Chapitre 2

I

Résolution des équations d'optimalités

II

III

Conditions d'optimalité pour une optimisation sans contrainte

Résolution numérique d'équations non linéaires

I) Conditions d'optimalité pour une optimisation sans contrainte

A) Optimalité locale

Soit le problème d'optimisation sans contrainte $\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$ avec $f:\mathbb{R}^n \to \mathbb{R}$

Soit **x*** un minimum local

2 fois différentiable dans un voisinage V de x*

Conditions nécessaires d'optimalité locale

Condition du **premier ordre** ⇒ $\nabla f(\mathbf{x}^*)$ est nul

Les vecteurs **x*** sont alors appelés **points critiques** ou **points stationnaires** de f:

- **x*** ⇔ minimum local
- **x*** ⇔ maximum local
- **x*** ⇔ **point selle**

remarque *Cette condition joue un rôle central en optimisation*

Condition du **second ordre** ⇒ $\nabla^2 f(\mathbf{x}^*)$ semi définie positive

remarque *Condition difficile à vérifier systématiquement car nécessite le calcul fastidieux des dérivées secondes*

I) Conditions d'optimalité pour une optimisation sans contrainte

A) Optimalité locale

Conditions suffisantes d'optimalité locale

remarque *Cette condition exclut l'existence de point selle.*

x* est minimum local $\quad \nabla f(\mathbf{x}^*) = 0$

$\nabla^2 f(\mathbf{x}^*)$ définie positive

I

II

III

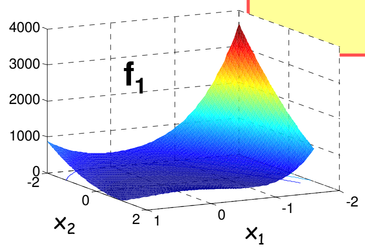
Considérons les 3 fonctions objectifs suivantes

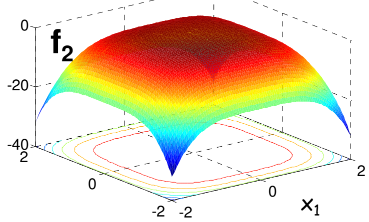
$f_1(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ au point $(1 \quad 1)^T$

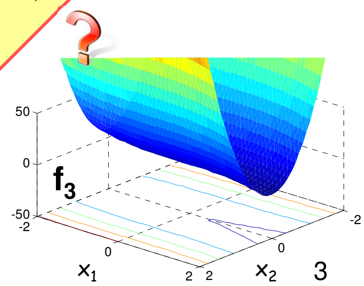
$f_2(x_1, x_2) = -x_1^4 - x_2^4$ au point $(0 \quad 0)^T$

$f_3(x_1, x_2) = 50x_1^2 - x_2^3$ au point $(0 \quad 0)^T$

De quel type de point s'agit-il ?







B) Optimalité globale

Soit une fonction continue $f : \mathbb{R}^n \to \mathbb{R}$ et soit $\in \mathbb{R}^n$ un minimum local de f.

Si f est une fonction convexe, alors
x* est un minimum global de f

Si, de plus, f est strictement convexe, alors
x* est l'unique minimum global de f

I

II

III

C) Cas des fonctions quadratiques

Considérons le problème d'optimisation suivant : $\min_{x \in \mathbb{R}^n} f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T Q \mathbf{x} + \mathbf{b}^T \mathbf{x} + c$

$\mathbf{b} \in \mathbb{R}^n,\ c \in \mathbb{R}$ Q matrice symétrique n x n

Valeur de c ⇨ Aucun impact sur la solution

$$\min_{x \in \mathbb{R}^n} f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T Q \mathbf{x} + \mathbf{b}^T \mathbf{x}$$

- Q non semi définie positive
  - ➡ Pas de solution
- Q définie positive
  - ➡ $\mathbf{x}^* = -Q^{-1}\mathbf{b}$ est l'unique solution

remarque *Le cas où Q est semi définie positive mais singulière est complexe et non traité ici.*

