors le résultat suivant.

Théorème II.4.9 (PA=LU)**.**

Pour toute matrice inversible A, il existe une matrice de permutation P, une matrice triangulaire inférieure L (avec une diagonale de 1) et une matrice triangulaire supérieure U telles que $PA = LU$

Remarque II.4.10.
Il n'y a certainement pas unicité de cette décomposition puisque les transpositions à faire pour rendre la matrice "sans pivot nul" ne sont pas uniques.

Démonstration. — Etant donné une matrice inversible A, on a vu lors de l'algorithme de Gauss que l'on peut toujours au besoin échanger des lignes pour que la matrice satisfasse aux conditions $\det A_k \neq 0$, $k = 1...n$. Cela revient à multiplier globalement A à gauche par un produit de matrices de transposition de type $H_{(i,j)}$. Le produit de ces matrices est une matrice de permutation P. □

II.4.2. Algorithme de décomposition LU. —

Algorithme de décomposition LU
Etant donné une matrice $A = (a_{i,j})$ inversible de taille n qui satisfait les conditions du théorème LU, cet algorithme trouve les matrices $L = (l_{i,j})$ triangulaire inférieure (avec diagonale de 1) et $U = (u_{i,j})$ triangulaire supérieure telles $A = LU$.
Entrée : la matrice A,
Sortie : les matrices L et U.
Etape 1 [initialisation] définir L et U égales à la matrice identité de taille n,
Etape 2 [grande boucle] pour j de 1 à n faire
- [boucle pour U] pour k de j à n, faire

$$u_{j,k} \leftarrow a_{j,k} - \sum_{m=1}^{j-1} l_{j,m} u_{m,k},$$

fin pour
- [boucle pour L] pour k de $j+1$ à n, faire

$$l_{k,j} \leftarrow \frac{1}{u_{j,j}} (a_{k,j} - \sum_{m=1}^{j-1} l_{k,m} u_{m,j}),$$

fin pour
Etape 3 [fin] sortir L et U et fin.

Démonstration. — Cela résulte des n^2 relations obtenues en écrivant $LU = A$ dans un ordre précis.

Les places $(1,1), \ldots, (1,n)$ permettent de trouver $u_{1,1}, \ldots, u_{1,n}$

$$u_{1,1} = a_{1,1}, \ldots, u_{1,n} = a_{1,n},$$

puis $(2,1), \ldots, (n,1)$ permettent de trouver $l_{2,1}, \ldots, l_{n,1}$

$$l_{2,1}u_{1,1} = a_{2,1}, \ldots, l_{n,1}u_{1,1} = a_{n,1},$$

puis $(2,2), \ldots, (2,n)$ permettent de trouver $u_{2,2}, \ldots, u_{2,n}$

$$l_{2,1}u_{1,2} + u_{2,2} = a_{2,2}, \ldots, l_{2,1}u_{1,n} + u_{2,n} = a_{2,n},$$

puis $(3,2), \ldots, (n,2)$ permettent de trouver $l_{3,2}, \ldots, l_{n,2}$

$$l_{3,1}u_{1,2} + l_{3,2}u_{2,2} = a_{3,2}, \ldots, l_{n,1}u_{1,2} + l_{n,2}u_{2,2} = a_{n,2},$$

etc □

Exemple II.4.11.

On reprend la matrice de la méthode de Gauss (exemples II.3.2, II.3.3, II.3.4):

$$A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ l_{2,1} & 1 & 0 \\ l_{3,1} & l_{3,2} & 1 \end{pmatrix} \begin{pmatrix} u_{1,1} & u_{1,2} & u_{1,3} \\ 0 & u_{2,2} & u_{2,3} \\ 0 & 0 & u_{3,3} \end{pmatrix}$$

On obtient successivement $u_{1,1} = 2$, $u_{1,2} = 1$, $u_{1,3} = 1$, $l_{2,1} = \frac{1}{2}$, $l_{3,1} = \frac{1}{2}$, ... On obtient finalement

$$L = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/2 & 1/3 & 1 \end{pmatrix} \qquad \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3/2 & 1/2 \\ 0 & 0 & 4/3 \end{pmatrix}$$

Complexité. Le nombre d'opérations à faire est pour chaque j de 1 à n,
[boucle sur U] : $(n - j + 1) \times (j - 1$ produits, $j - 1$ sommes),
[boucle sur L] : $(n - j) \times (j - 1$ produits, $j - 1$ sommes et un quotient).
On obtient donc

$$\sum_{j=1}^{n} 2(n-j+1)(j-1) + (n-j)(2j-1) = \sum_{j=0}^{n-1} 2j(n-j) + \sum_{j=1}^{n} 2j(n-j) - \sum_{j=1}^{n} (n-j)$$
$$= 4n \sum_{j=1}^{n-1} j - 4 \sum_{j=1}^{n-1} j^2 - \sum_{j=1}^{n} j = 4n^2(n-1)/2 - 4(n-1)n(2n-1)/6 - n(n+1)/2 \asymp \frac{2n^3}{3}.$$

Pour la résolution complète du sytème linéaire il faut rajouter $2n^2$ opérations pour la résolution des deux sytèmes triangulaires ce qui fait au total toujours un équivalent de $\boxed{N_{op}(n) \asymp \frac{2n^3}{3}}$ opérations.

II.4.3. Lien avec la méthode du pivot de Gauss. — On a obtenu au bout de l'algorithme de Gauss (dans le cas où tous les pivots sont non nuls), $MA = U$ qui est une matrice triangulaire supérieure. On déduit que $A = M^{-1}U$. On va voir en-dessous que M^{-1} est en fait triangulaire inférieure avec une diagonale de 1, ce qui signifie tout simplement, compte tenu de l'unicité de la décomposition, que $L = M^{-1}$ et $U = U$ (!). Or U est connue à la fin de la phase de descente. Il suffit donc de connaître $M^{-1} = (M_{n-1} \dots M_1)^{-1} = M_1^{-1} \dots M_{n-1}^{-1}$.

Un calcul simple (exercice) montre que d'une part, l'inverse de la matrice $M_k = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & m_{k+1,k} & & \\ & & \vdots & \ddots & \\ & & m_{n,k} & & 1 \end{pmatrix}$

utilisée pour réduire A par la méthode de Gauss est $M_k^{-1} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -m_{k+1,k} & & \\ & & \vdots & \ddots & \\ & & -m_{n,k} & & 1 \end{pmatrix}$. On peut vérifier ensuite que le produit d'une matrice de la forme $P = \begin{pmatrix} 1 & 0 & \dots & 0 & & & \\ p_{2,1} & 1 & \dots & 0 & & & \\ \vdots & \vdots & & & & & \\ p_{k,1} & p_{k,2} & \dots & 1 & & & \\ p_{k+1,1} & p_{k+1,2} & \dots & 0 & 1 & & \\ \vdots & \vdots & & \vdots & \vdots & \ddots & \\ p_{n,1} & p_{n,2} & \dots & 0 & 0 & \dots & \dots & 1 \end{pmatrix}$ <u>à droite</u> par $P_k = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & p_{k+1,k} & & \\ & & \vdots & \ddots & \\ & & p_{n,k} & \dots & 1 \end{pmatrix}$ donne $PP_k = \begin{pmatrix} 1 & 0 & \dots & 0 & & & & \\ p_{2,1} & 1 & \dots & 0 & & & & \\ \vdots & \vdots & & & & & & \\ p_{k,1} & p_{k,2} & \dots & 1 & & & & \\ p_{k+1,1} & p_{k+1,2} & \dots & p_{k+1,k} & 1 & & & \\ \vdots & \vdots & & \vdots & \vdots & \ddots & & \\ p_{n,1} & p_{n,2} & \dots & p_{n,k} & 0 & \dots & \dots & 1 \end{pmatrix}$.

On en déduit que $M^{-1} = M_1^{-1} \dots M_{n-1}^{-1}$ est une matrice triangulaire inférieure dont la diagonale est formée de 1 et que ses éléments au-dessous de la diagonale sont les **opposés** des $m_{i,j}$ utilisés pour réduire A par la méthode de Gauss. Ainsi, si on connaît tous les multiplicateurs $m_{i,j}$ et la matrice T, on connaît de fait L et U.

Si l'expression de M^{-1} est simple, en revanche celle de M ne l'est pas.

A retenir. L'algorithme du pivot de Gauss donne, dans le cas où l'on n'a pas fait d'échanges de lignes, directement la matrice L (opposée du triangle inférieur sous la diagonale) et la matrice U (triangle supérieur). Cela peut se vérifier sur l'exemple (exemples II.3.2, II.3.3, II.3.4) traité.
Donc, soyons clair la décomposition LU ou le pivot de Gauss sont deux présentations différentes d'une même technique matricielle, elles ont la même complexité $\frac{2n^3}{3}$:

- version Gauss : on part de $AX = b$, on fait des manipulations de lignes ce qui revient à mutiplier à gauche par M, du coup on a, $MAx = Mb$ soit $Ux = Mb$ que l'on résout.
- version LU : on part de $A = LU = M^{-1}U$, du coup on a $LUx = b$, on résout $Ly = b$ soit $y = Mb$ puis on résout $Ux = y$ soit $Ux = Mb$. En fait, LU est la version matricielle de Gauss.

On notera bien que l'on ne stocke pas M qui n'a pas d'expression sympathique mais plutôt M^{-1} qui est triangulaire inférieure avec une diagonale de 1. L'obtention de $y = Mb$ ne se fait pas en multipliant par M mais en résolvant le système triangulaire $Ly = b$.

Exemple II.4.12.

On reprend $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$. (Cf. exemple II.3.4). Par Gauss, à la fin de la phase de descente, on a obtenu: $\begin{pmatrix} 2 & 1 & 1 \\ \boxed{-1/2} & 3/2 & 1/2 \\ \boxed{-1/2} & \boxed{-1/3} & 4/3 \end{pmatrix}$. On en déduit $L = M^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/2 & 1/3 & 1 \end{pmatrix}$ et $U = \begin{pmatrix} 2 & 1 & 1 \\ 0 & 3/2 & 1/2 \\ 0 & 0 & 4/3 \end{pmatrix}$.

Si on s'amuse par curiosité à calculer vraiment M on trouve: $M = \begin{pmatrix} 1 & 0 & 0 \\ -1/2 & 1 & 0 \\ -1/3 & -1/3 & 1 \end{pmatrix}$.

Complément: décomposition LDR. Toute matrice U triangulaire supérieure peut se factoriser de façon unique en $U = DR$ où D est une matrice diagonale et R est une matrice triangulaire supérieure ayant des 1 en diagonale.
On en déduit que toute matrice A admettant une factorisation LU se factorise de façon unique en $A = LDR$ où D est une matrice diagonale, L est une matrice triangulaire inférieure ayant des 1 en diagonale, R est une matrice triangulaire supérieure ayant des 1 en diagonale.

II.4.4. Décomposition LU, cas des matrices bandes. — On observe très souvent dans les problèmes de mathématiques appliquées des matrices creuses (i.e. contenant beaucoup de zéros) et notamment des matrices bande.

Définition II.4.13.

Une matrice $A \in \mathcal{M}_n(\mathbb{K})$ est dite matrice bande, de demi-largeur $p \in \mathbb{N}$ si $a_{i,j} = 0$ pour tous i, j tels que $|i - j| > p$. La largeur totale de la bande est alors $2p + 1$.

D'un point de vue informatique, par gain de place, pour stocker de telles matrices, on ne stocke bien sûr que les coefficients non nuls.

Exemple II.4.14.

- La matrice tridiagonale $\begin{pmatrix} 1 & 2 & 0 & 0 \\ 3 & 4 & 2 & 0 \\ 0 & 3 & 1 & 2 \\ 0 & 0 & 3 & 1 \end{pmatrix}$ est une matrice bande de demi-largeur 1.
- Une matrice bande classique obtenue par discrétisation de l'opérateur dérivée seconde:

$$A = \begin{pmatrix} 2 & -1 & 0 & \dots & \dots & \\ -1 & 2 & -1 & 0 & & \dots \\ 0 & -1 & 2 & -1 & 0 & \dots \\ \vdots & & & & & \vdots \\ 0 & \dots & \dots & & -1 & 2 \end{pmatrix}$$

Proposition II.4.15.

La factorisation LU conserve la structure bande des matrices.

Démonstration. — Simple exercice. □

La factorisation LU est donc bien pratique pour ces matrices. Cela réduit considérablement le nombre d'inconnues dans la décomposition LU.

II.5. Calcul de l'inverse d'une matrice, de son déterminant

II.5.1. Calcul de l'inverse d'une matrice. — Les algorithmes précédents (LU et Gauss) permettent aussi de calculer l'inverse d'une matrice.

Pour calculer l'inverse d'une matrice A inversible, il suffit de résoudre les n systèmes linéaires

$$Ax = e_i, \quad i = 1, \dots, n.$$

où $e_i = {}^t(0, \dots, 0, 1, 0, \dots, 0)$ est le vecteur colonne des coordonnées dans la base canonique du i-ème vecteur de cette base. Si c_i désigne la solution de $Ax = e_i$ pour i de 1 à n alors, $(c_1, \dots, c_n)$ sont les colonnes de A^{-1}.

1ère méthode. Une manière de calculer cette inverse par la méthode du pivot est de concaténer la matrice A et la matrice identité I_n, $A_I = (A|I_n)$. On met en œuvre alors la méthode du pivot, ce qui donne en multipliant le tout par une certaine matrice M, une matrice $MA = T$ triangulaire supérieure et on obtient pour le second membre $MI_n = M$.
Il suffit alors de résoudre le système triangulaire $Tx = b_i$ pour chaque vecteur colonne b_i de la matrice M. On obtient ainsi les n vecteurs colonnes de la matrice A^{-1}.
La complexité de calcul de l'inverse par cette méthode est alors:

- pour la phase de descente avec I_n à droite : $\sum_{k=1}^{n-1}(n-k)2(n-k+k) \asymp n^3$ (en tenant compte de la presence de nombreux zeros à droite;
- pour les n remontées : $n \times n^2 = n^3$.

Au total $\boxed{N_{op}(n) \asymp 2n^3}$.

2ème méthode. On peut aussi à partir de L et U, résoudre les n systémes $LUx = e_i$, tous comptes faits, on obtient aussi $N_{op}(n) \asymp 2n^3$.

3ème méthode. On peut également à partir de L et U inverser L puis U puis multiplier les deux inverses, le coût total de cette méthode est:

- $2n^3/3$ pour LU (comme Gauss);
- calcul des inverses de L et U, $2n^3/3$;
- $2n^3/3$ pour le produit $U^{-1}L^{-1}$.

Au total encore $\boxed{N_{op}(n) \asymp 2n^3}$.

4ème méthode : Gauss-Jordan. Une variante de la méthode de Gauss appelée méthode de Gauss-Jordan consiste à obtenir par pivotage sur A non plus simplement une matrice triangulaire T mais directement la matrice identité.

On part comme précédemment de la matrice concaténée $(A|I)$. Au lieu d'aboutir à une matrice triangulaire comme dans le cas de la méthode de Gauss classique, on travaille le système pour obtenir à gauche la matrice identité. A droite, la matrice finale est alors A^{-1}.
Ce qui change par rapport à l'algorithme classique de Gauss c'est:

1. on normalise chaque ligne pivot, en divisant par le pivot, ceci pour avoir le 1 en position diagonale;
2. avec le pivot, on fait apparaître des zéros sur toute la colonne (au-dessous et puis au-dessus du 1).

La complexité de calcul de l'inverse par cette méthode est alors $\sum_{k=1}^{n-1}(n-k+k)2(n-k+k) \asymp 2n^3$ (en tenant compte de la presence de nombreux zeros à droite. Encore une fois, $\boxed{N_{op}(n) \asymp 2n^3}$.

Exemple II.5.1 (Gauss-Jordan)**.**

On veut inverser la matrice $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$. On a:

$$\left(\begin{array}{ccc|ccc} 2 & 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \end{array}\right) \to \left(\begin{array}{ccc|ccc} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 1 & 2 & 1 & 0 & 1 & 0 \\ 1 & 1 & 2 & 0 & 0 & 1 \end{array}\right) \to \left(\begin{array}{ccc|ccc} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{3}{2} & \frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & \frac{1}{2} & \frac{3}{2} & -\frac{1}{2} & 0 & 1 \end{array}\right) \to$$

$$\left(\begin{array}{ccc|ccc} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 1 & \frac{1}{3} & -\frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{2} & \frac{3}{2} & -\frac{1}{2} & 0 & 1 \end{array}\right) \to \left(\begin{array}{ccc|ccc} 1 & 0 & \frac{1}{3} & \frac{2}{3} & -\frac{1}{3} & 0 \\ 0 & 1 & \frac{1}{3} & -\frac{1}{3} & \frac{2}{3} & 0 \\ 0 & 0 & \frac{4}{3} & -\frac{1}{3} & -\frac{1}{3} & 1 \end{array}\right) \to \left(\begin{array}{ccc|ccc} 1 & 0 & \frac{1}{3} & \frac{2}{3} & -\frac{1}{3} & 0 \\ 0 & 1 & \frac{1}{3} & -\frac{1}{3} & \frac{2}{3} & 0 \\ 0 & 0 & 1 & -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} \end{array}\right) \to$$

$$\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} \\ 0 & 1 & 0 & -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} \\ 0 & 0 & 1 & -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} \end{array}\right)$$

On en déduit: $A^{-1} = \begin{pmatrix} \frac{3}{4} & -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} \\ -\frac{1}{4} & -\frac{1}{4} & \frac{3}{4} \end{pmatrix}$.

Remarque II.5.2.

On constate que toutes les méthodes ont une complexité $N_{op}(n) \asymp 2n^3$, on remarquera que c'est TROIS fois plus que celle de la résolution (par Gauss ou LU) d'un systéme linéaire. C'est pourquoi il faut retenir le principe suivant déjà dit :

Pour résoudre un système linéaire, on n'a pas du tout intérêt en général à calculer l'inverse de la matrice.

II.5.2. Calcul du déterminant d'une matrice. — On a déjà remarqué que le déterminant de la matrice A s'obtient comme sous-produit des calculs de triangularisation puisque l'on a $\det A = \det A'$ si aucune permutation de lignes ou de colonnes n'est faite. On sait qu'une transposition de lignes ou de colonnes change le signe du déterminant, on aura donc $\det A = \pm \det T$ suivant la parité du nombre de transpositions effectuées. Le déterminant s'obtient donc au signe près en faisant le produit des différents pivots; il est nul dès que le rang du système est inférieur à n.

II.6. Méthode de Cholesky

La méthode de Cholesky[3] concerne le cas particulier mais assez fréquent des matrices **symétriques réelles définies positives**. La factorisation de Cholesky consiste, pour une matrice symétrique définie positive A, à déterminer une matrice triangulaire inférieure B telle que $A = B\,{}^tB$. C'est en fait une factorisation "LU" symétrisée.

La matrice B est en quelque sorte une "racine carrée" de A. Cette décomposition permet notamment de calculer la matrice inverse A^{-1}, de calculer le déterminant de A (égal au carré du produit des éléments diagonaux de B) ou encore de simuler une loi multinormale (loi de Gauss en plusieurs variables). Elle est aussi utilisée en chimie quantique pour accélérer les calculs.

Remarque II.6.1 (Rappel)**.**
La matrice réelle A est définie positive si et seulement si $\forall x \in \mathbb{R}^n$, ${}^txAx \geq 0$ avec égalité seulement si $x = 0$.
Si A est définie positive alors A est inversible (on a $Ax = 0 \Rightarrow {}^txAx = 0 \Rightarrow x = 0$).

On signale le résultat suivant de Sylvester[4].

Proposition II.6.2 (Critère de Sylvester)**.**

Soit $A = (a_{i,j})_{1\leq i,j\leq n} \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique. La matrice A est définie positive si et seulement si: pour tout k de 1 à n, $\det(a_{i,j})_{1\leq i,j\leq k} > 0$. Autrement dit, si tous les mineurs principaux sont strictement positifs.

Démonstration. — Cf. TD. □

Remarque II.6.3.
Cela explique pourquoi une matrice définie positive admet toujours une décomposition LU.

Théorème II.6.4 (Factorisation de Cholesky)**.**

Une matrice réelle symétrique définie positive A admet une unique factorisation $A = B\,{}^tB$ dite de Cholesky, où B est une matrice inversible triangulaire inférieure ayant des éléments diagonaux positifs.

Démonstration. — **a) Existence.** On a déjà vu qu'une matrice admettant une décomposition LU admettait aussi une décompostion $A = LDR$ où R est triangulaire supérieure avec une diagonale de 1 et $D = \mathrm{Diag}(d_1, \ldots, d_n)$. On peut facilement voir qu'une telle décomposition est unique. En effet si $A = LDR = L'D'R' = L(DR) = L'(D'R')$ on obtient deux décompositions LU de A donc $L = L'$ et $DR = D'R'$. Or la décomposition $U = DR$ est unique donc $D = D'$ et $R = R'$.
On a vu aussi que les matrices symétriques réelles définies positives admettent une d'écomposition

[3] André-Louis Cholesky, 1875-1918, polytechnicien français, ingénieur topographe et géodésien, commandant d'artillerie, met au point le procédé qui porte son nom en travaillant sur des applications de la géodésie (science qui mesure et représente la surface terrestre)
[4] James Joseph Sylvester, 1814-1897, mathématicien anglais.

