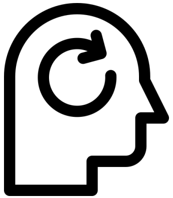
Optimisation non différentiable

Niveau M2

Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Jour 1 :
Partie 1

Rappels d’Optimisation

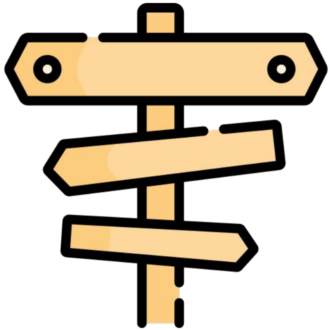


Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Motivation - Pourquoi étudier l'optimisation ?

L'optimisation est partout !

- ➢ Trouver le chemin le plus court avec votre GPS
- ➢ Entraîner un modèle de Machine Learning pour prédire le cours de la bourse
- ➢ Maximiser les profits d'une entreprise en allouant au mieux les ressources
- ➢ Concevoir des médicaments plus efficaces
- ➢ … et bien d'autres exemples !

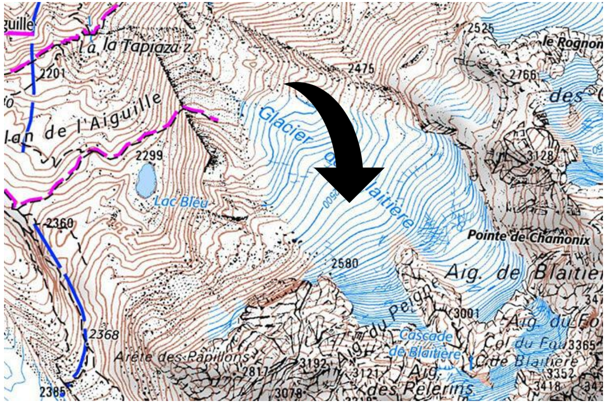


Qu'est-ce que l'optimisation ?

Définition intuitive : Trouver la "meilleure" solution parmi un ensemble de solutions possibles.

Illustration :

- Imaginez une montagne : l'optimisation consiste à trouver le sommet (maximum) ou le point le plus bas de la vallée (minimum).



Types de problèmes d'optimisation - 1/2

Continue

Les variables peuvent prendre n'importe quelle valeur dans un intervalle.

Optimiser la température d'un four, la vitesse d'une voiture.

Discrète

Les variables ne peuvent prendre que des valeurs discrètes (souvent des entiers).

Choisir le nombre de machines à utiliser, le nombre d'employés à recruter.

Sans contraintes

Aucune restriction sur les valeurs des variables.

Trouver le minimum de la fonction $f : f(x) = x^2$

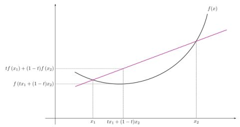
Avec contraintes

Les variables doivent respecter certaines conditions.

Maximiser le volume d'une boîte avec une surface de carton limitée.

Types de problèmes d'optimisation - 2/2

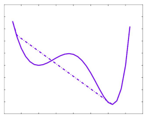
Convexe



Un seul optimum global, plus facile à résoudre.

Exemple : Minimiser f : $f(x) = x^2$

Non convexe



Plusieurs optima locaux, plus difficile à résoudre.

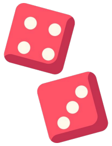
Exemple : Minimiser g : $g(x) = x^4 - 2x^3 + x + 10$

Déterministe

Tous les paramètres du problème sont connus avec certitude.

Exemple : Calculer la trajectoire d'un projectile.

Stochastique



Présence d'aléas.

Exemple : Optimiser un portefeuille d'actions en bourse.

Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Notions fondamentales - 1/2

- **Fonction objectif :** La fonction mathématique qu'on veut minimiser ou maximiser.
  - *Exemple : Coût, profit, temps, erreur.*

$$f(x, y) = 20x - 0.1x^2 + 50y - 0.2y^2$$

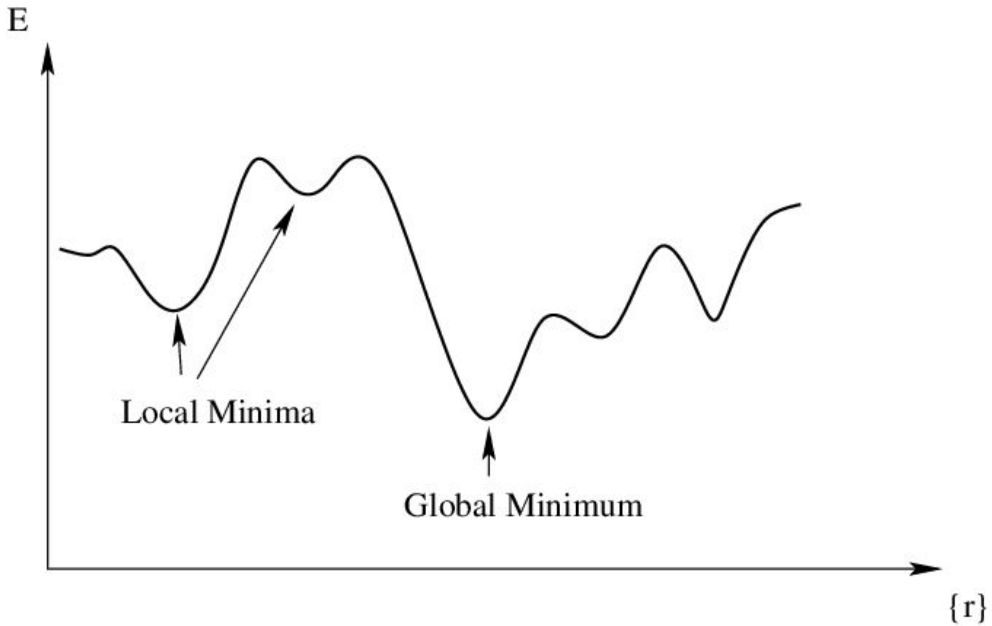
- **Espace de recherche :** L'ensemble des solutions possibles.
  - *Exemple : Tous les nombres réels, les entiers positifs, un intervalle donné.*

