

MÉTHODE DE NEWTON

La recherche des zéros d'une fonction f de \mathbb{R} dans \mathbb{R} est un problème classique en mathématiques. Lorsque la résolution exacte de l'équation $f(x) = 0$ n'est pas possible, il faut envisager des méthodes d'analyse numérique permettant d'approcher les solutions. La méthode la plus élémentaire, que nous avons déjà rencontrée dans le chapitre sur la continuité, est la dichotomie. Dans cette annexe, nous décrivons une méthode alternative, dite de Newton–Raphson, que l'on peut présenter comme un procédé d'approximation des solutions fondé sur le principe de linéarisation.

a. Description de l'algorithme

Soient $a, b \in \mathbb{R}$ tels que $a < b$. On considère une fonction $f : [a; b] \rightarrow \mathbb{R}$ de classe \mathcal{C}^1 dont la dérivée ne s'annule pas et dont les valeurs en a et b sont de signes opposés (i.e. $f(a)f(b) < 0$).

Sous ces conditions, le théorème de la bijection assure que f admet un unique zéro α dans l'intervalle ouvert $]a; b[$.

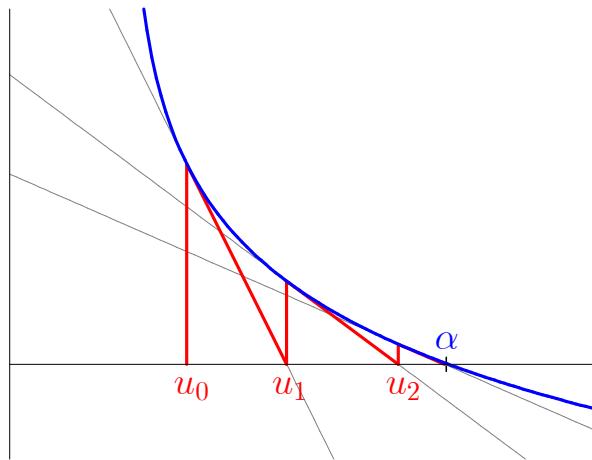
Pour estimer α , on part d'une valeur $u_0 \in [a; b]$, si possible assez proche de α .

La fonction f étant de classe \mathcal{C}^1 , on sait, qu'en tout point, la courbe de f ressemble à sa tangente. D'où l'idée de linéariser la fonction f , c'est-à-dire de confondre f et la fonction affine dont le graphe est la tangente à f en u_0 . En recherchant le point d'annulation u_1 de cette fonction affine, on peut espérer se rapprocher de α .

À partir de u_1 , on peut recommencer le processus en trouvant le point d'annulation u_2 de la fonction affine dont le graphe est la tangente à f en u_1 .

Etc.

Graphiquement, à chaque étape, on recherche le point d'intersection entre l'axe des abscisses et la tangente à \mathcal{C}_f au point de coordonnées $(u_n, f(u_n))$. Cela donne la position de u_{n+1} .



Pour tout $n \geq 0$, l'équation de la tangente en $(u_n, f(u_n))$ est $y = f(u_n) + f'(u_n)(x - u_n)$, donc

$$\forall n \geq 0, \quad u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)}.$$

B. Convergence de l'algorithme

L'intérêt principal de l'algorithme de Newton est sa convergence quadratique locale, qui est démontrée (sous des hypothèses raisonnables) dans l'énoncé suivant. En termes imaginés mais peu précis, cela signifie que le nombre de chiffres significatifs corrects des itérés double à chaque étape.

Proposition 1

Soit $f : [a; b] \rightarrow \mathbb{R}$ une fonction de classe \mathcal{C}^2 telle que $f(a) < 0$, $f(b) > 0$, $f' > 0$ et $f'' \geq 0$. Alors

- (i) la fonction f admet un unique zéro, noté α , dans $]a; b[$;
- (ii) pour tout $u_0 \in [\alpha; b]$, la suite récurrente $(u_n)_{n \geq 0}$ satisfaisant, pour tout $n \geq 0$, la relation $u_{n+1} = u_n - f(u_n)/f'(u_n)$ converge vers α ;
- (iii) il existe $k \in \mathbb{R}_+^*$ tel que $\forall n \geq 0$, $0 \leq u_{n+1} - \alpha \leq k(u_n - \alpha)^2$, ce qui implique que $\forall n \geq 0$, $0 \leq u_n - \alpha \leq (k(u_0 - \alpha))^{2^n}/k$.

