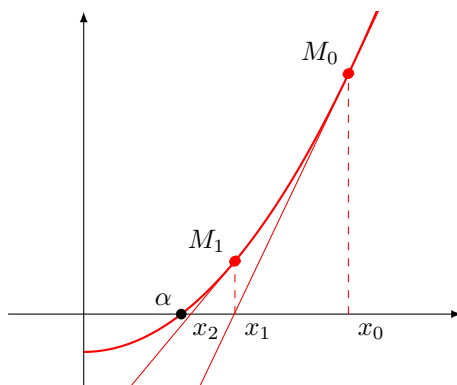


La méthode de Newton

La méthode de Newton est une méthode qui permet de déterminer une valeur approchée d'une équation du type $f(x) = 0$ quand f vérifie certaines conditions. Elle consiste à remplacer la courbe d'une fonction par ses tangentes.

Soit f une fonction deux fois dérivable sur un intervalle I dont la dérivée et la dérivée seconde sont strictement positives sur I . On suppose que l'équation $f(x) = 0$ possède une unique solution α sur I . Pour approcher α , on commence avec une première estimation x_0 supérieure à α . On considère alors la tangente à la courbe de f au point $M_0(x_0, f(x_0))$. Elle coupe l'axe des abscisses en un point d'abscisse x_1 . On recommence ensuite le même procédé avec la tangente au point $M(x_1, f(x_1))$: elle coupe l'axe des abscisses en un point d'abscisse x_2 . Etc.



La méthode de Newton consiste à étudier la suite (x_n) . Le but est ici de montrer que (x_n) converge vers α .

1. (a) Donner l'équation de la tangente à la courbe de f au point M_0 .
- (b) Montrer que $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$.
- (c) En utilisant la convexité de f , montrer que $x_1 \geq \alpha$.
- (d) Montrer de la même manière que pour tout entier naturel n , $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ et que $x_n \geq \alpha$.
- (e) Montrer que la suite (x_n) est décroissante et en déduire qu'elle converge vers un réel ℓ .
- (f) Montrer que $\ell = \alpha$.

2. Application au calcul de $\sqrt{2}$. Soit f la fonction définie sur $[1; 2]$ par $f(x) = x^2 - 2$.
 - (a) Vérifier que f est deux fois dérivable sur $[1; 2]$ et que f' et f'' sont strictement positives sur $[1; 2]$.
 - (b) Montrer qu'il existe une unique solution à l'équation $f(x) = 0$ sur l'intervalle $[1; 2]$. Quelle est cette solution ?
 - (c) Recopier et compléter le programme écrit en langage Python ci-dessous afin qu'il renvoie la valeur de x_n .
 - (d) Tester ce programme pour avec $x_0 = 2$ et afficher les 10 premiers termes de la suite. En déduire une estimation de $\sqrt{2}$.

```
def f(x):
    return ....

def fprime(x):
    return .....

def newton(x,n):
    for i in range(0,n):
        x = .....
    return x
```