Méthode de Kacmarz

Julie Parreaux

2018-2019

Référence du développement : Aucune référence connue.

Leçons où on présente le développement : 159 (Forme linéaire et dualité) ; 162 (Systèmes d'équations linéaires) ; 233 (Analyse numérique matricielle).
Leçons où on peut en parler : 208 (Espace vectoriel normé).

1 Introduction

La méthode de Kacmarz permet de résoudre des systèmes linéaires sans décomposé la matrice. Elle réalise une suite de projection sur des hyperplan bien choisi. Cette méthode permet alors de résoudre des systèmes d'équations sans calculs de valeurs propres ni d'inverse : on utilise uniquement des projections.

2 Méthode de Kacmarz

Objectif : Résoudre $Ax = b$ avec $A \in \mathrm{GL}_n(\mathbb{R})$ sans décomposition ni calcul de la matrice inverse mais en effectuant une suite de projection.

Schéma du développement

1. Introduction des notations.
2. Lemme 1 : caractérisation de l'unique solution via les hyperplans.
3. Lemme 2 : caractérisation des projections que l'on utilise.
4. Présentation de la méthode (Dessin).
5. Théorème : convergence de la méthode.
   (a) Convergence de la suite des approximations.
   (b) Cette limite vaut 0.

Notations : On se place sur $\mathbb{R}^n$ muni du produit scalaire euclidien $\langle , \rangle$.

— $\overline{x}$ est l'unique solution (car A est inversible)

— $A = \begin{bmatrix} {}^t a_1 \\ \vdots \\ {}^t a_N \end{bmatrix}$ $\quad u_i = \frac{a_i}{||a_i||} \in \mathbb{R} \quad \beta_i = \frac{b_i}{a_i} \in \mathbb{R}$ (où ${}^t a_i$ est la $i^{\text{ième}}$ ligne de la matrice A).

— $H_i = \{z \in \mathbb{R} | {}^t u_i z = \beta_i\} = \beta_i u_i + \mathrm{Vect}(u_i)^{\perp}$ un hyperplan affine (la première égalité est la définition, la deuxième se montre par double inclusion :

  — $z \in H_i$, ${}^t u_i z = \beta_i +^t u_i y$ car ${}^t u_i y = 0$ avec $y \in \mathrm{Vect}(u_i)^{\perp}$ et $z = u_i \beta_i + y \in \beta_i u_i + \mathrm{Vect}(u_i)^{\perp}$

  — $z \in \beta_i u_i + \mathrm{Vect}(u_i)^{\perp}$, alors il existe $y \in \mathrm{Vect}(u_i)^{\perp}$ tel que $z = \beta_i u_i + y$, d'où ${}^t u_i z = \beta_i +^t u_i y = \beta_i$.

).

— Π_i la projection orthogonale (vectorielle) sur $\mathrm{Vect}(a_i)^\perp$ que l'on confondra avec leur matrice M_i.

Lemme 1. *On a* $\{\overline{x}\} = \cap_{i=1}^n H_i$.

Démonstration. On a alors :

$$\begin{array}{rcll} z \in \cap_{i=1}^n H_i & \Leftrightarrow & \forall i, {}^t u_i z = \beta_i & \text{(définition de } H_i\text{)} \\ & \Leftrightarrow & \forall i, {}^t a_i z = b_i & \text{(multiplication par } ||a_i||\text{)} \\ & \Leftrightarrow & Az = b & \text{(définition de } Az = b\text{)} \\ & \Leftrightarrow & z = \overline{x} & \text{(définition de } \overline{x}\text{)} \end{array}$$

□

Lemme 2. *Soit* $u \in \mathbb{R}^n$ *de norme un. La matrice* M *de la projection orthogonale sur* $\mathrm{Vect}(u)^\perp$ *est égale à* $I_N - u^t u$*. De plus* $|||M||| = 1$ *et pour tout* $x \notin \mathrm{Vect}(u)^\perp$, $\|Mx\| < \|x\|$.

Démonstration. Soit $x \in \mathbb{R}^n$.