LU, donc LDR.
Sachant que ${}^tA = A$ (car A symétrique), on tire $LDR = {}^tR\,D\,{}^tL$ et par unicité de la décomposition LDR on obtient que $L = {}^tR$ ou encore $R = {}^tL$. Ce qui donne $A = L\,D\,{}^tL$.
La matrice A est définie positive, donc A est inversible et par suite $D = \mathrm{Diag}(d_1, \ldots, d_n)$ et L également. De plus $\forall x \in \mathbb{R}^n \setminus \{0\}$, ${}^txAx > 0 \Leftrightarrow {}^t({}^tLx)\,D\,({}^tLx) > 0 \Leftrightarrow \forall y \in \mathbb{R}^n \setminus \{0\}$, ${}^tyDy > 0 \Rightarrow d_i > 0$ pour tout $i = 1, \ldots, n$. (A est la matrice dans la base canonique de $\mathbb{R}^n$ d'une forme quadratique q définie positive. On voit que D représente toujours la même forme quadratique q mais dans une nouvelle base associée à la matrice de passage ${}^tL$. Donc les coefficients diagonaux de D sont forcément positifs non nuls puisque q est définie positive.)
Il suffit à présent de poser $D = \Delta^2$ où $\Delta = \mathrm{Diag}(\sqrt{d_1}, \ldots, \sqrt{d_n})$ puis $B = L\Delta$. La matrice B est inversible (puisque L et Δ le sont) et clairement triangulaire inférieure. On a alors $A = B\,{}^tB$.
b) Unicité. Supposons qu'une telle décomposition existe, $A = B\,{}^tB$. On peut toujours factoriser B sous la forme $B = L\Delta$ avec Δ diagonale positive. Du coup, on $A = L(\Delta^2)\,{}^tL$. Par unicité de la décomposition LDR on a forcément unicité de L et de $D = \Delta^2$. Comme Δ est à diagonale positive forcément il est unique. □

II.6.1. Algorithme de calcul pour la factorisation $A = B\,{}^tB$ d'une matrice A symétrique définie positive. — Il suffit en posant des coefficients $b_{i,j}$ indéterminés sur la partie triangulaire inférieure de B d'écrire les $n(n+1)/2$ égalités résultant de l'égalité matricielle $A = B\,{}^tB$.
On obtient sur la partie triangulaire inférieure de A pour tous $1 \leq j \leq i \leq n$ les relations

$$a_{i,j} = \sum_{k=1}^{j} b_{i,k} b_{j,k}.$$

On en déduit un procédé de calcul par exemple colonne par colonne.

On va écrire l'égalité $A = B\,{}^tB$ en suivant les colonnes de A.

Colonne $j = 1$. On fait varier i de 1 à n:

$$\begin{aligned}
a_{1,1} &= b_{1,1}^2 && \Rightarrow b_{1,1} = \sqrt{a_{1,1}} \\
a_{2,1} &= b_{1,1}b_{2,1} && \Rightarrow b_{2,1} = \frac{a_{2,1}}{b_{1,1}} \\
&\vdots && \vdots \\
a_{n,1} &= b_{1,1}b_{n,1} && \Rightarrow b_{n,1} = \frac{a_{n,1}}{b_{1,1}}
\end{aligned}$$

Cela a permis de calculer la colonne $j = 1$ de B.
Supposons que l'on a calculé les $(j-1)$ premières colonnes de B.

Colonne j. On fait varier i de j à n:

$$\begin{aligned}
a_{j,j} &= \sum_{k=1}^{j} b_{j,k}^2 \qquad \Rightarrow b_{j,j} = \sqrt{a_{j,j} - \sum_{k=1}^{j-1} b_{j,k}^2} \\
a_{j+1,j} &= \sum_{k=1}^{j} b_{j,k} b_{j+1,k} \qquad \Rightarrow b_{j+1,j} = \frac{\left(a_{j+1,j} - \sum_{k=1}^{j-1} b_{j,k} b_{j+1,k}\right)}{b_{j,j}} \\
&\vdots \qquad\qquad\qquad\qquad \vdots \\
a_{n,j} &= \sum_{k=1}^{j} b_{j,k} b_{n,k} \qquad \Rightarrow b_{n,j} = \frac{\left(a_{n,j} - \sum_{k=1}^{j-1} b_{j,k} b_{n,k}\right)}{b_{j,j}}
\end{aligned}$$

On a ainsi calculé la j-ème colonne de B. On procède ainsi jusqu'à $j = n$.

Algorithme (décomposition de Cholesky)
Etant donné une matrice A symétrique définie positive, calcule la matrice B triangulaire inférieure telle que $A = B\,{}^tB$, (on ne s'intéresse qu'au calcul du triangle inférieur)
Etape 0 [initialisation] définir B égale à la matrice identité de taille n,
Etape 1 [grande boucle sur j]
pour j de 1 à n faire
• [calcul de $b_{j,j}$]
$b_{j,j} \leftarrow a_{j,j}$,
pour $k = 1$ à $j-1$ faire $b_{j,j} \leftarrow b_{j,j} - b_{j,k}^2$, fin pour
[se fait en une seule instruction via un produit scalaire]
$b_{j,j} \leftarrow (b_{j,j})^{1/2}$,
• [calcul des $b_{i,j}$ pour $i > j$]
pour $i = j+1$ à n faire
$b_{i,j} \leftarrow a_{i,j}$,
pour $k = 1$ à $j-1$ faire $b_{i,j} \leftarrow b_{i,j} - b_{i,k} b_{j,k}$, fin pour
[se fait en une seule instruction via un produit scalaire]
$b_{i,j} \leftarrow b_{i,j}/b_{j,j}$
fin pour
fin pour
Etape 2 [fin] sortir B.

Complexité. Pour chaque j de 1 à n:

1. calcul de $b_{j,j}$: $j-1$ produits (carrés), $j-1$ sommes, 1 racine carrée,
2. pour i de $j+1$ à n , calcul de $b_{i,j}$: $j-1$ produits, $j-1$ sommes, une division.

Il y a pour chaque j de 1 à n, $(n-j+1)$ fois $(2j-1)$ opérations élémentaires. Cela donne (poser $k = n-j+1$):

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

Au final $\boxed{N_{op}(n) \asymp \frac{n^3}{3}.}$

On constate que cet algorithme est deux fois plus rapide que celui de Gauss ou LU. Ceci s'explique par le fait que par rapport à une factorisation de type LU on a, par symétrie, deux fois moins de variables. Il sera particulièrement indiqué pour de telles matrices.

II.6.2. Applications de la facorisation de Cholesky. —

1. Résolution du système $AX = b$: on résout d'abord $By = b$ puis ${}^tBx = y$.
   Sur l'exemple $A = \begin{pmatrix} 4 & 2 & 4 \\ 2 & 2 & 4 \\ 4 & 4 & 12 \end{pmatrix}$ et $b = {}^t(1,1,1)$ on trouve aisément $B = \begin{pmatrix} 2 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 2 & 2 \end{pmatrix}$.
   On résout d'abord $By = b$ ce qui donne $y = {}^t(1/2, 1/2, -1/2)$ puis ${}^tBx = y$ ce qui donne $x = {}^t(0, 1, -1/4)$.
2. Calcul du déterminant : on a $\det A = (\det B)^2$.
3. Calcul de l'inverse : $A^{-1} = {}^tB^{-1}B^{-1}$.

II.7. Méthode QR (triangularisation orthogonale)

L'idée est de se ramener encore à un système triangulaire en utilisant une matrice orthogonale Q facile à inverser ($Q^{-1} = {}^tQ = Q^\star$). Qui plus est, une matrice orthogonale est bien conditionnée [cf TD1].

Théorème II.7.1 (Factorisation *QR*)**.**

Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice réelle carrée inversible. Il existe un unique couple (Q, R) où Q est une matrice orthogonale et R une matrice triangulaire supérieure dont tous les éléments diagonaux sont strictement positifs, tel que $A = QR$.

Remarque II.7.2.

- Cette factorisation se généralise aux matrices rectangulaires et cela sera utile pour la résolution des problèmes de moindres carrés. Cf. proposition II.8.9.
- La même factorisation existe dans le cas complexe, il faut remplacer orthogonale par unitaire.

Démonstration. — **Existence.** On note $c_1, c_2, \dots c_n$ les vecteurs colonnes de A.

$$A = \left(\boxed{c_1}\boxed{c_2}\dots\boxed{c_n}\right).$$

Ils forment une base de $\mathbb{R}^n$ puisque A est inversible. On orthonormalise cette base par la méthode de Gram-Schmidt que nous rappelons:

- on pose $p_1 = c_1$ et $q_1 = \frac{p_1}{\|p_1\|}$ (q_1 est donc un vecteur de norme 1), puis
- on pose $p_2 = \alpha_{1,2}q_1 + c_2$ avec $r_{2,1}$ tel que p_2 soit orthogonal à q_1 (donc à c_1). L'égalité $\langle q_1, p_2\rangle = 0$ se traduit par: $\alpha_{1,2} = -\langle q_1, c_2\rangle$.
  On normalise p_2 ce qui donne $q_2 = \frac{p_2}{\|p_2\|}$. Puis
- on pose $p_3 = \alpha_{1,3}q_1 + \alpha_{2,3}q_2 + c_3$ tel que p_3 orthogonal à $\mathrm{Vect}(q_1, q_2)$ (donc p_3 orthogonal à $\mathrm{Vect}(c_1, c_2)$). Les égalités $\langle q_1, p_3\rangle = \langle q_2, p_3\rangle = 0$ se traduisent par: $\alpha_{1,3} = -\langle q_1, c_3\rangle$ et $\alpha_{2,3} =$

$-\langle q_2, c_3\rangle$.
On normalise p_3, ce qui donne $q_3 = \dfrac{p_3}{\|p_3\|}$. Puis

– etc..

Par construction, la matrice $Q = \left(\boxed{q_1}\boxed{q_2}\dots\boxed{q_n}\right)$ constitué par les vecteurs colonnes q_i est orthogonale: ${}^tQQ = I$.

Posons $R = \left(r_{i,j}\right) = {}^tQA = \begin{pmatrix} {}^tq_1 \\ {}^tq_2 \\ \vdots \\ {}^tq_n \end{pmatrix}\begin{pmatrix} c_1 & c_2 & \cdots & c_n \end{pmatrix}$. Ainsi, $r_{i,j} = {}^tq_i\, c_j = \langle q_i, c_j\rangle$. Or p_i donc q_i est orthogonal à $\mathrm{Vect}(c_1, \cdots, c_{i-1})$ pour tout $2 \le i \le n$. Donc $r_{i,j} = 0$ pour tout $i > j$, c'est à dire R est une matrice triangulaire supérieure de la forme: $R = \begin{pmatrix} r_{1,1} & r_{1,2} & \dots & r_{1,n} \\ 0 & r_{2,2} & \dots & r_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & & r_{n,n} \end{pmatrix}$. On notera que pour tout $i < j$, on a $r_{i,j} = -\alpha_{i,j}$ (calculé dans la méthode de Gram-Schmidt). Sur la diagonale on a d'une part $r_{1,1} = \langle q_1, c_1\rangle = \|c_1\|\langle q_1, q_1\rangle = \|c_1\| > 0$ tandis que pour $1 \le i \le n$, puisque $c_i = p_i + \sum_{k=1}^{i-1} r_{k,i}q_k$ et que $\langle q_i, q_k\rangle = 0$ si $k < i$: $r_{i,i} = \langle q_i, c_i\rangle = \langle q_i, p_i\rangle = \|p_i\| > 0$. (Calculé dans la méthode de Gram-Schmidt).
Finalement l'égalité $R = {}^tQA$ se traduit par l'égalité matricielle $QR = A$, où Q et R ont les propriétés voulues. D'où l'existence de la décomposition.

Unicité. Si $A = Q_1R_1 = Q_2R_2$ alors $T = Q_2^{-1}Q_1 = R_2R_1^{-1}$ est à la fois triangulaire supérieure et orthogonale.
Cela implique que T soit diagonale avec des éléments diagonaux égaux à ± 1. En effet, rappelons qu'une matrice orthogonale a ses colonnes deux à deux orthogonales et de norme 1.
Comme R_1 et R_2 ont des éléments diagonaux positifs, il en est de même pour T donc $T = I_n$ ce qui donne finalement $R_1 = R_2$ et $Q_1 = Q_2$. □

Exemple II.7.3.

On considère $A = \begin{pmatrix} 1 & -1 & 2 \\ -1 & 1 & 0 \\ 0 & -2 & 1 \end{pmatrix}$ de sorte que $c_1 = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$, $c_2 = \begin{pmatrix} -1 \\ 1 \\ -2 \end{pmatrix}$ et $c_1 = \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix}$.

– On a $\|c_1\| = \sqrt{2}$, donc $q_1 = \dfrac{c_1}{\|c_1\|} = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \\ 0 \end{pmatrix}$. Ainsi $Q = \begin{pmatrix} 1/\sqrt{2} & * & * \\ -1/\sqrt{2} & * & * \\ 0 & * & * \end{pmatrix}$ et $R = \begin{pmatrix} \sqrt{2} & * & * \\ 0 & * & * \\ 0 & 0 & * \end{pmatrix}$.

– on pose $p_2 = \alpha_{1,2}q_1 + c_2$. L'égalité $\langle q_1, p_2\rangle = 0$ se traduit par: $\alpha_{1,2} = -\langle q_1, c_2\rangle = \sqrt{2}$ de sorte que $p_2 = {}^t(0,0,-2)$. Aussi $\|p_2\| = 2$, $q_2 = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}$ et par suite $Q = \begin{pmatrix} 1/\sqrt{2} & 0 & * \\ -1/\sqrt{2} & 0 & * \\ 0 & -1 & * \end{pmatrix}$ et $R = \begin{pmatrix} \sqrt{2} & -\sqrt{2} & * \\ 0 & 2 & * \\ 0 & 0 & * \end{pmatrix}$.

– on pose $p_3 = \alpha_{1,3}q_1 + \alpha_{2,3}q_2 + c_3$. Les égalités $\langle q_1, p_3\rangle = \langle q_2, p_3\rangle = 0$ se traduisent par: $\alpha_{1,3} = -\langle q_1, c_3\rangle = -\sqrt{2}$ et $\alpha_{2,3} = -\langle q_2, c_3\rangle = 1$. Ainsi $p_3 = {}^t(1,1,0)$, donc $\|p_3\| = \sqrt{2}$ et $q_3 = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{pmatrix}$. En conséquence $A = QR$ avec $Q = \begin{pmatrix} 1/\sqrt{2} & 0 & 1/\sqrt{2} \\ -1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & -1 & 0 \end{pmatrix}$ et $R = \begin{pmatrix} \sqrt{2} & -\sqrt{2} & \sqrt{2} \\ 0 & 2 & -1 \\ 0 & 0 & \sqrt{2} \end{pmatrix}$.

Complexité. La résolution du système $Ax = b$ par méthode QR nécessite ($Ax = b \Leftrightarrow Rx = {}^tQb$):

– factorisation de Gram-Schmidt : $N_{op}(n) \asymp 2n^3$,
– calcul de ${}^tQb$: $N_{op}(n) \asymp 2n^2$,
– résolution du système triangulaire : $N_{op}(n) \asymp n^2$.

Au final, $\boxed{N_{op}(n) \asymp 2n^3.}$ On remarque que c'est trois fois plus lent que la méthode de Gauss, donc cette méthode sera peu utilisée pour les systèmes linéaires carrés.
En revanche elle se généralise très bien aux systèmes rectangulaires et sera donc utilisée pour la résolution des problèmes aux moindres carrés, on y reviendra.

Remarque II.7.4.
En pratique on n'utilise pas la méthode de Gram-Schmidt pour la factorisation QR car cette méthode n'est pas numériquement stable. On lui préfère la méthode dite de **Householder**, plus stable (cf. TP). En outre, la complexité est un peu meilleure : $N_{op}(n) \asymp 4n^3/3$.

Remarque II.7.5.

Les fonctions `np.linalg.qr` ou `sc.linalg.qr` de Numpy ou de Scipy appliquées à une matrice (rectangulaire) A, retournent le couple `q,r` de la décomposition QR de A mais avec des conventions de normalisation non usuelles: les éléments diagonaux de la matrice R ne sont pas nécessairement positifs.

II.8. Application aux problèmes de moindres carrés

II.8.1. Introduction. — Supposons que l'on ait une fonction f inconnue dont on connaît seulement les valeurs en un certain nombre de points $t_1, \ldots, t_m$.

t_i	t_1	t_2	t_3	$\ldots$	t_m
$y_i = f(t_i)$	y_1	y_2	y_3	$\ldots$	y_m

On fait une hypothèse : f appartient à un certain espace vectoriel V de fonctions (pas forcément polynomiales), de dimension finie n dont on connaît une base $(\phi_1, \ldots, \phi_n)$. On sait alors que l'on peut

exprimer f comme combinaison linéaire des éléments de la base sous la forme $f = x_1\phi_1 + \cdots + x_n\phi_n$, il reste simplement à déterminer les coefficients x_i.

En pratique, les ϕ_i peuvent être n'importe quelles fonctions, par exemple des fonctions puissances $1, t, t^2, \ldots$ ou des fonctions circulaires $\sin(t), \sin(2t), \sin(3t), \ldots$ ou bien encore des exponentielles e^t, e^{2t}, e^{3t}.

Si le nombre de points m où f est connue est égal à la dimension de l'espace vectoriel n il y a, en général, une solution unique (n équations, n inconnues) car on arrive alors à un système de Cramer. En pratique cependant, comme il y a souvent des erreurs de mesure sur les $y_i = f(t_i)$, on dispose en général d'un nombre de points $m > n$. Dans ce cas, la détermination des coefficients n'a pas, en général, de solution car il y a plus d'équations que d'inconnues (système sur-déterminé). Cf. exemple II.8.1.

Exemple II.8.1.

On dispose du tableau suivant de valeurs concernant la fonction f

t_i	1	3	4	6	7
$y_i = f(t_i)$	−2.1	−0.9	−0.6	0.6	0.9

On cherche une droite passant par ces 5 points. On recherche donc $f \in V = \mathbb{R}_1[t]$ (espace des polynômes de degré ≤ 1) sous la forme $f = x_1\phi_1 + x_2\phi_2$ avec $\phi_1(t) = 1$, $\phi_2(t) = t$. On veut donc que $\begin{cases} x_1\phi_1(1) + x_2\phi_2(1) = -2.1 \\ x_1\phi_1(3) + x_2\phi_2(3) = -0.9 \\ \vdots \\ x_1\phi_1(7) + x_2\phi_2(7) = 0.9 \end{cases}$, soit la résolution du système $Ax = b$ avec $A = \begin{pmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 4 \\ 1 & 6 \\ 1 & 7 \end{pmatrix}$ et $b = \begin{pmatrix} -2.1 \\ -0.9 \\ -0.6 \\ 0.6 \\ 0.9 \end{pmatrix}$. Si on représente les points dans le plan (cf. Fig. 1), on voit clairement que ce problème n'a pas de solution.

Il faut en fait envisager un autre point de vue qui est plutôt de chercher dans l'espace vectoriel V la fonction $f^\star$ qui approche le mieux f. Ceci est lié au choix d'une norme capable de mesurer ce qu'on entend par bonne approximation.

D'un point de vue algèbre linéaire, au lieu de chercher la solution unique du système $Ax = b$, on va chercher le vecteur $x^\star$ qui minimise $\|Ax - b\|$.

$$\|Ax^\star - b\| = \min_{x \in \mathbb{R}^n} \|Ax - b\|.$$

Cette démarche est une généralisation de la résolution d'un système linéaire $Ax = b$ de Cramer aux cas où A n'est pas inversible.

On va choisir la norme euclidienne, ce qui va conduire à une minimisation d'une somme de carrés d'où le nom du problème. Cf. exemple II.8.2.

Exemple II.8.2.

On dispose du tableau suivant de valeurs concernant la fonction f

t_i	1	3	4	6	7
$y_i = f(t_i)$	−2.1	−0.9	−0.6	0.6	0.9

On cherche une droite d'équation $y = x_1 + x_2 t$ qui s'adapte le mieux aux valeurs du tableau, (cf. Fig. 1). On va rechercher $x^\star = {}^t(x_1, x_2)$ tel que $\|Ax^\star - b\|_2 = \min_{x\in\mathbb{R}^2} \|Ax - b\|_2$ où A et b sont donnés dans l'exemple II.8.1. On obtient ainsi la **droite de régression linéaire** dessinée sur la Fig. 1.

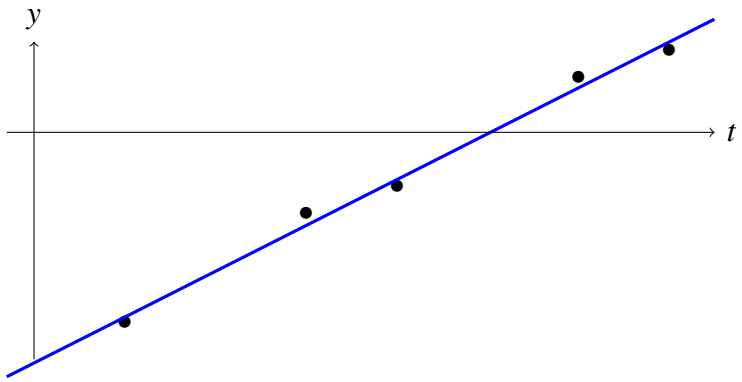


FIGURE 1. Nuage de points et droite de régression linéaire

Définition II.8.3.