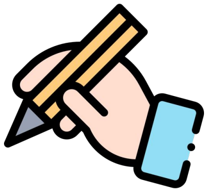
Notions fondamentales - 2/2

➢ **Solution optimale**
La solution qui minimise ou maximise la fonction objectif.

➢ **Minimum/maximum local**
Optimal dans un voisinage restreint.

➢ **Minimum/maximum global**
Optimal parmi toutes les solutions possibles.

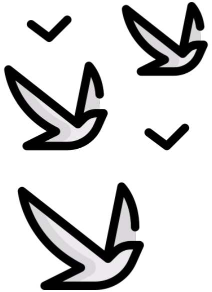




Quelques exercices pour commencer :

Rendez-vous au TD 1, exercices I.1 et I.2

Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

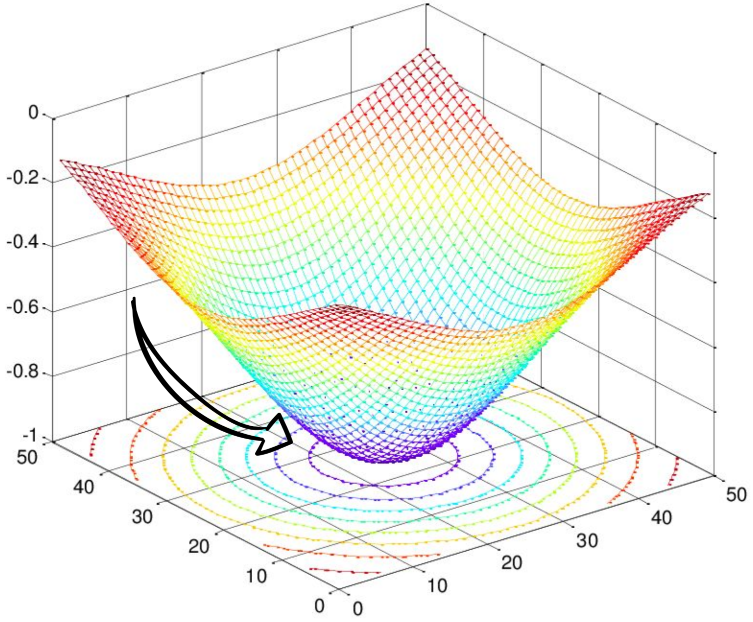


Jour 1 :
Partie 2

Optimisation sans contraintes

Optimisation sans contraintes

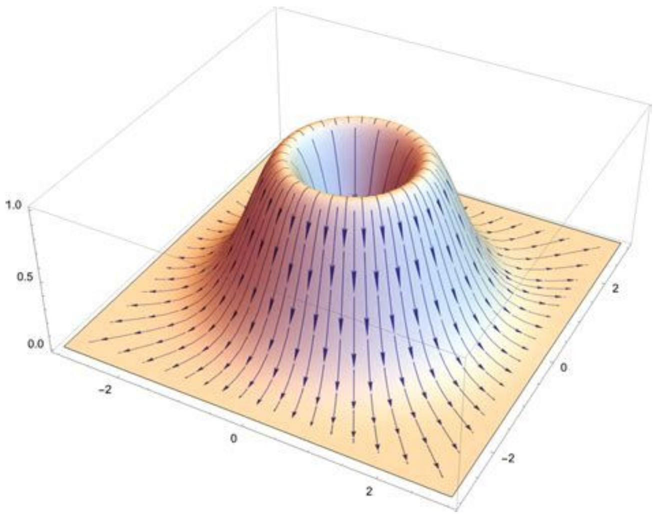
Objectif : Trouver le minimum ou le maximum d'une fonction sans aucune restriction sur les variables.



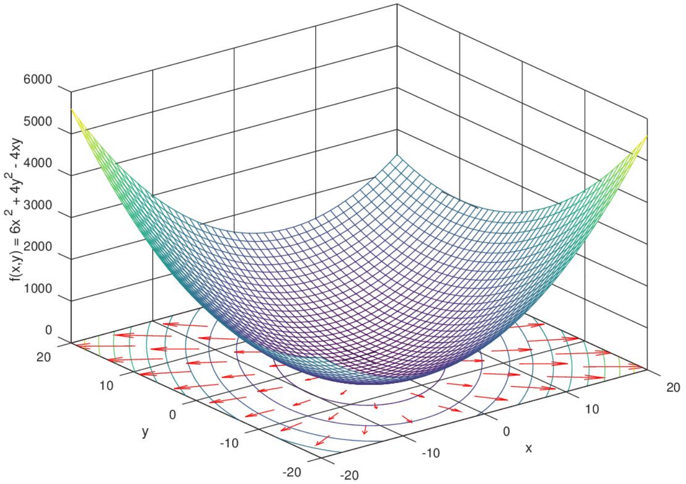
Gradient

Le **gradient** est le vecteur indiquant la direction de la plus forte pente. $\nabla f(x)$

Interprétation géométrique : Le gradient est perpendiculaire aux courbes de niveau.