II) Résolution numérique d'équations non linéaires

Base des algorithmes d'optimisation

Dans la suite, problèmes à une inconnue ou plusieurs $f(x)=0$ pour $x \in \mathbb{R}$, $f(\mathbf{x})=0$ pour $\mathbf{x} \in \mathbb{R}^n$

A) Rappel sur les développements de Taylor

f fonction d'une variable

$$\exists \alpha \text{ tq } f(x+d)=f(x)+\frac{d}{1!}\frac{\partial f}{\partial x}(x)+\frac{d^2}{2!}\frac{\partial^2 f}{\partial x^2}(x)+\cdots+\frac{d^n}{n!}\frac{\partial^n f}{\partial x^n}(x)+\frac{d^{n+1}}{(n+1)!}\frac{\partial^{n+1} f}{\partial x^{n+1}}(x+\alpha d)$$

Soit une fonction $f:\mathbb{R}^n \to \mathbb{R}$ continûment différentiable sur une sphère ouverte S centrée en $\mathbf{x}$

1er ordre

Pour tout **d** tq $\mathbf{x}+\mathbf{d} \in S$

$$f(\mathbf{x}+\mathbf{d})=f(\mathbf{x})+\mathbf{d}^T\nabla f(\mathbf{x})+o(\|\mathbf{d}\|)$$

$$\exists \alpha \in [0,1] \quad \text{tq} \quad f(\mathbf{x}+\mathbf{d})=f(\mathbf{x})+\mathbf{d}^T\nabla f(\mathbf{x}+\alpha\mathbf{d})$$

remarque

*Cette expression est connue sous le nom de **théorème de la moyenne***

2ième ordre

Pour tout **d** tq $\mathbf{x}+\mathbf{d} \in S$

$$f(\mathbf{x}+\mathbf{d})=f(\mathbf{x})+\mathbf{d}^T\nabla f(\mathbf{x})+\frac{1}{2}\mathbf{d}^T\nabla^2 f(\mathbf{x})\mathbf{d}+o(\|\mathbf{d}^2\|)$$

$$\exists \alpha \in [0,1] \quad \text{tq} \quad f(\mathbf{x}+\mathbf{d})=f(\mathbf{x})+\mathbf{d}^T\nabla f(\mathbf{x})+\frac{1}{2}\mathbf{d}^T\nabla^2 f(\mathbf{x}+\alpha\mathbf{d})\mathbf{d}$$

II) Résolution numérique d'équations non linéaires

B) Méthode de Newton

Equation non linéaire
Résolution complexe !!!

Transformation

Equation linéaire
Résolution simple

Modèle linéaire
(linéarisation de f)

Application de la formule de Taylor sur une fonction non linéaire

Fonction à une variable

Soit une fonction $f:\mathbb{R} \to \mathbb{R}$ différentiable

Modèle linéaire de f en $\hat{x}$ est noté $m_{\hat{x}}(x):\mathbb{R} \to \mathbb{R}$ défini par :

$$m_{\hat{x}}(x)=f(\hat{x})+(x-\hat{x})f'(\hat{x})$$

Fonction à plusieurs variables

Soit une fonction $f:\mathbb{R}^n \to \mathbb{R}^m$ continûment différentiable

Modèle linéaire de f en $\hat{\mathbf{x}}$ est noté $m_{\hat{x}}(\mathbf{x}):\mathbb{R}^n \to \mathbb{R}^m$ défini par :

$$m_{\hat{x}}(\mathbf{x})=f(\hat{\mathbf{x}})+\nabla f(\hat{\mathbf{x}})^T(\mathbf{x}-\hat{\mathbf{x}})=f(\hat{\mathbf{x}})+J(\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})$$

I
II
III

Algorithme de Newton (1 variable)