Soit V un espace vectoriel de fonctions numériques. On appelle **problème aux moindres carrés discrets** attaché aux points $t_1, \dots, t_m$, à la fonction f donnée en $t_1, \dots, t_m$, $y_i = f(t_i)$, et aux fonctions linéairement indépendantes $(\phi_1, \dots, \phi_n) \in V^n$ avec $m > n$, la recherche de la fonction $f^\star = \sum_{i=1}^{n} x_i \phi_i \in V$ telle que $\|f^\star - f\|^2 := \sum_{i=1}^{m} (f^\star(t_i) - f(t_i))^2 = \sum_{i=1}^{m} (f^\star(t_i) - y_i)^2$ soit minimale parmi les fonctions de $g \in V$, c'est à dire: $\|f^\star - f\| = \inf_{g\in V} \|g - f\|$.

Dans tout ce qui suit, on suppose que le nombre n de fonctions ϕ_j, est inférieur au nombre m de points t_i, $m > n$.

Exemple II.8.4.

Dans l'exemple II.8.1, on a $V = \mathbb{R}_1[t]$, $\phi_1(t) = 1$, $\phi_2(t) = t$. On recherche $f^\star = x_1\phi_1 + x_2\phi_2$ qui minimise $\sum_{i=1}^{m} (f^\star(t_i) - y_i)^2$. Il s'agit donc de minimiser en x_1 et x_2 l'expression:

$$F(x_1, x_2) = \sum_{i=1}^{m} (x_1 + x_2 t_i - y_i)^2.$$

Cette recherche de la "droite de régression linéaire" $y = x_1 + x_2 t$ peut se faire par les méthodes classiques de calcul différentiel: l'application $F : (x_1, x_2) \in \mathbb{R}^2 \mapsto F(x_1, x_2) \geq 0$ est différentiable (c'est une fonction polynomiale de degré 2 en (x_1, x_2)) et minorée (par 0), elle admet donc un minimum où le gradient s'annule. Le couple (x_1, x_2) est donc solution du système $\dfrac{\partial F}{\partial x_1} = \dfrac{\partial F}{\partial x_2} = 0$.
Pour mémoire la droite de régression linéaire a pour équation: $y = \hat{\alpha} t + \hat{\beta}$ où $\hat{\alpha} = \dfrac{\mathrm{cov}(T, Y)}{\mathrm{var(T)}}$ et $\hat{\beta} = \overline{Y} - \hat{\alpha}\overline{T}$, et où $\overline{Y}$ et $\overline{T}$ sont les moyennes des y_i et t_i respectivement.
Mais nous allons voir ci-dessous une façon algébrique de résoudre le problème en toute généralité.

Pour faire la recherche de la fonction $f^\star$, on regarde $f^\star$ comme un vecteur de l'espace vectoriel $V = \mathrm{Vect}(\phi_1, \ldots, \phi_n)$ de coordonnées $x = {}^t(x_1, \ldots, x_n)$ dans la base $(\phi_1, \ldots, \phi_n)$, $f^\star = \sum_{i=1}^{n} x_i \phi_i$. L'expression à minimiser est:

$$\|f^\star - f\|^2 := \sum_{i=1}^{m} (f^\star(t_i) - y_i)^2 = \|Ax - b\|_2^2,$$

où $b = {}^t(y_1, \ldots, y_m)$ et A est la matrice à $m > n$ lignes et n colonnes $A = \left(\phi_j(t_i)\right)_{1 \leq i \leq m, 1 \leq j \leq n} \in \mathcal{M}_{m,n}(\mathbb{R})$:

$$A := \begin{pmatrix} \phi_1(t_1) & \phi_2(t_1) & \ldots & \phi_n(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \ldots & \phi_n(t_2) \\ \vdots & & & \\ \phi_1(t_m) & \phi_2(t_m) & \ldots & \phi_n(t_m) \end{pmatrix}.$$

Tout se passe donc dans l'espace vectoriel $E = \mathbb{R}^m$ (on a Ax et b dans R^m) et il s'agit simplement de trouver un vecteur $x \in \mathbb{R}^n$ qui minimise (ce qu'on appelle) le **résidu** $R = Ax - b$, c'est à dire: trouver $x^\star \in \mathbb{R}^n$ tel que $\|Ax^\star - b\|_2 = \min_{x \in \mathbb{R}^n} \|Ax - b\|_2$.

L'espace vectoriel $E = \mathbb{R}^m$ est un espace vectoriel de dimension finie qui, muni du produit scalaire naturel $\langle x, y \rangle = \sum_{i=1}^{m} x_i y_i$, est un espace euclidien. Dans cet espace euclidien, les vecteurs colonnes u_i de la matrice A engendrent un sous-espace vectoriel $F = \mathrm{Im}\, A = \mathrm{Vect}\,\{Ax, x \in \mathbb{R}^n\} \subset \mathbb{R}^m$ de dimension $n < m$ (si A est de rang maximal n, sinon la dimension est moindre).

Le problème qui nous occupe est finalement celui de trouver dans E la distance minimale entre le vecteur b et le sous-espace vectoriel F.

II.8.2. Problème aux moindres carrés discrets et équation normale. — On va utiliser le théorème suivant de géométrie euclidienne élémentaire suivant.

Théorème II.8.5.

Soient E un espace euclidien et F un sous-espace vectoriel de E de dimension finie. Pour tout vecteur $u \in E$ il existe $v_0 \in F$ tel que

$$d(u, F) := \inf_{v \in F} \|u - v\| = d(u, v_0).$$

Le vecteur v_0 est exactement le projeté orthogonal de u sur F, c'est à dire l'unique vecteur $p(u)$ qui vérifie:

$$p(u) \in F \quad \text{et} \quad (u - p(u)) \perp F.$$

Démonstration. — La démonstration est classique. On note $F^\perp$ l'orthogonal[(5)] de F. On a donc $E = F \oplus F^\perp$, de sorte qu'on peut décomposer le vecteur u suivant F et $F^\perp$: $u = p(u) + (u - p(u))$ où $p(u)$ désigne le projeté orthogonal de u sur F. Cf. Fig. 2. Pour tout $v \in F$, on a $p(u) - v \in F$ de sorte que $\langle u - p(u), p(u) - v\rangle = 0$ car $u - p(u) \in F^\perp$. Aussi:

$$\begin{aligned}\|u - v\|^2 = \|u - p(u) + p(u) - v\|^2 &= \|u - p(u)\|^2 + \|p(u) - v\|^2 + 2\langle u - p(u), p(u) - v\rangle \\ &= \|u - p(u)\|^2 + \|p(u) - v\|^2 \geq \|u - p(u)\|^2,\end{aligned}$$

avec égalité si et seulement si $v = p(u)$. □

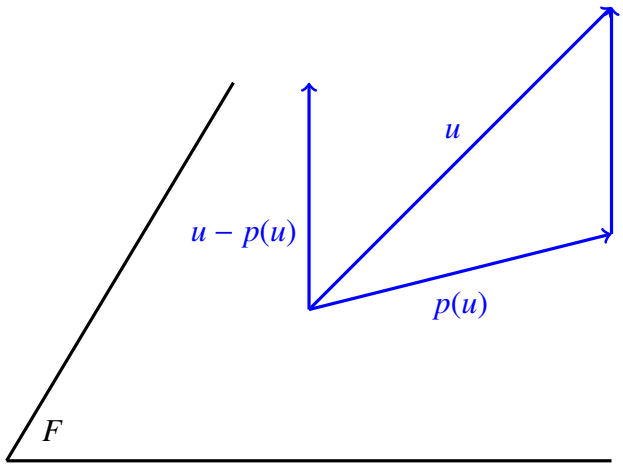


FIGURE 2.

Théorème II.8.6.

Soit $A \in \mathcal{M}_{m,n}(\mathbb{R})$ une matrice à $m > n$ lignes et n colonnes et $b \in \mathbb{R}^m$. Le problème au moindre carrés,

$$\text{(II.1)} \qquad \text{trouver } x^\star \in \mathbb{R}^n \text{ tel que } \|Ax^\star - b\|_2 = \min_{x\in\mathbb{R}^n} \|Ax - b\|_2,$$

admet une unique solution $x^\star \in \mathbb{R}^n$ si $\mathrm{Ker}(A) = \{0\}$ ou, de manière équivalente, si $\mathrm{rang}(A) = n$. Dans ce cas $x^\star$ vérifie l'équation: ${}^tAA\,x^\star = {}^tAb$ où ${}^tAA \in \mathcal{M}_n(\mathbb{R})$ est symétrique définie positive.

Démonstration. — Le problème qui nous occupe est de trouver un vecteur $x^\star = {}^t(x_1, \ldots, x_n)$ tel que $Ax^\star$ soit le projeté orthogonal du vecteur b sur $F = \mathrm{Vect}\{Ax, x \in \mathbb{R}^n\} = \mathrm{Vect}(u_1, \cdots, u_n)$ où les u_i désignent les vecteurs colonnes de A.

(5) L'orthogonal $F^\perp$ de $F \subset E$ est l'ensemble des vecteurs de E orthogonaux à tous les vecteurs de F : $F^\perp = \{v \in E \mid \langle v, u\rangle = 0, \forall u \in F\}$. L'orthogonal $F^\perp$ de F est un sous-espace vectoriel de E, supplémentaire directe de F : $E = F \oplus F^\perp$.

En appliquant le théorème II.8.5, on sait que ce vecteur $x = x^\star$ vérifie $(Ax^\star - b) \perp F$, soit encore ${}^t u_i\,(Ax^\star - b) = 0$ pour tout i de 1 à n. Ceci se traduit globalement par ${}^tA(Ax^\star - b) = 0$, soit encore

$$ {}^tAA\,x^\star = {}^tAb. $$

Or la matrice carrée $G = {}^tAA \in \mathcal{M}_n(\mathbb{R})$ est symétrique définie positive, donc inversible. En effet: ${}^t x({}^tAA)x = {}^t(Ax)\,Ax = \|Ax\|_2^2 \geq 0$ et $\|Ax\|_2^2 = 0 \Leftrightarrow Ax = 0 \Leftrightarrow x = 0$ car $\operatorname{Ker} A = \{0\}$.
Le système a donc une unique solution, $x^\star = G^{-1}({}^tAb)$. □

Définition II.8.7 (Equation normale)**.**

Le système carré $n \times n$ d'équations $\boxed{{}^tAA\,x = {}^tAb}$ s'appelle l'équation **normale** du problème (II.1).

Remarque II.8.8.

- Si $\operatorname{Ker} A \neq \{0\}$, c'est à dire si $\operatorname{rang}(A) < n$, on peut montrer qu'il existe toujours au moins une solution mais elle n'est pas unique. Deux solutions diffèrent d'un vecteur de $\operatorname{Ker} A$.
- Si $\operatorname{Ker} A = \{0\}$, la matrice $A^\dagger = ({}^tAA)^{-1}\,{}^tA$ s'appelle la **pseudo-inverse** de A. Ainsi $x^\star = A^\dagger b$.

II.8.3. Retour sur le problème introductif. — On recherchait $f^\star \in V = \operatorname{Vect}(\phi_1, \ldots, \phi_n)$, de coordonnées $x = {}^t(x_1, \ldots, x_n)$ dans la base $(\phi_1, \ldots, \phi_n)$, qui minimise l'expression: $\sum_{i=1}^{m} (f^\star(t_i) - y_i)^2 = \|Ax - b\|_2^2$, où $b = {}^t(y_1, \ldots, y_m)$ et $A := \begin{pmatrix} \phi_1(t_1) & \phi_2(t_1) & \ldots & \phi_n(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \ldots & \phi_n(t_2) \\ \vdots & & & \\ \phi_1(t_m) & \phi_2(t_m) & \ldots & \phi_n(t_m) \end{pmatrix} \in \mathcal{M}_{m,n}(\mathbb{R})$.

Méthode de l'équation normale. D'un point de vue pratique on calcule successivement:

- la matrice $G = {}^tAA$ qui n'est autre que la matrice dite de Gram, $G = \left(\langle \phi_i, \phi_j \rangle\right)_{1 \leq i,j \leq n}$ pour le produit scalaire discret[(6)] $\langle f, g \rangle = \sum_{j=1}^{m} f(t_j)g(t_j)$ défini sur $V = \operatorname{Vect}(\phi_1, \cdots, \phi_n)$
- le vecteur $y = {}^tAb = (\langle f, \phi_i \rangle)_{1 \leq i \leq n}$.

On résout ensuite le système linéaire $Gx = y$. La matrice G étant réelle symétrique définie positive, la méthode la plus rapide est la décomposition de Cholesky. On peut aussi évidemment utiliser la méthode de Gauss ou LU.

Une alternative par décomposition QR. On peut aussi appliquer la méthode QR à la matrice A. On suppose $m > n$ et $\operatorname{Ker} A = \{0\}$, on cherche la solution unique du problème aux moindres carrés.

En étendant la méthode de la décomposition QR précédente aux matrices rectangulaires, on peut établir le résultat suivant:

[(6)] Ce produit scalaire discret précédent est donc bien un produit scalaire sur $V = \operatorname{Vect}(\phi_1, \cdots, \phi_n)$ mais attention c'est simplement une forme bilinéaire symétrique, positive (pas définie) si on l'étend à un espace de fonctions plus générales

Proposition II.8.9.

Soit A une matrice réelle, $m \times n$ avec $m > n$, de rang maximal n. Il existe une matrice orthogonale Q de taille $m \times m$ et une matrice R de taille $m \times n$ de la forme $R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$ avec R_1 matrice carrée $n \times n$ triangulaire supérieure inversible dont tous les éléments diagonaux sont positifs telles que $A = QR$.

Démonstration. — On reprend la preuve du théorème II.7.1. Il suffit d'orthonormaliser à la Gram-Schmidt les n colonnes de A. On obtient un début de base orthonormale $(q_1, \dots, q_n)$ qu'on complète ensuite avec des vecteurs $(q_{n+1}, \dots, q_m)$ pour en faire une base orthonormale de $\mathbb{R}^m$. □

On va appliquer ce résultat au problème de moindre carrés (II.1). On considère la décomposition $A = QR$ avec Q orthogonale, ${}^tQQ = I$, et on écrit:

$$\begin{aligned} \|Ax - b\|_2^2 &= \|QRx - b\|_2^2 = \|Q(Rx - {}^tQb)\|_2^2 \\ &= {}^t\Big(Q(Rx - {}^tQb)\Big)\Big(Q(Rx - {}^tQb)\Big) \\ &= {}^t(Rx - {}^tQb)\,{}^tQQ\,(Rx - {}^tQb) = \|Rx - {}^tQb\|_2^2. \end{aligned}$$

On pose ${}^tQb = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$ où c_1 est de taille n et c_2 est de taille $m - n$. Du coup

$$\|Ax - b\|_2^2 = \|R_1x - c_1\|_2^2 + \|c_2\|_2^2$$

et le mimimum est atteint en $x^\star$ vérifiant $R_1x^\star = c_1$. Cet $x^\star$ est bien unique puisque R_1 est inversible.

Exemple II.8.10.

On prend l'exemple II.8.1. La base choisie est $\phi_1(t) = 1$, $\phi_2(t) = t$ et on a le tableau:

t	1	3	4	6	7
$\phi_1(t)$	1	1	1	1	1
$\phi_2(t)$	1	3	4	6	7
$f(t)$	−2.1	−0.9	−0.6	0.6	0.9

Ainsi $A = \begin{pmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 4 \\ 1 & 6 \\ 1 & 7 \end{pmatrix}$. et $b = {}^t(-2.1, -0.9, -0.6, 0.6, 0.9)$.

1. **Méthode de l'équation normale.** On calcule la matrice $G = {}^tAA = \begin{pmatrix} \langle \phi_1, \phi_1 \rangle & \langle \phi_1, \phi_2 \rangle \\ \langle \phi_2, \phi_1 \rangle & \langle \phi_2, \phi_2 \rangle \end{pmatrix}$ par les produits scalaires. Par exemple $\langle \phi_1, \phi_2 \rangle = \sum_{j=1}^{5} \phi_1(t_j)\phi_2(t_j) = 1 + 3 + 4 + 6 + 7 = 21$. Au final: $G = \begin{pmatrix} 5 & 21 \\ 21 & 111 \end{pmatrix}$. On calcule le vecteur $y = {}^tAb = \begin{pmatrix} -2.1 \\ 2.7 \end{pmatrix}$. La résolution de l'équation $Gx = y$ par méthode de Cholesky fournit l'unique solution: $x^\star = {}^t(-2.5421, 0.5053)$. La

fonction solution du problème au moindre carré est donc;

$$f^\star(t) = -2.5421 + 0.5053t.$$

2. **Méthode QR.** On calcule la décomposition QR de $A = \left(\boxed{a_1}\,\boxed{a_2}\right)$.

   – On a $\|a_1\| = \sqrt{5}$, donc $q_1 = \dfrac{a_1}{\|a_1\|} = \dfrac{1}{\sqrt{5}}{}^t(1,1,1,1,1)$. Ainsi $Q = \begin{pmatrix} 1/\sqrt{5} & * & * & * & * \\ 1/\sqrt{5} & * & * & * & * \\ 1/\sqrt{5} & * & * & * & * \\ 1/\sqrt{5} & * & * & * & * \\ 1/\sqrt{5} & * & * & * & * \end{pmatrix}$ et $R = \begin{pmatrix} \sqrt{5} & * \\ 0 & * \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$.

   – on pose $p_2 = \alpha_{1,2}q_1 + a_2$. L'égalité $\langle q_1, p_2\rangle = 0$ donn $\alpha_{1,2} = -\langle q_1, a_2\rangle = -\dfrac{21}{\sqrt{5}}$. Ainsi $p_2 = \dfrac{1}{5}{}^t(-16,-6,-1,9,14)$ et $\|p_2\| = \sqrt{114/5}$ et $q_2 = \dfrac{p_2}{\|p_2\|} = \dfrac{1}{\sqrt{570}}{}^t(-16,-6,-1,9,14)$. Par suite $Q = \begin{pmatrix} 1/\sqrt{5} & -16/\sqrt{570} & * & * & * \\ 1/\sqrt{5} & -6/\sqrt{570} & * & * & * \\ 1/\sqrt{5} & -1/\sqrt{570} & * & * & * \\ 1/\sqrt{5} & 9/\sqrt{570} & * & * & * \\ 1/\sqrt{5} & 14/\sqrt{570} & * & * & * \end{pmatrix}$ et $R = \begin{pmatrix} \sqrt{5} & 21/\sqrt{5} \\ 0 & \sqrt{114/5} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$.

On calcule $c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = Q^\star b$ avec $c_1 = \begin{pmatrix} -0.9388 \\ 0.1054 \end{pmatrix} \in \mathbb{R}^2$ puis on résout l'équation $R_1 x = c_1$. On trouve: $x = {}^t(-2.5421, 0.5053)$. C'est bien la même solution.

Remarque II.8.11.
Si par curiosité, on s'autorise à avoir un terme en t^2, on obtient alors

$$f^\star(x) = -0.012t^2 - 2.68t + 0,6.$$

Le terme en t^2 est faible comme attendu, cf. Fig. 1.

II.8.4. Conclusion. — Pour la résolution d'un problème aux moindres carrés (II.1), on vient de voir deux méthodes:

- par l'équation normale avec la méthode de Cholesky,
- par la décomposition QR de A.

Laquelle choisir ?
Celle de Cholesky est plus rapide. Cependant si la matrice A est mal conditionnée (avec une notion

de conditionnement généralisé qui utilise la pseudo-inverse, cf. remarque II.8.8) il vaut mieux utiliser la méthode QR. En effet pour l'équation normale, c'est la matrice $G = {}^tAA$ qui sert et ${}^tAA$ sera encore plus mal conditionnée que A. En revanche par QR, $\mathrm{cond}_2(A) = \mathrm{cond}_2(R)$ car Q orthogonale (cf. proposition I.4.3). Le conditionnement n'empire pas.

CHAPITRE III

SYSTÈMES LINÉAIRES : MÉTHODES ITÉRATIVES STATIONNAIRES

III.1. Principes et résultats généraux

Le chapitre précédent a été consacré à des méthodes directes de résolution de systèmes linéaires, basées sur l'élimination de Gauss. Celle-ci devient assez vite impraticable pour des grands systèmes: outre le problème de stockage (capacité mémoire) elle requiert un nombre d'opérations de l'ordre de N^2m si N est la taille et m la largeur de bande de la matrice. Les méthodes itératives ont été proposées pour remédier à ces difficultés. Elles ont l'avantage d'utiliser uniquement la matrice du système en tant qu'opérateur linéaire, c'est-à-dire par son action sur un vecteur. Il est donc possible de réduire le stockage de la matrice à ses éléments non nuls. Le gain de mémoire par rapport au solveur direct est d'autant plus élevé que le système est grand et que la matrice possède une grande largeur de bande. Cependant ces solveurs sont extrêmement sensibles aux propriétés de la matrice du système et en particulier à son nombre de conditionnement.

Dans la suite $\mathbb{K}$ est égal à $\mathbb{R}$ ou $\mathbb{C}$.

III.1.1. Principe général. — On considère le système linéaire

$$Ax = b \tag{III.1}$$

avec $A \in \mathcal{M}_n(\mathbb{K})$ inversible et $x, b \in \mathbb{K}^n$. On recherche l'unique solution $x^\star \in \mathbb{K}^n$ de l'équation. Le principe des méthodes développées dans ce chapitre repose sur l'idée de décomposer la matrice A en la différence de 2 matrices,

$$A = M - N \tag{III.2}$$

où :

1. $M \in \mathcal{M}_n(\mathbb{K})$ est une matrice inversible,
2. le système linéaire $My = c$ est simple à résoudre, avec un coût de calcul faible, typiquement M diagonale ou triangulaire.

