Polytech'Nice Sophia-Antipolis
Optimisation - MAM4 - Année 2022-2023

TP 1 : Méthodes de gradient

Gradient à pas fixe... sur le papier

Soit $\beta \in \mathbb{R}$, et la fonction f_β définie sur $\mathbb{R}^2$ par :

$$f_\beta(x_1, x_2) = x_1^2 + x_2^2 + \beta x_1 x_2$$

1. Calculer le gradient et le hessien de f_β ;

$$\nabla f_\beta(x_1, x_2) = \begin{pmatrix} 2x_1 + \beta x_2 \\ 2x_2 + \beta x_1 \end{pmatrix}; \quad \nabla^2 f_\beta(x_1, x_2) = \begin{pmatrix} 2 & \beta \\ \beta & 2 \end{pmatrix}$$

2. Pour quelles valeurs de β la fonction f_β est-elle convexe ? strictement convexe ?

   Nous avons : $Tr(\nabla^2 f_\beta(x_1, x_2)) = 4 > 0$ et $det(\nabla^2 f_\beta(x_1, x_2)) = 4 - \beta^2 > 0$ si et seulement si $|\beta| < 2$ auquel cas la fonction est fortement convexe. Pour $\beta = 2$, on a seulement la convexité (mais pas stricte, trouver un contre-exemple).

3. On cherche à minimiser f_β sur $\mathbb{R}$ à l'aide de la méthode de gradient à pas constant. On note $u^{(k)} = (x_1^{(k)}, x_2^{(k)})$ et $u^{(k+1)} = (x_1^{(k+1)}, x_2^{(k+1)})$ deux itérés successifs.

   (a) On note $t > 0$ le pas de descente, constant. Écrire sous une forme $u^{(k+1)} = Au^{(k)}$ la relation entre $u^{(k)}$ et $u^{(k+1)}$:
   $u^{(k+1)} = u^{(k)} - t\nabla f_\beta(u^{(k)})$.
   Expliciter la matrice A en fonction de t et de β.

   On trouve facilement

   $$A = \begin{pmatrix} 1 - 2t & -\beta t \\ -\beta t & 1 - 2t \end{pmatrix}$$

   (b) Énoncer une condition nécessaire et suffisante sur β pour que la méthode de gradient à pas constant $t = \frac{1}{2}$ converge.

   Puisque nous avons une suite géométrique, la condition nécessaire et suffisante pour la convergence est que le rayon spectral $\rho(A) < 1$, c'est à dire, pour $t = \frac{1}{2}$: $|\beta| < 2$ ($Tr(A) = 0$ et $det(A) = -\beta^2/4$ donnent comme valeurs propres $+\beta/2$ et $-\beta/2$ d'où $\rho(A) = |\beta|/2$).

Gradient à pas optimal... sur le papier

On considère la fonctionnelle

$$J(v) = \frac{1}{2}(Av, v) - (b, v), \text{ avec } v \in \mathbb{R}^n,\ b \in \mathbb{R}^n \text{ et } A \text{ une matrice symétrique définie positive.}$$

Soient λ_1 la plus petite valeur propre de A et λ_n la plus grande. On se propose de minimiser $J(v)$ sur $\mathbb{R}^n$.

A Propriétés de J

(a) Montrer que J est α-convexe en précisant la valeur de α.

$\alpha = \lambda_1$

(b) Calculer $J'(v)$. Montrer que J' est lipschitzienne par rapport à v et préciser la constante M.

$M = \lambda_n$

(c) Justifier l'existence et l'unicité d'un minimum u de J sur $\mathbb{R}^n$ et le caractériser.

J quadratique α-convexe, donc (a) coercive d'où existence, et (b) strictement convexe d'où unicité.

B Algorithme de gradient à pas optimal

(a) Ecrire l'algorithme du gradient à pas optimal pour la fonctionnelle J définie au début de l'exercice.

(b) Justifier sa convergence.

J vérifie les hypothèses du théorème : α-convexe et de gradient lipschitzien.

(c) Exprimer ρ_k en fonction de $w_k = Au_k - b$.

On écrit que deux gradients successifs sont orthogonaux, soit :
$\langle \nabla J(u^{(k+1)}), \nabla J(u^{(k)}) \rangle = 0$ et comme
$\nabla J(u^{(k)}) = Au^{(k)} - b = w_k$ et $\nabla J(u^{(k+1)}) = A(u^{(k)} - \rho_k w_k) - b$, on trouve :

$$\rho_k = \frac{\langle w_k, w_k \rangle}{\langle Aw_k, w_k \rangle}$$

C Un cas particulier

On considère $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \in \mathbb{R}^2$, $A = \begin{pmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{pmatrix}$, avec $0 < \alpha_1 < \alpha_2$ et $b = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$.

On considère l'algorithme du gradient à pas optimal qu'on initialise par $u_0 = \begin{pmatrix} u_{1,0} \\ u_{2,0} \end{pmatrix}$.