Trouver une approximation de la solution de l'équation $f(x)=0$

Principe de l'algorithme

- Départ : Donner une approximation de la solution $x_0 = \hat{x}$
- Calculer le modèle linéaire en $\hat{x}$ ⇨ $m_{\hat{x}}(x) = f(\hat{x}) + (x-\hat{x})f'(\hat{x})$
- Calculer la racine x^+ de ce modèle linéaire ⇨ $f(\hat{x}) + (x-\hat{x})f'(\hat{x}) = 0$ ⇨ $x^+ = \hat{x} - \dfrac{f(\hat{x})}{f'(\hat{x})}$
- Si x^+ n'est pas solution alors considérer x^+ comme une nouvelle approximation et recommencer

remarque

Il est courant que la méthode ne génère pas de solution stricte telle que $f(x^+)=0$ *et on arrête généralement l'algorithme lorsque l'utilisateur estime que la solution est « suffisamment proche » soit lorsque* $|f(x^+)| \le \varepsilon$

I
II
III

Exemple 1 $f(x) = x^2 - 2 = 0$ ⇨ $f'(x) = 2x$

Posons : $x_0 = 2$ et $\varepsilon = 10^{-15}$

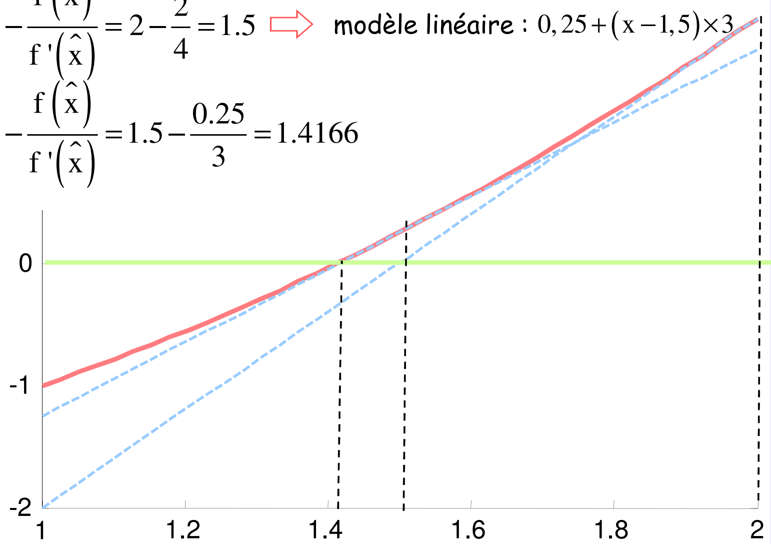
- Itération k=0 $x_0^+ = 2$ ⇨ modèle linéaire : $2 + (x-2)\times 4$
- Itération k= 1 $x_1^+ = \hat{x} - \dfrac{f(\hat{x})}{f'(\hat{x})} = 2 - \dfrac{2}{4} = 1.5$ ⇨ modèle linéaire : $0,25 + (x-1,5)\times 3$
- Itération k= 2 $x_2^+ = \hat{x} - \dfrac{f(\hat{x})}{f'(\hat{x})} = 1.5 - \dfrac{0.25}{3} = 1.4166$