On va alors approcher la solution $x^\star$ de (III.1) à l'aide d'une suite (x_k), $x_k \in \mathbb{K}^n$, définie par récurrence:

$$x_0 \text{ donné}, \qquad x_{k+1} = M^{-1}Nx_k + M^{-1}b \tag{III.3}$$

Remarques.

1. Dans un tel schéma itératif, les matrices M et N sont fixées une fois pour toute, elle ne dépendent pas de n. C'est la raison pour laquelles ces méthodes sont dites itératives **stationnaires**.

2. L'application $x \in \mathbb{K}^n \mapsto M^{-1}Nx_k + M^{-1}b$ est continue. Donc, si la suite (x_k) converge vers $y \in \mathbb{K}^n$, alors $My = Ny + b$ par passage à la limite, c'est à dire $Ay = b$. Donc $y = x^\star$ par unicité de la solution.
3. On ne va pas calculer M^{-1}, on aura juste à savoir résoudre l'équation $\boxed{Mx_{k+1} = Nx_k + b}$.
4. Les méthodes itératives sont sensibles au conditionnement de M.

Pour les raisons qui suivent, on introduit la définition suivante:

Définition III.1.1.

La matrice $P = M^{-1}N$ est appelée **matrice d'itération** de la méthode itérative (III.3).

Nous avons la condition nécessaire et suffisante suivante de convergence.

Proposition III.1.2 (Condition nécessaire et suffisante de convergence)**.**

La suite (III.3) converge vers la solution $x^\star$ du système $Ax = b$ pour tout choix de x_0 si et seulement si la matrice d'itération $P = M^{-1}N$ vérifie $\rho\left(P\right) < 1$.

Remarque III.1.3.
On rappelle que $\rho(P)$ est le rayon spectral de P, cf. définition I.2.10. On rappelle que $\rho(P) \leq \|P\|$ pour toute norme $\|.\|$ subordonnée à une norme vectorielle, cf. proposition I.2.12.

Démonstration. — Posons $e_k = x^\star - x_k \in \mathbb{K}^n$ l'erreur à l'étape k. On a

$$\begin{cases} Mx_{k+1} = Nx_k + b \\ Mx^\star = Nx^\star + b \end{cases} \Rightarrow e_{k+1} = M^{-1}Ne_k \Rightarrow e_k = \left(M^{-1}N\right)^k e_0$$

La suite (x_k) converge vers 0 pour tout choix de x_0 ssi $\left(M^{-1}N\right)^n e_0 \to 0$ pour tout choix de e_0, ce qui est le cas ssi $\rho\left(M^{-1}N\right) < 1$ par la proposition I.3.1. □

III.1.2. Critère d'arrêt. — Contrairement aux méthodes directes, les méthodes itératives nécessitent de définir un critère d'arrêt. Cela nécessite une évaluation de l'erreur $\boxed{e_k = x^\star - x_k}$ au rang k. Deux types de tests sont utilisés. (Par la suite $b \neq 0$ donc $x^\star \neq 0$).

1. Un type de test souvent utilisé repose sur l'évaluation du **résidu** $\|r_k\| = \|Ax_k - b\|$. Notons que

$$e_k = A^{-1}(Ax^\star - Ax_k) = A^{-1}(b - Ax_k)$$

de sorte $\|e_k\| \leq \|A^{-1}\|.\|r_k\|$ pour une norme matricielle $\|.\|$ donnée sur $\mathbb{K}^n$. Ainsi, si la norme $\|A^{-1}\|$ n'est pas trop grande, l'erreur e_k entre la solution approchée et la solution exacte est de l'ordre du vecteur résidu. Le test simple d'arrêt est alors $\boxed{\|r_k\| < \varepsilon}$ pour un **seuil de tolérance** $\varepsilon > 0$ fixé.

On pourra cependant préfèrer contrôler l'erreur relative $\dfrac{\|e_k\|}{\|x^\star\|}$. Comme $b = Ax^\star$ on a $\|b\| \leq \|A\|.\|x^\star\|$ et par suite: $\dfrac{\|e_k\|}{\|x^\star\|} \leq \dfrac{\|A\|.\|e_k\|}{\|b\|} \leq \text{cond}(A)\dfrac{\|r_k\|}{\|b\|}$. Le terme $\dfrac{\|r_k\|}{\|b\|}$ s'appelle le

résidu normalisé et la condition d'arrêt est $\boxed{\frac{\|r_k\|}{\|b\|} < \varepsilon}$ pour un seuil de tolérance $\varepsilon > 0$ fixé.

2. Le test précédent est assez couteux en calcul. On pourra alors plus simplement considérer l'estimateur $\|x_k - x_{k-1}\|$ de l'erreur $\|e_k\|$. En effet, puisque $e_k = Pe_{k-1}$ (où $P = M^{-1}N$ est la matrice d'itération), on a:

$$\|e_k\| \leq \|P\|.\|e_{k-1}\| \leq \|P\|.\|e_k + x_k - x_{k-1}\| \leq \|P\|.\Big(\|e_k\| + \|x_k - x_{k-1}\|\Big).$$

Ainsi, en supposant que $\|P\| < 1$, on obtient: $\|e_k\| \leq \dfrac{\|P\|}{1 - \|P\|}\|x_k - x_{k-1}\|$.

Le test simple d'arrêt est alors $\boxed{\|x_k - x_{k-1}\| < \varepsilon}$ pour un seuil de tolérance $\varepsilon > 0$ fixé.

III.2. Méthode de Jacobi

Nous présentons la méthode de Jacobi[(1)] comme premier exemple de méthode itérative.

Définition III.2.1 (Méthode de Jacobi)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$ une matrice inversible dont la diagonale D est inversible. La **méthode de Jacobi** consiste à choisir $M = D$ et $N = D - A$. La matrice d'itération $M^{-1}N$ est notée $P_J = P_J(A)$.

Exemple III.2.2.

Soit $A = \begin{pmatrix} 4 & 2 & 1 \\ -1 & -3 & -1 \\ 1 & 0 & 2 \end{pmatrix}$. Cette matrice A est inversible car à diagonale $D = \mathrm{Diag}(4, -3, 2)$ strictement dominante (et donc D est inversible). La méthode itérative de Jacobi de résolution du système $Ax = b \in \mathbb{R}^3$ consiste à écrire $A = D - (D - A)$ et à considérer la suite $(x_k \in \mathbb{R}^3)$ définie par $x_0 \in \mathbb{R}^3$ et $Dx_{k+1} = (D - A)x_k + b$. La matrice d'itération est $P_J = D^{-1}(D - A) = \mathrm{Diag}(1/4, -1/3, 1/2)\begin{pmatrix} 0 & -2 & -1 \\ 1 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1/2 & -1/4 \\ -1/3 & 0 & -1/3 \\ -1/2 & 0 & 0 \end{pmatrix}$

La proposition suivante fournit une condition suffisante de convergence de la méthode de Jacobi.

Proposition III.2.3 (Condition suffisante de convergence pour Jacobi)**.**

Si $A \in \mathcal{M}_n(\mathbb{K})$ est une matrice à diagonale D strictement dominante, alors $\rho(P_J(A)) < 1$, donc la méthode de Jacobi converge pour tout choix de x_0.

Démonstration. — Puisque $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{K})$ est une matrice à diagonales trictement dominante, alors $D = \mathrm{Diag}(a_{1,1}, \cdots, a_{n,n})$ est inversible et $D^{-1} = \mathrm{Diag}(a_{1,1}^{-1}, \cdots, a_{n,n}^{-1})$.

On a $P_J = D^{-1}(D - A) = I - D^{-1}A = (b_{ij})$ avec $b_{ii} = 0$ et $b_{ij} = -\dfrac{a_{ij}}{a_{ii}}$ si $i \neq j$. Par suite, pour tout

(1) Carl G. J. Jacobi, 1804-1851, mathématicien allemand

$i \in \{1, \cdots, n\}$, $\sum_{j=1}^{n} |b_{ij}| = \frac{1}{|a_{ii}|} \sum_{j \neq i}^{n} |a_{ij}| < 1$ car $|a_{i,i}| > \sum_{j \neq i}^{n} |a_{ij}|$ par hypothèse. Ainsi $\|P_J\|_\infty < 1$, donc $\rho(P_J) < 1$ (cf. Proposition I.2.12). □

Algorithme de Jacobi. Pour $k \in \mathbb{N}$ on pose $x_k = {}^t(x_{1,k}, \cdots, x_{n,k}) \in \mathbb{K}^n$.

- Passage $\boxed{x_k \to x_{k+1}}$. La i-ème coordonnée $x_{i,k+1,i}$ de x_{k+1} est donnée par: $\boxed{x_{i,k+1} = \frac{b_i - \sum_{j \neq i} a_{ij} x_{j,k}}{a_{ii}}}$. Ce calcul nécessite $(n-1)$ multiplications, $(n-1)$ additions, une division, soit $(2n-1)$ opérations élémentaires. Le passage $x_k \to x_{k+1}$ nécessite donc $n(2n-1)$ opérations élémentaires.

- test d'arrêt $\boxed{\frac{\|r_k\|}{\|b\|} < \varepsilon}$ basé sur le calcul du résidu normalisé, où $r_k = {}^t(r_{1,k}, \cdots, r_{n,k}) = Ax_k - b$. Dans le cas de Jacobi, on a $Dx_{k+1} = (D-A)x_k + b$, donc $r_k = Ax_k - b = D(x_k - x_{k+1})$. Ainsi: $\boxed{r_{i,k} = a_{ii}(x_{i,k} - x_{i,k+1})}$. Le calcul de r_k et de sa norme nécessitent $O(n)$ opérations élémentaires.

Complexité. Le calcul précédent montre que $N_{op}(n) \asymp 2(\text{nombre d'itération})n^2$ opérations, ce qui est très compétitif par rapport aux méthodes directes en $O(n^3)$ si la convergence est rapide (donc si $\rho(P_j) \ll 1$).

Algorithme de Jacobi
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(K)$ inversible à diagonale $\text{Diag}(a_{1,1}, \cdots, a_{n,n})$ inversible, qui satisfait les conditions de la proposition III.1.2 cet algorithme trouve la solution $x^\star$ de $Ax = b$ avec estimation de l'erreur basée sur le calcul du résidu normalisé.
Entrée : la matrice A, le vecteur b, le seuil de tolérance $\varepsilon > 0$.
Sortie : l'estimation numérique de $x^\star$.
Etape 1 [initialisation] donnée d'un x initial et d'un $s \geq \varepsilon$, calcul de norme(b)
Etape 2 [boucle] tant que $s \geq \varepsilon$ faire
pour i de 1 à n, faire
$y_i = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}}$,
$r_i = a_{ii}(x_i - y_i)$,
fin pour
$s = \frac{\text{norme}(r)}{\text{norme}(b)}$
$x \leftarrow y$
fin tant
Etape 3 [fin] sortir x et fin.

III.3. Méthode de Gauss-Seidel

Deuxième exemple de méthode itérative, celle de Gauss-Seidel[(2)].

[(2)] Philipp Ludwig von Seidel, 1821-1896, mathématicien allemand.

Définition III.3.1 (Méthode de Gauss-Seidel)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$ une matrice inversible dont diagonale D est inversible. On note $A = D + L + U$ où L (resp. U) est la partie triangulaire inférieure stricte (resp. partie triangulaire supérieure stricte) de $A = \begin{pmatrix} \ddots & & U \\ & D & \\ L & & \ddots \end{pmatrix}$. La **méthode de Gauss-Seidel** consiste à choisir $M = D + L$ et $N = -U$. La matrice d'itération $M^{-1}N$ est notée $P_{GS} = P_{GS}(A)$.

Remarque III.3.2.

On a l'équivalence: D inversible $\Leftrightarrow$ M inversible. Donc la méthode de Gauss-Seidel est bien définie.

Proposition III.3.3 (Condition suffisante de convergence pour Gauss-Seidel)**.**

Soit $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{K})$ une matrice à diagonale D strictement dominante. Alors $\rho\Big(P_{GS}(A)\Big) < 1$, donc la méthode de Gauss-Seidel converge pour tout choix de x_0.

Démonstration. — Afin d'obtenir une estimation de $\|M^{-1}N\|_\infty$, on s'intéresse à la solution de $My = Nx$ pour un $x \neq 0$ donné : on a

$$a_{i,i}y_i + \sum_{j<i} a_{i,j}y_j = -\sum_{j>i} a_{i,j}x_j \Leftrightarrow a_{i,i}y_i = -\sum_{j>i} a_{i,j}x_j - \sum_{j<i} a_{i,j}y_j.$$

Soit $\ell \in [1,n]$ tel que $|y_\ell| = \|y\|_\infty = \sup_{i\in[1,n]} |y_i|$. On a d'après ce qui précéde,

$$|a_{\ell,\ell}|.|y_\ell| \leq \sum_{j>\ell} |a_{\ell,j}|.\|x\|_\infty + \sum_{j<\ell} |a_{\ell,j}|.\|y\|_\infty$$

de sorte que $\left(|a_{\ell,\ell}| - \sum_{j<\ell} |a_{\ell,j}|\right)\|y\|_\infty \leq \left(\sum_{j>\ell} |a_{\ell,j}|\right)\|x\|_\infty$. Puisque A est à diagonale strictement dominante on a: $\sum_{j>\ell} |a_{\ell,j}| < |a_{\ell,\ell}| - \sum_{j<\ell} |a_{\ell,j}|$. Par suite:

- soit $\sum_{j>\ell} |a_{\ell,j}| = 0$ auquel cas $y = 0$ et $0 = \|y\|_\infty < \|x\|_\infty$ (car $x \neq 0$);
- soit $\sum_{j>\ell} |a_{\ell,j}| \neq 0$ et alors $\|y\|_\infty < \|x\|_\infty$.

En conclusion, pour tout $x \neq 0$, $\|y = M^{-1}Nx\|_\infty < \|x\|_\infty$, donc $\|M^{-1}N\|_\infty < 1$ ce qui implique $\rho\Big(P_{GS}(A)\Big) < 1$ (cf. Proposition I.2.12). □

Proposition III.3.4 (Condition suffisante de convergence pour Gauss-Seidel)**.**

Soit $A = (a_{ij}) \in \mathcal{M}_n(\mathbb{K})$. Si A est hermitienne (ou symétrique si $\mathbb{K} = \mathbb{R}$) définie positive, alors la méthode de Gauss-Seidel est bien définie et $\rho\Big(P_{GS}(A)\Big) < 1$, donc la méthode de Gauss-Seidel converge pour tout choix de x_0.

Démonstration. — On fait la preuve dans le cas $A = (a_{ij}) \in \mathcal{M}_n(\mathbb{R})$ symétrique définie positive: ${}^tA = A$ et pour tout $x \in \mathbb{R}^n \setminus \{0\}$, ${}^txAx > 0$. Dans ce cas $a_{i,i} = {}^te_iAe_i > 0$ pour tout $i \in [1, n]$ (les e_i sont les vecteurs de la base canonique de $\mathbb{R}^n$), $D = \mathrm{Diag}(a_{1,1}, \cdots, a_{n,n})$ est donc définie positive et $M = D + L$ est inversible. La méthode de Gauss-Seidel est donc bien définie. Par ailleurs ${}^tM + N = D + {}^tL - U = D$ car ${}^tL = U$.
Comme A est symétrique définie positive, on peut considérer la norme vectorielle $\|x\|_A$ définie par: pour tout $x \in \mathbb{R}^n$, $\|x\|_A^2 = {}^txAx$. On note encore par $\|.\|_A$ la norme matricielle subordonnée correspondante. Pour tout $x \in \mathbb{R}^n \setminus \{0\}$:

$$\begin{aligned}\|M^{-1}Nx\|_A^2 &= {}^t(M^{-1}Nx)AM^{-1}Nx = {}^tx\,{}^t(M-A)\,({}^tM)^{-1}AM^{-1}(M-A)x\\ &= \Big({}^tx - {}^tx\,{}^tA({}^tM)^{-1}\Big)A(x - M^{-1}Ax)\\ &= \|x\|_A^2 - {}^txAM^{-1}Ax - {}^tx\,{}^tA({}^tM)^{-1}Ax + {}^tx\,{}^tA({}^tM)^{-1}AM^{-1}Ax\end{aligned}$$

Il vient:

$$\begin{aligned}\|M^{-1}Nx\|_A^2 &= \|x\|_A^2 + {}^tx\,{}^tA({}^tM)^{-1}\Big[A - M - {}^tM\Big]M^{-1}Ax\\ &= \|x\|_A^2 - {}^tx\,{}^tA({}^tM)^{-1}\Big[{}^tM + N\Big]M^{-1}Ax = \|x\|_A^2 - \|M^{-1}Ax\|_D^2\end{aligned}$$

On en déduit que $\|M^{-1}Nx\|_A < \|x\|_A$ si $x \neq 0$, par suite $\|M^{-1}N\|_A = \sup_{\|x\|_A=1} \|M^{-1}Nx\|_A < 1$, donc $\rho\Big(M^{-1}N\Big) \leq \|M^{-1}N\|_A < 1$. (Proposition I.2.12). □

Algorithme de Gauss-Seidel: Pour $k \in \mathbb{N}$ on pose $x_k = {}^t(x_{1,k}, \cdots, x_{n,k}) \in \mathbb{K}^n$.

- Passage $\boxed{x_k \to x_{k+1}}$. L'algorithme s'écrit $(D + L)x_{k+1} = b - Ux_k$, soit aussi $Dx_{k+1} = b - Lx_{k+1} - Ux_k$ de sorte que la i-ème coordonnée $x_{i,k+1}$ de x_{k+1} est donnée par : $\boxed{x_{i,k+1} = \dfrac{b_i - \sum_{j<i} a_{i,j}x_{j,k+1} - \sum_{j>i} a_{i,j}x_{j,k}}{a_{ii}}}$. Le passage $x_k \to x_{k+1}$ nécessite $n(2n-1)$ opérations élémentaires.
  Dans la méthode de Gauss-Seidel, dès que la i-ème coordonnée de x_{k+1} est calculée, la i-ème coordonnée de x_k devient inutile pour la suite : on peut donc l'écraser par la i-ème coordonnée de x_{k+1} dès que celle-ci est calculée, ce qui est intéressant en gain de place mémoire.

- test d'arrêt $\boxed{\dfrac{\|r_k\|}{\|b\|} < \varepsilon}$ basé sur le calcul du résidu normalisé $r_k = Ax_k - b$. Ce calcul est coûteux, en $O(n^2)$ opérations élémentaires (comparer à Jacobi).

Algorithme de Gauss-Seidel
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(K)$ inversible à diagonale $\mathrm{Diag}(a_{1,1}, \cdots, a_{n,n})$ inversible, qui satisfait les conditions de la proposition III.1.2, cet algorithme trouve la solution $x^\star$ de $Ax = b$ avec estimation de l'erreur basée sur le calcul du résidu normalisé.
Entrée : la matrice A, le vecteur b, le seuil de tolérance $\varepsilon > 0$.
Sortie : l'estimation numérique de $x^\star$.
Etape 1 [initialisation] donnée d'un x initial et d'un $s \geq \varepsilon$, calcul de norme(b)
Etape 2 [boucle] tant que $s \geq \varepsilon$ faire
pour i de 1 à n, faire
$$x_i \leftarrow \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j - \sum_{j=i+1}^{n} a_{ij}x_j}{a_{ii}},$$
fin pour
$r = Ax - b$
$s = \dfrac{\text{norme}(r)}{\text{norme}(b)}$
fin tant
Etape 3 [fin] sortir x et fin.

Remarque III.3.5.
Quoi que très faciles à programmer, les méthodes de Jacobi et de Gauss-Seidel peuvent être très lentes à converger pour certains systèmes. Afin d'accélerer cette convergence, des variantes sont utilisées comme les méthodes de relaxation que nous aborderons en exercice.

On peut comparer la vitesse de convergence des méthodes de Jacobi et de Gauss-Seidel, au moins dans le cadre (courant en pratique) des matrices tridiagonales:

Proposition III.3.6.

Si $A \in \mathcal{M}_n(\mathbb{K})$ est tridiagonale, on a $\rho\left(P_{GS}(A)\right) = \rho\left(P_J(A)\right)^2$.

Démonstration. — En TD. □

Ainsi, sous les conditions de la proposition III.1.2 et pour une matrice A tridiagonale, la méthode de Gauss-Seidel convergera 2 fois plus vite que la méthode de Jacobi.

CHAPITRE IV

RECHERCHE DE VALEURS PROPRES

IV.1. Introduction

Avec la résolution des systèmes linéaires, la recherche des valeurs propres (et de vecteurs propres associés) d'une matrice est l'autre problème majeur de l'algèbre linéaire, avec un champ d'applications important en physique (théorie du signal, mécanique quantique, etc..), en statistique (ACP, ACM ..), etc..

On sait que les valeurs propres d'une matrice $A \in \mathcal{M}_n(\mathbb{K})$ sont les racines du polynôme caractéristique $\det(\lambda I - A) = \lambda^n - \text{Trace}(A)\lambda^{n-1} + \cdots + (-1)^n \det(A)$. Ce fait amène deux constats:

1. le calcul du polynôme caractéristique équivaut au calcul des $n-1$ invariants de similitudes (les coefficients de ce polynôme), essentiellement la somme de déterminants: on a vu aux chapitres précédents les liens entre ce type de calculs et ceux portant sur la résolution numérique directe ou itérative de systèmes linéaires;
2. comme (selon Abel[(1)] et Galois[(2)]) il ne peut pas y avoir en général de formules algébriques donnant les racines d'un polynôme, il ne peut donc pas exister de méthode directe fournissant le spectre d'une matrice (cela voudrait dire qu'on a une méthode de calcul exacte en un nombre fini d'opérations élémentaires).