$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \vdots \end{bmatrix}$$



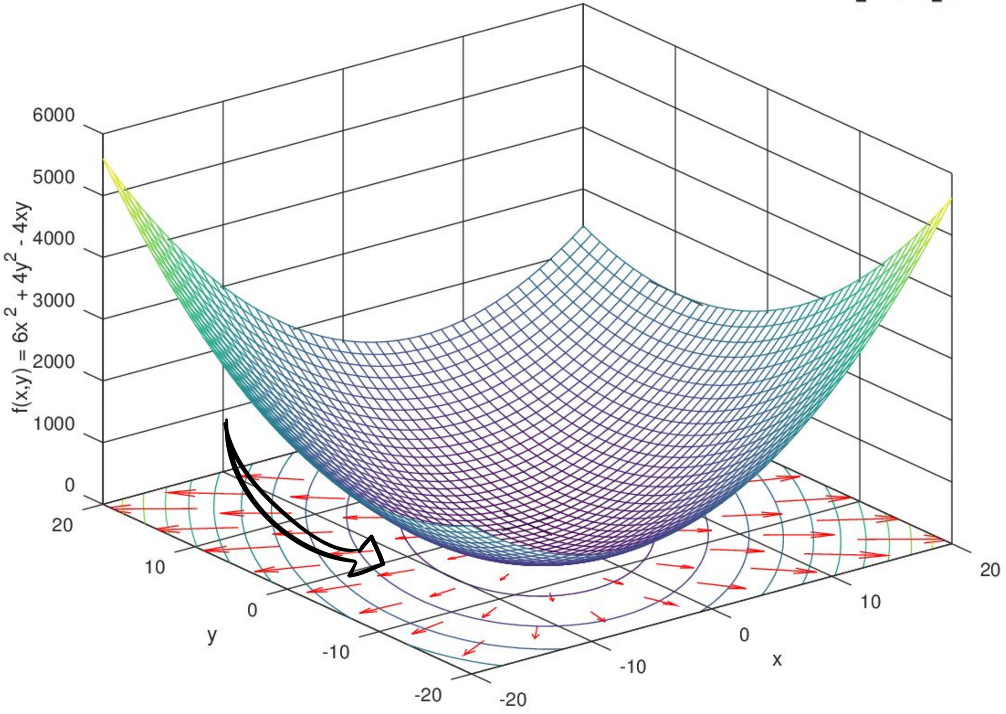
Optimalité du premier ordre

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \vdots \end{bmatrix}$$

Point critique : Un point où le gradient est nul

$$\nabla f(x) = 0$$

Condition nécessaire d'optimalité : Si est un minimum (ou maximum) local, alors est un point critique.



Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

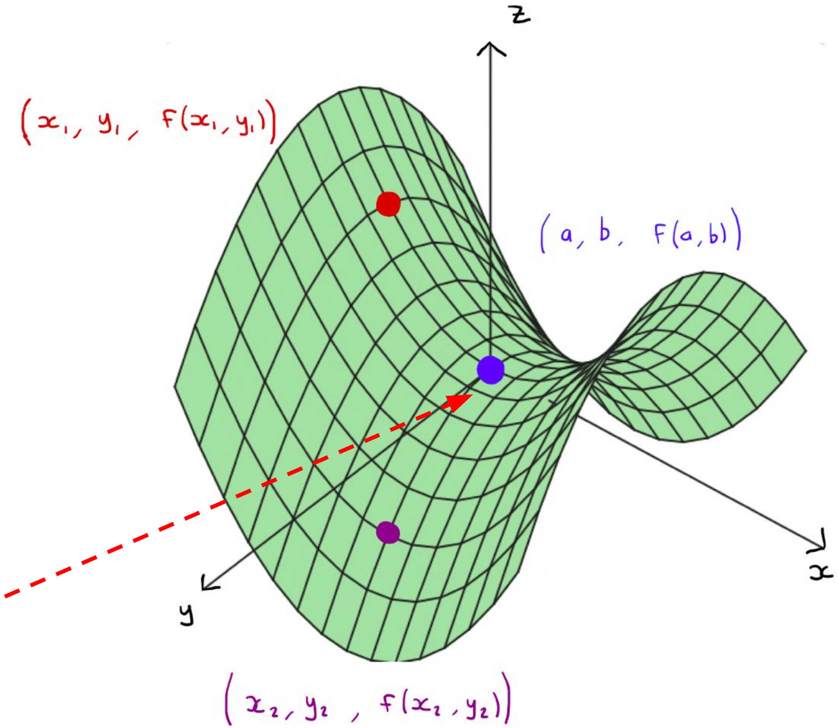
Optimalité du premier ordre

Point critique : Un point où le gradient est nul

$$\nabla f(x) = 0$$

Condition nécessaire d'optimalité : Si est un minimum (ou maximum) local, alors est un point critique.