Convergence au bout de 5 itérations

$x_5^+ = 1.41421356$

$f(x_5^+) = 4,441\times 10^{-16}$

$f'(x_5^+) = 2,828427$



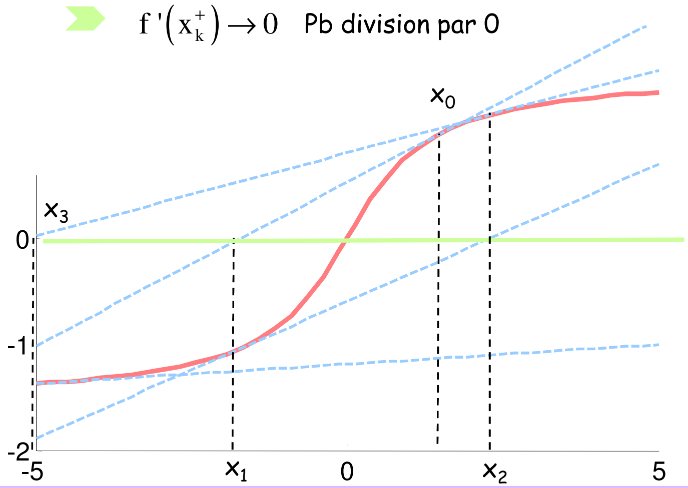
B) Méthode de Newton

Exemple 2 $f(x) = \arctan(x) = 0 \Rightarrow f'(x) = \dfrac{1}{1+x^2}$

Posons : $x_0 = 1{,}5$ et $\varepsilon = 10^{-15}$

k	x_k	$f(x_k)$	$f'(x_k)$
0	1,50	0,98	0,31
1	-1,69	-1,03	0,26
2	2,32	1,16	0,15
3	-5,11	-1,37	3,7E-2
4	32,29	1,54	9,6E-4
5	-1,57E3	-1,57	4,03E-7
6	8,89E6	1,57	6,6E-14
7	-2,38E13	-1,57	1,76E-27
…			

- $|x_k^+|$ augmente à chaque itération
- $f(x_k^+)$ semble osciller
- $f'(x_k^+) \to 0$ Pb division par 0



Pas de convergence

B) Méthode de Newton

Convergence 2 théorèmes …

Erreur du modèle linéaire

Soit un intervalle ouvert $X \subseteq \mathbb{R}$ et une fonction f dont la dérivée est continue au sens de Lipschitz (constante M) alors pour tout $\hat{x}, x^+ \in X$ on a :

$$\left|f(x^+) - m_{\hat{x}}(x^+)\right| \leq M \frac{(x^+ - \hat{x})^2}{2}$$

L'erreur commise par le modèle **dépend du degré de linéarité de f** : si f est presque linéaire (M faible) l'erreur commise est faible sinon l'erreur est plus importante

Convergence de la méthode de Newton

De plus supposons qu'il existe $\rho > 0$ tel que $|f'(x)| \geq \rho$ et $x^* \in X$ tel que $f(x^*) = 0$ alors il existe $\eta > 0$ tel que si $|x_0 - x^*| < \eta$ (avec $x_0 \in X$)

la suite définie par $x_{k+1} = x_k - \dfrac{f(x_k)}{f'(x_k)}$ converge vers x*

de la façon suivante : $|x_{k+1} - x^*| \leq \dfrac{M}{2\rho}|x_k - x^*|^2$

…ce qu'il faut retenir de ces théorèmes

- Si la fonction n'est pas trop non linéaire
  Continuité de Lipschitz ⇒ Plus M est petit plus la fonction est linéaire

- Si la dérivée de f n'est pas trop proche de 0
  Hypothèse $|f'(x)| \geq \rho$ ⇒ sinon division par 0 ou convergence lente

- Si l'approximation initiale x_0 n'est pas trop éloignée de la racine x*
  Hypothèse $|x_0 - x^*| < \eta$ ⇒ Plus la fonction est linéaire, plus M est petit et donc plus
  $\eta \propto 1/M$ la méthode pourra démarrer de x_0 éloigné de x*

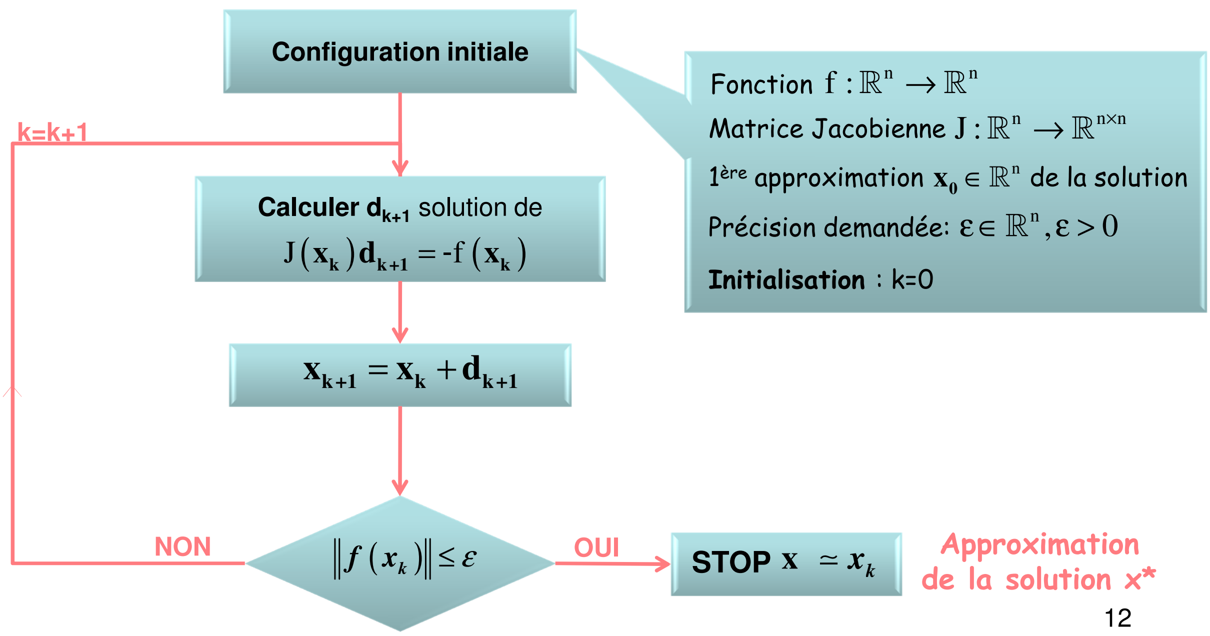
alors la méthode de Newton **converge très vite vers la solution**

