

ENS CACHAN, DÉPARTEMENT DE MATHS

OPTIMISATION NUMÉRIQUE M1, ANNÉE 2014-2015

3 OCTOBRE 2014

TP 1 – MÉTHODES DE GRADIENT, GRADIENT CONJUGUÉ ET QUASI-NEWTON

NB. Il est conseillé d'utiliser la fonction `inline` ou une déclaration de fonction “anonyme” (anonymous) pour les petites fonctions, par exemple :

```
[...]
J = @(u1, u2) (u1-1)^2 + 100.0 * (u1^2 - u2)^2;
[...]
```

1. FONCTION DE ROSENBROCK ET GRADIENT À PAS FIXE

La fonction de Rosenbrock fait partie des cas tests modèles de validation des méthodes d'optimisation numérique continue (programmation non linéaire). Elle est définie par

$$J(u_1, u_2) = (u_1 - 1)^2 + 100(u_1^2 - u_2)^2.$$

La recherche du minimum est rendue relativement difficile en raison d'une “vallée” très aplatie. Le minimum est clairement atteint en $u^* = (1, 1)$, de valeur $J^* = 0$.

1. Tracer les lignes de niveau de J dans le rectangle $(u_1, u_2) \in [-1.5, 1.5] \times [-0.5, 1.5]$ (utiliser `contourf` par exemple).

2. Appliquer la méthode de gradient à pas fixe

$$u^{k+1} = u^k - \rho \nabla J(u^k).$$

On considérera les valeurs numériques $u^0 = (-1, 1)$ et $\rho = 2 \times 10^{-3}$. Tracer graphiquement les itérés de la méthode de gradient à pas fixe. Afficher le nombre d'itérations qu'il faut pour atteindre le critère d'arrêt suivant $J(u^{k+1}) - J^* < 10^{-4}$.

3. Recommencer avec $\rho = 0.0045$ puis avec $\rho = 0.01$. Que se passe-t-il ?

4. Recommencer avec le gradient normalisé

$$u^{k+1} = u^k - \rho \frac{\nabla J(u^k)}{\|\nabla J(u^k)\|}$$

avec $\rho = 0.05$ et en affichant les 200 premières itérations. Cet algorithme a-t-il des chances de converger ?

2. RECHERCHE LINÉAIRE (LINEAR SEARCH), RÈGLE D'ARMIJO

Pour une direction de descente d^k ($\langle d^k, \nabla J(u^k) \rangle < 0$), la recherche linéaire exacte consiste à trouver le $\alpha^* > 0$ qui réalise le minimum de

$$\min_{\alpha \geq 0} J(u^k + \alpha d^k).$$

En pratique, cela est inatteignable et d'ailleurs inutile : il s'agit avant tout pour l'algorithme de “descente” correctement à chaque itération. Il existe plusieurs algorithmes de recherche inexacte : règle de Wolfe, d'Armijo, d'Armijo modifiée, etc. On présente ici la règle d'Armijo (1966).

— On déclare 3 paramètres : $\beta \in (0, 1)$, $m \in (0, \frac{1}{2})$ et $L > 0$. La valeur de L est représentative de la constante de Lipschitz de la fonction J dans un compact K de recherche.

— On calcule un α_t “test” comme suit :

$$\alpha_t = - \frac{\langle \nabla J(u^k), d^k \rangle}{L \|d^k\|^2}$$

— On effectue le test de “descente raisonnable”

$$J(u^k + \alpha_t d^k) \leq J(u^k) + m \alpha_t \langle \nabla J(u^k), d^k \rangle.$$

- Si le test est positif, alors on affecte $\alpha^k \leftarrow \alpha_t$ et la recherche est terminée.
- Sinon on affecte $\alpha_t \leftarrow \beta \alpha_t$ et on recommence le test ci-dessus.

On montre que la recherche d'Armijo nécessite un nombre fini de tests.

Pour les travaux pratiques, on considère à nouveau la fonction de Rosenbrock du premier exercice avec $u^0 = (-1, 1)$. Utiliser la recherche linéaire inexacte avec règle d'Armijo pour le calcul du pas variable α^k dans la méthode de gradient

$$u^{k+1} = u^k - \alpha^k \nabla J(u^k).$$

On utilisera les paramètres $L = 100$ pour la fonction de Rosenbrock, ainsi que $m = 0.4$ et $\beta = 0.5$.