La recherche des valeurs propres relèvent donc nécessairement de méthodes itératives. Certaines de ces méthodes sont "partielles", c'est à dire ne fournissent qu'une partie du spectre (typiquement la valeur propre dominante); d'autres méthodes s'intéressent à la totalité du spectre.

IV.2. Quelques rappels et résultats sur la réduction de matrices

Nous débutons ce chapitre par quelques notions et résultats connus ou complémentaires relatifs à la réduction de matrice. Par la suite $\mathbb{K} = \mathbb{R}$ ou $\mathbb{C}$.

Définition IV.2.1.

Soit $A, B \in \mathcal{M}_n(\mathbb{K})$ deux matrices.

- Les matrices A et B sont équivalentes s'il existe deux matrices inversibles $P, Q \in GL_n(\mathbb{K})$ telle que $B = QAP$.

(1) Niels Henrik Abel, 1802-1829, mathématicien norvégien.
(2) Évariste Galois, 1811-1832, mathématicien français.

– Les matrices A et B sont semblables s'il existe une matrice inversible $P \in GL_n(\mathbb{K})$ telle que $B = P^{-1}AP$.

Théorème IV.2.2 (Réduction de matrices)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$.

1. *$\lambda \in \mathbb{K}$ est valeur propre associée au vecteur propre $u \in \mathbb{K}^n$, $Au = \lambda u \Leftrightarrow \lambda$ est racine du polynôme caractéristique* $\det(\lambda I - A)$.
2. *A admet n valeurs propres simples $(\lambda_1, \cdots, \lambda_n) \Rightarrow A$ se diagonalise dans une base de vecteurs propres $(u_1, \cdots, u_n)$: $D = P^{-1}AP$ avec $D = \mathrm{Diag}(\lambda_1, \cdots, \lambda_n)$ et $P = (u_1, \ldots, u_n)$.*
3. *Le polynôme caractéristique de A est scindé dans $\mathbb{K}$, $\det(\lambda I - A) = (\lambda_1 - \lambda)^{n_1} \cdots (\lambda_p - \lambda)^{n_p} \Rightarrow \mathbb{K}^n$ est somme directe des espaces caractéristiques $\mathbb{K}^n = \mathrm{Ker}(A - \lambda_1 I)^{n_1} \oplus \cdots \oplus \mathrm{Ker}(A - \lambda_p I)^{n_p} \Rightarrow A$ est semblable à une matrice triangulaire supérieure $T = D + N$ dont la partie diagonale D est formée des valeurs propres de A, chacune répétée selon sa multiplicité. De plus $DN = ND$.*

Exemple IV.2.3.

On considère la matrice $A = \begin{pmatrix} 1 & 0 & 1 \\ -1 & 2 & 1 \\ 1 & -1 & 1 \end{pmatrix}$. Le polynôme caractéristique est scindé, de la forme $\det(\lambda I - A) = (\lambda - 1)^2(\lambda - 2)$, de sorte que $\lambda = 2$ est une valeur propre simple, $\lambda = 1$ une valeur propre double. La matrice A est donc semblable à une matrice triangulaire supérieure T: il existe P inversible telle que $P^{-1}AP = T = \begin{pmatrix} 2 & * & * \\ 0 & 1 & * \\ 0 & 0 & 1 \end{pmatrix}$. Explicitons cela. L'espace propre $\mathrm{Ker}(A - 2I)$ est de dimension 1 (c'est normal puisque 2 est valeur propre simple) et $\mathrm{Ker}(A - 2I) = \mathrm{Vect}(u_1)$ avec $u_1 = {}^t(1, 0, 1)$. L'espace propre $\mathrm{Ker}(A - I)$ est aussi de dimension 1, explicitement $\mathrm{Ker}(A - I) = \mathrm{Vect}(u_2)$ avec $u_2 = {}^t(1, 1, 0)$. Comme 1 est une valeur propre double, la matrice A n'est pas diagonalisable mais uniquement triangularisable. Enfin l'espace caractéristique $\mathrm{Ker}(A - I)^2$ est de la forme $\mathrm{Ker}(A - I)^2 = \mathrm{Vect}(u_2, u_3)$ avec $u_3 = {}^t(1, 1, 1)$ et $(A - I)u_3 = u_2$. Par suite, si $P = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} = (u_1, u_2, u_3)$ est la matrice de passage de la base canonique à la base (u_1, u_2, u_3) de $\mathbb{R}^3$, on a $P^{-1}AP = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} = T$ (Jordanisation).

Nous présentons la décomposition de Schur[(3)] dont l'utilisation est plus théorique que pratique. Nous verrons plus loin (théorème IV.6.1) le lien entre cette décomposition et l'une des méthodes de recherche numérique du spectre d'une matrice.

[(3)] Issai Schur, 1875-1941, mathématicien russe

Théorème IV.2.4 (Décomposition de Schur)**.**

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

1. *Il existe une matrice unitaire $Q \in \mathcal{M}_n(\mathbb{C})$ et une matrice triangulaire supérieure $T \in \mathcal{M}_n(\mathbb{C})$ telles que $Q^\star AQ = T$. Les élements diagonaux de T sont toutes les valeurs propres de A, chacune répétée selon sa multiplicité.*
2. *Si $A \in \mathcal{M}_n(\mathbb{R})$ et si son polynôme caractéristique est scindé dans $\mathbb{R}$, alors on peut prendre $T \in \mathcal{M}_n(\mathbb{R})$ et $Q \in \mathcal{M}_n(\mathbb{R})$ orthogonale.*

Remarque IV.2.5.

1. On rappelle que $Q^\star = {}^t\overline{Q}$ et que $Q \in \mathcal{M}_n(\mathbb{C})$ unitaire veut dire $Q^\star Q = I$.
2. On rappelle que $Q \in \mathcal{M}_n(\mathbb{R})$ orthogonale veut dire ${}^tQQ = I$.

Démonstration. — On fait une preuve par récurrence sur n. Si $n = 1$ le résultat est vrai. Supposons le résultat vrai pour $n-1$. Soit $A \in \mathcal{M}_n(\mathbb{C})$. Considérons l'une (quelconque) de ses valeurs propres λ et soit $u_1 \in \mathbb{C}^n$ un vecteur propre associé tel que $\|u_1\|_2 = 1$. On complète ce vecteur en une base orthonormée $(u_1, u_2, \cdots, u_n)$ de $\mathbb{C}^n$. On pose P la matrice de passage de la base canonique $(e_1, e_2, \cdots, e_n)$ de $\mathbb{C}^n$ à la base $(u_1, u_2, \cdots, u_2)$, $P = \begin{pmatrix} u_{1,1} & \cdots & u_{n,1} \\ \vdots & & \vdots \\ u_{1,n} & \cdots & u_{n,n} \end{pmatrix}$. La matrice P est unitaire (i.e. $P^{-1} = P^\star$ et on rappelle que $P^\star = {}^t\overline{P}$ est la matrice adjointe). Posons $B = P^\star AP$.
On a $Be_1 = P^\star APe_1 = P^\star Au_1 = P^\star(\lambda u_1) = \lambda e_1$ de sorte que

$$B = \begin{pmatrix} \lambda & b \\ 0 & B_{n-1} \end{pmatrix}, \qquad B_{n-1} \in \mathcal{M}_{n-1}(\mathbb{C}), \qquad b = (b_1, \cdots, b_{n-1}) \in \mathbb{C}^{n-1}$$

On applique à B_{n-1} l'hypothèse de récurrence : il existe $Q_{n-1} \in \mathcal{M}_{n-1}(\mathbb{C})$ unitaire et $T_{n-1} \in \mathcal{M}_n(\mathbb{C})$ triangulaire supérieure telles que $Q_{n-1}^\star B_{n-1} Q_{n-1} = T_{n-1}$. On en déduit:

$$\begin{pmatrix} 1 & 0 \\ 0 & Q_{n-1}^\star \end{pmatrix} \begin{pmatrix} \lambda & b \\ 0 & B_{n-1} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & Q_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & bQ_{n-1} \\ 0 & Q_{n-1}^\star B_{n-1} Q_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & bQ_{n-1} \\ 0 & T_{n-1} \end{pmatrix} = T$$

où $T \in \mathcal{M}_n(\mathbb{C})$ est triangulaire supérieure. On note que $Q_n = \begin{pmatrix} 1 & 0 \\ 0 & Q_{n-1} \end{pmatrix} \in \mathcal{M}_n(\mathbb{C})$ est unitaire et que $T = Q_n^\star B Q_n$, soit aussi $B = Q_n T Q_n^\star$. Par suite, $A = PQ_n T Q_n^\star P^\star = QTQ^\star$ avec $Q = PQ_n \in \mathcal{M}_n(\mathbb{C})$ unitaire. Enfin $\det(\lambda I - A) = \det(\lambda I - T)$, donc les éléments diagonaux de T sont les valeurs propres de A. □

Exemple IV.2.6.

On considère de nouveau la matrice $A = \begin{pmatrix} 1 & 0 & 1 \\ -1 & 2 & 1 \\ 1 & -1 & 1 \end{pmatrix}$ de l'exemple IV.2.3. On y a vu que A est semblable à la matrice triangulaire $T_1 = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$, précisément $P^{-1}AP = T_1$ avec

$P = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} = (u_1, u_2, u_3)$. Cependant la matrice P n'est pas orthogonale. On va donc transformer la base (u_1, u_2, u_3) en une base orthogonale (v_1, v_2, v_3), resp. orthonormée (w_1, w_2, w_3), par le procédé de Gram-Schmidt (on note $\langle ., . \rangle$ le produit scalaire euclidien):

1. $v_1 = u_1 = {}^t(1, 0, 1)$, $w_1 = \dfrac{v_1}{\|v_1\|_2} = \dfrac{1}{\sqrt{2}}{}^t(1, 0, 1)$ soit aussi $w_1 = \dfrac{1}{\sqrt{2}}u_1$;
2. $v_2 = u_2 - \langle u_2, w_1\rangle w_1 = {}^t\left(\dfrac{1}{2}, 1, -\dfrac{1}{2}\right)$, $w_2 = \dfrac{v_2}{\|v_2\|_2} = \dfrac{1}{\sqrt{6}}{}^t(1, 2, -1)$ soit aussi $w_2 = \dfrac{2}{\sqrt{6}}u_2 - \dfrac{1}{\sqrt{6}}u_1$;
3. $v_3 = u_3 - \langle u_3, w_1\rangle w_1 - \langle u_3, w_2\rangle w_2 = \dfrac{1}{3}{}^t(-1, 1, 1)$, d'où
   $w_3 = \dfrac{v_3}{\|v_3\|_2} = \dfrac{1}{\sqrt{3}}{}^t(-1, 1, 1)$ soit aussi $w_3 = \dfrac{3}{\sqrt{3}}u_3 - \dfrac{1}{\sqrt{3}}u_1 - \dfrac{2}{\sqrt{3}}u_2$.

Appelons $Q = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{3}} \\ 0 & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \end{pmatrix}$ la matrice de passage de la base canonique à la base (w_1, w_2, w_3): c'est une matrice orthogonale (par construction) qui, par le procédé de Gram-Schmidt, se déduit de la matrice P par multiplication à droite par une matrice triangulaire supérieure T_2, la matrice de passage de la base (u_1, u_2, u_3) à la base (w_1, w_2, w_3). Explicitement par nos calculs précédents:

$$Q = PT_2 \quad \text{avec} \quad T_2 = \begin{pmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{3}} \\ 0 & \frac{2}{\sqrt{6}} & -\frac{2}{\sqrt{3}} \\ 0 & 0 & \frac{3}{\sqrt{3}} \end{pmatrix}$$

Finalement $T = Q^{-1}AQ = T_2^{-1}P^{-1}APT_2 = T_2^{-1}T_1T_2$ est une matrice triangulaire supérieure, comme produit de telles matrices. Explicitement sur notre exemple: $T = \begin{pmatrix} 2 & -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} \\ 0 & 1 & \frac{3}{\sqrt{2}} \\ 0 & 0 & 1 \end{pmatrix}$.

Remarque: ce procédé fournit une autre preuve du théorème IV.2.4.

On complète cette section par la décomposition SVD.

Théorème IV.2.7 (Décomposition en valeurs singulières d'une matrice)**.**

Soit $n, m \in \mathbb{N}^\star$ et $A \in \mathcal{M}_{n,m}(\mathbb{C})$ une matrice rectangulaire. On note $p = \min\{n, m\}$.

1. *Il existe deux matrices unitaires $Q \in \mathcal{M}_n(\mathbb{C})$ et $P \in \mathcal{M}_m(\mathbb{C})$ telles que:*
   (a) $Q^\star AP = \begin{pmatrix} D \\ 0_{n-m,m} \end{pmatrix}$ *si* $n \geq m$,
   (b) $Q^\star AP = \begin{pmatrix} D & 0_{n,m-n} \end{pmatrix}$ *si* $m \geq n$,
   où $D = \mathrm{Diag}(\sigma_1, \cdots, \sigma_p)$ est une matrice carrée $p \times p$ diagonale à coefficients $\sigma_i \geq 0$ réels positifs ou nuls. De plus les p coefficients diagonaux σ_i d'une telle décomposition sont uniques, à une permutation près.
2. *Si $A \in \mathcal{M}_{n,m}(\mathbb{R})$, les matrices $Q \in \mathcal{M}_n(\mathbb{R})$ et $P \in \mathcal{M}_m(\mathbb{R})$ peuvent être choisies orthogonales.*

Démonstration. — On va supposer $n \geq m$ pour se fixer les idées.
On note que la matrice $A^\star A \in \mathcal{M}_m(\mathbb{C})$ est hermitienne, donc ses valeurs propres sont réelles; de plus elles sont positives ou nulles : si λ est une valeur propre de $A^\star A$ associée au vecteur propre u, alors $u^\star A^\star A u = \lambda u^\star u$ où $u^\star = {}^t\overline{u}$, donc $\lambda = \dfrac{\|Au\|_2^2}{\|u\|_2^2} \geq 0$. De plus, $A^\star A$ se diagonalise dans une base orthonormée de vecteurs propres : il existe $P \in \mathcal{M}_m(\mathbb{C})$ unitaire telle que $P^\star A^\star A P = D^2$ où $D \in \mathcal{M}_m(\mathbb{R})$ est une matrice diagonale $D = \mathrm{Diag}(\sigma_1, \cdots, \sigma_m)$ avec $\sigma_i \geq 0$, les $\sigma_1^2, \cdots, \sigma_m^2$ étant les valeurs propres de $A^\star A$ comptées avec leur multiplicité.
Faisons une remarque : notons par $u_i \in \mathbb{C}^n$ les m vecteurs colonnes de la matrice $AP \in \mathcal{M}_{n,m}(\mathbb{C})$. L'égalité $(AP)^\star AP = D^2$ se traduit par : pour $i, k = 1, .., m$,

1. $(u_i)^\star u_i = \sigma_i^2$,
2. $(u_i)^\star u_k = 0$ si $i \neq k$.

Autrement dit, les vecteurs u_i sont soit nuls (si $\sigma_i = 0$), soit sont orthogonaux deux-à-deux.
On suppose qu'on a choisi P de sorte que les valeurs propres nulles de $A^\star A$ soient rangées en premier dans D^2 : il existe un entier $0 \leq r \leq m$ tel que $\sigma_i = 0$ si $i \leq r$ et $\sigma_i > 0$ pour $i > r$.
Pour $i > r$, on note $w_i = \dfrac{1}{\sigma_i} u_i \in \mathbb{C}^n$ la i-ieme colonne de $AP \in \mathcal{M}_{n,m}(\mathbb{C})$ divisée par σ_i. Par la remarque précédente, on sait que la famille $(w_{r+1}, \cdots, w_m)$ forme une famille orthonormée de $\mathbb{C}^n$. On la complète en une base orthonomée de $\mathbb{C}^n$: $(w_1 \cdots, w_r, w_{r+1}, \cdots, w_m, w_{m+1}, \cdots, w_n)$. On pose alors $Q \in \mathcal{M}_n(\mathbb{C})$ la matrice unitaire formée par ces n vecteurs w_k pour obtenir le résultat voulu: $Q^\star AP = \begin{pmatrix} D \\ 0_{n-m,m} \end{pmatrix}$.
Il nous reste à montrer que dans une telle décomposition, les éléments diagonaux σ_i sont uniques. Supposons donc qu'on ait $Q^\star AP = \begin{pmatrix} D \\ 0_{n-m} \end{pmatrix}$. Alors $P^\star A^\star Q Q^\star AP = P^\star A^\star AP$ est une matrice symétrique dont les valeurs propres sont réelles positives et uniquement déterminées. Or $P^\star A^\star Q Q^\star AP = \begin{pmatrix} D & 0_{m,n-m} \end{pmatrix} \begin{pmatrix} D \\ 0_{n-m,m} \end{pmatrix} = D^2$.

Cas $m \geq n$: on pose $B = A^\star$. Par le résultat précédent, il existe 2 matrices unitaires $Q \in \mathcal{M}_m(\mathbb{C})$ et $P \in \mathcal{M}_n(\mathbb{C})$ telles que $Q^\star BP = \begin{pmatrix} D \\ 0_{m-n,n} \end{pmatrix}$. Par suite $P^\star AQ = \begin{pmatrix} D & 0_{n,m-n} \end{pmatrix}$ □

Exemple IV.2.8.

On pose $A = \begin{pmatrix} 1 & -1 \\ -1 & 2 \\ 2 & 1 \end{pmatrix} \in \mathcal{M}_{3,2}(\mathbb{R})$. La matrice symétrique ${}^tAA = \begin{pmatrix} 6 & -1 \\ -1 & 6 \end{pmatrix}$ se diagonalise sous la forme ${}^tP\,{}^tA\,AP = \begin{pmatrix} 5 & 0 \\ 0 & 7 \end{pmatrix}$ où $P = \begin{pmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{pmatrix}$ est la matrice orthogonale dont les vecteurs colonnes donnent une base orthonormée de de $\mathbb{R}^2$ formée des vecteurs propres de ${}^tAA$ associés aux valeurs propres respectives 5 et 7. La matrice AP est de la forme $AP = \begin{pmatrix} 0 & \sqrt{2} \\ \frac{\sqrt{2}}{2} & -\frac{3\sqrt{2}}{2} \\ \frac{3\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{pmatrix} = (u_1, u_2)$ où (u_1, u_2)

forme une famille libre orthogonale par construction. On les normalise en posant: $w_1 = \begin{pmatrix} 0 \\ \frac{\sqrt{2}}{2\sqrt{5}} \\ \frac{3\sqrt{2}}{2\sqrt{5}} \end{pmatrix}$ et $w_2 = \begin{pmatrix} \frac{\sqrt{2}}{\sqrt{7}} \\ -\frac{3\sqrt{2}}{2\sqrt{7}} \\ \frac{\sqrt{2}}{2\sqrt{7}} \end{pmatrix}$ de sorte que (w_1, w_2) forme une famille orthonormale de $\mathbb{R}^3$. On la complète en une base (w_1, w_2, w_3) orthornomée de $\mathbb{R}^3$ avec (par exemple) $w_3 = w_1 \wedge w_2 = \begin{pmatrix} \frac{\sqrt{5}}{\sqrt{7}} \\ \frac{3}{\sqrt{5}\sqrt{7}} \\ -\frac{1}{\sqrt{5}\sqrt{7}} \end{pmatrix}$ ($\wedge$ est le produit vectoriel), ce qui fournit la matrice orthogonale $Q = (w_1, w_2, w_3) = \begin{pmatrix} 0 & \frac{\sqrt{2}}{\sqrt{7}} & \frac{\sqrt{5}}{\sqrt{7}} \\ \frac{\sqrt{2}}{2\sqrt{5}} & -\frac{3\sqrt{2}}{2\sqrt{7}} & \frac{3}{\sqrt{5}\sqrt{7}} \\ \frac{3\sqrt{2}}{2\sqrt{5}} & \frac{\sqrt{2}}{2\sqrt{7}} & -\frac{1}{\sqrt{5}\sqrt{7}} \end{pmatrix}$ et on a ${}^tQAP = \begin{pmatrix} \sqrt{5} & 0 \\ 0 & \sqrt{7} \\ 0 & 0 \end{pmatrix}$.

On remarquera qu'on aurait tout aussi bien pu prendre la matrice $Q = (w_1, w_2, -w_3)$ dans notre exemple : dans le théorème IV.2.7, les matrices Q et P ne sont pas uniques.

Définition IV.2.9 (SVD).

Une décomposition d'une matrice $A \in \mathcal{M}_{n,m}(\mathbb{K})$ donnée sous la forme $A = QSP^\star$ avec Q, P unitaires ($\mathbb{K} = \mathbb{C}$) ou orthogonales ($\mathbb{K} = \mathbb{R}$) et $S \in \mathcal{M}_{n,m}(\mathbb{R})$ de la forme $S = \begin{pmatrix} D \\ 0_{n-m} \end{pmatrix}$ ou $S = \begin{pmatrix} D & 0_{n-m} \end{pmatrix}$ avec $D = \mathrm{Diag}(\sigma_1, \cdots, \sigma_p)$, $p = \min\{n, m\}$, diagonale à coefficients réels positifs ou nuls, est appelée **décomposition en valeurs singulières** de A, où **décomposition SVD** (singular values decomposition). Les $\sigma_i \geq 0$ sont les **valeurs singulières** de A: ce sont les racines carrées des valeurs propres de $A^\star A$ ($\mathbb{K} = \mathbb{C}$) ou de ${}^tAA$ ($\mathbb{K} = \mathbb{R}$).