> ***Attention !***
> *Un point critique n'est pas forcément un minimum ou un maximum (**point selle**).*



Hessienne

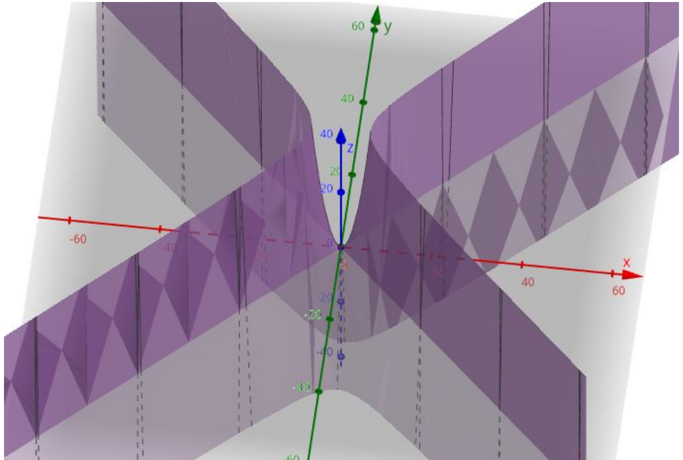
La Hessienne est la matrice des dérivées partielles secondes. $\nabla^2 f(x)$

Interprétation géométrique : La Hessienne décrit la courbure de la fonction.

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

Par exemple, le gradient et la hessienne de la fonction $f(x,y)=x^2-y^2$ s'écrivent

$$\nabla f(x,y) = \begin{pmatrix} 2x \\ -2y \end{pmatrix} \quad \text{et} \quad \nabla^2 f(x,y) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}.$$



Optimalité du second ordre

Conditions suffisantes d'optimalité :

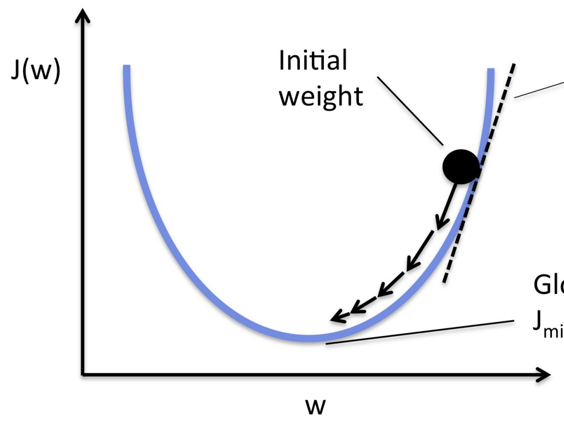
- ➢ Si $\nabla f(x^*) = 0$ et $\nabla^2 f(x^*)$ est définie positive, alors x^* est un minimum local strict.
- ➢ Si $\nabla f(x^*) = 0$ et $\nabla^2 f(x^*)$ est définie négative, alors x^* est un maximum local strict.

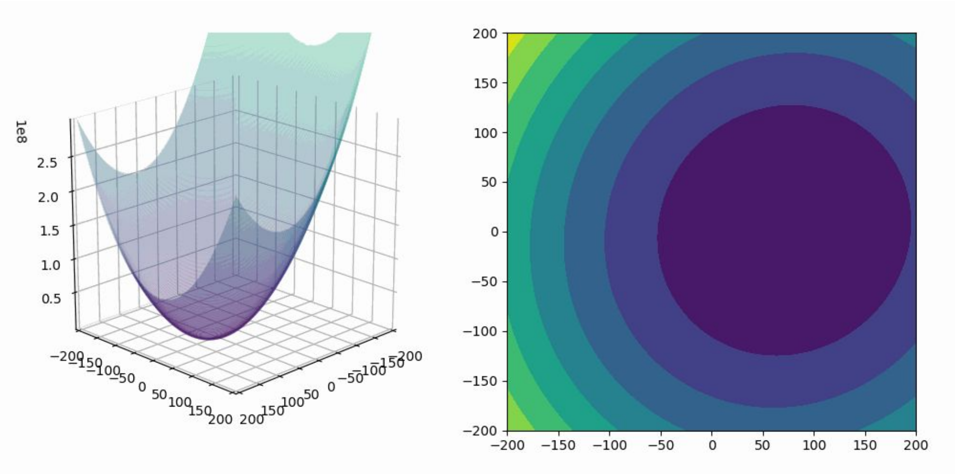
Convexité :

- ➢ Une fonction est convexe si sa Hessienne est semi-définie positive en tout point.
- ➢ Pour une fonction convexe, tout minimum local est global.

Méthodes de descente - Principe général

Idée : Partir d'un point initial et se déplacer itérativement dans la direction de descente (opposée au gradient) pour minimiser la fonction.





Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Descente de gradient

Algorithme :

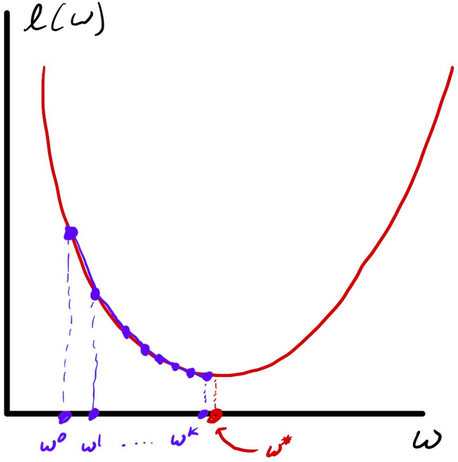
1. Initialiser . x_0
2. Répéter jusqu'à convergence :

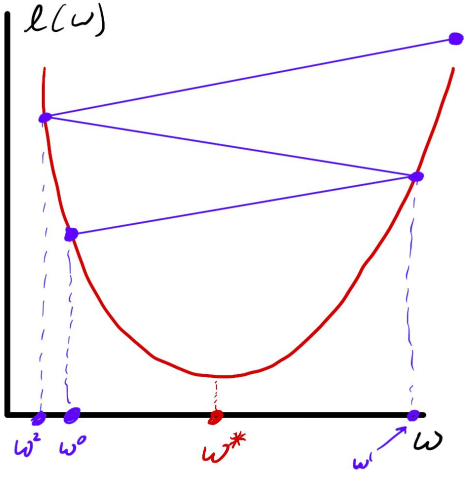
    - Calculer le gradient . $\nabla f(x_k)$
    - Choisir un pas de descente . α_k
    - Mettre à jour : . $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$

