Série 1 (Corrigé)

Semaine prochaine, Pour ce vendredi : apprenez les vidéos des sections 1.2-1.3.

Mardi aux exercices : Parcourez les sections 1.2-1.3 des notebooks Jupyter et resolvez les exercices qui y sont proposés.

Partiellement en classe

Exercice 1

a) Trouver une fonction $f(x)$ correspondant à chacune des descriptions suivantes :

1. $f(x)$ est discontinue en 0.
2. $f(x)$ est discontinue en 0 mais contient un zero en x=2.
3. $f(x)$ est strictement positive.
4. $f(x)$ est positive ou nulle $\forall x \in \mathbb{R}$.
5. $f(x)$ a au moins deux zéros.
6. $f(x)$ change de signe mais ne passe pas par zero.

b) Pour quelle(s) fonction(s) du point a) peut-on espérer trouver un zéro avec la méthode de bissection.

c) Montrer que l'erreur commise avec la méthode de la bissection est bornée comme suit :

$$|e^{(k)}| := |x^{(k)} - \alpha| < \frac{b-a}{2^{k+1}}. \tag{1}$$

d) Considérons la fonction $sin(x)$ dans l'intervalle $[\pi/2, 3\pi/2]$. Combien d'iterations sont-elles nécessaires pour trouver un zero avec une tolérance de $\varepsilon = 10^{-7}$.

e) Si dans l'algorithme on divise l'intervalle en deux parties de tailles non égales, quel en sera l'effet ? Donner une borne théorique pour le pire cas en supposant que l'intervalle est divisé en utilisant 1/3 et 2/3 de l'intervalle original.

Sol. :

a) Voici des exemples possibles pour la fonction $f(x)$:

1. $f(x) = \frac{1}{x}$

2. $f(x) = 1 - \frac{2}{x}$

3. $f(x) = |x| + 1$

4. $f(x) = x^2$

5. $f(x) = (x-1)(x-2)$

6.

$$f(x) = \begin{cases} 1 & if\ x \geq 0 \\ -1 & if\ x < 0 \end{cases} \tag{2}$$

b) Par rapport aux fonctions ci dessous, a) c) d) f) ne peut pas utiliser la méthode de bissection. Pour b) et e) ça depends du choix de l'intervalle initial.

c) La preuve de la borne se fait par récurrence en observant que le zéro α de la fonction se trouve soit dans l'intervalle $[a, x^{(k)}]$, soit dans l'intervalle $[x^{(k)}, b]$.

d) Le point millieux de l'intervalle est π, qui est un zero α de la fonction. Donc il suffit de faire une iteration. La borne donnée en cours est la suivante

$$|e^{(k)}| := |x^{(k)} - \alpha| < \frac{b-a}{2^{k+1}}. \tag{3}$$

Pour garantir que $|e^{(k)}| < \varepsilon$, il suffit d'effectuer k_{min} iterations, où k_{min} est le plus petit entier tel que

$$k_{min} > log_2 \left(\frac{b-a}{\varepsilon}\right) - 1. \tag{4}$$

Dans ce cas précis, $k_{min} = 24$.

e) Cela va affecter la vitesse de convergence en fonction de la valeur du zero α : dans un cas favorable la taille de l'intervalle va décroitre plus vite et dans un cas défavorable plus lentement. Pour définir la borne théorique, on considère le cas où le zéro est toujours situé dans l'intervalle le plus grand, c'est à dire celui qui correspond au 2/3 de l'intervalle précédent. Ainsi la borne devient :

$$|e^{(k)}| := |x^{(k)} - \alpha| < \left(\frac{2}{3}\right)^{k+1} (b-a). \tag{5}$$

Exercice 2

Soit $f : [1, 3] \to \mathbb{R}$, $f(x) = x^3 - 2x - 5$. On observe que :

$$f(1) = -6 < 0 \quad \text{et} \quad f(3) = 16 > 0 \ .$$

On a sûrement au moins un zéro $x^* \in [1, 3]$.

1. Montrer l'unicité du zéro $x^* \in [1, 3]$.
2. Écrire la méthode de Newton pour la fonction f.
3. En interprétant cette méthode comme une méthode de point fixe, montrer qu'elle est d'ordre 2. *(cette partie sera a faire après avoir vu la section 1.4 des vidéos)*

Sol. :

1. L'unicité est garantie grâce au fait que $f(x)$ est strictement croissante dans $[1, 3]$, car sa dérivée première $f'(x) = 3x^2 - 2$ est $f'(x) > 0 \ \forall x \in [1, 3]$.

