TRAVAUX PRATIQUES : OPTIMISATION SANS CONTRAINTES

Faits : Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice *symétrique* ; $A = (a_{i,j})_{i,j=1,\dots,n}$.
• Les conditions suivantes sont équivalentes :
– A est *définie positive*,
– Toutes les valeurs propres de A sont strictement positives,
– le polynome caractéristique de A, $p_A(\lambda) = \sum_{k=0}^{n}(-1)^k a_k \lambda^k$, avec $\forall k$, $a_k > 0$,
– $\det A_k > 0$, pour tout $k = 1, 2, \dots, n$, où $A_k = (a_{i,j})_{i,j=1,\dots,k}$.
– (seulement lorsque $n = 2$), $tr(A) > 0$ et $\det(A) > 0$.

• Les conditions suivantes sont équivalentes :
– A est *semi-définie positive*,
– Toutes les valeurs propres de A sont positives,
– le polynome caractéristique de A, $p_A(\lambda) = \sum_{k=0}^{n}(-1)^k a_k \lambda^k$, avec $\forall k$, $a_k \geqslant 0$,
– Pour tout $k = 1, 2, \dots, n$, tous les mineurs principaux d'ordre k de A sont positifs.
– (seulement lorsque $n = 2$), $tr(A) \geq 0$ et $\det(A) \geq 0$.

Exercice 1

Soient f, g les fonctions définies sur $\mathbb{R}^2$ par :

$$f(x,y) = x^3 + y^3 - 9xy + 27$$

$$g(x,y) = x^4 + y^4 - (x+y)^2$$

1 - a) - Déterminer les extremas locaux et globaux de f sur $\mathbb{R}^2$.
b) - Déterminer les extremas locaux et globaux de g sur $\mathbb{R}^2$.

2) Retrouvez les minima de f et g sous `Matlab` en utilisant la fonction `fminunc`. En guise d'exemple, pour la fonction f, le calcul de f ainsi que de son gradient sont fournis à Matlab dans le fichier `F1.m` :

```
function [f,gradf]=F1(x)
f=x(1)^3+x(2)^3-9*x(1)*x(2)+27;
if nargout > 1
    gradf(1)=3*x(1)^2-9*x(2);
    gradf(2)=3*x(2)^2-9*x(1);
end
```

On définira les options suivantes pour l'algorithme de minimisation :

```
options=optimset('LargeScale','on','GradObj','on');
```

On utilisera les commandes suivantes, la signification des paramètres employés se trouve dans l'aide sur la fonction `fminunc` :

```
x0=[1,1];
[x,fval,exitflag,output,grad,hessian]=fminunc(@F1,x0,options)
```

On essaiera plusieurs points initiaux, l'option de sortie `output` permet de voir l'influence de ce choix.

Exercice 2

f est la fonction définie sur $\mathbb{R}^3$ par :

$$f(x_1, x_2, x_3) = x_1^2 + x_2^2 + x_3^2 - x_1 x_2 - x_2 x_3 + 3x_1 - x_2 + 2x_3$$

1) Justifier de l'existence d'un extremum global que l'on caractérisera à l'aide d'un système d'équations linéaires.
2) Déterminer les extrema globaux de f sous `matlab` sans utiliser la toolbox d'optimisation.
3) Retrouver ce résultat en appliquant la fonction `quadprog` de la toolbox d'optimisation. Il faut alors écrire f sous la forme :

$$f(x) = \frac{1}{2} x^t \ H \ x + b \ x$$

Exercice 3

Soit $t \in \mathbb{R}$; on considère la fonction quadratique définie sur $\mathbb{R}^2$:

$$f_t(x, y) = \frac{t}{2}(x^2 + y^2) + xy + x - y$$

1) Déterminer tous les extrema de $(x, y) \longrightarrow g(x, y) = \exp(f_t(x, y))$; on discutera en fonction du paramètre t.
2) On se restreint à $t = 0$, -1 et -2. Tracer la surface représentative de f_0, f_{-1}, f_{-2} sur le carré $[-10, 10] \times [-10, 10]$. Par exemple le tracé pour f_0 s'obtient par :

```
[X,Y]=meshgrid([-10:0.5:10]);
graphf0=X.*Y+X-Y;
surf(X,Y,graphf0)
```

3) Que donne la fonction `fminunc` de Matlab appliquée à f_0? On lui demandera de nous fournir le gradient et la matrice hessienne.