vitesse ⇒ à chaque itération la nouvelle distance à la solution est de l'ordre de grandeur du carré de l'ancienne

⇔ **Convergence quadratique**

Algorithme de Newton (n variables)
Généralisation des résultats précédents

Trouver une approximation de la solution de l'équation $f(\mathbf{x}) = 0$



Exemple 1

Résoudre le système d'équations suivant: $\begin{cases} (x_1+1)^2 + x_2^2 = 2 \\ e^{x_1} + x_2^3 = 2 \end{cases}$

Posons : $\mathbf{x_0} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ et $\varepsilon = 10^{-15}$

Définissons: $f(\mathbf{x}) = \begin{pmatrix} (x_1+1)^2 + x_2^2 - 2 \\ e^{x_1} + x_2^3 - 2 \end{pmatrix}$ ⇨ $J(\mathbf{x}) = \begin{pmatrix} 2(x_1+1) & 2x_2 \\ e^{x_1} & 3x_2^2 \end{pmatrix}$

- k=0 ⇨ $f(\mathbf{x_0}) = \begin{pmatrix} 3 \\ e-1 \end{pmatrix} \approx \begin{pmatrix} 3 \\ 1,718 \end{pmatrix}$, $J(\mathbf{x_0}) = \begin{pmatrix} 4 & 2 \\ e & 3 \end{pmatrix}$ et $\|f(\mathbf{x_0})\| = \sqrt{3^2 + 1,718^2} \approx 3,45$

$$J(\mathbf{x_0})\mathbf{d_1} = \begin{pmatrix} 4 & 2 \\ e & 3 \end{pmatrix}\begin{pmatrix} d_1^1 \\ d_1^2 \end{pmatrix} = -f(\mathbf{x_0}) = -\begin{pmatrix} 3 \\ 1,7182 \end{pmatrix} \Rightarrow \begin{pmatrix} d_1^1 \\ d_1^2 \end{pmatrix} = \begin{pmatrix} 4 & 2 \\ e & 3 \end{pmatrix}^{-1}\begin{pmatrix} 3 \\ 1,7182 \end{pmatrix} = \begin{pmatrix} -0,847 \\ 0,195 \end{pmatrix}$$

- k suivant …

$$\Rightarrow \mathbf{x_1} = \mathbf{x_0} + \mathbf{d_1} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} -0,8477 \\ 0,1953 \end{pmatrix} = \begin{pmatrix} 0,1523 \\ 1,1953 \end{pmatrix}$$

k	x_1	x_2	$f_1(x_k)$	$f_2(x_k)$	$\|\|f(x_k)\|\|$
1	0,1523	1,1953	0,7566	0,8722	1,1547
2	-1,0837E-2	1,0361	5,1968E-2	1,0151E-1	1,14E-1
…	…	…	…	…	…
6	-1,5320E-16	1	-2,22E-16	0	2,22E-16