L'appellation "SVD" apparaît dans un article de A.Horn[4] en 1951. Le théorème IV.2.7, aussi appelé "théorème de Eckart-Young) a été démontré par C. Eckart[5] et G. Young[6] en 1939. La décomposition SVD est l'un des outils essentiels en analyse des données et en apprentissage (machine learning), comme les étudiants du parcours DS pourront le constater[7].

Nous verrons un peu plus loin une méthode particulièrement efficace de recherche numérique de l'ensemble des valeurs singulières d'une matrice: la méthode de Jacobi, cf. théorème IV.5.8.

[4] Alfred Horn, 1918-2001, mathématicien américain
[5] Carl H. Echart, 1902-1971, physicien américain
[6] Gale Young, physicien américain
[7] Cf. par curiosité l'article https://arxiv.org/abs/1510.08532

IV.3. Recherche de valeurs propres et sensibilités numériques

Considérons la matrice $A = \begin{pmatrix} 0 & 0 & \cdots & \varepsilon \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix} \in \mathcal{M}_n(\mathbb{R})$ dont le polynôme caractéristique est $(\lambda^n - \varepsilon)$. Les valeurs propres λ_i sont nulles si $\varepsilon = 0$, égales aux racines n-ième de ε sinon. Si (par exemple) $n = 10$ et $\varepsilon = 10^{-10}$, alors $|\lambda_i| = 0.1$ de sorte qu'une petite variation sur A peut entraîner une variation très sensible sur les valeurs propres. On précise cela dans le théorème de Bauer[8]-Fike suivant:

Théorème IV.3.1.

Soit $A \in \mathcal{M}_n(\mathbb{C})$ diagonalisable et $P \in \mathcal{M}_n(\mathbb{C})$ inversible telle que $P^{-1}AP = \mathrm{Diag}(\lambda_1, \cdots, \lambda_n)$. Soit $\|\,.\,\|$ une norme sur $\mathcal{M}_n(\mathbb{C})$ induite par une norme vectorielle. Alors pour toute perturbation $\Delta A \in \mathcal{M}_n(\mathbb{C})$ de A, on a $\mathrm{Sp}(A + \Delta A) \subset \bigcup_{i=1}^{n} D_i$ avec $D_i = \{\mu \in \mathbb{C} \mid |\mu - \lambda_i| \leq \mathrm{cond}(P) \times \|\Delta A\|\}$.

*Autrement dit, **la sensibilité du problème aux valeurs propres dépend du conditionnement de la matrice de passage** P et non du conditionnement de la matrice A.*

Démonstration. — Le nombre $\mu \in \mathbb{C}$ est valeur propre de $A + \Delta A$ ssi $A + \Delta A - \mu A$ est une matrice singulière (=non inversible), c'est à dire que $\mathrm{Diag}(\lambda_1 - \mu, \cdots, \lambda_n - \mu) + P^{-1}\Delta AP$ est singulière:

1. soit $\mathrm{Diag}(\lambda_1 - \mu, \cdots, \lambda_n - \mu)$ est singulière, c'est à dire $\mu = \lambda_i$ pour un certain $j = 1, \cdots, n$. En particulier $\mu \in D_j \subset \bigcup_{i=1}^{n} D_i$;
2. sinon $\mathrm{Diag}(\lambda_1 - \mu, \cdots, \lambda_n - \mu)$ est inversible et $I + \mathrm{Diag}\,(\frac{1}{\lambda_1 - \mu}, \cdots, \frac{1}{\lambda_n - \mu})P^{-1}(\Delta A)P$ est singulière. Ceci implique[9] que $1 \leq \|\,\mathrm{Diag}(\frac{1}{\lambda_1 - \mu}, \cdots, \frac{1}{\lambda_n - \mu})P^{-1}(\Delta A)P\|$ puis que

$$1 \leq \|\,\mathrm{Diag}\,(\frac{1}{\lambda_1 - \mu}, \cdots, \frac{1}{\lambda_n - \mu})\|.\|P^{-1}\|.\|\Delta A\|.\|P\|$$

et enfin $1 \leq \max_i\{\frac{1}{|\lambda_i - \mu|}\}\,\mathrm{cond}(P).\|\Delta A\|$ car $\|\,.\,\|$ est une norme subordonnée à une norme vectorielle. Autrement dit $\min_i\{|\lambda_i - \mu|\} \leq \mathrm{cond}(P) \times \|\Delta A\|$, ce qui achève la preuve.

□

Remarque IV.3.2.

On rappelle que si P est une matrice carrée unitaire ou orthogonale ($P^\star P = I$), alors $\mathrm{cond}_2(P) = \|P\|_2\|P^{-1}\|_2 = \sqrt{\rho(P^\star P)^2} = 1$.

Si A est une matrice carrée symétrique ou hermitienne, alors A se diagonalise à l'aide d'un changement de base orthogonal ou unitaire (c'est même vrai si A est une matrice normale, c'est à dire vérifiant $A^\star A = AA^\star$): il existe P orthogonale ou unitaire telle que $P^\star AP = \mathrm{Diag}(\lambda_1, \cdots, \lambda_n)$

[8] Friedrich L. Bauer, 1924-2015, informaticien et mathématicien allemand

[9] Si $B = -\mathrm{Diag}\,(\frac{1}{\lambda_1 - \mu}, \cdots, \frac{1}{\lambda_n - \mu})P^{-1}(\Delta A)P$ vérifie $\|B\| < 1$, alors la matrice $C = \sum_{k=0}^{\infty} B^n$ est bien définie (convergence normale) et $(I - B)C = C(I - B) = I$, donc $I - B$ est inversible.

et $\mathrm{cond}_2(P) = 1$ (cf. proposition I.2.2). Le théorème IV.3.1 nous dit que la recherche numérique des valeurs propres d'une telle matrice sera peu sensible aux perturbations numériques ΔA.

IV.4. Méthodes partielles de recherche de valeurs propres

Dans les méthodes présentées dans cette partie, on ne s'intéresse qu'à la valeur propre de plus grand ou plus petit module, ou encore à la valeur propre la plus proche d'une valeur donnée : on parle de méthodes partielles. Ces méthodes permettent de trouver le vecteur propre correspondant.

IV.4.1. Méthode de la puissance. —

IV.4.1.1. Méthode de la puissance. — La méthode de la puissance permet de déterminer la valeur propre λ_1 de module maximal, alors dite **dominante**, d'une matrice diagonalisable $A \in \mathcal{M}_n(\mathbb{K})$, ($\mathbb{K} = \mathbb{R}$ ou $\mathbb{C}$), sous l'hypothèse que A possède une seule valeur propre simple de module maximal.

Notons $\lambda_1, \cdots, \lambda_n$ les valeurs propres de A comptées avec mutiplicité. On suppose donc que $\boxed{|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|}$. A la valeur propre simple λ_1 on associe le vecteur propre unitaire $u_1 \in K^n$: $Au_1 = \lambda_1 u_1$ et $\|u_1\|_2^2 = \langle u_1, u_1 \rangle = 1$.

Remarque IV.4.1.

1. Rappel. On note $\langle u, v \rangle$ le produit scalaire canonique hermitien ($\mathbb{K} = \mathbb{C}$) ou euclidien ($\mathbb{K} = \mathbb{R}$) des vecteurs u et v, c'est à dire $\langle u, v \rangle = u^\star v$ avec $u^\star = {}^t\overline{u}$ ($u \in \mathbb{C}^n$) ou $u^\star = {}^t u$ ($u \in \mathbb{R}^n$).
2. On notera que si $Au_1 = \lambda_1 u_1$ et que $\|u_1\|_2 = 1$, alors $\langle u_1, Au_1 \rangle = \lambda_1 \|u_1\|_2^2 = \lambda_1$.
3. Si $A \in \mathcal{M}_n(\mathbb{R})$ on a $\lambda_1 \in \mathbb{R}$ sous nos hypothèses (même si A est diagonalisable sur $\mathbb{C}$). Exercice: pourquoi ?

Pour construire le couple (λ_1, u_1), on part d'un vecteur unitaire $v_0 \in \mathbb{K}^n$, $\|v_0\|_2 = 1$ quelconque et on construit la suite $(\mu_k, v_k)_{k \geq 0} \in \mathbb{K} \times \mathbb{K}^n$ par récurrence:

$$\text{(IV.1)} \qquad \begin{cases} \widetilde{v}_{k+1} &= Av_k \\ v_{k+1} &= \dfrac{\widetilde{v}_{k+1}}{\|\widetilde{v}_{k+1}\|_2} \\ \mu_k &= \langle v_k, \widetilde{v}_{k+1} \rangle \Leftrightarrow \mu_k = \langle v_k, Av_k \rangle \end{cases}$$

On prétend que sous certaines hypothèses, v_k tend vers un vecteur v_∞ colinéaire à u_1 quand $k \to +\infty$, donc que Av_k tend vers $\lambda_1 v_\infty$ de sorte que $\mu_k = \langle v_k, Av_k \rangle$ tend vers $\lambda_1 \|v_\infty\|_2^2 = \lambda_1$.

La méthode (IV.1) ainsi définie s'appelle **méthode de la puissance** car v_k est proportionnel à $A^k v_0$, plus exactement $\boxed{v_k = \dfrac{A^k v_0}{\|A^k v_0\|_2}}$ comme on le voit facilement par récurrence: on a $v_0 = \dfrac{A^0 v_0}{\|A^0 v_0\|_2}$ car v_0 est unitaire (initialisation). Par ailleurs si $v_k = \beta A^k v_0$ avec $\beta = \dfrac{1}{\|A^k v_0\|_2} > 0$, alors $\widetilde{v}_{k+1} = \beta A^{k+1} v_0$ et $v_{k+1} = \dfrac{\widetilde{v}_{k+1}}{\|\widetilde{v}_{k+1}\|_2} = \dfrac{\beta A^{k+1} v_0}{\|\beta A^{k+1} v_0\|_2} = \dfrac{\beta}{|\beta|} \dfrac{A^{k+1} v_0}{\|A^{k+1} v_0\|_2} = \dfrac{A^{k+1} v_0}{\|A^{k+1} v_0\|_2}$ (hérédité).

Théorème IV.4.2 (Convergence de la méthode de la puissance)**.**

Soit $A \in \mathcal{M}_n(\mathbb{K})$ une matrice diagonalisable dans $\mathcal{M}_n(\mathbb{K})$. Soient $\lambda_1, \cdots, \lambda_n$ ses valeurs propres et $(u_1, \cdots, u_n)$ une base de vecteurs propres unitaires associés.

Si $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$ et si $v_0 = \sum_{i=1}^{n} a_i u_i$ est un vecteur unitaire ($\|v_0\|_2 = 1$) n'appartenant pas à $\mathrm{Vect}(u_2, \cdots, u_n)$ (c'est à dire $a_1 \neq 0$), alors la suite $(\mu_k)_{k\geq 1}$ définie par (IV.1) converge vers λ_1. De plus, il existe des constantes $C, C' \geq 0$ indépendantes de k et il existe une suite $(\varepsilon_k \in \mathbb{K})_{k\geq 1}$ de scalaires de module 1, $|\varepsilon_k| = 1$, tels que :

(IV.2)
$$\begin{cases} \|\varepsilon_k v_k - u_1\|_2 & \leq & C\left(\dfrac{|\lambda_2|}{|\lambda_1|}\right)^k \\ |\mu_k - \lambda_1| & \leq & C'\left(\dfrac{|\lambda_2|}{|\lambda_1|}\right)^k \end{cases}$$

Remarque IV.4.3.
La convergence est 2 fois plus rapides si A est une matrice symétrique. Cf. TD.

Démonstration. — On part de la décomposition $v_0 = \sum_{i=1}^{n} a_i u_i$ qui implique (rappel: $a_1 \neq 0$):

$$A^k v_0 = \sum_{i=1}^{n} a_i A^k u_i = \sum_{i=1}^{n} a_i \lambda_i^k u_i = a_1 \lambda_1^k \left(u_1 + \sum_{i=2}^{n} \frac{a_i}{a_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k u_i \right) = a_1 \lambda_1^k w_k$$

Comme $v_k = \dfrac{A^k v_0}{\|A^k v_0\|_2}$ on a $w_k = \tau_k v_k$ où $\tau_k = \varepsilon_k \|w_k\|_2$ avec ε_k un scalaire de module 1.

Comme $\left|\dfrac{\lambda_i}{\lambda_1}\right| < 1$ pour $i \in [2, n]$, on remarque que $w_k \to u_1$ quand $k \to +\infty$: w_k (donc v_k) s'aligne sur u_1 à l'infini. Par inégalité triangulaire, comme $\|u_2\|_2 = \cdots = \|u_n\|_2 = 1$:

$$\|w_k - u_1\|_2 \leq \| \sum_{i=2}^{n} \frac{a_i}{a_1} \left(\frac{\lambda_i}{\lambda_1}\right)^k u_i \| \leq \sum_{i=2}^{n} \frac{|a_i|}{|a_1|} \left(\frac{|\lambda_i|}{|\lambda_1|}\right)^k$$

puis en utilisant le fait que $|\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$:

$$\|w_k - u_1\|_2 \leq C_1 \left(\frac{|\lambda_2|}{|\lambda_1|}\right)^k \text{ avec } C_1 = \sum_{i=2}^{n} \frac{|a_i|}{|a_1|}.$$

On écrit à présent $w_k = \tau_k v_k = \varepsilon_k v_k + \varepsilon_k(\|w_k\|_2 - 1)v_k = \varepsilon_k v_k + \varepsilon_k(\|w_k\|_2 - \|u_1\|_2)v_k$. Il vient:

$$\|\varepsilon_k v_k - u_1\|_2 \leq \|w_k - u_1\|_2 + \Big| \|w_k\|_2 - \|u_1\| \Big| \leq \|w_k - u_1\|_2 + \|w_k - u_1\|_2 \leq 2C_1 \left(\frac{|\lambda_2|}{|\lambda_1|}\right)^k$$

A présent $\mu_k = \langle v_k, A v_k \rangle$ de sorte que $\langle \varepsilon_k v_k, A(\varepsilon_k v_k) \rangle = |\varepsilon_k|^2 \mu_k = \mu_k$ car $|\varepsilon_k| = 1$, tandis que $\lambda_1 = \langle u_1, A u_1 \rangle$ car $\|u_1\|_2 = 1$. On obtient:

$$\mu_k - \lambda_1 = \langle \varepsilon_k v_k - u_1, A(\varepsilon_k v_k) \rangle + \langle u_1, A(\varepsilon_k v_k - u_1) \rangle$$

puis par Cauchy-Schwarz[(10)] , puisque $\|u_1\|_2 = \|v_k\| = 1$:

$$|\mu_k - \lambda_1| \leq 2\|\varepsilon_k v_k - u_1\|_2 \times \|A\|_2 \leq 4\|A\|_2 C_1 \left(\frac{|\lambda_2|}{|\lambda_1|}\right)^k.$$

□

IV.4.1.2. Méthode de la puissance et test d'arrêt. — Afin de proposer un test d'arrêt effectif pour la méthode de la puissance, nous énonçons (en réel) un résultat technique, précédé de la remarque suivante:

Remarque IV.4.4.
Si λ est valeur propre de la matrice $A \in \mathcal{M}_n(\mathbb{K})$, alors λ est aussi valeur propre de sa transposée ${}^tA$. En effet $\det(\lambda I - A) = \det\left({}^t(\lambda I - A)\right) = \det(\lambda I - {}^tA)$, c'est à dire A et ${}^tA$ ont même polynôme caractéristique.

Lemme IV.4.5.

Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice diagonalisable et soit $\lambda \in \mathbb{R}$ une valeur propre simple de A. Il existe donc deux vecteurs unitaires $u, z \in \mathbb{R}^n$ tels que $Au = \lambda u$ et ${}^tAz = \lambda z$.
Pour $\varepsilon \geq 0$ on pose $A(\varepsilon) = A + \varepsilon E$ où $E \in \mathcal{M}_n(\mathbb{R})$ est une matrice vérifiant $\|E\|_2 = 1$. Alors il existe une fonction $\varepsilon \mapsto (\lambda(\varepsilon), u(\varepsilon)) \in \mathbb{R} \times \mathbb{R}^n$ de classe C^1 definie sur un voisinage de 0 telle que $\lambda(0) = \lambda$, $u(0) = u$, $A(\varepsilon)u(\varepsilon) = \lambda(\varepsilon)u(\varepsilon)$, $\|u(\varepsilon)\|_2 = 1$. De plus $\left|\frac{d\lambda}{d\varepsilon}(0)\right| \leq \frac{1}{|\langle z, u\rangle|}$.
En conséquence $|\lambda(\varepsilon) - \lambda| \lessapprox \frac{\varepsilon}{|\langle z, u\rangle|}$ pour $\varepsilon \simeq 0$.

Démonstration. — Le polynôme caractéristique $P : (\varepsilon, \mu) \in \mathbb{R} \times \mathbb{R} \mapsto P(\varepsilon, \mu) = \det(\mu I - A(\varepsilon))$ est une application C^∞ qui vérifie $P(0, \lambda) = 0$ et $\frac{\partial P}{\partial \mu}(0, \lambda) \neq 0$ car λ est valeur propre simple. Par le théorème des fonctions implicites, il existe un voisinage $U \subset \mathbb{R}$ de 0, un voisinage $V \subset \mathbb{R}$ de λ et une fonction $\varepsilon \in U \mapsto \lambda(\varepsilon) \subset V$ de classe $C^\infty(I)$ tels que $\lambda(0) = \lambda$ et $\begin{cases} (\varepsilon, \mu) \in U \times V \\ P(\varepsilon, \mu) = 0 \end{cases} \Rightarrow \mu = \lambda(\varepsilon)$ valeur propre simple de $A(\varepsilon)$.
Plus généralement, l'application $\Psi : (\varepsilon, \mu, X) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^n \mapsto \left(\|X\|_2^2 - 1, A(\varepsilon)X - \mu X\right) \in \mathbb{R} \times \mathbb{R}^n$ est $C^\infty(\mathbb{R} \times \mathbb{R} \times \mathbb{R}^n)$ et vérifie $\Psi(0, \lambda, u) = (0, 0)$. Par ailleurs, pour tout $(\mu, Y) \in \mathbb{R} \times \mathbb{R}^n$:

$$\Psi(0, \lambda + \delta, u + Y) = \left(2\,{}^tuY, AY - \lambda Y - \delta u\right) + \left(\|Y\|_2^2, \mu u\right)$$

(on rappelle que $\|X\|_2^2 = {}^tXX$) de sorte que la différentielle partielle de Ψ suivant (μ, X) en $(0, \lambda, u)$ est

$$D_{(\mu,X)}\Psi_{(0,\lambda,u)} : (\delta, Y) \in \mathbb{R} \times \mathbb{R}^n \mapsto \left(2\,{}^tuY, AY - \lambda Y - \delta u\right) \in \mathbb{R} \times \mathbb{R}^n$$

Montrons que cette endomorphisme est inversible. Pour cela nous recherchons son noyau: $D_{(\mu,X)}\Psi_{(0,\lambda,u)}(\delta, Y) = 0$ ssi $\begin{cases} {}^tuY = 0 \\ AY - \lambda Y - \delta u = 0 \Leftrightarrow (A - \lambda I)Y = \delta u \end{cases}$. De l'égalité $(A - \lambda I)Y = \delta u$ on tire $(A - \lambda I)^2 Y = 0$. Or $\mathrm{Ker}(A - \lambda I)^2 = \mathrm{Ker}(A - \lambda I) = \mathrm{Vect}(u)$ car λ est valeur propre simple. Donc il existe $\alpha \in \mathbb{R}$ tel que $Y = \alpha u$. Comme ${}^tuY = 0$ on tire que $\alpha = 0$. Donc $Y = 0$ puis $\delta = 0$. En conclusion $\mathrm{Ker}\, D_{(\mu,X)}\Psi_{(0,\lambda,u)} = \{0\}$ et $D_{(\mu,X)}\Psi_{(0,\lambda,u)}$ est inversible. En application du théorème des fonctions

(10) Rappel de Cauchy-Schwarz: $|\langle x, y\rangle| \leq \|x\|_2.\|y\|_2$

implicites: il existe un voisinage $U \subset \mathbb{R}$ de 0, un voisinage $\mathscr{V} \subset \mathbb{R} \times \mathbb{R}^n$ de (λ, u) et une application $\varepsilon \in U \mapsto \Big(\lambda(\varepsilon), u(\varepsilon)\Big) \subset \mathscr{V}$ de classe $C^\infty(I)$ tels que $\Big(\lambda(0), u(0)\Big) = \Big(\lambda, u\Big)$ et $\left\{ \begin{array}{l} (\varepsilon, \mu, X) \in U \times \mathscr{V} \\ \Psi(\varepsilon, \mu, X) = 0 \end{array} \right. \Rightarrow$ $(\mu, X) = \Big(\lambda(\varepsilon), u(\varepsilon)\Big)$ où $\lambda(\varepsilon)$ est valeur propre simple de $A(\varepsilon)$ et $u(\varepsilon)$ est un vecteur propre unitaire associé.
On part maintenant de l'équation $\Big(A + \varepsilon E\Big)u(\varepsilon) = \lambda(\varepsilon)u(\varepsilon)$ et on dérive:

$$Eu(\varepsilon) + \Big(A + \varepsilon E\Big)\frac{du}{d\varepsilon}(\varepsilon) = \frac{d\lambda}{d\varepsilon}(\varepsilon)u(\varepsilon) + \lambda(\varepsilon)\frac{du}{d\varepsilon}(\varepsilon).$$

