

# Optimisation

## Chapitre 6 : Problèmes aux moindres carrés

Joseph GERGAUD, Serge GRATTON & Daniel RUIZ

10 septembre 2024

L'objectif de ce chapitre est de résoudre les problèmes aux moindres carrés non-linéaires.

$$(P) \begin{cases} \text{Min} & f(\beta) = \frac{1}{2} \|r(\beta)\|^2 \\ & \beta \in \mathbb{R}^p. \end{cases}$$

Considérons le problème aux moindres carrés linéaire

$$(P) \begin{cases} \text{Minf}(\beta) = \frac{1}{2} \|y - X\beta\|^2 \\ \beta \in \mathbb{R}^p. \end{cases}$$

Ce problème admet une solution. En effet, ce problème est équivalent à résoudre

$$(P') \begin{cases} \text{Ming}(\gamma) = \frac{1}{2} \|y - \gamma\|^2 \\ \gamma \in \text{Im } X \subset \mathbb{R}^n. \end{cases}$$

Comme  $\text{Im } X$  est un fermé, non-vidé et que  $g$  est 0-coercive, on a l'existence d'une solution.

**Remarque 6.3.1.** Le problème  $(P')$  est en fait le problème de la projection orthogonale du vecteur  $y$  sur  $\text{Im } X$ . Il possède une unique solution car  $g$  est strictement convexe. Par contre le problème initial  $(P)$  possède une ou une infinité de solutions suivant que le rang de  $X$  est  $p$  ou est strictement inférieur à  $p$ .

Le problème  $(P)$  est un problème convexe et différentiable, par suite une solution est caractérisée par la condition nécessaire d'ordre 1, qui conduit au système d'équations suivant, aussi appelé dans ce cas *équations normales* :

$$\nabla f(\beta) = X^T X \beta - X^T y = 0. \quad (1)$$

**Remarque 6.3.2.** Considérons ici la matrice  $X$  comme l'expression d'une application linéaire de  $\mathbb{R}^p$  à valeurs dans  $\mathbb{R}^n$ . On a alors

$$\mathbb{R}^p = \text{Ker } X^\perp \oplus \text{Ker } X$$

$$\mathbb{R}^n = \text{Im } X \oplus \text{Im } X^\perp = \text{Im } X \oplus \text{Ker}(X^T)$$

$$\beta^* = X^+ y$$

$$\gamma^* = \text{Proj}_{\text{Im } X}(y)$$

$$X^+ X = \text{Proj}_{(\text{Ker } X)^\perp}$$

$$X X^+ = \text{Proj}_{\text{Im } X}$$

$$\text{Si } \text{rank}(X) = p, \text{ alors } \text{Ker } X = \{\vec{0}\} \text{ et } X^+ = (X^T X)^{-1} X^T$$

Objectif : résoudre  $f(x) = 0$ .

l'intersection de la tangente à  $f$  en  $x^{(k)}$  avec l'axe des abscisses (cf. la figure 1) est donnée par la solution de  $f(x^{(k)}) + f'(x^{(k)}).(x - x^{(k)}) = 0$ , soit :

$$x = x^{(k)} - \frac{1}{f'(x^{(k)})} \cdot f(x^{(k)})$$

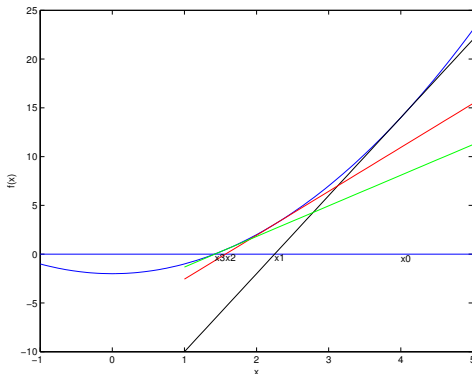


FIGURE 1 – *Algorithme de Newton.*

**Initialisation :**

choisir  $x^{(0)} \in \mathbb{R}$

choisir  $\varepsilon > 0$  et *MaxIter*

$k := 0$

**Corps :****répéter**

Résoudre  $f(x^{(k)}) + f'(x^{(k)})(x - x^{(k)}) = 0$ , soit  $x^{(k+1)}$  la solution

$k := k + 1$

**jusqu'à** ( $|f(x^{(k)})| < \varepsilon |f(x^{(0)})|$ ) ou ( $k = \text{MaxIter}$ )

**Remarque 6.4.1.**

- i) ici  $f'(x^{(k)})$  appartient à  $\mathbb{R}$ .
- ii) l'algorithme peut se "bloquer" si  $f'(x^{(k)}) = 0$ , et donc dans ce cas l'algorithme ne fournit pas de solution.
- iii) cet algorithme ne converge pas toujours.

