Travaux Pratiques : Méthode de Newton pour l'Optimisation

Master - Dr. Kaoutar SENHAJI

Objectifs du TP

- Comprendre le principe de la méthode de Newton pour l'optimisation.
- Apprendre à utiliser la méthode sur des problèmes convexes et non convexes.
- Implémenter la méthode de Newton en Python et analyser les résultats.
- Comparer la méthode de Newton à la méthode de descente de gradient.

—

1 Introduction (30 minutes)

Principe de la méthode de Newton

La méthode de Newton est une méthode itérative d'optimisation qui utilise à la fois le gradient et la matrice hessienne de la fonction objectif pour trouver son minimum. La mise à jour des variables est donnée par :

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}\nabla f(\mathbf{x}_k),$$

où :

- $\mathbf{H}$ est la matrice hessienne (les dérivées secondes) de f au point $\mathbf{x}_k$.
- $\nabla f(\mathbf{x}_k)$ est le gradient au point $\mathbf{x}_k$.

Avantages et limitations

- Convergence rapide pour des fonctions convexes quadratiques.
- Coût élevé pour l'inversion de la matrice hessienne.
- Nécessité d'une hessienne définie positive pour garantir la descente.

—

2 Exercice 1 : Modélisation mathématique (30 minutes)

Considérons une fonction quadratique :

$$f(x, y) = ax^2 + by^2 + cxy + dx + ey + f.$$

1. Calculer le gradient $\nabla f(x, y)$.
2. Calculer la matrice hessienne $\mathbf{H}$.
3. Écrire la règle de mise à jour de la méthode de Newton pour les variables x et y.
4. Vérifier si $\mathbf{H}$ est définie positive pour des coefficients spécifiques ($a = 1, b = 1, c = 0, d = -2, e = -2, f = 0$).

—

3 Exercice 2 : Résolution manuelle (30 minutes)

Considérons la fonction quadratique suivante :

$$f(x, y) = x^2 + y^2 - 2x - 2y.$$

1. Initialiser $x_0 = 0, y_0 = 0$.
2. Calculer le gradient $\nabla f(x_0, y_0)$ et la matrice hessienne $\mathbf{H}$ au point initial.
3. Effectuer une itération de la méthode de Newton.
4. Vérifier si la solution obtenue correspond au minimum théorique $(1, 1)$.

—

4 Exercice 3 : Implémentation Python (1h)

Objectif

Implémentez l'algorithme de la méthode de Newton pour résoudre numériquement des problèmes d'optimisation.

Étapes

1. Implémenter une fonction Python pour calculer $f(x, y)$, $\nabla f(x, y)$, et $\mathbf{H}$.
2. Implémenter une boucle pour l'algorithme de Newton :
   - Condition d'arrêt basée sur la norme du gradient.
   - Affichage des valeurs de x, y et $f(x, y)$ à chaque itération.
3. Visualiser les trajectoires des itérations sur un graphique des contours de $f(x, y)$.

—

5 Exercice 4 : Étude avancée (30 minutes)

Appliquez la méthode de Newton à une fonction non quadratique :

$$f(x, y) = \sin(x) + \cos(y) + x^2 + y^2.$$

1. Implémentez la méthode pour cette fonction.
2. Étudiez la convergence et les éventuels problèmes liés à la non-convexité.
3. Comparez les performances avec la méthode de descente de gradient.

—

6 Conclusion (30 minutes)

1. Discutez des avantages de la méthode de Newton par rapport à la descente de gradient.
2. Proposez des extensions comme la méthode de quasi-Newton (BFGS).
3. Posez des questions ouvertes pour inciter à approfondir les variantes.

—

Annexes

- Code Python de base (fourni si nécessaire).
- Représentations graphiques pour la comparaison des méthodes.

—