Descente de gradient

Choix du pas de descente :

- Pas fixe
- Pas décroissant
- Recherche linéaire (e.g., méthode de Wolfe)





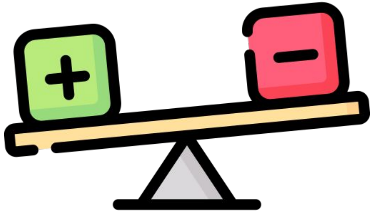
Descente de gradient

Avantages

Simple à implémenter, peu coûteuse en calcul.

Inconvénients

Convergence peut être lente, sensible au choix du pas.



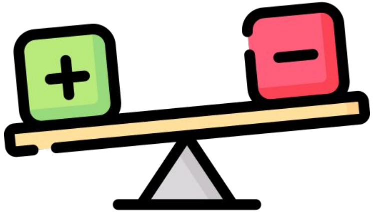
Descente de gradient

Avantages

Simple à implémenter, peu coûteuse en calcul.

Inconvénients

Convergence peut être lente, sensible au choix du pas.



Variantes :
Descente de gradient stochastique, gradient conjugué, etc.

Méthode de Newton

Principe : Utiliser une approximation quadratique de la fonction pour déterminer la direction de descente.

Formule de mise à jour :

$$x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k)$$

$$H(f) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

Méthode de Newton

Avantages : Convergence rapide près de la solution optimale.

Inconvénients :

- Nécessite le calcul de la Hessienne et de son inverse, coûteux en calcul.
- Suppose que la fonction est deux fois différentiable

Variantes : Méthodes quasi-Newton (BFGS) pour approximer la Hessienne.

Exemple 1 - Minimisation d'une fonction quadratique

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

1. Calcul du gradient

Le gradient $\nabla f(x, y)$ est donné par :

$$\nabla f(x, y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

Calculons chaque dérivée :

$$\frac{\partial f}{\partial x} = 2x - 2y + 1$$

$$\frac{\partial f}{\partial y} = 4y - 2x - 2$$

Ainsi, le gradient est :

$$\nabla f(x, y) = (2x - 2y + 1, 4y - 2x - 2)$$

Exemple 1 - Minimisation d'une fonction quadratique

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

2. Calcul de la Hessienne

La matrice Hessienne H est donnée par les dérivées secondes :

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

Calculons les dérivées secondes :

$$\frac{\partial^2 f}{\partial x^2} = 2$$

$$\frac{\partial^2 f}{\partial y^2} = 4$$

$$\frac{\partial^2 f}{\partial x \partial y} = -2$$

Donc, la matrice Hessienne est :

$$H = \begin{bmatrix} 2 & -2 \\ -2 & 4 \end{bmatrix}$$

Exemple 1 - Minimisation d'une fonction quadratique

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

3. Application de la descente de gradient

La méthode de la descente de gradient met à jour les variables selon la règle :

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \end{bmatrix} - \alpha \nabla f(x_k, y_k)$$

où α est le taux d'apprentissage.

4. Méthode de Newton

La méthode de Newton met à jour les variables selon la règle :

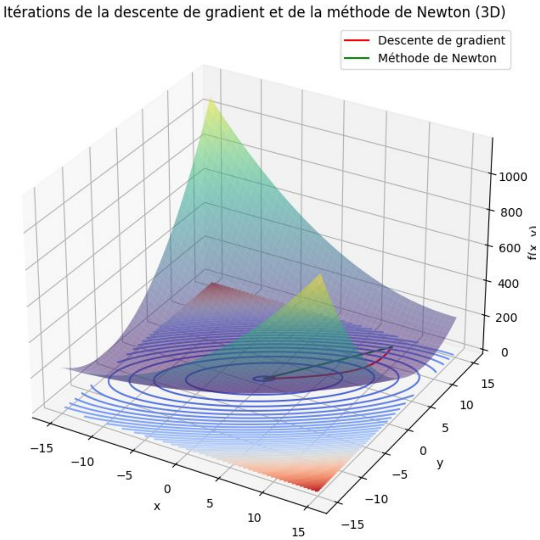
$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \end{bmatrix} - H^{-1} \nabla f(x_k, y_k)$$