De la même façon qu'en dimension 1 on obtient l'algorithme en calculant  $x^{(k+1)}$  à partir de  $x^{(k)}$  donné en annulant la "meilleure" approximation affine de la fonction  $f$  au voisinage de  $x^{(k)}$ , c'est-à-dire en résolvant le système linéaire à  $n$  équations et à  $n$  inconnues suivant :

$$f(x^{(k)}) + J_f(x^{(k)})(x - x^{(k)}) = 0$$

qui admet une unique solution si  $J_f(x^{(k)})$  est inversible.

**Remarque 6.4.2.** On présente très souvent l'algorithme de Newton sous la forme de la mise à jour du point courant donnée par 2. Cette équation est très utile pour la théorie, mais est bien évidemment à bannir pour une implémentation informatique.

$$x^{(k+1)} := x^{(k)} - [J_f(x^{(k)})]^{-1} \cdot f(x^{(k)}) \quad (2)$$

**Remarque 6.4.3.** lorsque  $n = 1$ ,  $f'(x^{(k)})$  est un réel et  $[f'(x^{(k)})]^{-1} = 1/f'(x^{(k)})$ , et nous retrouvons la mise à jour exposée précédemment.

en conclusion nous obtenons l'algorithme suivant :

**Initialisation :**

choisir  $x^{(0)} \in \mathbb{R}^n$

choisir  $\varepsilon > 0$  et *MaxIter*

$k := 0$

**Corps :**

**répéter**

Résoudre  $f(x^{(k)}) + J_f(x^{(k)})(x - x^{(k)}) = 0$ , soit  $x^{(k+1)}$  la solution

$k := k + 1$

**jusqu'à** ( $\|f(x^{(k)})\| < \varepsilon\|f(x^{(0)})\|$ ) ou ( $k = \text{MaxIter}$ )

**Remarque 6.4.4.**

- i) cet algorithme se “bloque” si  $J_f(x^{(k)})$  n'est pas inversible.
- ii) le test d'arrêt  $\|f(x^{(k)})\| < \varepsilon\|f(x^{(0)})\|$  signifie en fait que toutes les composantes de  $f(x^{(k)})$  sont “proches” de 0 en relatif.



**Exercice 6.4.1.** On considère la fonction

$$\begin{aligned}\text{soit } f : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto x^2 - a \text{ avec } a > 0.\end{aligned}$$

1. Donner l'itération de Newton pour résoudre  $f(x) = 0$ .

**Exercice 6.4.2.** On considère la fonction

$$\begin{aligned}f : \mathbb{R}^2 &\longrightarrow \mathbb{R}^2 \\ \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &\longmapsto \begin{pmatrix} x_1 + x_2 - 3 \\ x_1^2 + x_2^2 - 9 \end{pmatrix}\end{aligned}$$

alors  $f(x) = 0$  si et seulement si  $x = (0, 3)^T$  ou  $x = (3, 0)^T$ .

1. Appliquer l'algorithme de Newton en partant du point  $x^{(0)} = (1, 5)^T$  et en prenant  $\varepsilon = 0,6$ .

## Théorème 6.4.1

Soit  $f$  une fonction définie sur un ouvert  $\Omega$  de  $\mathbb{R}^n$  à valeurs dans  $\mathbb{R}^n$  de classe  $C^2$  dans  $B(x^*, r) \subset \Omega$  et  $x^*$  un point de  $\Omega$  tel que  $f(x^*) = 0$ . On suppose que  $f'(x^*)$  est inversible, alors il existe  $\varepsilon > 0$  tel que pour tout point  $x^{(0)} \in B(x^*, \varepsilon)$ , l'algorithme de Newton est bien défini et la suite des itérés  $(x^{(k)})_k$  converge vers  $x^*$ . De plus la convergence est quadratique, c'est-à-dire qu'il existe  $c > 0$  tel que

$$\|x^{(k+1)} - x^*\| \leq c \|x^{(k)} - x^*\|^2 \quad (3)$$

	$f_1(x) = x^2 - 1$	$f_2(x) = (x - 1)^2$
$x^{(0)}$	2	2
$x^{(1)}$	1.25	1.5
$x^{(2)}$	1.025	1.25
$x^{(3)}$	1.0003048780488	1.125
$x^{(4)}$	1.0000000464611	1.0625
$x^{(5)}$	1.0	1.03125

TABLE 1 – Convergence quadratique pour  $f_1$  et linéaire pour  $f_2$ .

