Méthodes de type point fixe

Exercice 1. Calculer les points fixes x^* des fonctions suivantes :

1. $g(x) = (x^2 - 1)/3, x \in [-1, 1]$.
2. $g(x) = (x^2 - 1)/3, x \in [3, 4]$.

Dans chaque cas étudier $|g'(x^*)|$ et décider si la suite des itérés successifs converge ou non.

Exercice 2. L'objectif de cet exercice est d'analyser et de programmer la méthode de point fixe pour résoudre une équation du type $g(x) = x$, avec $g : [a, b] \to \mathbb{R}$.

Sous certaines hypothèses sur la fonction g, la suite $(x_n)_n$, des itérés successifs, définie par

$$\begin{cases} x_0 & \text{donné} \\ x_{n+1} = g(x_n), & n \geq 0, \end{cases}$$

converge vers le point fixe x^*.

1. Soit $\varepsilon > 0$. On pose $e_n = x_n - x^*, n \geq 0$ (l'erreur à l'itération n).

(a) Calculer le nombre d'itérations de l'algorithme de point fixe nécessaire pour avoir

$$|e_n| \leq \varepsilon.$$

(b) En supposant que $g'(x^*) \neq 0$, montrer que

$$e_{n+1} \approx g'(x^*)e_n.$$

(c) En déduire que

$$e_n \approx (g'(x^*))^n e_0.$$

(d) Conclure que si l'on a $|g'(x^*)| < 1$, alors e_n tend vers 0.

(e) Analyser le cas où g est telle que $-1 < g'(x^*) < 0$.

(f) A l'aide de représentations graphiques, illustrer les cas $|g'(x)| > 1, 0 < g'(x) < 1$ et $-1 < g'(x) < 0$.

2. Implémenter la méthode en utilisant comme test d'arrêt : $|x_n - x_{n-1}|/|x_n| > \varepsilon$ ou $n \geq N_{max}$ (où ε la tolérance et N_{max} le nombre maximal d'itérations autorisé).

3. A l'aide du programme précédent résoudre $x^2 - 2x - 3 = 0$. Utiliser $x_0 = 4, \varepsilon = 10^{-8}$ et comme fonction g les fonctions suivantes :

(a) $g_1(x) = \sqrt{2x + 3}$;

(b) $g_2(x) = \frac{3}{x-2}$;

(c) $g_3(x) = \frac{1}{2}(x^2 - 3)$.

4. Adapter votre programme pour qu'il renvoie la suite des itérés x_n et représenter graphiquement x_n vs n. Vérifier numériquement le résultat de la question **1.**(a) et analyser la convergence.

5. Résoudre $g(x) = x$ avec $g(x) = \sqrt{1+x}$ et $x_0 = 1.5$. Une fois trouver le point fixe x^*, et à l'aide de la suite des itérés, calculer, pour $n \geq 0$, $|e_n|$ et $|e_{n+1}/e_n|$. Quel est l'ordre de la méthode ? Donner une estimation de la vitesse de convergence.

   Indication : Une méthode converge à l'ordre p si $|e_{n+1}| \leq C|e_n|^p$, avec $C > 0$.

6. Adapter votre programme pour résoudre par la méthode de Newton $f(x) = 0$, avec $f(x) = x^2 - 2$. Quel est l'ordre de la méthode ? Donner une estimation de la vitesse de convergence.

Exercice 3. Soient $I \subset \mathbb{R}$ un intervalle et $f : I \to \mathbb{R}$ une fonction C^∞. On dit que f a une racine de multiplicité $m \in \mathbb{N}$ si f admet la factorisation suivante :

$$f(x) = (x - x^*)^m h(x),$$

avec h une fonction telle que $h(x^*) \neq 0$.

1. Montrer que x^* est une racine de multiplicité m si et seulement si

$$f^{(k)}(x^*) = 0, \quad 0 \leq k \leq m-1, \quad f^{(m)}(x^*) \neq 0.$$

   Indication : utiliser la formule de Leibniz.

2. Montrer que la fonction $f(x) = x \sin x, x \in \mathbb{R}$, possède une racine de multiplicité 2 en $x = 0$.

3. Montrer que dans le cas d'une racine de multiplicité $m > 1$, la méthode de Newton appliquée à cette fonction ne converge pas de façon quadratique.

   Indication : Utiliser la factorisation de f pour calculer $g'(x)$, avec $g(x) = x - f(x)/f'(x)$.

4. Soit $(x_n)_{n\geq 0}$ la suite définie par

$$x_n = x_{n-1} - m\frac{f(x_{n-1})}{f'(x_{n-1})}, \quad n \geq 1.$$

   Vérifier que cette suite converge vers une racine de f. Montrer que la convergence est quadratique.

Exercice 4. La méthode de Newton est généralisable au cas d'une fonction $F : \mathbb{R}^k \to \mathbb{R}^k$. La suite des itérés successifs est définie par

$$\begin{cases} \boldsymbol{x}_0 \in \mathbb{R}^n & \text{donné} \\ \boldsymbol{x}_{n+1} = \boldsymbol{x}_n - J_F(\boldsymbol{x}_n)^{-1}F(\boldsymbol{x}_n), & n \geq 0, \end{cases}$$

où $J_F(\boldsymbol{x}_n)$ est la matrice $(k \times k)$ jacobienne de F en $\boldsymbol{x}_n$.

Dans la pratique, le calcul du vecteur $J_F(\boldsymbol{x}_n)^{-1}F(\boldsymbol{x}_n)$ est très couteux si k est grand (calcul de l'inverse de la matrice jacobienne à chaque itération), il est donc généralement remplacé par la résolution du système linéaire $J_F(\boldsymbol{x}_n)\boldsymbol{z}_n = F(\boldsymbol{x}_n)$ ($\boldsymbol{z}_n = J_F(\boldsymbol{x}_n)^{-1}F(\boldsymbol{x}_n)$).

Ainsi, l'algorithme de la méthode de Newton pour un système linéaire devient

$$\begin{array}{l} \boldsymbol{x}_0 \text{ donné} \\ \text{Tant que } (||\boldsymbol{z}_n||_2 > \varepsilon \quad | \quad n \leq N_{max}) \\ \qquad \text{résoudre } J_F(\boldsymbol{x}_n)\boldsymbol{z}_n = F(\boldsymbol{x}_n) \\ \qquad \boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \boldsymbol{z}_n \end{array}$$

1. Programmer l'algorithme en utilisant la fonction `solve` pour la résolution du système.

2. Tester votre algorithme pour la résolution du système non-linéaire suivant :

$$\begin{cases} 3x_1^2 - x_2^2 = 0 \\ 3x_1 x_2 - x_1^3 = 1 \end{cases} \tag{1}$$

Commencer par identifier la fonction $F : \mathbb{R}^2 \to \mathbb{R}^2$ telle que telle que (1) soit équivalent à $F(\boldsymbol{x}) = \mathbf{0}$ puis calculer $J_F(x_1, x_2)$, i.e.

$$[J_F(x_1, x_2)]_{ij} = \frac{\partial f_i}{\partial x_j}, \quad 1 \leq i, j \leq 2.$$

Utiliser comme point initial $\boldsymbol{x}_0 = (1, 1)^T$ et une tolérance $\varepsilon = 10^{-8}$.