

Chapitre I

Introduction

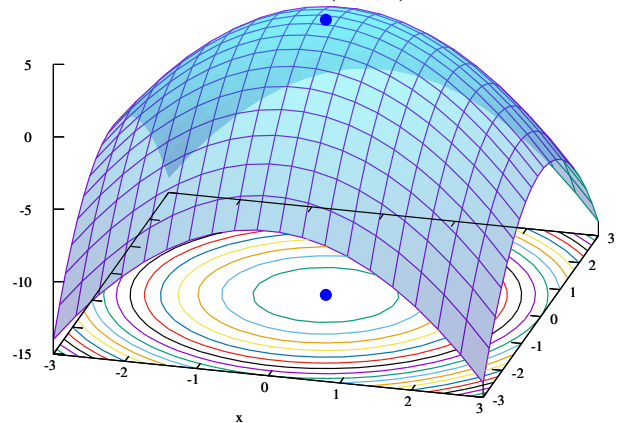
1 Définition

L'optimisation consiste à choisir :

- parmi un ensemble d'alternatives possibles,
- le meilleur élément (au sens d'un critère de choix),

Exemple :

- Trouver le meilleur élément (x, y) (= quoi chercher),
 - dans $[-3; +3] \times [-3; +3]$ (= ensemble des alternatives),
 - qui maximise la fonction $f(x, y) = -(x^2 + y^2) + 4$ (= critère de choix du meilleur (x, y)).
- (0,0,4)



Ici, le meilleur élément (=optimum) est le point $(0, 0)$.

2 Application générale

L'optimisation est utilisé dans de très nombreux domaines :

- en **mécanique**, dans la résolution d'équations aux dérivées partielles (chercher le 0), avec des contraintes géométriques (points respectant telles contraintes, ...)
- en **économie** et **finance**, minimisation des coûts, maximisation des profits ou de l'utilité par les consommateurs, minimisation des risques, maximisation des avoirs, ...
- en **ingénierie civile**, optimisation du cycle de vie des infrastructures, gestion des ressources, ...

- en **recherche opérationnelle**, pour optimiser la prise de décision (planification de trajectoire, optimisation de réseau ou de flux sur des réseaux, optimisation de placement, problème d'assignation de tâche, gestion de chaîne d'approvisionnement, planification, ...)
- ...

3 Application à la classification

Dans le cadre de la classification, à partir de données, il s'agit de regrouper un ensemble d'individus en des classes homogènes.

Plusieurs problèmes se posent :

- comment mesurer le regroupement des individus ?
- comment mesurer l'homogénéité des classes ?
- comment déterminer le nombre de classes ?

A cette fin, on définit :

- un **critère de distance** (= dissimilarité) entre deux individus mais aussi un critère de similarité.
- une **stratégie de classification** (= critère qui mesure si la classification réalisée est "bonne").

Les stratégies de classification utilisent la plupart du temps :

- le centre de gravité des classes,
- l'inertie intra-classe (mesure du "groupement" des individus dans la classe),
- l'inertie inter-classe (mesure de la "séparation" des classes entre elles)

Un critère est alors construit tel que lorsqu'il est optimisé :

- on maximise le groupement des individus dans une classe,
- on maximise la séparation des classes,

et au cours de l'optimisation, on peut modifier le nombre de classes :

- si deux classes ne sont pas assez séparées, alors on peut les fusionner.
- si dans une même classe, les individus ne sont pas homogènes, alors on peut la découper en deux.

La construction de la classification se fait avec un algorithme qui :

- à partir d'une certaine initialisation des classes,
- **répète** :
 - ◇ modifie les classes d'une manière permettant d'améliorer le critère,
 - ◇ détermine à quel classe appartient chaque individus,
 - ◇ calcule le critère de classification,
 - ◇ si le critère décroît, alors accepter la modification et recommencer.

Après convergence de l'algorithme, les classes obtenues qui minimisent le critère sont celles recherchées.

4 Classification

Optimisation fonctionnelle / paramétrique

- Inconnues = fonctions → Optimisation fonctionnelle
Optimisation en dimension infinie
Commande optimale
- Inconnues = entiers ou réels → Optimisation paramétrique
Optimisation en dimension finie
Programmation mathématique

Programmation mathématique

- Inconnues = entiers → Optimisation combinatoire
Programmation en nombres entiers
- Inconnues = réels → **Optimisation continue**
Programmation linéaire (LP)
Programmation non linéaire (NLP)
- Inconnues = entiers et réels → Programmation mixte

5 Contenu du cours

- **bases théoriques**
ensemble de définitions permettant de définir théoriquement le problème.
définition de contraintes linéaire et non linéaire.
condition d'optimalité.
- **optimisation sans contrainte**
= sans contrainte sur le domaine (méthodes de descente)
- **optimisation avec contrainte**
= avec contrainte sur le domaine
(simplexe, gradient, projeté, lagrangien)
- **métaheuristique**
cas où la solution est recherchée en utilisant des heuristiques.

6 Ressources

Ce cours est basé sur celui le cours de "Technique d'optimisation" de Max Cerf pour la plus grande partie.

Il est beaucoup plus complet que ce cours, et doit être utilisé comme première étape pour un approfondissement.

Il contient en plus une partie sur l'optimisation fonctionnelle (optimisation mathématique fonctionnelle) non abordée ici.

Note :

Pierre Delisle fait également un cours d'optimisation en M1 (master CHPS) mais s'intéresse essentiellement aux problèmes discrets d'optimisation (programmation linéaire, programmation dynamique, problème sur des flux et des graphes).

Il explore plus profondément ces problèmes que ne le fait le cours de Max Cerf. Voir avec lui afin d'y avoir accès.

7 Notes sur le cours

Le cours sera distribué seulement en version électronique à partir de la plateforme de l'URCA (mais redirigé vers github).

⇒ simplifie les téléchargements/mises à jour pour tout le monde.

Les évaluations de ce cours sont constituées :

- d'un DS à mi-parcours
- d'un DST à la fin du cours

Pendant le cours, ne pas hésiter à poser des questions si vous ne comprenez pas.

8 Conclusion

En conclusion de cette présentation, le but de ce cours est de vous donner :

- les outils conceptuels afin d'aborder les problèmes d'optimisation,
- les méthodes de résolutions avec et sans contrainte dans les cas les plus classique,
- les algorithmes les plus classiques de recherche d'optimum,

En d'autres termes, ce cours est une boîte à outil utilisable dans tous les cours où l'on cherche à optimiser un critère pour la classification, la décision ou l'apprentissage.

Exemples :

- pour la **classification**, voir l'exemple déjà donné,
- pour la **décision**, comme le choix minimisant un certain critère de risque,
- pour l'**apprentissage**, comme le choix de paramètres d'un modèle qui minimise l'écart entre le comportement du modèle et les données d'apprentissage.