— Le projeté orthogonal $\Pi(x) = Mx$ de x est caractérisé par :

$$\begin{cases} \Pi(x) \in \mathrm{Vect}(u)^\perp \\ \forall z \in \mathrm{Vect}(u)^\perp, \langle z, \Pi(x) - x\rangle = 0 \end{cases}$$

Notons $y = (I_n - u^t u)x = x - u^t ux$ et montrons que y vérifie la caractérisation de Mx.

— Montrons que $y \in \mathrm{Vect}(u)^\perp$. Pour cela montrons que $\langle u, y\rangle = 0$. On a

$$\begin{array}{rcll} \langle u, y\rangle & = & \langle u, x - u^t ux\rangle & \text{(Définition de } y\text{)} \\ & = & \langle u, x\rangle - \langle u, u^t ux\rangle & \text{(Linéarité du produit scalaire)} \\ & = & \langle u, x\rangle - \langle u, x\rangle & (u^t u = \|u\|_2^2 = 1 \text{ hypothèse)} \\ & = & 0 & \end{array}$$

— Soit $forall z \in \mathrm{Vect}(u)^\perp$, Montrons que $\langle z, x - y\rangle = 0$. On a

$$\begin{array}{rcll} \langle z, x - y\rangle & = & \langle z, x\rangle - \langle z, y\rangle & \text{(Linéarité du produit scalaire)} \\ & = & \langle z, x\rangle - \langle z, x\rangle - \langle z, u^t ux\rangle & \text{(Linéarité du produit scalaire et définition de } y\text{)} \\ & = & -\langle u^t uz, x\rangle & (u^t u = 1) \\ & = & 0 & ({}^t uz = 0 \text{ car } z \in \mathrm{Vect}(u)^\perp) \end{array}$$

Donc $y = \Pi(x)$ ce qui assure que $M = I_n - u^t u$.

— De plus, $\forall x \in \mathbb{R}^n$,

$$\|x\|^2 \underbrace{=}_{\text{Introduction de } \pm Mx} \|Mx + (I - M)x\|^2 \underbrace{=}_{\substack{\text{Pythagore} \\ (M,(I-M) \text{ orthogonaux})}} \|Mx\|^2 + \|(I - M)x\|^2 \geq \|Mx\|^2$$

donc $|||M||| \leq 1$ (car il existe u tel que $\|Mx\| = |||M||| \|x\|$) dont l'égalité est atteinte pour tout $x \in \mathrm{Vect}(u)^\perp$ et donc $|||M||| = 1$.

— Enfin, pour tout $x \notin \mathrm{Vect}(u)^\perp)$, $(I - M)x \notin 0$ et donc $\|Mx\| < \|x\|$.

□

On donne maintenant la méthode de Kackmarz (algorithme 1). On donne également une figure en dimension 2 (Figure 1). De plus, cette méthode peut s'écrire comme une suite récurrente définie par :

$$x_{k+1} = M_r x_k + u_r \beta_r \text{ où } r = (k+1) \mod n$$

Algorithm 1 Méthode de Kacmarz

while $||x_n - x_{n+1}|| \geq \epsilon$ **do**
  Projeter x_0 sur H_1 qui donne x_1
  Projeter x_1 sur H_2 qui donne x_2
  ...
  Projeter x_n sur H_1 qui donne x_{n+1}
end while

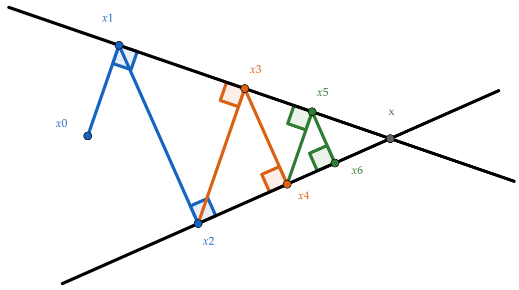

FIGURE 1 – Illustration de la méthode de Kacmarz en dimension 2.

Théorème 1. *La méthode de Kacmarz converge vers l'unique solution $\overline{x}$.*

Démonstration. Notons ϵ_k l'erreur à l'étape n correspondant de l'approximation de la solution par $\epsilon_k = x_k - \overline{x}$, et montrons que $\lim_{k\to\infty} ||\epsilon_k|| = 0$.