3. GRADIENT CONJUGUÉ

On rappelle que le gradient conjugué (GC) est une méthode de type quasi-Newton qui calcule des directions conjuguées au cours des itérations. On met à jour les états selon

$$u^{k+1} = u^k + \alpha^k d^k,$$

où d^k est la direction de recherche et α^k est le pas. Dans cet exercice, on pourra utiliser la règle d'Armijo pour la recherche linéaire, comme indiqué dans l'exercice précédent. Le calcul de la nouvelle direction de recherche est une combinaison linéaire entre $-\nabla J(u^{k+1})$ et d^k , et calibré de façon que

$$(d^{k+1})^T \nabla^2 J(u^k) d^k \approx 0.$$

Dans le cours, on a dérivé la formulation dite de Polak-Ribière

$$d^{k+1} = -\nabla J(u^{k+1}) + \beta^k d^k$$

avec

$$\beta^k = \frac{\langle \nabla J(u^{k+1}) - \nabla J(u^k), \nabla J(u^{k+1}) \rangle}{\|\nabla J(u^k)\|^2}.$$

Appliquer le gradient conjugué avec recherche linéaire utilisant la règle d'Armijo à la fonction de Rosenbrock du premier exercice. On partira de $u^0 = (-1, 1)$ et on tracera la trajectoire empruntée par les itérés de la méthode avec le même critère d'arrêt. On observera l'accélération de la convergence par rapport au gradient classique.

4. LE PROBLÈME DE LENNARD-JONES DE TAILLE N .

Ce problème a été donné à l'oral de modélisation de l'Agrégation de Maths.

Les problèmes de conformation atomique consistent à trouver les coordonnées spatiales de N noyaux atomiques formant une molécule d'énergie minimale. Dans le cas du problème de Lennard-Jones de taille N (en abrégé, problème LJ_N), la force d'interaction entre deux atomes identiques distants d'une longueur r est supposée issue d'un potentiel radial de van der Waals, s'écrivant sous forme adimensionnée :

$$V(r) = \frac{1}{r^{12}} - \frac{2}{r^6}.$$

Tracer la fonction V sur $]0, +\infty[$ pour se faire une idée. Lorsqu'une molécule est constituée de N atomes situés respectivement aux positions $X_1, \dots, X_N \in \mathbb{R}^3$, $X_i = (x_i, y_i, z_i)$, l'énergie potentielle totale du système est égale à

$$LJ_N(X_1, \dots, X_N) = \sum_{i=1}^N \sum_{j \in \{1, \dots, N\}, i < j} V(\|X_i - X_j\|).$$

On a ainsi une fonctionnelle de la variable $u = (X_1, \dots, X_N) \in \mathbb{R}^{3N}$. La configuration la plus stable de la molécule u^* correspond à un minimum global d'énergie potentielle. Lorsque $N > 4$, le problème de la recherche du minimum global de LJ_N devient complexe. Il est d'ailleurs conjecturé que le nombre de minima locaux de LJ_N croît exponentiellement avec N .

- (1) Appliquer à la fonctionnelle de Lennard-Jones l'algorithme du gradient conjugué en utilisant la règle d'Armijo pour la recherche linéaire.

On commencera avec $N = 4$ pour vérifier que la méthode est capable de trouver le minimum global dans ce cas (conformation en tétraédre, $LJ_4 = -6$). Pour l'initialisation, on choisira $X_1 = 0$ et X_i , $i \geq 1$ choisis aléatoirement dans la boule $B(0, N^{1/3})$.

- (2) Visualiser en 3D la configuration minimale obtenue. On tracera les noyau (`plot3`) ainsi que les segments joignant tous les (X_i, X_j) , $i < j$ (`line`).
- (3) Appliquer à nouveau l'algorithme pour $N = 13$. On sait dans ce cas que le minimum est atteint pour l'**icosaédre régulier** de côté de longueur 1. Répéter plusieurs fois l'algorithme avec différentes initialisations pour évaluer leur influence. Etes-vous satisfait ?