Correction

Exercice 1

1 - a) Le gradient et la matrice hessienne de f valent :

$$\nabla f(x,y) = \begin{pmatrix} 3x^2 - 9y \\ -9x + 3y^2 \end{pmatrix} \quad , \quad \nabla^2 f = \begin{pmatrix} 6x & -9 \\ -9 & 6y \end{pmatrix}$$

Les points critiques de f sont les points $O(0,0)$ et $A(3,3)$. En O, la matrice hessienne de f n'est pas semi-définie positive (le déterminant est négatif et la trace est nulle). Donc l'origine O n'est pas un minimum pour f.
En A, la matrice hessienne de f est définie positive (le déterminant et la trace sont strictement positifs). Le point critique A de f est donc l'unique minimum local de f sur $\mathbb{R}^2$.
Il n'y a pas d'extremum global puisque $\lim_{x\to+\infty} f(x,0) = +\infty$ et $\lim_{x\to-\infty} f(x,0) = -\infty$
1 - b) Le gradient et la matrice hessienne de g valent :

$$\nabla g(x,y) = \begin{pmatrix} 4x^3 - 2x - 2y \\ 4y^3 - 2x - 2y \end{pmatrix} \quad , \quad \nabla^2 g(x,y) = \begin{pmatrix} 12x^2 - 2 & -2 \\ -2 & 12y^2 - 2 \end{pmatrix}$$

Les points critiques de g sont les points $O(0,0)$, $A(1,1)$ et $B(-1,-1)$. La matrice Hessienne de g est définie positive en A et B (déterminant et trace > 0). Ainsi A et B sont des minima locaux. La matrice Hessienne est semi-définie négative en 0 (trace < 0 et déterminant nul). Cependant O n'est pas un extremum local comme on peut s'en apercevoir en comparant $g(0,0)$ avec $g(x,x)$ et $g(x,-x)$ lorsque x est dans un voisinage de 0.
g est coercive et admet donc un minimum global et aucun maximum global. Les points A et B sont les deux minima globaux de g ($g(A) = g(B)$).

Exercice 2

1) La matrice hessienne H de f est :

$$H = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$$

Elle est définie positive ($\det H_1 = 2$, $\det H_2 = 3$, $\det H_3 = 4$) et la fonction f est donc strictement convexe et admet un unique minimum global u solution du système linéaire $H.u = -b$ où ${}^t b = (3,-1,2)$.
2) Il suffit de résoudre ce système sous `matlab` :

```
H=[2 -1 0;-1 2 -1;0 -1 2]
b=[3;-1;2]
u=inv(H)*(-b)
fmin=1/2*u'*H*u+b'*u
```

Le minimum se trouve en $u^* = (-2.2500, -1.5000, -1.7500)$ en lequel la fonction vaut -4.3750.
3) La fonction `quadprog` donne ce minimum en 1 itération.

```
options=optimset('LargeScale','off');
H=[2 -1 0;-1 2 -1;0 -1 2];
b=[3 -1 2];
[x,fval,exitflag,output]=quadprog(H,b,[],[],[],[],[],[],x0,options)
% les 2 premiers champs définissent f(x)=1/2*x'*H*x+b*x
% les suivants sont analogues à linprog
```

Exercice 3

1) Puisque la fonction exp est strictement croissante tout extremum de $g(x,y)$ est extremum de f_t et réciproquement. La matrice Hessienne de f_t est :

$$H_t = \begin{pmatrix} t & 1 \\ 1 & t \end{pmatrix}$$

elle a pour déterminant $t^2 - 1$ et pour trace $2t$. Ainsi :
– pour $t > 1$ elle est définie positive: f_t a pour unique minimum $((1+t)/(1-t^2), -(1+t)/(1-t^2))$,
– pour $t = 1$ elle est semi-définie positive, et f_t n'a aucun extremum,
– pour $-1 < t < 1$ elle est non semi-définie (négative ou positive) et f_t n'a pas d'extremum
– pour $t = -1$ elle est semi-définie négative, f_t a une infinité de maxima sur la droite $y = x - 1$
– pour $t < -1$ elle est définie négative : f_t a un unique maximum $((1+t)/(1-t^2), -(1+t)/(1-t^2))$.

2) Tracés des surfaces représentatives :

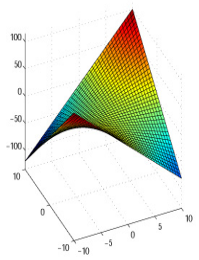

Figure 1: f_0

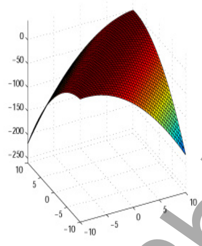

Figure 2: f_{-1}

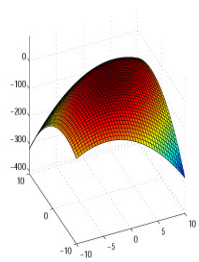

Figure 3: f_{-2}

3) On tape dans un fichier `F0.m` :

```
function [f,gradf]=F0(x)
f=x(1)*x(2)+x(1)-x(2);
if nargout >1
    gradf(1)=x(2)+1;
    gradf(2)=x(2)-1;
end
```

puis :

```
options=optimset('LargeScale','on','GradObj','on');
x0=[0,0];
[x,fval,exitflag,output,grad,hessian]=fminunc(@F0,x0,options)
```

`fminunc` retourne pour résultat :

```
x =   -10.0073    3.7297
```

en lequel le vecteur gradient est non nul. L'option `exitflag` est strictement positive, elle vaut 2 (`matlab` pense avoir trouvé une solution optimale). Le résultat est pourtant évidemment aberrant.