Exemple 2

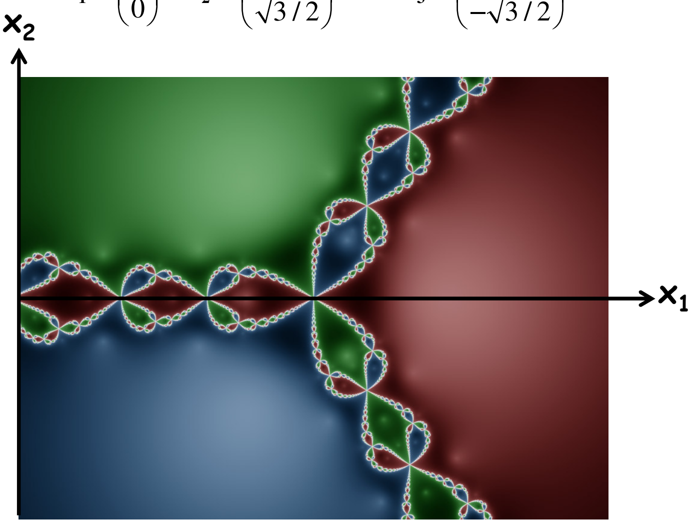
Fractale de Newton

Résoudre le système d'équations suivant: $\begin{pmatrix} x_1^3 - 3x_1x_2^2 - 1 \\ x_2^3 - 3x_1^2x_2 \end{pmatrix} = 0$

⇨ 3 solutions théoriques: $\mathbf{x}_1^* = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $\mathbf{x}_2^* = \begin{pmatrix} -1/2 \\ \sqrt{3}/2 \end{pmatrix}$, et $\mathbf{x}_3^* = \begin{pmatrix} -1/2 \\ -\sqrt{3}/2 \end{pmatrix}$

Utilisation de la méthode de Newton en partant d'un point $\mathbf{x_0}$:

- convergence vers $\mathbf{x}_1^*$
- convergence vers $\mathbf{x}_2^*$
- convergence vers $\mathbf{x}_3^*$



Frontières des bassins de convergence non définies de façon nette

Légère perturbation de $\mathbf{x_0}$ ⇔ Modification de la solution

⇨ Frontière de dimension **fractale**

II) Résolution numérique d'équations non linéaires

B) Méthode de Newton

Inconvénients de la méthode

I
II
III

Ne fonctionne pas à partir de n'importe quel $\mathbf{x_0}$
(hypothèse T10)

- Solution dépend du point de départ
- Si trop éloigné ⇨ Pas de convergence

Solution:
Méthodes globales

(Cf cours Fi-OptMath1)

Nécessite le calcul des dérivées à chaque itération
(algorithme T 12 par le calcul de J)

- Problème de l'existence de celle-ci
- Calcul lourd
- Pb si pas d'expression analytique de f
(expérience, exécution d'un logiciel etc ..)

Solution:
Méthodes quasi-Newton

II) Résolution numérique d'équations non linéaires

B) Méthode de quasi-Newton

Utilisation de la méthode de Newton sans le calcul des dérivées

Modèle linéaire sécant

I
II
III

Fonction à une variable

Soit une fonction $f : \mathbb{R} \rightarrow \mathbb{R}$ différentiable

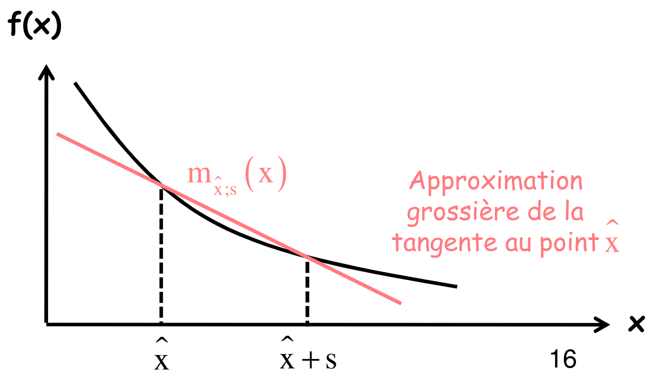
$$f'(x) = \lim_{s \to 0} \frac{f(x+s) - f(x)}{s} \quad \overset{\text{s petit}}{\Longleftrightarrow} \quad a_s = \frac{f(x+s) - f(x)}{s}$$