2. La méthode de Newton est donnée par

$$x^{(n+1)} = x^{(n)} - \frac{f(x^{(n)})}{f'(x^{(n)})} = x^{(n)} - \frac{(x^{(n)})^3 - 2x^{(n)} - 5}{3(x^{(n)})^2 - 2} \ .$$

3. On écrit la méthode de Newton comme la méthode de point fixe $x^{(n+1)} = g(x^{(n)})$ avec

$$g(x^{(n)}) = x^{(n)} - \frac{(x^{(n)})^3 - 2x^{(n)} - 5}{3(x^{(n)})^2 - 2} \ .$$

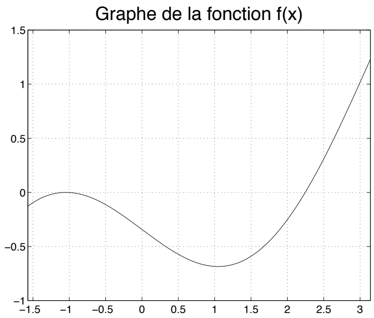
On a

$$g'(x) = \frac{6x\,(x^3 - 2x - 5)}{(3x^2 - 2)^2} \ ,$$

et clairement $g'(x^*) = 0$; la méthode est donc d'ordre 2.

Exercice 3

On veut calculer les zéros de l'équation $f(x) = \frac{x}{2} - \sin(x) + \frac{\pi}{6} - \frac{\sqrt{3}}{2} = 0$ dans l'intervalle $\left[-\frac{\pi}{2}, \pi\right]$. Le graphe de la fonction $f(x)$ est montré dans la figure suivante :



1. Peut-on appliquer la méthode de bissection pour calculer les deux racines ? Pourquoi ? Dans le cas où c'est possible, estimer le nombre minimal d'itérations nécessaires pour calculer le(s) zéro(s) avec une tolérance $tol = 10^{-10}$, après avoir choisi un intervalle convenable.
2. Écrire la méthode de Newton pour la fonction $f(x)$.
3. A l'aide du graphe de la fonction $f(x)$, déduire l'ordre de convergence de la méthode pour les deux zéros. *(cette partie sera a faire après avoir vu la section 1.4 des vidéos)*

Sol. :

1. *Grâce au graphe, on voit que la fonction $f(x)$ a un zéro $\alpha_1 \in \left(-\frac{\pi}{2}, 0\right)$ et un zéro $\alpha_2 \in \left(\frac{2\pi}{3}, \pi\right)$.*

   On peut appliquer la méthode de la bisection pour calculer le zéro $\alpha_2 \in \left(\frac{2\pi}{3}, \pi\right)$, mais on ne peut pas utiliser cette méthode pour calculer α_1, car la condition $f(a) \cdot f(b) < 0$ n'est pas satisfaite pour tout $a, b \in \left[-\frac{\pi}{2}, \alpha_2\right)$, $a < b$. On considère donc $\alpha_2 \in \left(\frac{2\pi}{3}, \pi\right)$. Pour estimer le nombre d'itérations, il suffit appliquer la formule

$$|e^{(k)}| \leq \frac{b-a}{2^{k+1}} ,$$

   avec $a = \frac{2\pi}{3}$ et $b = \pi$. Donc

$$k \geq \frac{\log_2\left(\frac{b-a}{tol}\right)}{\log_2 2} - 1 = \frac{\log_2 \frac{\pi}{3} + 10 \log_2 10}{1} - 1 = 32.286,$$

   et le nombre minimal d'itération pour satisfaire la tolérance est 33.

2. La méthode de Newton s'écrit :

$$x^{(k+1)} = x^{(k)} - \frac{\dfrac{x^{(k)}}{2} - \sin x^{(k)} + \dfrac{\pi}{6} - \dfrac{\sqrt{3}}{2}}{\dfrac{1}{2} - \cos x^{(k)}}.$$

Ordre de la convergence :

- *zéro* $\alpha_2 \in \left(\frac{2\pi}{3}, \pi\right)$ *: puisque* f *est deux fois différentiable,* $f(\alpha_2) = 0$ *et* $f'(\alpha_2) \neq 0$*, la convergence est quadratique ;*
- *zéro* $\alpha_1 \in \left(-\frac{\pi}{2}, 0\right)$ *: puisque* f *est deux fois différentiable,* $f(\alpha_1) = 0$*, mais* $f'(\alpha_1) = 0$ *(on voit sur le graphe que* α_1 *est un maximum local pour la fonction* f*), la convergence est seulement linéaire.*

Python

L'exercice suivant se trouve dans le notebook *1.2 Newton-Raphson*.