On en déduit en faisant $\varepsilon = 0$ que $Eu + \Big(A - \lambda I\Big)\frac{du}{d\varepsilon}(0) = \frac{d\lambda}{d\varepsilon}(0)u$ puis:

$$\langle z, Eu\rangle + \langle z, \Big(A - \lambda I\Big)\frac{du}{d\varepsilon}(0)\rangle = \frac{d\lambda}{d\varepsilon}(0)\langle z, u\rangle.$$

Mais $\Big({}^tA - \lambda\Big)z = 0$ entraîne ${}^tz\Big(A - \lambda I\Big) = 0$, donc $\langle z, \Big(A - \lambda I\Big)\frac{du}{d\varepsilon}(0)\rangle = 0$. Par conséquent: $\langle z, Eu\rangle = \frac{d\lambda}{d\varepsilon}(0)\langle z, u\rangle$. Comme $\|u\|_2 = \|z\|_2 = \|E\|_2 = 1$ on obtient que $\left|\frac{d\lambda}{d\varepsilon}(0)\right| \times |\langle z, u\rangle| = |\langle z, Eu\rangle| \leq 1$.
Il reste à montrer que $\langle z, u\rangle \neq 0$ pour pouvoir conclure. Il existe une matrice $P \in \mathcal{M}_n(\mathbb{R})$ inversible, formée des vecteurs (colonnes) propres de A telle que $P^{-1}AP = \mathrm{Diag}(\lambda, \lambda_2, \cdots, \lambda_n) \in \mathcal{M}_n(\mathbb{R})$ diagonale. On peut supposer que la première colonne de P soit le vecteur unitaire u. Par ailleurs ${}^tP\,{}^tA\,{}^t(P^{-1}) = {}^tD = D$, donc le premier vecteur colonne de ${}^t(P^{-1})$ est de la forme αz et on a donc $\alpha\langle z, u\rangle = 1$.
L'estimation $|\lambda(\varepsilon) - \lambda| \lessapprox \frac{\varepsilon}{|\langle z, u\rangle|}$ pour $\varepsilon \simeq 0$, se déduit du théorème des accroissements finis. □

Test d'arrêt pour la méthode de la puissance: On suppose ici que $A \in \mathcal{M}_n(\mathbb{R})$ est une matrice diagonalisable dans $\mathcal{M}_n(\mathbb{R})$ et que les hypothèses du théorème IV.4.2 sont satisfaites. En particulier λ_1 est valeur propre dominante de A associée à la valeur propre unitaire $u_1 \in \mathbb{R}^n$.
Soit $(\mu_k, v_k)_{k\geq 1} \in \mathbb{R}\times\mathbb{R}^n$ la suite définie par la méthode de la puissance (IV.1). On introduit le (vecteur) **résidu** au cran k,

$$r_k = Av_k - \mu_k v_k \in \mathbb{R}^n \tag{IV.3}$$

qui tend vers 0 quand $k \to +\infty$. En parallèle on introduit la matrice de rang 1 $E_k = -\frac{r_k({}^tv_k)}{\|r_k\|_2} \in \mathcal{M}_n(\mathbb{R})$.
Observons que pour tout $X \in \mathbb{R}^n$, $\|r_k({}^tv_k)X\|_2 = |({}^tv_k)X| \times \|r_k\|_2$ de sorte que $\|E_k\|_2 = \sup_{\|X\|_2=1} |({}^tv_k)X| = 1$ car $\|v_k\|_2 = 1$. Pour la même raison, $E_kv_k = -\frac{r_k}{\|r_k\|_2}$. On déduit de (IV.3):

$$\Big(A + \|r_k\|_2 E_k\Big)v_k = \mu_k v_k, \tag{IV.4}$$

autrement dit que v_k est vecteur propre unitaire associée à la valeur propre μ_k de la perturbation $A + \|r_k\|_2\|E_k\|_2$ de la matrice A. En application du théorème IV.4.5 on a, pour k assez grand:

$$|\mu_k - \lambda_1| \lessapprox \frac{\|r_k\|_2}{|\langle z_1, u_1\rangle|} \tag{IV.5}$$

où $z_1 \in \mathbb{R}^n$ est un vecteur unitaire vérifiant ${}^tAz_1 = \lambda_1 z_1$. Problème: comment estimer le **résidu normalisé** $\frac{\|r_k\|_2}{|\langle z_1, u_1\rangle|}$ alors qu'on ne connaît ni u_1 ni z_1. On sait cependant que la suite des vecteurs

unitaires v_k converge vers un vecteur colinéaire à u_1. On peut aussi faire de même pour estimer z_1 par une suite (w_k). On modifie donc le schéma (IV.1) en introduisant deux vecteurs unitaires v_0 et w_0 et:

$$\text{(IV.6)} \qquad \begin{cases} \widetilde{v}_{k+1} = Av_k, & \widetilde{w}_{k+1} = {}^tAw_k \\ v_{k+1} = \dfrac{\widetilde{v}_{k+1}}{\|\widetilde{v}_{k+1}\|_2}, & w_{k+1} = \dfrac{\widetilde{w}_{k+1}}{\|\widetilde{w}_{k+1}\|_2} \\ \mu_k = \langle v_k, \widetilde{v}_{k+1}\rangle, & r_k = \widetilde{v}_{k+1} - \mu_k v_k \\ s_k = \dfrac{\|r_k\|_2}{|\langle w_k, v_k\rangle|} \end{cases}$$

avec s_k fournissant une estimation de $e_k = |\mu_k - \lambda_1|$ (comparer à (IV.5), donc un critère d'arrêt effectif.

Remarque IV.4.6.
Si A est une matrice symétrique, le calcul de la suite (w_k) est évidemment inutile puisqu'on peut prendre $w_k = v_k$.

Remarque IV.4.7.
Dans la méthode de la puissance, il est nécessaire que le vecteur de départ v_0 (resp. w_0) ait une composante non nulle suivant le vecteur propre u_1 (resp. z_1). Il est par ailleurs necessaire que la matrice A ait une valeur propre simple dominante. Ces conditions sont impossibles à vérifier à priori. C'est pourquoi dans l'algorithme de la puissance ci-dessous, on couple le test d'arrêt avec un nombre d'itération maximal à ne pas dépasser.

Algorithme de la puissance
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{K})$ diagonalisable ayant une valeur propre simple dominante λ_1, cet algorithme renvoit un couple valeur propre-vecteur propre (λ_1, u_1) avec estimation de l'erreur basée sur le calcul du résidu normalisé.
Entrée : la matrice A, le seuil de tolérance $\varepsilon > 0$, un nombre d'itération maximal N.
Sortie : l'estimation numérique du couple (λ_1, u_1).
Etape 1 [initialisation] donnée des vecteurs unitaires initiaux v et w, d'un $s \geq \varepsilon$, de $k = 0$;
Etape 2 [boucle] tant que $s \geq \varepsilon$ et $k \leq N$ faire

$$\widetilde{v} = Av, \quad \widetilde{w} = {}^tAw$$
$$\mu = \langle v, \widetilde{v}\rangle >, \quad r = \widetilde{v} - \mu v$$
$$s \leftarrow \frac{\|r\|_2}{|\langle w, v\rangle|}$$
$$v \leftarrow \frac{\widetilde{v}}{\|\widetilde{v}\|_2}, \quad w \leftarrow \frac{\widetilde{w}}{\|\widetilde{w}\|_2}$$
$$k \leftarrow k + 1$$

fin tant
Etape 3 [fin] sortir (μ, v) et fin.

IV.4.2. Méthode de la puissance inverse. — On suppose ici que la matrice $A \in \mathcal{M}_n(\mathbb{C})$, (resp. $\mathcal{M}_n(\mathbb{R})$), est inversible et diagonalisable. On va de plus supposer que les valeurs propres $\lambda_1, \cdots, \lambda_n$

de A (associées à une base de vecteurs propres unitaires $(u_1, \cdots, u_n)$) vérifient

$$0 < |\lambda_1| < |\lambda_2| \leq |\lambda_3| \leq \cdots \leq |\lambda_n|$$

et on désire calculer (λ_1, u_1).
Comme A est diagonalisable, son inverse A^{-1} l'est également et admet les valeurs propres $\lambda_1^{-1}, \cdots, \lambda_n^{-n}$ (associées à la même base de vecteurs propres unitaires $(u_1, \cdots, u_n)$) qui satisfont:

$$|\lambda_1^{-1}| > |\lambda_2^{-1}| \geq |\lambda_3^{-1}| \geq \cdots \geq |\lambda_n^{-1}|$$

On peut donc appliquer la méthode de la puissance pour A^{-1} : c'est la méthode de la **puissance inverse**. Cela donne un algorithme de la forme:

$$\text{(IV.7)} \qquad \begin{cases} A\widetilde{v}_{k+1} &= v_k \quad (*) \\ v_{k+1} &= \dfrac{\widetilde{v}_{k+1}}{\|\widetilde{v}_{k+1}\|_2} \\ \mu_k &= \langle v_k, \widetilde{v}_{k+1} \rangle \end{cases}$$

où la suite (μ_{k-1}) tend vers λ_1^{-1}. On notera que par $(*)$, on doit procéder (plusieurs fois) à la résolution d'un système correspondant à la même matrice A : cf. les chapitres précédents pour les méthodes directes ou itératives à appliquer. (On ne calcule pas A^{-1} !).

Algorithme de la puissance inverse
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(K)$ inversible et diagonalisable, de valeurs propres vérifiant $0 < |\lambda_1| < |\lambda_2| \leq |\lambda_3| \leq \cdots \leq |\lambda_n|$, cet algorithme renvoit un couple valeur propre-vecteur propre (λ_1, u_1) avec estimation de l'erreur basée sur le calcul du résidu normalisé.
Entrée : la matrice A, le seuil de tolérance $\varepsilon > 0$, un nombre d'itération maximal N.
Sortie : l'estimation numérique du couple (λ_1, u_1).
Etape 1 [initialisation] donnée des vecteurs unitaires initiaux v et w, d'un $s \geq \varepsilon$, de $k = 0$;
Etape 2 [boucle] tant que $s \geq \varepsilon$ et $k \leq N$ faire
résoudre les systèmes $A\widetilde{v} = v, \quad {}^tA\widetilde{w} = w$
$\mu = \langle v, \widetilde{v} \rangle >, \quad r = \widetilde{v} - \mu v$
$s \leftarrow \dfrac{\|r\|_2}{|\langle w, v \rangle|}$
$v \leftarrow \dfrac{\widetilde{v}}{\|\widetilde{v}\|_2}, \quad w \leftarrow \dfrac{\widetilde{w}}{\|\widetilde{w}\|_2}$
$k \leftarrow k + 1$
fin tant
Etape 3 [fin] sortir $(\dfrac{1}{\mu}, v)$ et fin.

IV.5. Une méthode globale pour des matrices symétriques : la méthode de Jacobi

Nous détaillons dans cette partie une méthode globale de recherche de valeurs propres mais valable uniquement pour une matrice $A \in \mathcal{M}_n(\mathbb{R})$ symétrique. Nous introduisons au préalable les matrices de Givens[(11)]

[(11)] James Wallace Givens,1910-1993, mathématicien américain

Définition IV.5.1 (Matrice de rotation de Givens)**.**

Pour $(p,q) \in \mathbb{N}^\star \times N^\star$, $1 \leq p < q \leq n$, et pour $\theta \in [0, 2\pi[$, on appelle **matrice de rotation de Givens** $Q_{p,q}(\theta) \in \mathcal{M}_n(\mathbb{R})$ la matrice orthogonale définie par:

$$Q_{p,q}(\theta) = \begin{pmatrix} I_{p-1} & \vdots & 0 & \vdots & 0 \\ \cdots & c & \cdots & s & \cdots \\ 0 & \vdots & I_{q-p-1} & \vdots & 0 \\ \cdots & -s & \cdots & c & \cdots \\ 0 & \vdots & 0 & \vdots & I_{n-q} \end{pmatrix}$$

où $c = \cos(\theta)$ et $s = \sin(\theta)$.

Remarque IV.5.2.

La matrice de Givens $Q_{p,q}(\theta)$ est la matrice de la rotation d'angle $-\theta$ dans le plan engendré par les vecteurs (e_p, e_q) de la base canonique.

Soit $A = (a_{i,j} = a_{j,i}) \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique et soit $Q_{p,q}$ une matrice de Givens. Posons $B = (b_{i,j}) = Q_{p,q}^{-1} A Q_{p,q}(\theta) = {}^t Q_{p,q} A Q_{p,q}(\theta)$. Alors B est une matrice symétrique, $b_{i,j} = b_{j,i}$, telle que:

$$\text{(IV.8)} \qquad \begin{cases} b_{i,p} = c.a_{i,p} - s.a_{i,q} & \text{si } i \neq p,q \\ b_{i,q} = s.a_{i,p} + c.a_{i,q} & \text{si } i \neq p,q \\ b_{p,p} = c^2.a_{p,p} - 2cs.a_{p,q} + s^2.a_{q,q} \\ b_{q,q} = s^2.a_{p,p} + 2cs.a_{p,q} + c^2.a_{q,q} \\ b_{p,q} = (c^2 - s^2).a_{p,q} + cs.(a_{p,p} - a_{q,q}) \\ b_{i,j} = a_{i,j} \quad \text{sinon} \end{cases}$$

A un $O(1)$ près, le coût du calcul des coefficients de B est celui des $b_{i,p}$ (de l'ordre de $3n$ opérations) et des $b_{i,q}$ (idem), soit $N_{op}(n) \asymp 6n$ opérations.

Exemple IV.5.3.

Soit $A = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{1,2} & a_{2,2} & a_{2,3} \\ a_{1,3} & a_{2,3} & a_{3,3} \end{pmatrix}$ une matrice symétrique et $Q_{2,3} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & s \\ 0 & -s & c \end{pmatrix}$. On a:

$$B = {}^t Q_{2,3} A Q_{2,3} = \begin{pmatrix} a_{1,1} & c.a_{1,2} - s.a_{1,3} & s.a_{1,2} + c.a_{1,3} \\ c.a_{1,2} - s.a_{1,3} & c^2.a_{2,2} - 2cs.a_{2,3} + s^2.a_{3,3} & (c^2 - s^2).a_{2,3} + cs.(a_{2,2} - a_{3,3}) \\ s.a_{1,2} + c.a_{1,3} & (c^2 - s^2).a_{2,3} + cs.(a_{2,2} - a_{3,3}) & s^2.a_{2,2} + 2cs.a_{2,3} + c^2.a_{3,3} \end{pmatrix}$$

IV.5.1. Méthode de Jacobi : principe général. — La méthode de Jacobi[(12)] consiste à construire une suite

$$A^{(0)} = A, A^{(1)}, \cdots, A^{(k)} = \left(a_{i,j}^{(k)}\right), \cdots$$

[(12)] Carl Gustav Jacob Jacobi proposa cette méthode en 1846 !

de matrices symétriques, chacune étant orthogonalement semblable à la précédente via une matrice de Givens Q_k,

$$\text{(IV.9)} \qquad \begin{cases} A^{(k+1)} = {}^t Q^{(k)} A^{(k)} Q^{(k)} \\ Q^{(k)} = Q^{(k)}_{p(k),q(k)}(\theta_k) \end{cases},$$

avec Q_k choisie de telle sorte que la suite $\left(A^{(k)}\right)$ va converger vers une matrice diagonale. On a ainsi pour tout $k \in \mathbb{N}$:

$$A^k = {}^t\left(Q^{(1)} \cdots Q^{(k)}\right) A \left(Q^{(1)} \cdots Q^{(k)}\right)$$

A l'étape k, on choisit les indices $p(k) < q(k)$ de la matrice de Givens $Q^{(k)}$ de sorte que $\boxed{a^{(k)}_{p(k),q(k)} \neq 0}$ (si ce n'est pas possible c'est que $A^{(k)}$ est déjà une matrice diagonale semblable à A).
On choisit alors l'angle θ_k tel que $\boxed{a^{(k+1)}_{p(k),q(k)} = 0}$ c'est à dire, d'après (IV.8):

$$\text{(IV.10)} \qquad \begin{cases} (c^2 - s^2).a^{(k)}_{p(k),q(k)} + cs.(a^{(k)}_{p(k),p(k)} - a^{(k)}_{q(k),q(k)}) = 0 \\ \\ c = \cos(\theta_k), \quad s = \sin(\theta_k) \end{cases}$$

Cela mène à la condition, en remarquant que $\dfrac{c^2 - s^2}{2cs} = \dfrac{1}{\tan(2\theta_k)}$:

$$\text{(IV.11)} \qquad \begin{cases} \dfrac{1}{\tan(2\theta_k)} = \dfrac{a^{(k)}_{q(k),q(k)} - a^{(k)}_{p(k),p(k)}}{2a^{(k)}_{p(k),q(k)}} := \sigma_k \\ t = \dfrac{s}{c} = \tan(\theta_k), \quad \tan(2\theta_k) = \dfrac{2t}{1-t^2} \end{cases},$$

d'où l'équation: $t^2 + 2\sigma_k t - 1 = 0$.
Cette équation admet 2 racines, dont l'une est dans l'intervalle $[-1, 1[$ (question : pourquoi ?), correspondant à un choix d'angle $\theta_k \in [-\pi/4, \pi/4[$.
On obtient alors, pour ce choix de t:

$$\text{(IV.12)} \qquad c = \frac{1}{\sqrt{1+t^2}}, \quad s = tc.$$

Le calcul de c, s ne demande ainsi que $O(1)$ opérations, de sorte que le coût d'une itération pour la méthode de Jacobi est de $6n + O(1)$.
Remarques:

1. Notez qu'on n'a pas à calculer les angles θ_k !
2. L'élément annulé lors d'une itération redevient en général non nul à l'étape suivante.

Exemple IV.5.4.

On considère la matrice symétrique $A^{(0)} = A = \begin{pmatrix} 1 & 1 & -1 \\ 1 & -1 & 2 \\ -1 & 2 & 2 \end{pmatrix}$. Pour construire $A^{(1)}$ par la méthode de Jacobi, on va considérer le coefficient $a_{2,3} = 2$ correspondant au couple d'indices $(p_1, q_1) = (2, 3)$ qui a la vertu de maximiser les termes $\{|a_{i,j}|, i < j\}$. (Cf. ci-après).

On a $\sigma_1 = \dfrac{a_{3,3} - a_{2,2}}{2a_{2,3}} = \dfrac{3}{4}$. L'équation $t^2 + 2\sigma_1 t - 1 = 0$ admet 2 racines, dont la racine $t = \dfrac{1}{2} \in [-1, 1[$. Ceci correspond à:

$$c = \cos(\theta_1) = \frac{1}{\sqrt{1+t^2}} = \frac{2\sqrt{5}}{5}, \quad \text{et} \quad s = \sin(\theta_1) = tc = \frac{\sqrt{5}}{5}.$$

La matrice de Givens $Q^{(1)}$ est de la forme: $Q^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{2\sqrt{5}}{5} & \frac{\sqrt{5}}{5} \\ 0 & -\frac{\sqrt{5}}{5} & \frac{2\sqrt{5}}{5} \end{pmatrix}$. La matrice symétrique $A^{(1)} = {}^tQ^{(1)}AQ^{(1)}$ semblable à A est alors : $A^{(1)} = \begin{pmatrix} 1 & \frac{3\sqrt{5}}{5} & -\frac{\sqrt{5}}{5} \\ \frac{3\sqrt{5}}{5} & -2 & 0 \\ -\frac{\sqrt{5}}{5} & 0 & 3 \end{pmatrix}$

IV.5.2. Méthode de Jacobi classique : convergence. — La méthode de Jacobi dite **classique** consiste, à chaque cran k, à choisir le couple d'indices $(p(k), q(k))$ tel que $a^{(k)}_{p(k),q(k)}$ soit le coefficient extra-diagonal de $A^{(k)}$ de plus grande valeur absolue:

$$|a^{(k)}_{p(k),q(k)}| \geq |a^{(k)}_{i,j}|, \text{ pour tout } (i,j) \text{ tel que } i < j. \tag{IV.13}$$

(Cf. exemple IV.5.4). Cela nécessite $\dfrac{n(n-1)}{2}$ tests à chaque cran, ce qui peut être assez couteux pour des matrices de grandes tailles.

On va montrer que ce choix est optimal à l'aide de la proposition IV.5.7. On rappelle auparavant une définition.

Définition IV.5.5 (Norme de Frobénius)**.**

Soit $B \in \mathcal{M}_n(\mathbb{K})$. La **norme de Frobénius** $\|B\|_F$ de B est définie par :

$$\|B\|_F^2 = \text{Trace}({}^tBB) = \sum_{i=1}^{n}\sum_{j=1}^{n} |b_{i,j}|^2.$$

Remarque IV.5.6.

La norme de Frobénius est une norme matricielle non subordonnée à une norme vectorielle: noter que $\|I_n\|_F = \sqrt{n} \neq 1$ (si $n \geq 2$).

Proposition IV.5.7.