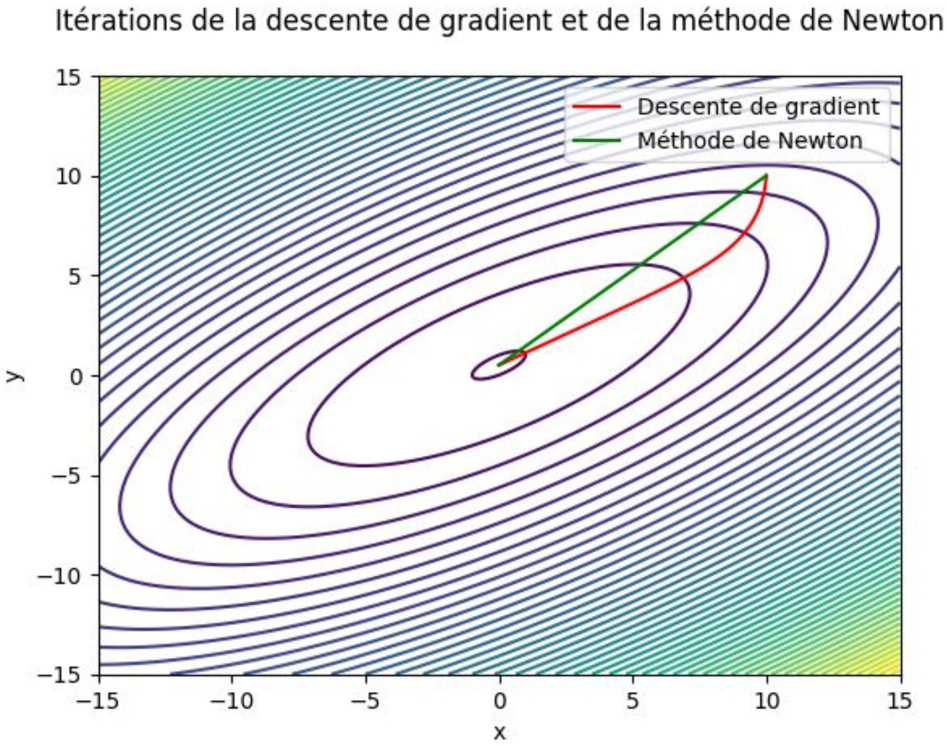
Où H^{-1} est l'inverse de la matrice Hessienne.

Exemple 1 - Minimisation d'une fonction quadratique

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

Le code exemple peut être trouvé [ici]



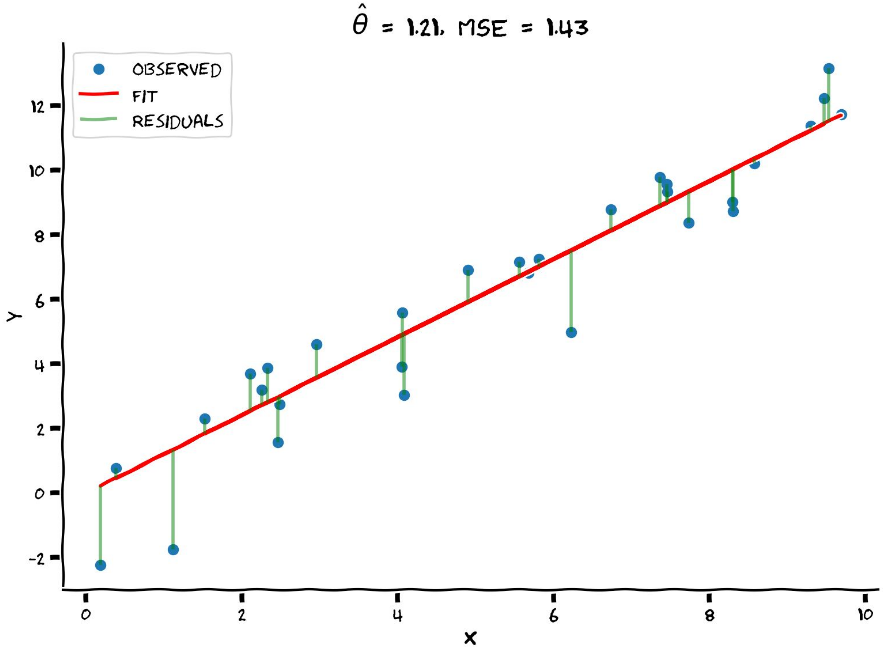


Exemple 2 - Régression linéaire

Problème : Ajuster une droite à un nuage de points en minimisant la somme des carrés des erreurs.

Fonction objectif : Erreur quadratique moyenne (MSE)

Résolution analytique
 vs. Descente de gradient



Exemple 2 - Régression linéaire

Objectif

L'objectif est de minimiser la somme des carrés des erreurs (ou résidus), qui est donné par :

$$\text{Erreur} = \sum (y_i - (mx_i + b))^2$$

où y_i est la valeur réelle pour chaque point x_i.

1. Résolution analytique

La méthode analytique pour ajuster une droite (y = mx + b) implique d'utiliser les formules pour le calcul des coefficients m (pente) et b (intercept) :

$$m = \frac{N(\sum xy) - (\sum x)(\sum y)}{N(\sum x^2) - (\sum x)^2}$$

$$b = \frac{\sum y - m(\sum x)}{N}$$

où N est le nombre de points, $\sum xy$ est la somme des produits des coordonnées, $\sum x$ est la somme des abscisses, et $\sum y$ est la somme des ordonnées.

Exemple 2 - Régression linéaire

2. Méthode de descente de gradient

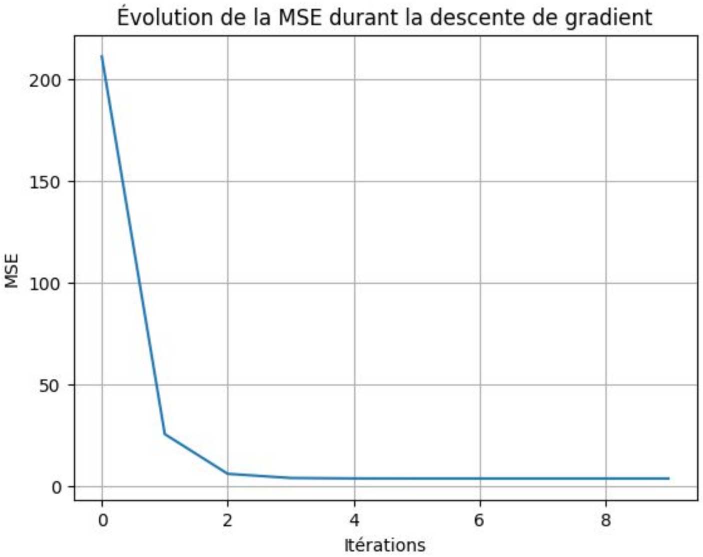
La méthode de descente de gradient consiste à itérer sur les coefficients m et b en suivant le gradient de la fonction de coût (MSE) par rapport à ces coefficients. Voici les étapes :