► L'ensemble  $\{x \in \Omega, J_f(x) \text{ inversible}\} = \mathbb{C}_\Omega\{x \in \Omega, \det \circ J_f(x) = 0\}$  est un ouvert car c'est le complémentaire de l'antécédent d'un fermé par une application continue. Par suite il existe  $\varepsilon_1 > 0$  tel que pour tout  $x \in B(x^*, \varepsilon_1)$ ,  $J_f(x)$  soit inversible. Comme  $f$  est  $C^2$  l'application qui à  $x \in B(x^*, \varepsilon_1)$  associe  $\| [J_f(x)]^{-1} \|$  est continue, on en déduit que pour  $0 < \varepsilon_2 < \varepsilon_1$  l'image par cette application de  $\overline{B(x^*, \varepsilon_2)}$  est un compact. Par suite, il existe  $\beta > 0$  tel que pour tout  $x \in \overline{B(x^*, \varepsilon_2)}$  on a  $\| [J_f(x)]^{-1} \| \leq \beta$ .

$$\|x^{(k+1)} - x^*\| = \|x^{(k)} - x^* - [J_f(x^{(k)})]^{-1} f(x^{(k)})\| \quad (4)$$

$$\leq \| [J_f(x^{(k)})]^{-1} \| \| f(x^{(k)}) - f(x^*) - J_f(x^{(k)})(x^{(k)} - x^*) \| \quad (5)$$

$$\leq \frac{1}{2} \| J_f(x^{(k)}) \|^{-1} \sup_{x \in B_f(x^*, \varepsilon_2)} \| \nabla^2 f(x) \| \|x^{(k)} - x^*\|^2. \quad (6)$$

Mais  $f$  est  $C^2$  par suite en posant  $\sup_{x \in B_f(x^*, \varepsilon_2)} \|\nabla^2 f(x)\| = \gamma$  on obtient

$$\|x^{(k+1)} - x^*\| \leq \frac{1}{2} \beta \gamma \|x^{(k)} - x^*\|^2. \quad (7)$$

Posons maintenant  $\varepsilon = \min(\varepsilon_2, 1/(\beta\gamma))$  et prenons  $x^{(0)} \in B(x^*, \varepsilon)$ , on obtient

$$\|x^{(1)} - x^*\| \leq \frac{1}{2} \|x^{(0)} - x^*\| \leq \varepsilon/2.$$

Donc  $x^{(1)} \in B(x^*, \varepsilon)$  et par récurrence

$$\|x^{(k)} - x^*\| \leq \frac{1}{2^k} \|x^{(0)} - x^*\|.$$

Par suite  $x^{(k)} \in B(x^*, \varepsilon)$  et la suite des itérés de Newton existe et converge vers  $x^*$ .  
Quand à la convergence quadratique elle vient de l'inéquation (7). ■

Nous rappelons que le problème qui nous intéresse ici est le suivant :

$$(P) \begin{cases} \text{Minf}(x) \\ x \in \mathbb{R}^n \end{cases}$$

On applique l'algorithme de Newton pour la recherche d'un zéro de l'équation  $g(x) = 0$  avec  $g(x) = \nabla f(x)$ . A chaque itération nous aurons donc à résoudre le système linéaire suivant :

$$\nabla^2 f(x^{(k)})(x - x^{(k)}) + \nabla f(x^{(k)}) = 0. \quad (8)$$

**Remarque 6.4.5.** Si  $\nabla^2 f(x^{(k)})$  n'est pas inversible alors l'algorithme se bloque.

**Remarque 6.4.6.** la meilleure approximation quadratique de la fonctionnelle  $f$  au voisinage du point  $x^{(k)}$  est donnée par :

$$q(x) = f(x^{(k)}) + (\nabla f(x^{(k)}) | x - x^{(k)}) + \frac{1}{2}(\nabla^2 f(x^{(k)})(x - x^{(k)}) | x - x^{(k)})$$

et nous avons  $\nabla q(x) = \nabla f(x^{(k)}) + \nabla^2 f(x^{(k)})(x - x^{(k)})$  et  $\nabla^2 q(x) = \nabla^2 f(x^{(k)})$ . Par suite si  $\nabla^2 f(x^{(k)})$  est définie positive  $q$  est convexe et rechercher le minimum de  $q(x)$  sur  $\mathbb{R}^n$  est équivalent à résoudre l'équation  $\nabla q(x) = 0$ . Mais cette dernière équation est équivalente à l'itération de Newton pour résoudre  $\nabla f(x) = 0$ . En conclusion notre algorithme recherche à chaque itération, lorsque  $\nabla^2 f(x^{(k)})$  est définie positive, le minimum de l'approximation à l'ordre 2 de la fonctionnelle  $f$ .

