

RÉSOLUTION D'ÉQUATION NON LINÉAIRE

Le but est de décrire des algo pour résoudre des équation non linéaire de type $f(x) = 0$.

1 Séparation des zéros

Le premier travail consiste à déterminer des intervalles $[a_i, b_i]$ tel que f possède une solution et un seul dans chaque intervalle.

La méthode la plus simple est d'utiliser une fonction continue strictement monotone sur $[a_i, b_i]$ tel que $f(a_i)f(b_i) < 0$. (On suppose que f est continue et dérivable des fois)

Quelques algo classique

2 La méthode de dichotomie (ou bisection)

Supposons que l'on a un zéros dans un intervalle $[a, b]$ (ie $f(a)f(b) < 0$).

- Si $f(\frac{a+b}{2}) = 0$, on a trouve le zeros.
- Sinon le zeros se trouve dans $[a, \frac{a+b}{2}]$ soit dans $[\frac{a+b}{2}, b]$.

Il est clair qu'une répétition de ce procédé donne un encadrement de plus en plus précis du zéros cherché et fournit donc un algo de calcul du zéros.

Algo (dichotomie)

Soit $f : [a_0, b_0] \mapsto \mathbb{R}$ continue monotone tel que $f(a_0)f(b_0) < 0$.

Pour $m = 0, 1, 2, \dots, N$ faire :

$$m = \frac{a_n + b_n}{2}$$

- Si $f(a_n)f(m) \leq 0$, $a_{n+1} = a_n, b_{n+1} = m$
- Sinon $a_{n+1} = m, b_{n+1} = b_n$

On a : $a_{n+1} - b_{n+1} = \frac{a_n - b_n}{2}$

Soit $a_n - b_n = \frac{a_0 - b_0}{2^n} \rightarrow 0$

On peut choisir le temps d'arrêt N pour que : $\frac{a_0 - b_0}{2^N} < \varepsilon$

3 Méthode de la sécante

Soit f admettant un zéro dans l'intervalle $[x_{-1}, x_0]$. Pour obtenir une première approximation x_1 de ce zéro, l'idée est de remplacer f par son interpolée linéaire sur $[x_{-1}, x_0]$.

Soit par

$$Y(x) = f(x_0) + (x - x_0) \frac{f(x_0) - f(x_{-1})}{x_0 - x_{-1}}$$

L'approximation x_1 est obtenu en résolvant $Y(x_1) = 0$ ie

$$x_1 = x_0 - f(x_0) \frac{x_0 - x_{-1}}{f(x_0) - f(x_{-1})}$$

Pour trouver une meilleur approximation, il suffit de répéter ce procédé a l'aide des points (x_n, x_{n+1})

Algo

Pour $n = 0, 1, 2, \dots$

$$x_{n+1} = x_n - f(x_n) \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

3.1 Critère d'arrêt

Une critère d'arrêt souvent utilisée consiste à choisir une tolérance ε à terminer l'algo lorsque $|x_{n+1} - x_n| < \varepsilon$

4 Méthode de Newton

Ici au lieu d'assimiler la courbe $y = f(x)$ à une sécante, on l'assimile à une tangente en un point $(x_n, f(x_n))$, soit la droite d'équation

$$Y = f(x_0) + f'(x_n)(x - x_n)$$

Algo

Pour $n = 0, 1, 2, \dots$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

5 Méthode de point fixe

Elle consiste d'abord à remplacer l'équation $f(x) = 0$ par une équation $g(x) = x$ ayant même solution.

Algo

Pour $n=0,1,2,\dots$

$$x_{n+1} = g(x_n)$$

5.1 Proposition

Soit $g : [a, b] \rightarrow [a, b]$ une fonction continue et x_0 dans $[a, b]$.

Si x_n converge vers x_∞ alors $x_\infty = g(x_\infty)$.

5.2 Convergence des algo

Soit $g : [a, b] \mapsto [a, b]$ dérivable et tel que $|g'(x)| \leq K$, pour tout $x \in [a, b]$ avec $0 \leq K < 1$, alors pour tout $x_0 \in [a, b]$, la suite définie par $x_{n+1} = g(x_n)$, pour tout n converge vers l'unique point fixe de g .