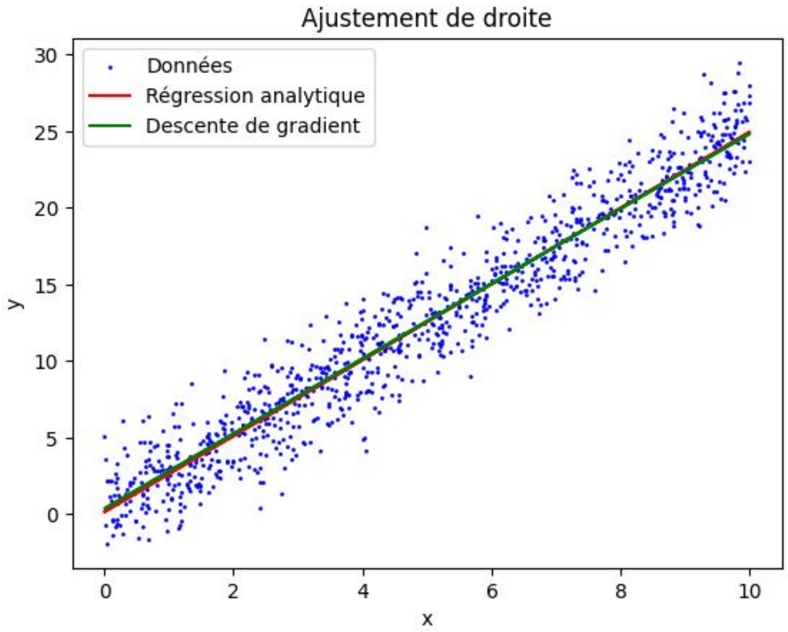
1. Initialiser m et b à des valeurs aléatoires.
2. Calculer la prédiction pour chaque point.
3. Calculer le MSE et le gradient.
4. Mettre à jour m et b en fonction du gradient.
5. Répéter jusqu'à convergence.

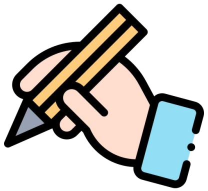
Exemple 2 - Régression linéaire

La fonction à minimiser : $(J(m,b) = \frac{1}{N}\sum_{i=1}^{N}(y_i - (mx_i + b))^2)$

Le code exemple peut être trouvé [ici]







Passons à la pratique !

Rendez-vous au TD 1, exercices partie II

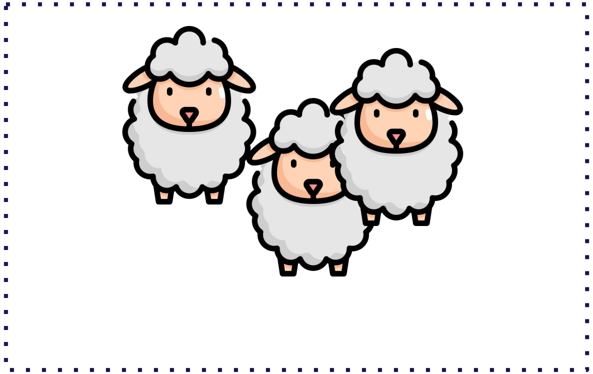
Jour 1 :
Partie 3

Optimisation avec contraintes

Optimisation avec contraintes

Objectif : Trouver le minimum ou le maximum d'une fonction tout en respectant des restrictions sur les variables.

Exemple : Un fermier souhaite clôturer un champ rectangulaire pour ses moutons. Il dispose de 100 mètres de clôture. Quelles dimensions doit-il choisir pour maximiser la surface du champ ?



Types de contraintes

Contraintes d'égalité

Une entreprise dispose d'un **budget** de 10 000€ pour sa campagne publicitaire.

Si x représente le montant dépensé en publicité en ligne et y le montant dépensé en publicité télévisée, la contrainte d'égalité s'écrit :

$$x + y = 10000$$

Contraintes d'inégalité

Un entrepôt a une **capacité de stockage** maximale de 1000 palettes.

Si x représente le nombre de palettes stockées, la contrainte d'inégalité s'écrit :

$x - 1000 \leq 0$ ou encore $x \leq 1000$

Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Méthodes de résolution (cas d'égalité)

Minimiser (ou maximiser) une fonction objectif sous la contrainte .

Substitution :

Exprimer une variable en fonction des autres à partir de la contrainte .

Substituer cette expression dans la fonction objectif pour obtenir un problème sans contrainte.

