MATHÉMATIQUES ET INFORMATIQUE
Sciences
Université Paris Cité

MASTER INFO 2022–2023

Optimisation Algorithmique :

TD4 : Méthode de Newton. Méthode de descente de gradient. Comparaisons.

Exercice 1

On considère la fonction définie sur $\mathbb{R}$ par $f(x) = \ln(e^x + e^{-x})$.

Écrire les fonctions `[x_h,v_h]=methode_gradient(x0,c,rho)` qui effectue une descente de gradient avec backtracking et
`[x_h,v_h]=methode_newton(x0)` qui utilise la décomposition de Cholesky pour calculer l'inverse de la Hessienne de f.

Tracer la suite $(x^{(\mathrm{k})})_k$ sur le graphe de f.

Utiliser comme valeurs initiales $x^{(0)} = 0,2$, $x^{(0)} = 1$ et $x^{(0)} = 1,1$.
Comparer les vitesses de convergence et commentez.

Exercice 2

On s'intéresse à la minimisation de la fonctionnelle coût $f(x) = \sum_{k=1}^{p} e^{<a^k,x>+b_k}$,

où $x \in \mathbb{R}^n$, $b = (b_1 \cdots b_p)^t \in \mathcal{M}(p,1)$ et on note A la matrice formé à partir des vecteurs $a^k \in \mathbb{R}^n$: $A = (a^1\, a^2 \cdots a^p)^t \in \mathcal{M}(p,n)$.

On va comparer la **méthode de descente de gradient** et la **méthode de** Newton sur cet exemple.

1. En Octave, écrire la fonction `objectif(x,A,b)` qui calcule $f(x)$, la fonction `gradient_obj(x,A,b)` qui calcule $\nabla f(x)$ et la fonction `hess_obj(x,A,b)` qui calcule $H_f(x)$.

   Pour pouvoir comparer les deux méthodes, on doit atteindre la même précision, pour cela chaque algorithme devra par exemple s'arrêter dès que la valeur courante de la fonction coût, `v`, est proche à 10^{-10} près, de la valeur minimale obtenue par
   `grad_obj = @(x) gradient_obj(x,A,b); [x_min,vg_min] = fsolve(grad_obj,x0);`
   Implémenter ce test d'arrêt dans la fonction `methode_gradient()` et la fonction `methode_newton()`.
2. Écrire la fonction `[x_h,v_h] = methode_gradient(x0,c,rho,A,b,v_min)` qui utilise la méthode de descente du gradient en utilisant un "backtracking".
   Les vecteurs `x_h`, resp. `v_h`, contiennent respectivement l'historique des points et des valeurs.

Faire quelques tests.

3. Écrire la fonction `[x_h,v_h]=methode_newton(x0,A,b,v_min)` qui utilise la décomposition de Cholesky pour déterminer le pas de Newton.

4. Applications : prendre d'abord $n = 2$, $p = 3$ et

$$f(x) = e^{x_1+3x_2-0.1} + e^{x_1-3x_2-0.1} + e^{-x_1-0.1}\,.$$

Pour la valeur initiale prendre `x0=[-2 3]'`.
Comparer les performances et le comportement des deux méthodes en affichant des lignes de niveau et le trajet de la suite des points minimisants.
Tracer la précision du résultat ($-\log_{10}|$`v-v_min`$|$) en fonction du nombre d'itérations.

Recommencer avec d'autres valeurs de `x0`.

Exercice 3

Reprendre la fonctionnelle coût $f(x) = \displaystyle\sum_{k=1}^{p} e^{<a^k,x>+b_k}$,
où $x \in \mathbb{R}^n$, $b = (b_1 \cdots b_p)^t \in \mathcal{M}(p,1)$ et on note A la matrice formé à partir des vecteurs $a^k \in \mathbb{R}^n$: $A = (a^1\, a^2\, \cdots\, a^p)^t \in \mathcal{M}(p,n)$.
pour n grand et $m = 2n+1$.