Soit $\left(A^{(k)}\right)_{k\geq 0}$ la suite de matrices symétriques construite par la méthode de Jacobi. Soit $D^{(k)} = \text{Diag}(a^{(k)}_{1,1}, \cdots, a^{(k)}_{n,n})$ la partie diagonale de $A^{(k)}$ et $E^{(k)}$ sa partie extradiagonale : $A^{(k)} = D^{(k)} + E^{(k)}$. Alors:

$$\|E^{(k+1)}\|_F^2 = \|E^{(k)}\|_F^2 - 2(a^{(k)}_{p(k),q(k)})^2 \tag{IV.14}$$

où $\|E\|_F$ désigne la norme de Frobénius de la matrice E.

Démonstration. — On commence par remarquer que deux matrices semblables ont même trace, donc:

$$\|A^{(k+1)}\|_F^2 = \text{Trace}({}^tQ^{(k)}\,{}^tA^{(k)}\,A^{(k)}Q^{(k)}) = \text{Trace}({}^tA^{(k)}\,A^{(k)}) = \|A^{(k)}\|_F^2.$$

Par ailleurs, si $A = D + E$ est une matrice symétrique avec D diagonale et E extradiagonale, alors

$$\|A\|_F^2 = \text{Trace}\Big((D+E)(D+E)\Big) = \text{Trace}\Big(D^2 + DE + ED + E^2\Big).$$

Or $\text{Trace}(DE) = \text{Trace}(ED) = 0$ (les diagonales de DE et ED sont formées de 0). Donc, par linéarité de la trace: $\|A\|_F^2 = \|D\|_F^2 + \|E\|_F^2$. Par suite:

$$\|D^{(k+1)}\|_F^2 + \|E^{(k+1)}\|_F^2 = \|D^{(k)}\|_F^2 + \|E^{(k)}\|_F^2,$$

soit aussi:

$$\|E^{(k+1)}\|_F^2 = \|E^{(k)}\|_F^2 + \|D^{(k)}\|_F^2 - \|D^{(k+1)}\|_F^2.$$

Posons $D^{(k)} = \text{Diag}(a_{1,1}, \cdots, a_{n,n})$ et $D^{(k+1)} = \text{Diag}(b_{1,1}, \cdots, b_{n,n})$. On a $\|D^{(k)}\|_F^2 = \sum_{i=1}^{n} a_{i,i}^2$ tandis que par (IV.8):

$$\|D^{(k+1)}\|_F^2 = \sum_{i=1}^{n} b_{i,i}^2 = (c^2a_{p,p} - 2csa_{p,q} + s^2a_{q,q})^2 + (s^2a_{p,p} + 2csa_{p,q} + c^2a_{q,q})^2 + \sum_{i\neq p,q}^{n} a_{i,i}^2$$

avec la condition $b_{p,q} = 0$, équivalente à $cs(a_{p,p} - a_{q,q}) = (s^2 - c^2)a_{p,q}$. Il vient (en se rappelant que $c^2 + s^2 = 1$):

$$\begin{aligned}
\|D^{(k+1)}\|_F^2 - \|D^{(k)}\|_F^2 &= (c^2a_{p,p} - 2csa_{p,q} + s^2a_{q,q})^2 + (s^2a_{p,p} + 2csa_{p,q} + c^2a_{q,q})^2 - (a_{p,p}^2 + a_{q,q}^2)\\
&= [(c^2-1)a_{p,p} - 2csa_{p,q} + s^2a_{q,q}][(c^2+1)a_{p,p} - 2csa_{p,q} + s^2a_{q,q}]\\
&+ [s^2a_{p,p} + 2csa_{p,q} + (c^2-1)a_{q,q}][s^2a_{p,p} + 2csa_{p,q} + (c^2+1)a_{q,q}]\\
&= -[s^2(a_{p,p} - a_{q,q}) - 2csa_{p,q}][(c^2+1)a_{p,p} - 2csa_{p,q} + s^2a_{q,q}]\\
&+ [s^2(a_{p,p} - a_{q,q}) - 2csa_{p,q}][s^2a_{p,p} + 2csa_{p,q} + (c^2+1)a_{q,q}]\\
&= [s^2(a_{p,p} - a_{q,q}) - 2csa_{p,q}][(s^2 - c^2 - 1)(a_{p,p} - a_{q,q}) + 4csa_{p,q}]\\
&= a_{p,q}^2[s^2\frac{s^2-c^2}{cs} - 2cs][(s^2 - c^2 - 1)\frac{s^2-c^2}{cs} + 4cs] = 2a_{p,q}^2
\end{aligned}$$

□

Ainsi, selon (IV.14), la méthode de Jacobi classique équivaut à optimiser la décroissance de la suite $(\|E^{(k)}\|_F)$ des normes des matrices extradiagonales.

On a dans ce cadre le théorème de convergence suivant:

Théorème IV.5.8 (Convergence de la méthode de Jacobi classique)**.**

La suite $A^{(0)} = A, A^{(1)}, \cdots, A^{(k)} = \left(a_{i,j}^{(k)}\right), \cdots$ de matrices symétriques, construite par la méthode de Jacobi classique, converge vers une matrice diagonale $D = \text{Diag}(\lambda_1, \cdots, \lambda_n)$ où les λ_i sont les

valeurs propres de A avec leur multiplicité. De plus (avec les notation de la proposition IV.5.7):

$$\|A^{(k)} - D\|_F \leq \frac{\sqrt{2}\|E^{(0)}\|_F}{1-\rho}\rho^k, \quad \rho = \sqrt{1 - \frac{2}{n(n-1)}}$$

Démonstration. — Le choix optimal (IV.13) de $a^{(k)}_{p(k),q(k)}$ implique que

$$\|E^{(k)}\|_F^2 \leq n(n-1)|a^{(k)}_{p(k),q(k)}|^2$$

On en déduit à l'aide de (IV.14) que $\|E^{(k+1)}\|_F^2 \leq \left(1 - \frac{2}{n(n-1)}\right)\|E^{(k)}\|_F^2$ ce qui implique:

$$\|E^{(k)}\|_F \leq \rho^k \|E^{(0)}\|_F. \tag{IV.15}$$

En particulier $E^{(k)} \to 0$ quand $k \to +\infty$. Il reste à montrer la convergence de la suite $(D^{(k)})$. Par le calcul (IV.8) et le fait que $a^{(k+1)}_{p(k),q(k)} = 0$, on a :

$$a^{(k+1)}_{p(k),p(k)} - a^{(k)}_{p(k),p(k)} = t a^{(k)}_{p(k),q(k)}$$

avec t choisi de tel sorte que $|t| \leq 1$. Donc : $|a^{(k+1)}_{p(k),p(k)} - a^{(k)}_{p(k),p(k)}| \leq |a^{(k)}_{p(k),q(k)}|$.
Pour les mêmes raisons : $|a^{(k+1)}_{q(k),q(k)} - a^{(k)}_{q(k),q(k)}| \leq |a^{(k)}_{p(k),q(k)}|$. Comme les autres coefficients diagonaux sont inchangés, on a : $\|D^{(k+1)} - D^{(k)}\|_F \leq \sqrt{2}|a^{(k)}_{p(k),q(k)}| \leq \|E^{(k)}\|_F$ par (IV.14).
Comme $\|E^{(k)}\|_F \leq \rho^k\|E^{(0)}\|_F$, il vient:

$$\|D^{(l)} - D^{(k)}\|_F \leq \|E^{(0)}\|_F \frac{\rho^k}{1-\rho}, \quad l > k.$$

La suite $(D^{(k)})$ est donc de cauchy dans l'espace complet $\left(\mathcal{M}_n(\mathbb{R}), \|.\|_F\right)$, donc converge vers une matrice diagonale D. Comme $(E^{(k)})$ tend vers 0, on tire que $(A^{(k)})$ tend vers D. Enfin,

$$\|A^{(k)} - D\|_F^2 = \|D^{(k)} - D\|_F^2 + \|E^{(k)}\|_F^2 \leq \frac{2}{(1-\rho)^2}\|E^{(k)}\|_F^2 \leq \frac{2\|E^{(0)}\|_F^2}{(1-\rho)^2}\rho^{2k}.$$

□

IV.5.3. La méthode de Jacobi classique : l'algorithme. — Résumons l'algorithme de Jacobi classique. Si $A = A^{(0)} = (a_{i,j}) \in \mathcal{M}_n(\mathbb{R})$ est une matrice symétrique, cet algorithme consiste:

1. Etape 1, construction de $A^{(1)}$.
   (a) On recherche (p, q) tel que $|a_{p,q}| \geq |a_{i,j}|$ pour tout (i, j) tel que $i < j$.
   (b) Si $a_{p,q} = 0$ alors A est diagonale et le travail est fini. Dans ce cas on pose $c = 1$ et $s = 0$ ce qui correspond à poser $Q_{p,q} = I$ comme matrice de Givens.
   Sinon on calcule $\sigma = \dfrac{a_{q,q} - a_{p,p}}{2a_{p,q}}$ puis les racines $t_\pm = -\sigma \pm \sqrt{\sigma^2 + 1}$ de l'équation $t^2 + 2\sigma t - 1 = 0$, on retient la racine $t \in [-1, 1[$. Autrement dit on pose

$$\begin{cases} t = -\sigma + \sqrt{\sigma^2+1} & \text{si} \quad \sigma > 0 \\ t = -\sigma - \sqrt{\sigma^2+1} & \text{si} \quad \sigma \leq 0 \end{cases}$$

   On pose $c = \dfrac{1}{\sqrt{1+t^2}}$ et $s = tc$.
   (c) On calcule la matrice $A^{(1)} = {}^t Q_{p,q} A Q_{p,q}$ à l'aide des formules (IV.8).
2. etc..

Il reste à fixer le **test d'arrêt**. Pour cela on s'appuie sur l'inégalité (IV.15). C'est à dire, pour un seuil de tolérance $\varepsilon > 0$ fixé, on s'arrête dès que $\|E^{(k)}\|_F \leq \varepsilon \|E\|_F$ où $E^{(k)}$ et E sont les matrices extradiagonales de $A^{(k)}$ et A respectivement.

Remarque IV.5.9.

Si $A = (a_{i,j})$ est symétrique et si E désigne sa partie extradiagonale, on a $\|E\|_F = \left(2 \sum_{i=2}^{n} \sum_{1 \leq j < i} |a_{i,j}|^2 \right)^{1/2}$.

Dans le test d'arrêt on travaillera plutôt avec $\boxed{\Psi(A) := \frac{1}{\sqrt{2}} \|E\|_F = \left(\sum_{i=2}^{n} \sum_{1 \leq j < i} |a_{i,j}|^2 \right)^{1/2}}$.

On décrit l'algorithme en plusieurs fonctions.

L'algorithme Jacobi1 décrit la recherche du couple optimal (p, q) de la méthode de Jacobi classique, ainsi que les valeurs $c = \cos(\theta)$ et $s = \sin(\theta)$ de la matrice de Givens $Q_{p,q}(\theta)$.

Algorithme Jacobi1
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{R})$ symétrique, cet algorithme retourne le couple (p, q) $(p < q)$ tel que $|a_{p,q}| = \max_{i<j} |a_{i,j}|$. L'algorithme retourne également les valeurs c et s de la matrice de Givens.
Entrée : la matrice symétrique A.
Sortie : la liste (p, q, c, s).
Etape 1 [initialisation] $(p, q) = (1, 2)$, $m = |a_{1,2}|$;
Etape 2 [boucle] pour i de 1 à $(n-1)$ faire
  [boucle] pour j de $(i+1)$ à n faire
    si $|a_{i,j}| > m$ faire
      $(p, q) \leftarrow (i, j)$, $m \leftarrow |a_{i,j}|$
    fin si
  fin pour
fin pour
Etape 2 [Calcul de c, s] si $a_{p,q} = 0$ faire
  $c = 1$, $s = 0$
sinon faire
  $\sigma = \dfrac{a_{q,q} - a_{p,p}}{2a_{p,q}}$
  si $\sigma > 0$ faire
    $t = -\sigma + \sqrt{\sigma^2 + 1}$
  sinon faire
    $t = -\sigma - \sqrt{\sigma^2 + 1}$
  fin si
fin si
$c = \dfrac{1}{\sqrt{1 + t^2}}$, $s = tc$
Etape 3 [fin] sortir p, q, c, s et fin.

L'algorithme Jacobi2 décrit la conjugaison d'une matrice symétrique A par une matrice de Givens Q.

Algorithme Jacobi2
Etant donné $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{R})$ symétrique et une matrice de Givens $Q = Q_{p,q}(\theta)$ caractérisée par $(p < q)$, $c = \cos(\theta)$ et $s = \sin(\theta)$, cet algorithme retourne la matrice $B = {}^tQAQ$.
Entrée : la matrice symétrique A, la liste (p, q, c, s). Sortie : la matrice ${}^tQAQ$.
Etape 1 [initialisation] $B = (b_{i,j})$ une copie de A;
Etape 2 [boucle] pour i de 1 à n faire
$b_{i,p} \leftarrow c.a_{i,p} - s.a_{i,q}, \quad b_{p,i} \leftarrow b_{i,p}$
$b_{i,q} \leftarrow s.a_{i,p} + c.a_{i,q}, \quad b_{q,i} \leftarrow b_{i,q}$
fin pour
$b_{p,p} \leftarrow c^2.a_{p,p} - 2cs.a_{p,q} + s^2.a_{q,q}$
$b_{q,q} \leftarrow s^2.a_{p,p} + 2cs.a_{p,q} + c^2.a_{q,q}$
$b_{p,q} \leftarrow (c^2 - s^2).a_{p,q} + cs.(a_{p,p} - a_{q,q}), \quad b_{q,p} \leftarrow b_{p,q}$
Etape 3 [fin] sortir B et fin.

L'algorithme Jacobi3 calcule $\Psi(A) = \frac{1}{\sqrt{2}}\|E\|_F$ où E est la partie extradiagonale de la matrice A.

Algorithme Jacobi3
Etant donné $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{R})$ symétrique, cet algorithme retourne $\Psi(A) = \frac{1}{\sqrt{2}}\|E\|_F$ où E est la partie extradiagonale de A.
Etape 1 [initialisation] $\Psi = 0$
Etape 2 [boucle] pour i de 2 à n faire
pour j de 1 à $(i-1)$ faire
$\Psi \leftarrow \Psi + |a_{i,j}|^2$
fin pour
fin pour
$\Psi \leftarrow \sqrt{\Psi}$
Etape 3 [fin] sortir Ψ et fin.

L'algorithme suivant est l'algorithme de Jacobi proprement dit.

Algorithme de Jacobi
Etant donné une matrice $A = (a_{i,j}) \in \mathcal{M}_n(\mathbb{R})$ symétrique, cet algorithme la liste de ses valeurs propres $(\lambda_1, \cdots, \lambda_n)$.
Entrée : la matrice symétrique A, le seuil de tolérance $\varepsilon > 0$.
Sortie : l'estimation numérique des valeurs propres $(\lambda_1, \cdots, \lambda_n)$.
Etape 1 [initialisation] $T = Jacobi3(A), \quad S = \varepsilon.T$
Etape 2 [boucle] tant que $T \geq S$ faire
$p, q, c, s = Jacobi1(A)$
$A \leftarrow Jacobi2(A, p, q, c, s)$
$T \leftarrow Jacobi3(A)$
fin tant
Etape 3 [fin] sortir Diag(A) et fin.

Remarque IV.5.10.
Dans cette méthode on obtient une évaluation numérique du spectre de la matrice A. Connaissant les valeurs propres, on peut alors en déduire (numériquement) des vecteurs propres (si on le souhaite) par une adaptation de la méthode de la puissance (e.g., méthode de la puissance inverse avec shift - cf td).

IV.6. Une autre méthode globale : la méthode QR

IV.6.1. Principe général. — Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice inversible. Une telle matrice possède une unique décomposition de la forme $A = QR$ où Q est une matrice orthogonale et R est une matrice triangulaire supérieure dont tous les éléments diagonaux sont strictement positifs. (Cf. théorème II.7.1).

Le principe de la méthode QR de recherche des valeurs propres de la matrice A repose sur la propriété précédente : on construit suite de matrices $(A_k)_{k\geq 0}$ avec $A_0 = A$. Au cran k, on fait la décomposition QR de A_k, $A_k = Q_k R_k$ puis on définit $A_{k+1} = R_k Q_k$:

$$\begin{cases} A_k = Q_k R_k \quad \text{(décomposition QR de } A_k\text{)} \\ \\ A_{k+1} = R_k Q_k \end{cases}$$

On voit ainsi que $A_{k+1} = {}^t Q_k A_k Q_k$, c'est à dire que A_k et A_{k+1} sont orthogonalement semblables. Plus généralement:

$$A_k = {}^t(Q_1 \cdots Q_k) A (Q_1 \cdots Q_k)$$

Posons $P_k = Q_1 \cdots Q_k$. Si la suite (P_k) de matrices orthogonales converge vers une matrice P, alors P est nécessairement une matrice orthogonale (l'application $\Phi : P \in \mathcal{M}(\mathbb{R}) \mapsto {}^t QQ - I \in \mathcal{M}(\mathbb{R})$ est continue. On a $\Phi(P_k) = 0$ pour tout k, donc $\Phi(P_k) \underset{k\to\infty}{\longrightarrow} \Phi(P) = 0$. Autre manière de dire: le groupe des matrices orthogonales (c'est $\Phi^{-1}(0)$) est compact, en particulier fermé). Dans ce cas $T = {}^t PAP$ est orthogonalement semblable à A, donc a mêmes valeurs propres. Cette limite T n'est en général pas une matrice diagonale car cela voudrait dire que A est une matrice normale (i.e. ${}^t AA = A\,{}^t A$). On espère plutôt (ce qui est en général le cas) que la suite (A_k) converge vers une matrice T triangulaire. On notera que dans certains cas on ne peut espérer une telle convergence : par exemple si A est une matrice orthogonale, alors $A = QR$ avec $Q = A$ et $R = I$ et la suite (A_k) est stationnaire.

IV.6.2. Méthode QR : un résultat de convergence. — L'analyse de la convergence de la méthode QR fait encore aujourd'hui l'objet de recherche. Divers résultats sont connus, dont le résultat suivant qui est non sans lien avec la décomposition de Schur: selon le théorème IV.2.4, si $A \in \mathcal{M}_n(\mathbb{R})$ admet un polynôme caractéristique scindé dans $\mathbb{R}$, alors A admet une décomposition de Schur de la forme ${}^t PAP = T$ où $T \in \mathcal{M}_n(\mathbb{R})$ est triangulaire supérieure et $P \in \mathcal{M}_n(\mathbb{R})$ orthogonale. Cela rend donc plausible la convergence de la méthode QR dans un tel cadre. Plus précisément:

Théorème IV.6.1.

Soit $A \in \mathcal{M}_n(\mathbb{R})$ *une matrice diagonalisable inversible, de valeurs propres réelles* $\lambda_1, \cdots, \lambda_n$ *simples telles que* $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0$. *Alors la suite* $\left(A_k = (t_{i,j}^{(k)})\right)$ *définie par la méthode QR converge et* $\lim_{k\to\infty} A_k = \begin{pmatrix} \lambda_1 & t_{1,2} & \cdots & t_{1,n} \\ 0 & \lambda_2 & t_{2,3} & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{pmatrix}$. *De plus* $|t_{i,i-1}^{(k)}| = \underset{k\to\infty}{O}\left(\left|\frac{\lambda_i}{\lambda_{i-1}}\right|^k\right)$, $i \in [2, n]$.

Démonstration. — Admis □

Remarque IV.6.2.

On l'a déjà dit (remarque II.7.4), la décomposition QR est couteuse et numériquement instable. Plutôt que la méthode QR décrite ci-avant, on réduit auparavant en pratique la matrice A à étudier sous la forme dite de **Hessenberg** au moyens de matrices de Householder.

IV.7. Pour aller plus loin : ouvrages recommandés

- G. Allaire, S.M. Kaber, Algèbre linéaire numérique. Ellipses (2002).
- G. Allaire, S.M. Kaber, Introduction à Scilab: exercices pratiques corrigés d'algèbre linéaire. Ellipses (2002).
- D. Serre, Les matrices: théorie et pratique. Dunod (2001).

Présentation de l'auteur

Eric Delabaere est professeur de mathématiques à l'Université d'Angers depuis 2000. Il y a enseigné à tous les niveaux : licence, master, préparation aux concours de l'enseignement, formations doctorales. Il a précédemment occupé un poste de chercheur au CNRS (1989-2000). Il est docteur en mathématiques de l'Université de Nice-Sophia Antipolis (1991) et lauréat du concours de l'agrégation de mathématiques (1988). Il est l'auteur d'une vingtaine d'articles de recherche publiés dans des revues internationales ainsi que de plusieurs livres et chapitres de livres spécialisés. Ses recherches relèvent de l'analyse complexe, elles portent principalement sur diverses théories dites de sommation des séries (résurgence, hyperasymptotique,..) et à leurs applications à la physique théorique, en particulier celles des hautes énergies, la mécanique quantique, la théorie quantique des champs, etc.. Il a des collaborations principalement aux USA, UK, Japon, VietNam. Il a occupé divers postes de responsabilité, au ministère de l'enseignement supérieur et de la recherche comme conseiller formations, ainsi qu'à l'Université d'Angers comme vice-président en charge des formations et de la vie étudiante, administrateur de l'Université d'Angers, directeur adjoint de la faculté des sciences, responsable formation continue et relations extérieures de la faculté des sciences, co-responsable du master Data Science de mathématiques dont il a dirigé la création, etc.. Il est actuellement vice-président en charge de la politique ressources humaines et du dialogue social à l'Université d'Angers.