DEVOIR 2

MAT-2410: Optimisation — Automne 2014

A remettre vendredi le 7 novembre avant 16h00.

L'objectif du devoir est de mettre en oeuvre les algorithmes de minimisation traités en classe pour les problèmes d'optimisation sans contrainte. Il s'agira d'appliquer ces algorithmes dans des situations concrètes et de comparer les performances des différents algorithmes.

Notes

- Le devoir doit obligatoirement être réalisé à l'aide du logiciel Matlab.
- On remettra une copie papier du travail comportant les fichiers de Matlab ainsi que les réponse aux questions. De plus, on enverra par courriel une copie électronique des fichiers Matlab (M-file, script, etc.) de sorte que le professeur soit capable d'exécuter les programmes Matlab.

Question 1.

L'objectif de cette question est résoudre un problème classique du calcul des variations. Il s'agit du problème de la Brachystochrone résolue par Johann Bernoulli (1667-1748). Le problème consiste à déterminer la trajectoire qui minimise le temps parcouru d'une particule qui glisse le long d'un fil uniquement par l'action de la gravité dont les extrémités sont fixés. Dans ce qui suit, nous allons supposer que les extrémités sont les points $(0, 0)$ et $(1, \alpha)$ avec $\alpha > 0$ et nous allons choisir un système d'axes pour lequel y est orienté vers le bas. Si on note par $y(x)$ la trajectoire cherchée, le problème consiste à calculer la fonction y qui minimise l'intégrale (le temps parcouru)

$$T(y) = \int_0^1 \sqrt{\frac{1 + (dy/dx)^2}{2gy(x)}}\, dx$$

où $g = 9.81$ est la constante gravitationnelle.

Pour résoudre ce problème de manière numérique, nous allons nous limiter à la classe des trajectoires linéaires par morceaux, c'est-à-dire une ligne brisée formée de segments de droites. Pour cela, on fixe une valeur de n et l'on décompose l'intervalle $[0, 1]$ en $n + 1$ sous-intervalles $[x_{i-1}, x_i]$ où $\Delta x = 1/(n + 1)$ et $x_i = i\Delta x$ pour $i = 0, 1, 2, \ldots, n + 1$. Il s'agit de calculer les valeurs y_i de la trajectoire en ces points x_i. Le fait que les extrémités soient fixées, nous donne que $y_0 = 0$ et $y_{n+1} = \alpha$.

La fonctionnelle $T(y)$ peut être approchée par la fonction $f(y_1, y_2, \ldots, y_n)$ définie par

$$f(y_1, y_2, \ldots, y_n) = \sum_{i=1}^{n+1} \frac{2\sqrt{(\Delta x)^2 + (y_i - y_{i-1})^2}}{\sqrt{2gy_i} + \sqrt{2gy_{i-1}}}$$

a) Montrer que pour une trajectoire $y(x)$ linéaire par morceaux définie par les valeurs nodales y_i aux noeuds x_i, on a bien l'égalité

$$T(y) = f(y_1, y_2, \ldots, y_n).$$

b) Evaluer le gradient de cette fonction f.

Pour les calculs numériques, on prendra $n = 20, 40$. De plus, on fera plusieurs essais avec $\alpha = 0.1, 0.5, 1.0$. On adoptera le critère d'arrêt suivant

$$\|\nabla f(y^{(k)})\| < \epsilon = 10^{-6}.$$

et la droite reliant les points $(0,0)$ et $(0,\alpha)$ comme solution initiale: $y_i = \alpha x_i$.

c) Résoudre le même problème par l'algorithme du gradient conjugué

- $y^{(0)}$ donné, évaluer $r^{(0)} = -\nabla f(y^{(0)})$ et poser $d^{(0)} = r^{(0)}$.
- Tant que $||r^{(k)}|| > \epsilon$ et $k < NMAX$ faire:
- déterminer $\rho^{(k)}$ par la recherche linéaire de type sécante.
- Poser $y^{(k+1)} = y^{(k)} + \rho^{(k)} d^{(k)}$
- Calculer $r^{(k+1)} = -\nabla f(y^{(k+1)})$
- Poser $\beta^{(k)} = \dfrac{\left(r^{(k+1)}, r^{(k+1)} - r^{(k)}\right)}{\left(r^{(k)}, r^{(k)}\right)}$
- Poser $d^{(k+1)} = r^{(k+1)} + \beta^{(k)} d^{(k)}$
- Si $(d^{(k+1)}, r^{(k+1)}) \leq 0$ on ré-initialise la direction de descente: $d^{(k+1)} = r^{(k+1)}$.

d) Représenter graphiquement les solutions, selon les valeurs de α pour $n = 20$ seulement. Décrire la convergence des algorithmes pour chaque essai.