On note $u_k = \begin{pmatrix} u_{1,k} \\ u_{2,k} \end{pmatrix}$ le k-ème itéré.

(a) Exprimer $J(v)$ en fonction de v_1 et v_2. Quel est l'optimum (minimum) de J sur $\mathbb{R}^2$?

$J(v) = \alpha_1 v_1^2 + \alpha_2 v_2^2$ dont le minimum global sur $\mathbb{R}^2$ est atteint pour $u = (0, 0)$.

(b) Montrer que $\rho_k = \dfrac{\alpha_1^2 u_{1,k}^2 + \alpha_2^2 u_{2,k}^2}{\alpha_1^3 u_{1,k}^2 + \alpha_2^3 u_{2,k}^2}$.

Il suffit de remarquer que $w_k = \begin{pmatrix} \alpha_1 u_{1,k} \\ \\ \alpha_2 u_{2,k} \end{pmatrix}$ et que donc $Aw_k = \begin{pmatrix} \alpha_1^2 u_{1,k} \\ \\ \alpha_2^2 u_{2,k} \end{pmatrix}$

(c) Montrer que $u_{1,k+1} = \dfrac{\alpha_2^2(\alpha_2 - \alpha_1) u_{1,k} u_{2,k}^2}{\alpha_1^3 u_{1,k}^2 + \alpha_2^3 u_{2,k}^2}$ et $u_{2,k+1} = \dfrac{\alpha_1^2(\alpha_1 - \alpha_2) u_{2,k} u_{1,k}^2}{\alpha_1^3 u_{1,k}^2 + \alpha_2^3 u_{2,k}^2}$.

Il suffit d'écrire $u_{1,k+1} = u_{1,k} - \rho_k \alpha_1 u_{1,k}$ (et idem pour la composante 2) et de remplacer ρ_k par la valeur trouvée ci-dessus.

(d) Si $\alpha_1 = \alpha_2$, en combien d'itérations l'algorithme converge-t-il ?

Dans ce cas, l'algorithme converge en une itération, quelle que soit la valeur initiale de u_k.

(e) Si $\alpha_1 \neq \alpha_2$, à quelle condition l'algorithme converge-t-il en un nombre fini d'itérations ?

Si une des deux composantes de la valeur initiale de u_k est nulle, l'algorithme converge en une itération. Sinon, il n'atteint jamais (c'est à dire que asymptotiquement quand $k \to +\infty$) la valeur $u = (0, 0)$.

(f) Quel autre algorithme peut-on utiliser, qui converge en un nombre fini d'itérations dans tous les cas ? Illustrer géométriquement la différence de comportement avec l'algorithme de la question **B**.

Le gradient conjugué bien sûr... !

TP Méthodes de gradient... pratique sur machine

Le but de ce TP est de programmer, valider et expérimenter l'algorithme du gradient à pas fixe GF et à optimal GO. Pour faciliter les comparaisons, les 2 variantes seront implémentées dans le même programme.

Pour les expérimentations et validations, on considèrera les fonctions suivantes :

$$J_1(v) = \sum_{i=1}^{i=N} (v_i - 1)^2, \quad J_2(v) = \sum_{i=1}^{i=N} (v_i - i)^2, \quad J_R(v) = \sum_{i=1}^{i=N-1} \{(v_{i+1} - v_i^2)^2 + (v_i - 1)^2\}$$

Préciser les gradients, et les solutions optimales exactes pour ces 3 fonctions coût.

Etape 1.
Rappeler les trois versions GF, GV et GO, et identifier l'ensemble des données utilisateur, numériques, et de contrôle.

Etape 2.
Ecrire le programme GF et les modules liés (de calcul de la fonction coût et de son gradient pour les fonctions J_1, J_2, J_R).

Etape 3.
Effectuer les tests de validation de GF pour les fonctions J_1 et J_2 ($N = 10, 20, 40$) : On prendra un pas fixe $t = 1$, qu'observe t-on et pourquoi ? puis prendre $t = 0.5$.

Etape 4.
Construire l'approximation parabolique de la fonction $f(t) = J(u_k - t\nabla J(u_k))$, qui utilise $f(0)$, $f'(0)$ et $f(t_{k-1})$. S'en servie pour implémenter une approximation du calcul du pas optimal t_k. Valider GO sur les cas J_1 et J_2. Comparer ensuite GF et GO sur J_R (tracer, sur le même graphique, les 2 courbes de convergence des coûts en fonction des itérations).

Etape 5.
Etudier les méthodes GF et GO sur le cas de la fonction $J_H(x, y) = (x^2+y-2)^2+(y^2-2x+1)^2$. On prendra comme guess initial $u_0 = (0, 0)$ puis $u_0 = (1.5, -1.5)$. Que constatez-vous ? Comment valider ce constat ?