5.3 Définition

Soient x_n une suite convergente vers x_∞ , s'il existe $p \in \mathbb{N}$ et $c \neq 0$ tel que

$$\lim_{n \rightarrow +\infty} \frac{|x_{n+1} - x_\infty|}{|x_n - x_\infty|^p} = c$$

On dit que la convergence est d'ordre p , " c " est appelée constante d'erreur asymptotique.

5.4 Méthode de Newton : Le retour

D'après la formule de Taylor d'ordre 2 en supposant g suffisamment régulière

$$x_{n+1} - x_\infty = g(x_n) - g(x_\infty)$$

$$x_{n+1} - x_\infty = g(x_n - x_\infty)g'(x_\infty) + \frac{(x_n - x_\infty)^2 g''(\varepsilon_n)}{2}$$

avec $\varepsilon_n \in [x_n, x_\infty]$ Pour n assez grand on a donc $x_{n+1} - x_\infty \simeq (x_n - x_\infty)g'(x_\infty)$ et vitesse de convergence est d'autant plus grande que $g'(x_\infty)$ est plus petit.

Le cas le plus favorable est lorsque $g'(x_\infty) = 0$ et si M est un majorant de $g''(x)$ sur $[a, b]$.

On a $|x_{n+1} - x_\infty| \leq \frac{M|x_n - x_\infty|^2}{2}$

La convergence est alors d'ordre 2. Si on revient à la méthode de Newton, on voit en fait qu'il s'agit d'un algo du point fixe pour la fonction

$$g(x) = x - \frac{f(x)}{f'(x)}.$$

La dérivée de g est donnée par $g'(x) = \frac{f(x)f''(x)}{(f'(x))^2}$.

Si $f'(x_\infty) \neq 0$ on a $g'(x_\infty) = 0$ car $f(x_\infty) = 0$.

Ceci montre que la méthode de Newton converge de façon quadratique (si elle converge).

5.5 Méthode de point fixe : Le retour

Les résultats de ce paragraphe sont une généralisation en dimension 1. La fonction g à cette fois une fonction $g : \mathbb{R}^n \mapsto \mathbb{R}^n$ (ici on prend $n = 2$) et la variable $x \in \mathbb{R}^2$.

Comme précédemment on construit une suite définie par

$$\begin{cases} x^0 \\ x^{k+1} = g(x^k) \end{cases}$$

pour x^0 donnée. Quand tout ce passe bien cette suite converge vers \hat{x} vérifiant $\hat{x} = g(\hat{x})$. On a donc besoin de généraliser la notion de fonction contractante.

5.5.1 Définition

Soit $E \subset \mathbb{R}^n$, la fonction $g : \mathbb{R}^n \mapsto \mathbb{R}^n$ est dite contractante sur E , $\exists \lambda \in [0, 1[$ tel que pour tout $x, y \in E$, on a $\|g(x) - g(y)\| \leq \lambda \|x - y\|$

5.5.2 Théoreme (point fixe)

Soit $g : \mathbb{R}^n \mapsto \mathbb{R}^n$ contractante, alors il existe un unique $\hat{x} \in \mathbb{R}^n$ tel que $\hat{x} = g(\hat{x})$ et la suite (vecteurs) définie par

$$\begin{cases} x^0 \\ x^{k+1} = g(x^k) \end{cases}$$

pour x^0 donnée et $k \geq 0$ converge vers \hat{x}

5.5.3 Définition

Une fonction $g : \mathbb{R}^n \mapsto \mathbb{R}^n$ est différentiable en x si g admet des dérivées partielles en x et si la matrice Jacobienne $D_g(x)$ vérifie :

$$g(y) = g(x) + D_g(x)(y - x) + \|y - x\| \varepsilon(y - x)$$

où $\varepsilon(y - x) \rightarrow 0$ quand $y \rightarrow x$.

5.5.4 Méthode de Newton pour 2 équation non linéaire

On cherche à résoudre le système d'équation $f(x) = 0$. Comme pour le cas $n = 1$, l'idée de la méthode de Newton consiste à considérer l'approximation affine de f en x^k .

Si f est différentiable, le développement de Taylor donne :

$$f(x^k + h) = f(x^k) + D_f(x^k)h + \|h\| \varepsilon(h)$$

On détermine le vecteur h tel que

$$\begin{aligned} \left\| \begin{array}{l} f(x^k) + D_f(x^k)h \\ x^{k+1} \end{array} \right. &= 0 \\ &= x^k + h \end{aligned}$$