

Travaux pratiques – Optimisation sans contrainte

Le but de cette séance est de se familiariser avec diverses méthodes d'optimisation sans contrainte. Cela inclut une découverte de la bibliothèque d'optimisation *optimize* de *SciPy*.

On souhaite optimiser les fonctions suivantes :

$$f(x_1, x_2) = (x_1^3 - x_2)^2 + 2 (x_2 - x_1)^4$$
$$g(x_1, x_2) = 3x_1^2 + 8 x_1 x_2 + 9 x_2^2 + 3 x_1 - 4 x_2$$

1- Implémenter les méthodes simples :

- Méthode de gradient à pas constant
- Méthode de gradient à pas optimisé en utilisant la fonction *optimize.line_search* pour l'obtention du pas.

2- Comparer ces méthodes avec les méthodes de

- Méthode de Newton : *optimize.fmin_ncg*.
- Gradient conjugué : *optimize.fmin_cg*
- Quasi-newtonienne de Broyden, Fletcher, Goldfarb, et Shanno (BFGS) semblable à celle de Davidon, Fletcher et Powell (DFP) mentionnée en cours : *optimize.fmin_bfgs*
- Force brute : *optimize.brute*

Dans chaque cas, on essaiera d'améliorer les performances de la méthode en modifiant les paramètres par défaut, en ayant recours au calcul exacte du gradient, de la matrice Hessienne,....

On comparera le nombre d'itérations, la précision et on illustrera la vitesse de convergence.