Exercice 4

On cherche les zéros de la fonction

$$f(x) = \frac{1}{2}\sin\left(\frac{\pi x}{2}\right) + 1 - x\ .$$

1. Vérifier qu'il y a au moins un zéro α dans l'intervalle $[0, 2]$.
2. Écrire la méthode de Newton pour trouver le zéro α de la fonction $f(x)$ et calculer la première itération à partir de la valeur initiale $x^{(0)} = 1$.
3. Calculer les zéro α de la fonction f avec la méthode de Newton (fonction `newton` que vous devrez écrire)

```python
def  newton( F, dF, x0, tol, nmax ) :
    # NEWTON Find the zeros of a nonlinear equations.
    # NEWTON(F,DF,X0,TOL,NMAX) tries to find the zero X of the
    # continuous and differentiable function F nearest to X0 using
    # the Newton method. DF is a function which take X and return the
       derivative of F.
    # If the search fails an error message is displayed.
    #
    # returns the value of the
    # residual R in X,the number of iterations N required for computing
       X and
    # INC the increments computed by Newton.

    return x, r, n, inc
```

Choisir $x^{(0)} = 1$ comme points de départ pour la méthode et utiliser une tolérance $tol = 10^{-8}$ sur la valeur absolue de l'incrément [1] entre deux itérations successives $|x^{(k+1)} - x^{(k)}|$.

Sol. :

1. *On voit que $f(0) = 1 > 0$ et $f(2) = -1 < 0$; puisque $f(x)$ est continue dans l'intervalle $[0, 2]$, on peut conclure qu'il y a au moins un zéro $x^* \in (0, 2)$.*
2. *La méthode de Newton est :*

$$x^{(n+1)} = x^{(n)} - \frac{\frac{1}{2}\sin\left(\frac{\pi x^{(n)}}{2}\right) + 1 - x^{(n)}}{\frac{\pi}{4}\cos\left(\frac{\pi x^{(n)}}{2}\right) - 1}\ , \quad n = 0, 1, 2, \ldots \tag{6}$$

1. Dans le cas de la méthode de Newton l'incrément approche assez bien l'erreur commise.

Si l'on considère $x^{(0)} = 1$, en posant $n = 0$ dans la méthode (6), on obtient

$$x^{(1)} = x^{(0)} - \frac{\frac{1}{2}\sin\left(\frac{\pi x^{(0)}}{2}\right) + 1 - x^{(0)}}{\frac{\pi}{4}\cos\left(\frac{\pi x^{(0)}}{2}\right) - 1} = 1 - \frac{\frac{1}{2}\sin\left(\frac{\pi}{2}\right)}{\frac{\pi}{4}\cos\left(\frac{\pi}{2}\right) - 1} = 1.5\ .$$

3.

```python
def  newton( F, dF, x0, tol, nmax ) :
    '''
    NEWTON Find the zeros of a nonlinear equations.
    NEWTON(F,DF,X0,TOL,NMAX) tries to find the zero X of the
    continuous and differentiable function F nearest to X0 using
    the Newton method. DF is a function which take X and return the
        derivative of F.
    If the search fails an error message is displayed.

    Outputs : [x, r, n, inc, x_sequence]
    x : the approximated root of the function
    r : the absolute value of the residualin X
    n : the number of iterations N required for computing X and
    inc : the increments computed by Newton.
    x_sequence : the sequence computed by Newton
    '''

    # Initial values
    n = 0
    xk = x0

    # initialisation of loop components
    # increments (in abs value) at each iteration
    inc = []
    # in case we wish to plot the sequence
    x = [x0]

    # diff : last increment,
    diff = tol + 1 # initially set larger then tolerance

    # Loop until tolerance is reached
    while ( diff >= tol and n <= nmax ) :
        # Newton iteration
        deltax  = F(xk) / dF(xk)
        xk1 = xk - deltax

        # increments
        diff = np.abs(deltax)
        inc.append (diff)
```

```
        # prepare the next loop
        n = n + 1
        xk = xk1
        x.append(xk)

    # Final residual
    rk1 = np.abs(F(xk1))

    # Warning if not converged
    if n > nmax :
        print('Newton stopped without converging to the desired
            tolerance ')
        print('because the maximum number of iterations was reached')

    return xk1, rk1, n, inc, np.array(x)
```

```
f  = lambda x : 0.5*np.sin(np.pi*x/2)+1-x
df = lambda x : 0.25*np.pi*np.cos(np.pi*x/2)-1

x0   = 1
tol  = 1e-8
nmax = 10

zero, residual, niter, inc, x = newton(f, df, x0, tol, nmax)

print(f'The zero computed is {zero:1.4f}')
print(f'Newton stoppedconverged in {niter} iterations');
print(f'with a residual of {residual:1.4e}.\n');
# The zero computed is 1.4031
# Newton stoppedconverged in 5 iterations
# with a residual of 0.0000e+00.
```