La fonction à optimiser s'écrit ici

$$f(\beta) = \frac{1}{2} \|r(\beta)\|^2 = \frac{1}{2} \sum_{i=1}^n r_i^2(\beta).$$

Nous avons donc

$$\nabla f(\beta) = \sum_i r_i(\beta) \nabla r_i(\beta) = J_r(\beta)^T r(\beta) \quad (9)$$

$$\nabla^2 f(\beta) = \sum_i r_i(\beta) \nabla^2 r_i(\beta) + \sum_i \nabla r_i(\beta) \nabla r_i(\beta)^T \quad (10)$$

$$= S(\beta) + J_r(\beta)^T J_r(\beta) \quad (11)$$

(12)

L'itération de l'algorithme de Newton s'écrit donc

$$\beta^{(k+1)} = \beta^{(k)} - [S(\beta^{(k)}) + J_r(\beta^{(k)})^T J_r(\beta^{(k)})]^{-1} J_r(\beta^{(k)})^T r(\beta^{(k)}) \quad (13)$$

On linéarise les résidus autour du point  $\beta^{(k)}$  et ainsi de se ramener à un problème aux moindres carrés linéaire. Posons  $s = \beta - \beta^{(k)}$ , on cherche donc à chaque itération à résoudre

$$(P_k) \begin{cases} \text{Min } f_k(s) = \frac{1}{2} \|r(\beta^{(k)}) + J_r(\beta^{(k)})s\|^2 \\ s \in \mathbb{R}^p, \end{cases}$$

ce qui est équivalent à résoudre les équations normales

$$J_r(\beta^{(k)})^T J_r(\beta^{(k)})s + J_r(\beta^{(k)})^T r(\beta^{(k)}) = 0. \quad (14)$$

Donc, si  $J_r(\beta^{(k)})^T J_r(\beta^{(k)})$  est inversible, on peut écrire l'itération de Gauß-Newton :

$$\beta^{(k+1)} = \beta^{(k)} - [J_r(\beta^{(k)})^T J_r(\beta^{(k)})]^{-1} J_r(\beta^{(k)})^T r(\beta^{(k)}). \quad (15)$$

### Remarque 6.5.1.

- i) La différence entre les deux algorithmes réside dans l'absence du terme  $S(\beta)$  dans l'équation (13), terme qui contient les matrices hessiennes des résidus.
- ii) Il y a deux avantages à l'algorithme de Gauß-Newton par rapport à l'algorithme de Newton :
  - On n'a pas besoin de calculer les matrices hessiennes des résidus ;
  - Contrairement à l'algorithme de Newton, on peut toujours trouver une solution  $\beta^{(k)}$  à  $(P_k)$ .

**Exemple 6.5.1.** Considérons le problème suivant

$$(P) \begin{cases} \text{Min } f(x) = \frac{1}{2}((x_1^2 - x_2)^2 + (1 - x_1)^2) \\ x \in \mathbb{R}^2 \end{cases}$$

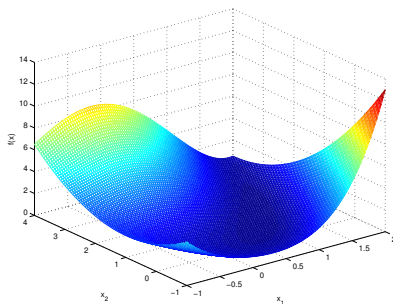


FIGURE 2 – Fonction  $f$  à minimiser pour l'exemple 6.5.1.



La figure 3 donne les courbes de niveaux de la fonction à minimiser ainsi que les itérés successifs

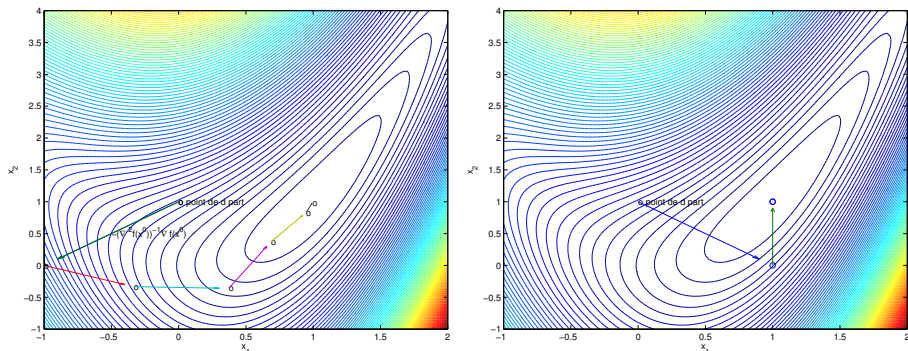


FIGURE 3 – *Algorithme de Newton à gauche et de Gauß-Newton à droite pour l'exemple 6.5.1.*