Question 2.

Dans cette dernière partie du devoir, nous allons résoudre le même problème de minimisation que celui de la question 1.

$$\min_{x \in \mathbb{R}^n} f(x)$$

par une variante de l'algorithme de Newton connue sous le nom de méthode de Levenberg-Marquardt. On a adopté la notation $x = (y_1, y_2, \dots, y_n)$.

Voici l'algorithme:

- x_0 donné,
- Tant que $\|\nabla f(x_k)\| > \epsilon = 10^{-6}$ et $k < NMAX$ faire:
- On cherche la plus petite constante non-négative μ_k pour laquelle la matrice $\mu_k I + H(x_k)$ admet des valeur propres plus grandes ou égales à $\delta = 10^{-4}$. On rappelle que $H(x_k)$ dénote la matrice Hessienne de f au point x_k.
- On pose la matrice $M_k = \mu_k I + H(x_k)$
- $x_{k+1} = x_k - \rho_k \; M_k^{-1} \nabla f(x_k)$ où $\rho_k > 0$ est choisi par la recherche linéaire d'Armijo. La direction de descente correspond à $d_k = -M_k^{-1} \nabla f(x_k)$. La recherche linéaire doit débuter avec $\rho_k = 1$.

On adoptera le critère d'arrêt suivant

$$\|\nabla f(x_k)\| < \epsilon = 10^{-6}.$$

(a) Calculer la matrice Hessienne de la fonction f de la question 1.

(b) Appliquer la méthode de Levenberg-Marquardt pour résoudre le problème de la Brachystochrone pour $n = 20, 40$ et $\alpha = 0.1, 0.5, 1.0$. Décrire la convergence de l'algorithme pour tous les essais.

(c) Commenter la performance de la méthode par rapport à l'algorithme de la question 1.

Question 3.

Dans cette question, nous allons résoudre des problèmes de moindres carrés non linéaires par une variante de l'algorithme de Newton connue sous le nom de méthode de Gauss-Newton.

Les problèmes de moindres carrés non linéaires sont des problèmes d'optimisation de la forme

$$\min_{x \in \mathbb{R}^n} f(x) \qquad \Longleftrightarrow \qquad \min_{x \in \mathbb{R}^n} \frac{\|F(x)\|^2}{2}$$

où $F : \mathbb{R}^n \longrightarrow \mathbb{R}^m$ est une fonction différentiable. Si $m > n$, le système non linéaire $F(x) = 0$ n'admet pas en général de solution car il y a trop d'équations par rapport au nombre d'inconnues. Ceci motive le problème d'optimisation ci-dessus.

(a) Evaluer le gradient ∇f en fonction de F et de ces dérivées premières.

(b) Evaluer la matrice Hessienne de f en fonction de F et de ces dérivées premières et secondes.

Algorithme de Gauss-Newton

- x_0 donné,
- Tant que $\|\nabla f(x_k)\| > \epsilon = 10^{-6}$ et $k < NMAX$ faire:
- Calculer d_k solution de
$$M_k d_k = -\nabla f(x_k)$$
où M_k est la partie de la matrice Hessienne de f (voir (b)) ne contenant que les dérivées premières de la fonction F.
- $x_{k+1} = x_k + \rho_k \, d_k$ où $\rho_k > 0$ est calculé par la recherche linéaire d'Armijo en partant de $\rho = 1$.

Appliquer l'algorithme de Gauss-Newton pour résoudre les problèmes suivants.

(c) Le problème de Rosenbrock

$$\min_{(x_1, x_2) \in \mathbb{R}^2} 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

On prendra comme point de départ $(-1.5, 1.5)$. Illuster le parcours suivi par les itérés pour aller vers le point minimisant. Commenter la convergence de la méthode.

(d) Le problème de régression non linéaire suivant

$$\min_{(a_1,a_2,\lambda_1,\lambda_2)} \frac{1}{2}\sum_{i=1}^{m}(a_1 e^{-\lambda_1 t_i} + a_2 e^{-\lambda_2 t_i} - y_i)^2$$

On prendra les données

$$\begin{array}{lcl} (t_1,\ldots,t_m) & = & [0, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0] \\ (y_1,\ldots,y_m) & = & [8.0010, 4.6933, 3.3918, 2.8315, 2.5487, 2.3709, 2.2353, 2.1186, 2.0135] \end{array}$$

On représentera graphiquement sur une même figure la fonction
$g(t) = a_1 e^{-\lambda_1 t} + a_2 e^{-\lambda_2 t}$ avec les paramètres solutions ainsi que les points expérimentaux.
Discuter de la convergence de l'algorithme.