5. MÉTHODE DE LEVENBERG-MARQUARDT ET GAUSS-NEWTON. APPLICATION À LA RÉGRESSION NON LINÉAIRE

(A effectuer s'il vous reste du temps).

L'algorithme de Levenberg-Marquardt est une méthode robuste et efficace de résolution des problèmes de moindres carrés et de régression non-linéaires. Il est spécifique à la minimisation d'une somme de fonctions au carré et présente le grand avantage de ne pas nécessiter de calcul des dérivées secondes.

Dans le problème classique d'ajustement des données, où le but est de trouver les paramètres β d'un certain modèle $y = f(x, \beta)$ permettant le meilleur ajustement aux observations (x_i, y_i) , on cherche à minimiser le problème aux moindres carrés régularisé

$$J(\beta) = \frac{1}{2} \sum_{i=1}^m [y_i - f(x_i, \beta)]^2 + \frac{1}{2} \mu \|\beta\|^2$$

avec $\mu > 0$ "assez petit". Dans cet exercice, on considère une approximation définie comme superposition de K gaussiennes

$$f(x, \beta) = \sum_{j=1}^K \alpha_j e^{-\frac{1}{2} \frac{(x-x_j^0)^2}{\sigma_j^2}}.$$

Les paramètres à ajuster ici sont les α_i , les σ_i et les x_i^0 , et on a

$$\beta = (\alpha_1, \dots, \alpha_K, \sigma_1, \dots, \sigma_K, x_1^0, \dots, x_K^0)^T \in \mathbb{R}^{3K}.$$

En introduisant les fonctions de résidu $r_i = y_i - f(x_i, \beta)$, $J(\beta) = \frac{1}{2} \sum_{i=1}^m r_i^2$, on a

$$\nabla J(\beta) = \sum_{i=1}^m \nabla r_i(\beta) r_i(\beta) + \mu \beta$$

et

$$\nabla^2 J(\beta) = \sum_{i=1}^m \nabla^2 r_i(\beta) r_i(\beta) + \sum_{i=1}^m \nabla r_i(\beta) \nabla r_i(\beta)^T + \mu I.$$

La méthode de Gauss-Newton est une méthode de Quasi-Newton où la matrice Hessienne de J est approchée simplement par

$$\sum_i \nabla r_i(\beta) \nabla r_i(\beta)^T + \mu I.$$

Cette approximation est assez raisonnable puisque les écarts $r_i(\beta)$ sont espérés petits pour un bon modèle de régression. Le procédé itératif est donc

$$\beta^{k+1} = \beta^k - \left(\sum_{i=1}^m \nabla r_i(\beta^k) \nabla r_i(\beta^k)^T + \mu I \right)^{-1} \nabla J(\beta^k).$$

La méthode de Levenberg-Marquardt est une variante améliorée où la constante de régularisation μ devient une suite adaptative de nombres $\mu^k > 0$ et où l'adaptation accélère la convergence et la robustesse de la méthode.

Dans ces travaux pratiques, on considérera simplement

$$\mu = \varepsilon \operatorname{tr} \left(\sum_{i=1}^m \nabla r_i(\beta^k) \nabla r_i(\beta^k)^T \right)$$

avec $\varepsilon = 10^{-5}$.

1. Mettre en oeuvre la méthode de Gauss-Newton. On considérera les valeurs numériques suivantes pour les x_i et y_i :

x_i	1	2	3	4	5	6	7	8
y_i	0.127	0.2	0.3	0.25	0.32	0.5	0.7	0.9

On choisira $K = 2$. On initialisera par $\alpha_1 = \alpha_2 = 1$, $\sigma_1 = \sigma_2 = 1$, $x_1^0 = 3$ et $x_2^0 = 6$. Tracer le nuage de points et l'approximation au cours des itérations de la boucle d'optimisation.

2. Appliquer la méthode de Gauss-Newton à la fonction de Rosenbrock $f(x, y) = 10(y - x^2)^2 + (1 - x)^2$. Tracer les points des itérés de la méthode de Gauss-Newton.