■ (i) On a déjà dit que c'est une conséquence immédiate du théorème de la bijection.

- (ii) La fonction $g : x \mapsto x - f(x)/f'(x)$ est de classe \mathcal{C}^1 sur $[a; b]$ et $\forall x \in [a; b]$, $g'(x) = f(x)f''(x)/f'(x)^2$, ce qui démontre que $\forall x \in [\alpha; b]$, $g'(x) \geq 0$ et donc que g est croissante sur $[\alpha; b]$. Cela permet de démontrer, par récurrence (immédiate), que $\forall n \geq 0$, $\alpha \leq u_{n+1} \leq u_n \leq b$, ce qui prouve, au passage, que la suite $(u_n)_{n \geq 0}$ existe bel et bien. Par ailleurs, cela démontre que $(u_n)_{n \geq 0}$ est convergente, vers $\ell \in [\alpha; b]$, en tant que suite décroissante minorée. En passant à la limite dans la relation de récurrence $\forall n \geq 0$, $u_{n+1} = u_n - f(u_n)/f'(u_n)$, on obtient $\ell = \ell - f(\ell)/f'(\ell)$, c'est-à-dire $f(\ell) = 0$ ou encore $\ell = \alpha$ puisque α est le seul point d'annulation de f sur $[a; b]$.
- (iii) Pour tout $n \in \mathbb{N}$, on applique la formule de Taylor–Lagrange à f entre α et u_n , ce qui donne l'existence de $c \in]\alpha; u_n[$ tel que

$$\forall n \geq 0, \quad f(\alpha) - f(u_n) - f'(u_n)(\alpha - u_n) = \frac{f''(c)}{2}(u_n - \alpha)^2.$$

En tenant compte du fait que $f(\alpha) = 0$ et en divisant par $f'(u_n)$, on obtient

$$\forall n \geq 0, \quad u_{n+1} - \alpha = \frac{f''(c)}{2f'(u_n)}(u_n - \alpha)^2,$$

ce qui donne

$$\forall n \geq 0, \quad u_{n+1} - \alpha \leq \frac{M_2}{2m_1}(u_n - \alpha)^2,$$

où $m_1 = \min_{[a; b]}(f') > 0$ et $M_2 = \max_{[a; b]}(f'') > 0$ (je vous laisse réfléchir au cas $M_2 = 0$). C'est le résultat attendu avec $k = M_2/(2m_1)$ puisque $\forall n \geq 0$, $u_n - \alpha \geq 0$.

La dernière inégalité s'obtient par une récurrence immédiate. ■

En choisissant u_0 assez proche de α (c'est-à-dire tel que $k(u_0 - \alpha) < 1$), la convergence de u_n vers α est bien quadratique.

C. Un exemple : l'algorithme de Babylone

Un cas particulier de la méthode de Newton est l'algorithme de Babylone, aussi connu sous le nom de méthode de Héron : il s'agit, pour calculer la racine carrée de $a \in \mathbb{R}_+$, d'appliquer la méthode de Newton à la fonction $f(x) = x^2 - a$. On obtient alors une méthode d'approximation de la solution \sqrt{a} donnée par la formule itérative suivante :

$$\forall n \geq 0, \quad u_{n+1} = \frac{1}{2} \left(u_n + \frac{a}{u_n} \right).$$

Cet algorithme est convergent pour tout $u_0 > 0$.

Pour $a = 2$ et $u_0 = 2$, le terme u_6 a déjà ses 48 premières décimales qui coïncident avec $\sqrt{2}!!$