Approximation

Modèle linéaire sécant de f en $\hat{x}$ noté
$m_{\hat{x};s}(x) : \mathbb{R} \rightarrow \mathbb{R}$ et défini par :

$$m_{\hat{x};s}(x) = f(\hat{x}) + \frac{f(\hat{x}+s) - f(\hat{x})}{s}(x - \hat{x})$$

$$m_{\hat{x};s}(x) = f(\hat{x}) + a_s(x - \hat{x}) \quad \text{où} \quad s \neq 0$$



B) Méthode de quasi-Newton

I
II
III

Fonction à plusieurs variables

Soit une fonction $f:\mathbb{R}^n \to \mathbb{R}^m$ continûment différentiable

$$\nabla_i f(\mathbf{x}) = \lim_{s\to 0} \frac{\mathbf{f}(x_1,\dots x_i+s,\dots x_n)-\mathbf{f}(x_1,\dots x_i,\dots x_n)}{s}$$

s petit ⇕ **Approximation**

$$\frac{\mathbf{f}(x_1,\dots x_i+s,\dots x_n)-\mathbf{f}(x_1,\dots x_i,\dots x_n)}{s}$$

Modèle linéaire sécant de **f** en $\hat{\mathbf{x}}$ noté $m_{\hat{\mathbf{x}};A}(\mathbf{x}):\mathbb{R}^n \to \mathbb{R}^m$ et défini par :

$$m_{\hat{\mathbf{x}};A}(\mathbf{x}) = f(\hat{\mathbf{x}}) + A(\mathbf{x}-\hat{\mathbf{x}})$$

où $A \in \mathbb{R}^{n\times m}$ correspond à une approximation grossière de la matrice Jacobienne

B) Méthode de quasi-Newton

I
II
III

Algorithme de quasi-Newton (1 variable)

Trouver une approximation de la solution de l'équation $f(x)=0$

Même principe que la méthode de Newton avec

$$x^+ = \hat{x} - \frac{f(\hat{x})}{f'(\hat{x})} \iff x^+ = \hat{x} - \frac{f(\hat{x})}{a_s(\hat{x})}$$

Détermination de la valeur de s:

- Méthode de Newton **par différence finie**

$$s = \begin{cases} \tau\hat{x} & \text{si } |\hat{x}| \geq 1 \\ \tau & \text{sinon} \end{cases}$$

Par exemple $\tau = 10^{-7}$

remarque

τ petit ⇨ peu de différence avec la méthode Newton

τ grand ⇨ Convergence beaucoup plus lente

- Méthode de Newton **sécante**

But: économiser des évaluations supplémentaires de fonction f couteuses en temps de calcul

Choix de s tel que a_s s'écrive pour l'itération k

$$a_s = \frac{f(x_{k-1}) - f(x_k)}{x_{k-1} - x_k}$$

II) Résolution numérique d'équations non linéaires

B) Méthode de quasi-Newton

Algorithme de quasi-Newton (n variables)

Méthode de Newton par différence finie

Trouver une approximation de la solution de l'équation $f(\mathbf{x}) = \mathbf{0}$

Configuration initiale

Fonction $f : \mathbb{R}^n \to \mathbb{R}^n$
Paramètre $\tau > 0$
1ère approximation $\mathbf{x_0} \in \mathbb{R}^n$ de la solution
Précision demandée: $\varepsilon \in \mathbb{R},\ \varepsilon > 0$
Initialisation : k=0

Former $s \in \mathbb{R}^n$

$$s_j = \begin{cases} \tau (x_k)_j \text{ si } |(x_k)_j| \geq 1 \\ \tau \quad \text{si } 0 \leq (x_k)_j < 1 \\ -\tau \text{ si } -1 \leq (x_k)_j < 0 \end{cases} \qquad j = 1, 2, \ldots, n$$

