Optimisation numérique (4eme année)

2021/2022

Fiche de TP n°1
UNCONSTRAINED OPTIMIZATION

L'Objectif de ce travail est se familiariser avec l'utilisation de Matlab pour faire du calcul symbolique i.e. calcul analytique en fonction de variables.

Pour déclarer la fonction $f(x) = \frac{1}{2}x^2 - \sin(x)$:

```
syms f(x) x
f(x)=(1/2 )*(x^2)-sin(x)
```

Maintenant, taper :

```
f(1)
vpa(f(1), 2)
vpa(f(1), 3)
vpa(f(1), 5)
```

Comparer les résultats, que remarquer vous ?

Que fait les commandes suivantes :

```
gradient(f(x))
diff(f(x))
taylor(f(x))
taylor(f(x),x,pi/2,'Order',3)
```

Pour la substitution de variables (par une autre variable ou une valeur numérique), on utilise la commande « subs »

Exemples : `subs(f, x, 3) ; ou ; subs(f, x, y)`

Pour tracer le graphe de la fonction f(`x`) `: fplot(f)`

On peut manipuler la taille des axes par : `xlim([-10 10])`
`ylim([-1e3 1e3])`

Optimisation numérique (4eme année)

2021/2022

Fiche de TP n°1
UNCONSTRAINED OPTIMIZATION

Cas de fonctions à plusieurs variables :

```
subs(f(x1, x2, x3), [x1;x2;x3],[1;2;3])
g=gradient(f(x1, x2, x3))
fsurf(sin(x) + cos(y))
fcontour(sin(x) + cos(y))
```

Que fait le script suivant :

```
syms x1 x2 x3 f(x1, x2, x3) alpha
f(x1, x2, x3)=(x1-4)^4+ (x2-3)^2+ 4*(x3+5)^4

x=[4; 2; -1];
g=gradient(f(x1, x2, x3));
grad=vpa(subs(g, [x1, x2, x3], x.'));

    a=0.004200;
    phi=subs(f(x1, x2, x3), [x1;x2;x3],x-alpha*grad)
    Q_1=vpa(subs(diff(phi),alpha,a));
    Q_2=vpa(subs(diff(diff(phi)),alpha,a));
    a=a-Q_1/Q_2
x=x-a*grad
```

Optimisation numérique (4eme année)

2021/2022

Fiche de TP n°1
UNCONSTRAINED OPTIMIZATION

En se basant sur les commandes précédentes, écrire des scripts pour résoudre ce qui suit.
Résoudre par la méthode de Newton, la fonction suivante :
$f(x) = \frac{1}{2}x^2 - \sin(x)$, avec $x_0 = 0.5$, et la condition d'arrêt : $|x_{k+1} - x_k| < 10^{-5}$

Résoudre par la méthode de la sécante le problème suivant :

$$\min_{\alpha>0 \in \mathcal{R}} \phi(\alpha) = (2\alpha - 1)^2 + 4(4 - 1024\alpha)^4$$
$$\alpha_0 = 0.0042$$

Résoudre par la méthode de « Steepest Descent » le problème suivant :

$$\min_{x\in\mathcal{R}^3} f(x) = (x_1 - 4)^4 + (x_2 - 3)^2 + 4(x_3 + 5)^4$$

On donne $x^{(0)} = [4,2,-1]^T$

Trouver par la méthode du gradient conjugué le minimum de la fonction suivante :

$$f(x) = \frac{3}{2}x_1^2 + 2x_2^2 + \frac{3}{2}x_3^2 + x_1x_3 + 2x_2x_3 - 3x_1 - x_3$$

($x^{(0)} = (0,0,0)$)

Optimisation numérique (4eme année)

2021/2022

Fiche de TP n°1

UNCONSTRAINED OPTIMIZATION

Découvrir :

```
solve(x^4 == 1)

assume(x,'real')
assumeAlso( x > 0)
assumptions(x)
solve(x^4 == 1)
```

Simplification :

```
simplify(f(x));
```

Apply trigonometric :

```
combine(2*sin(x)*cos(x) + (1- cos(2*x))/2 + cos(x)^2,'sincos')
```

Find the functional composition h(x)= $f(g(x))$.

```
h = compose(g,f,x)
```

Find the derivative of d/dx f(x):

```
diff(f(x),x)
```

integral:

```
int(f(x),x)

int(f(x),0,1)

limit(f(x),x,pi/2,'right')

dsolve(diff(y) == -a*y)
```

Solve the system of coupled first order ODEs

```
z = dsolve(diff(x) == y, diff(y) == -x);
disp([z.x;z.y])
```

Find the eigenvalues of A.

```
lambda = eig(A)
```

Plot the 3D

```
fplot3(xt,yt,zt, [-10,10],'--r');
```

voir :

[Computational Mathematics in Symbolic Math Toolbox - MATLAB & Simulink Example (mathworks.com)]