

TP OPTIMIZATION: Gradient Decent with Backtraking Method

— MOUAD KLAI —

December 25, 2019

1 Importation des bibliothèque requises

```
[1]: import numpy as np
import numpy.linalg as l
```

2 Determination du pas par la méthode de backtrakcing

```
[2]: def backtraking(f,g,x,d,alpha,beta):
    t=1
    while(f(x)-f(x-t*g(x))-alpha*(l.norm(g(x))**2)<0): #Armijo
        t*=beta
    return t
```

3 Methode de decent de Gradient avec backtraking

```
[3]: def gradient_method_backtracking(f,g,x_0,beta,alpha,epsilon,k_max):
    x=x_0
    k=0
    while(l.norm(g(x))>epsilon):
        d=-g(x)
        k+=1 #nombre d'iteration
        x+=backtraking(f,g,x,d,alpha,beta)*d
        if k==k_max:
            break
    return x,f(x),g(x),k
```

4 Methode de decent de Gradient avec un pas fixe

```
[4]: def gradient_method_fixe(f,g,x_0,beta,alpha,epsilon,k_max,t): # t est le pas
    ↪ (fixe)
    x_k=x_0
    k=0
    c='converged'
    while(l.norm(g(x_k))>epsilon):
        k+=1 #nombre d'iteration
        x_k+=-t*g(x_k)
        if k==k_max:
            c="Maximum number of iterations="+str(k)+"reached"
            break
    return x_k,f(x_k),g(x_k),c,k
```

5 Des Tests :

Soit $f(x) = x^2$, et soit le point de départ suivant $x_0 = 5$

```
[5]: gradient_method_fixe(lambda x: x**2,lambda x: 2*x,5,0.5,0.01,0.001,1000,1)
```

```
[5]: (5, 25, 10, 'Maximum number of iterations=1000reached', 1000)
```

Remarque : Cette methode ne Converge pas avec $\alpha = 0.01$, $\beta = 0.5$ et un pas fixe $t_k = 1$

```
[6]: gradient_method_fixe(lambda x: x**2,lambda x: 2*x,5,0.5,0.01,0.001,1000,0.1)
```

```
[6]: (0.00042535295865117324,
      1.809251394333067e-07,
      0.0008507059173023465,
      'converged',
      42)
```

Remarque : Cette methode converge apres 42 iterations avec $\alpha = 0.01$, $\beta = 0.5$ et un pas fixe $t_k = 0.1$

```
[7]: gradient_method_backtracking(lambda x: x**2,lambda x: 2*x,5,0.5,0.01,1e-3,3000)
```

```
[7]: (0.0, 0.0, 0.0, 1)
```

Remarque : Cette methode est tres efficace et converge d'es la premiere iteration avec $\alpha = 0.01$, $\beta = 0.5$ et un pas determiner a chaque iteration par la methode du backtrakcing