Former $A_k \in \mathbb{R}^{n\times n}$ jème colonne

$$(A_k)_j = \frac{f(x_k + s_j e_j) - f(x_k)}{s_j}$$

avec $e_j = (0 \ \ldots \ 1 \ \ldots \ 0)$

Calculer $\mathbf{d_{k+1}}$ solution de $A_k \mathbf{d_{k+1}} = -f(\mathbf{x_k})$

$$\mathbf{x_{k+1}} = x_k + \mathbf{d_{k+1}}$$

$$\| f(x_k) \| \leq \varepsilon$$

NON : k=k+1

OUI : **STOP** $x^* \approx x_k$

Approximation de la solution x*

Fait appel à n+1 évaluations de f par itération ...

II) Résolution numérique d'équations non linéaires

B) Méthode de quasi-Newton

Méthode de Newton sécante

On impose au modèle linéaire d'interpoler la fonction en $\mathbf{x_k}$ et en $\mathbf{x_{k-1}}$ *(Cf T16)*

$m_{k;A}(\mathbf{x})$ tel que

$$\begin{cases} m_{k;A}(\mathbf{x_k}) = f(\mathbf{x_k}) \\ m_{k;A}(\mathbf{x_{k-1}}) = f(\mathbf{x_{k-1}}) = f(\mathbf{x_k}) + A_k(\mathbf{x_{k-1}} - \mathbf{x_k}) \end{cases}$$

Vérifié implicitement

Vérifié si

$$A_k(\mathbf{x_k} - \mathbf{x_{k-1}}) = f(\mathbf{x_k}) - f(\mathbf{x_{k-1}})$$

Equation sécante $A_k \mathbf{d_{k-1}} = \mathbf{y_{k-1}}$ **avec** $\mathbf{d_{k-1}} = \mathbf{x_k} - \mathbf{x_{k-1}}$, $\mathbf{y_{k-1}} = f(\mathbf{x_k}) - f(\mathbf{x_{k-1}})$

n^2 inconnues (élément de A)
n équations

Sous-déterminé

Infinité de solutions pour A si n>1

remarque *Géométriquement, il existe une infinité d'hyperplan qui passent par 2 points*

Choix de la matrice A_k qui vérifie l' équation sécante et qui est le plus proche du modèle établi à k-1

(Broyden 1965)

$$A_k = A_{k-1} + \frac{(y_{k-1} - A_{k-1} d_{k-1}) d_{k-1}^T}{d_{k-1}^T d_{k-1}}$$

B) Méthode de quasi-Newton

Configuration initiale

Fonction $f : \mathbb{R}^n \to \mathbb{R}^n$

1ère approximation $\mathbf{x}_0 \in \mathbb{R}^n$ de la solution

1ère approximation de A: $A_0 = I$

Précision demandée: $\varepsilon \in \mathbb{R},\ \varepsilon > 0$

Initialisation :

k=1

$\mathbf{x}_1 = \mathbf{x}_0 - A_0^{-1} f(\mathbf{x}_0)$

$\mathbf{y}_0 = f(\mathbf{x}_1) - f(\mathbf{x}_0)$

$\mathbf{d}_0 = \mathbf{x}_1 - \mathbf{x}_0$

Mise à jour de Broyden

$$A_k = A_{k-1} + \frac{(y_{k-1} - A_{k-1} d_{k-1}) d_{k-1}^T}{d_{k-1}^T d_{k-1}}$$

Calculer $\mathbf{d}_k$ solution de

$$A_k \mathbf{d}_k = -f(\mathbf{x}_k)$$

$$\mathbf{x}_{k+1} = x_k + d_k$$

$$y_k = \mathbf{f}(\mathbf{x}_{k+1}) - f(x_k)$$

$$\|f(x_k)\| \le \varepsilon$$

NON

k=k+1

OUI

STOP $\mathbf{x}^* \approx x_k$

Approximation de la solution x*

⇨ Ne converge pas forcement vers la même solution que la méthode de Newton ……

remarque

La matrice A ne doit pas nécessairement être une bonne approximation de la matrice Jacobienne pour que la méthode fonctionne bien et heureusement car c'est rarement le cas !!!

I

II

III