Université de Versailles	Année 2016-2017
http://dumas.perso.math.cnrs.fr/MA751.html	M1, semestre 1

CC1: correction

Exercice 1 –

1. On remarque que J est une fonction coercive c'est à dire:

$$\lim_{||x|| \to +\infty} J(x) = +\infty$$

On en déduit que J possède un minimum global (cf cours et TD). Celui-ci vérifie nécessairement $\nabla J(x) = 0$. Le seul point qui convient $X^* = (x, y, z)$ vérifie donc

$$\begin{cases} 4x^3 - 2 = 0 \\ 8y^3 + 1 = 0 \\ 4z^3 - 1 = 0 \end{cases}$$

c'est à dire

$$X^* = \left((\frac{1}{2})^{\frac{1}{3}}, -\frac{1}{2}, (\frac{1}{4})^{\frac{1}{3}} \right)$$

.

2. La direction de descente d_0 vérifie $H_0 d_0 = -g_0$ où H_0 est le Hessien de J en X_0 et d_0 son gradient. On a aisément $g_0 = (2, 9, 3)$ et

$$H_0 = \begin{pmatrix} 12 & 0 & 0 \\ 0 & 24 & 0 \\ 0 & 0 & 12 \end{pmatrix}$$

ce qui permet d'obtenir (inversion immédiate de H_0):

$$d_0 = (-\frac{1}{6}, -\frac{3}{8}, -\frac{1}{4})$$

d_0 est bien une direction de descente car

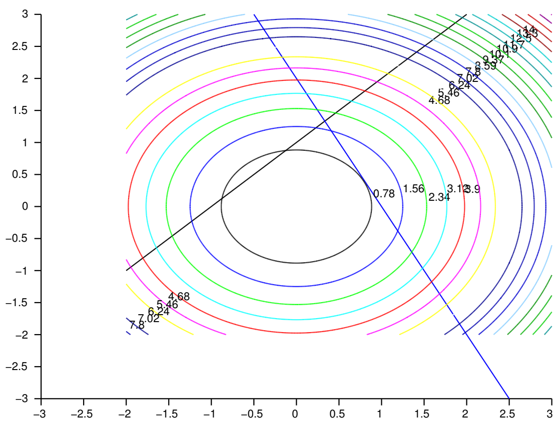
$$< d_0, g_0 >= -(\frac{1}{6}) * 2 - (\frac{3}{8}) * 9 - (\frac{1}{4}) * 3 < 0$$

3. On a

$$X_1 = X_0 + d_0 = (\frac{5}{6}, \frac{5}{8}, \frac{13}{4})$$

Exercice 2 –

1. On peut représenter la figure suivante:



   Elle représente les lignes de niveau de la fonction à minimiser et le domaine sur lequel on la minimise (quart de plan à droite). Le minimum correspond à un point qui est sur la droite de pente négative pour lequel la ligne de niveau est tangente à cette droite. On peut aussi le voir comme le point réalisant la distance de (0,0) à cette droite.

2. On veut minimiser une fonction coercive (voir exercice 1) sur un ensemble fermé. Comme précédemment, on peut affirmer qu'il y a au moins un minimum global sur cet ensemble.

3. On a

$$\mathcal{L}(x_1, x_2, \lambda_1, \lambda_2) = x_1^2 + x_2^2 + \lambda_1(-x_1 + x_2 - 1) + \lambda_2(-2x_1 - x_2 + 2)$$

   L'algorithme d'Uzawa consiste à faire évoluer un point $X_n = (x_1^n, x_2^n, \lambda_1^n, \lambda_2^n)$ en faisant successivement un pas de gradient pour minimiser $\mathcal{L}$ en (x_1, x_2) puis un pas de gradient pour maximiser $\mathcal{L}$ en $(\lambda_1, \lambda_2) \in \mathbb{R}_+^2$.

4. voir programme joint.

Exercice 3 –

1. La variable p_c est une probabilité de croisement. Avec la probabilité p_c, les lignes $2k-1$ et $2k$ de la nouvelle matrice sont issues d'un croisement de deux lignes de la matrice de départ. On simule pour cela une loi de Bernoulli de paramètre p_c.

2. L'aspect aléatoire intervient à trois niveaux dans ce code:

   - pour le choix des deux éléments sélectionnés pour le croisement (tirage aléatoire de deux lignes entre 1 et N_{pop}).

   - pour la probabilité de croisement (loi de Bernoulli, voir question précédente).

   - pour la génération des deux nouveaux éléments (pondération aléatoire des deux éléments de départ).

3. Plutôt que de choisir un coefficient α unique, on peut le rendre dépendant des coordonnées des deux éléments, ce qui donne alors:

```
if(rand()<pc) then
 for k=1:n
  alpha=rand()
  Acrois(2*k-1,n)=alpha*u1(k)+(1-alpha)*u2(k);
  Acrois(2*k,n)=(1-alpha)*u1(k)+alpha*u2(k);
 end
end
```