Étape 1 : Montrons que la suite $(\epsilon_n)_n$ converge. Montrons que la suite $(||\epsilon_k||)_k$ est décroissante. Soit $k \in \mathbb{N}$, on a :

$$\begin{array}{rcll} \epsilon_{k+1} & = & x_{k+1} - \overline{x} & \text{(définition de } (\epsilon_k)) \\ & = & (M_r x_k) + u_r\beta_r - \overline{x} & \text{(définition de } (x_k)) \\ & = & M_r\left(x_k + u_r\beta_r - \overline{x}\right) & (u_r\beta_r - \overline{x} \in \text{Vect}\left(u_r\right)^{\perp}) \\ & = & M_r\left(x_k - \overline{x}\right) & \text{(par la projection utilisée et } x \in \text{Vect}(u_i)^{\perp}, M_i x = x) \\ & = & = M_r\epsilon_k & \text{(définition de } (\epsilon_k)) \end{array}$$

Montrons les petits résultats de cette preuve.

— $u_r\beta_r - \overline{x} \in \text{Vect}\left(u_r\right)^{\perp}$: $\langle u_r\beta_r - \overline{x}, u_r\rangle = \beta_r \langle u_r, u_r\rangle - \langle \overline{x}, u_r\rangle$ par linéarité du produit scalaire. D'où, comme u_r est de norme un pour la norme euclidienne, $\langle u_r\beta_r - \overline{x}, u_r\rangle = \beta_r - \langle \overline{x}, u_r\rangle$. Comme $\overline{x} \in H_r$ (lemme 1), il existe $y \in\in \text{Vect}\left(u_r\right)^{\perp}$ tel que $\overline{x} = u_r\beta_r - y$. Par linéarité du produit scalaire, on a : $\langle u_r\beta_r - \overline{x}, u_r\rangle = \beta_r - \beta_r - \langle y, u_r\rangle = 0$.

— Si $x \in \text{Vect}(u_i)^{\perp}$, alors $M_i x = x$. Par le lemme 2, $M_i = I_n - u_i^t u_i$. On en déduit que $M_i x = (I_n - u_i^t u_i)x = x - u_i^t u_i x = x$ car $tu_i x = 0$ puisque $x \in \text{Vect}(u_i)^{\perp}$.

On en déduit que $||\epsilon_{k+1}|| \leq ||\epsilon_k||$ (car $||\epsilon_{k+1}|| \leq |||M_r||| ||\epsilon_k|| = ||\epsilon_k||$ par le lemme 2 puisque $|||M_r||| = 1$). Comme $(||\epsilon_k||)_k$ est une suite décroissante et minorée par 0 (par les propriétés de la norme qui est toujours positive), la suite converge vers une limite $l \in \mathbb{R}$.

Étape 2 : Montrons que cette limite vaut 0. On pose $T = M_n M_{n-1} \dots M_1$ (cela correspond à un cycle de notre algorithme (une boucle WHILE)). Montrons que $||T|| < 1$ (on prend une norme subordonnée à la norme euclidienne). Soit $x \in \mathbb{N}^n \setminus \{0\}$. On distingue alors deux cas.

— $\exists i \in [\![1, n]\!]$ tel que $||M_i \dots M_1 x|| < ||x||$. Dans ce cas, on peut écrire :

$$||Tx|| \leq \underbrace{||M_n|| \dots ||M_{i+1}||}_{\leq 1 \text{ (par le lemme 2)}} \underbrace{||M_i \dots M_1 x||}_{< ||x|| \text{(par hypothèse)}} < ||x||$$

— $\forall i \in [\![1, n]\!]$, $||M_i \dots M_1 x|| = ||x||$ (car $x \in \text{Vect}^{\perp}$). Dans ce cas, $||M_1 x|| = ||x||$ soit $M_1 x = x$. On en déduit que $\forall i, x \in \text{Vect}\left(u_i\right)^{\perp}$, donc $x = 0$ car $A \in GL_n(\mathbb{R})$. Contradiction.

On en déduit que $||\epsilon_{kn}|| \leq ||T^k|| ||\epsilon_0|| \to 0$ (par convergence d'une suite géométrique de raison inférieur à 1 et on trouve cette suite par récurrence). □

Remarque sur l'algorithme Sa complexité est en $O\left(n^2\right)$ et il converge pour toute matrice A même si elle n'est pas inversible.