Multiplicateurs de Lagrange :

Nouvelle variable λ (multiplicateur de Lagrange) permet de former le **Lagrangien** :

$$L(x, \lambda) = f(x) + \lambda g(x)$$

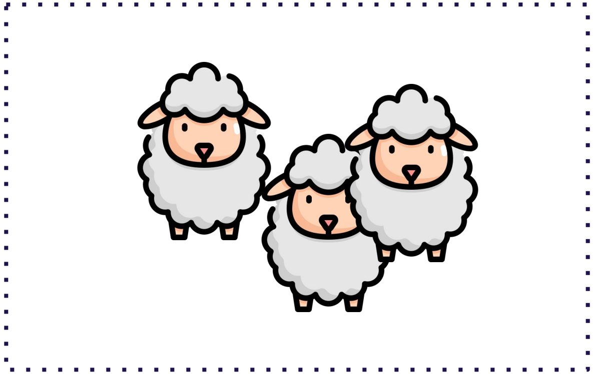
Points critiques du Lagrangien = solutions optimales

Méthodes numériques : méthodes de pénalisation, méthodes de projection…

Exemple : Maximiser une surface

Maximiser la surface d'un enclos rectangulaire avec **100 mètres de clôture**.

- **Variables :**
  - x : longueur du rectangle
  - y : largeur du rectangle



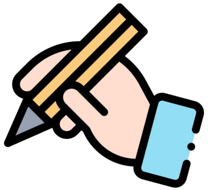
- **Fonction objectif :**
  - Surface à maximiser : $A(x, y) = x * y$

- **Contrainte :**
  - périmètre : $2x + 2y = 100$

Exemple : Maximiser une surface

Maximiser la surface d'un enclos rectangulaire avec **100 mètres de clôture**.

- **Variables :**
  - x : longueur du rectangle
  - y : largeur du rectangle



- **Fonction objectif :**
  - Surface à maximiser : $A(x, y) = x * y$

À vous ! Quelle est la surface optimale ?

Indice : procéder par substitution

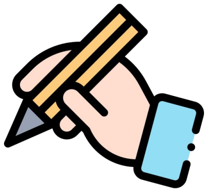
- **Contrainte :**
  - périmètre : $2x + 2y = 100$

Méthodes de résolution (cas d'inégalité)

Minimiser (ou maximiser) une fonction objectif sous la contrainte $h(x) < 0$

Méthodes de résolution :

- ➢ **Conditions KKT :** Conditions nécessaires d'optimalité utilisant le Lagrangien.
- ➢ **Pénalisation :** Ajouter une pénalité pour les contraintes violées.
- ➢ **Projection :** Projeter la solution sur l'ensemble admissible.
- ➢ **Primales-duales :** Résoudre le problème primal et dual simultanément.
- ➢ **...**



Passons à la pratique !

Rendez-vous au TD 1, exercices partie III

Jour 1 :
Partie 4

Optimisation avec Python

Importance des outils numériques

- ➢ **Résolution de problèmes complexes :** Les problèmes d'optimisation réels sont souvent de grande dimension et nécessitent des outils numériques pour être résolus.

- ➢ **Expérimentation et analyse :** Python permet de tester facilement différents algorithmes et paramètres, et d'analyser les résultats.

- ➢ **Prototypage rapide :** Python est un langage idéal pour le prototypage rapide d'algorithmes d'optimisation.

Librairies d'optimisation en Python

scipy.optimize :

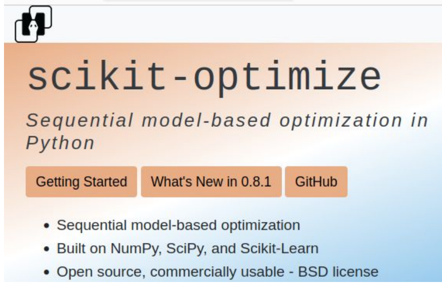
- minimize : Fonction générale pour minimiser une fonction scalaire.
- minimize_scalar : Pour les fonctions univariées.
- linprog : Pour la programmation linéaire.
- root : Pour trouver les racines d'une fonction.

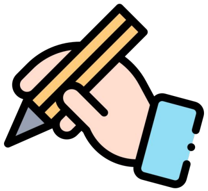
Autres librairies :

- cvxopt : Pour l'optimisation convexe.

- PuLP : Pour la programmation linéaire en nombres entiers.

- scikit-optimize : Pour l'optimisation de fonctions coûteuses à évaluer.





Passons maintenant à la pratique :

Rendez-vous au TD 1, exercices partie IV

Jour 2 :
Partie 1

Introduction à l’optimisation non différentiable
(convexe)

Le monde non lisse

Motivation : Pourquoi s'intéresser à l'optimisation non différentiable ?

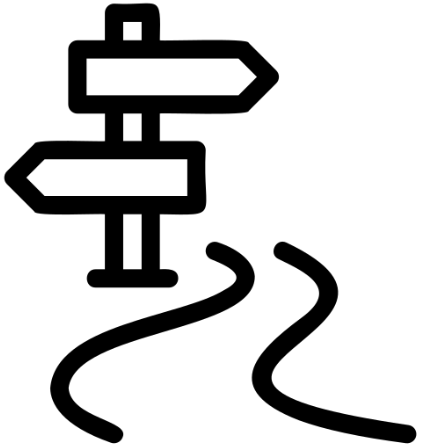
- De nombreux problèmes d'optimisation dans la vie réelle impliquent des fonctions non différentiables.
- Les méthodes classiques d'optimisation (gradient, Newton, etc.) ne fonctionnent pas pour ces problèmes.

Exemples de problèmes non différentiables :

➢ En Machine Learning : SVM, régularisation L1

➢ En ingénierie : conception optimale, planification de trajectoires

➢ En économie : optimisation de portefeuille, tarification



Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

Pourquoi les méthodes classiques ne fonctionnent pas ?

Rappel des méthodes classiques :

- ➢ Descente de gradient : utilise le gradient pour trouver la direction de descente.
- ➢ Méthode de Newton : utilise la Hessienne pour approximer la fonction par une parabole.

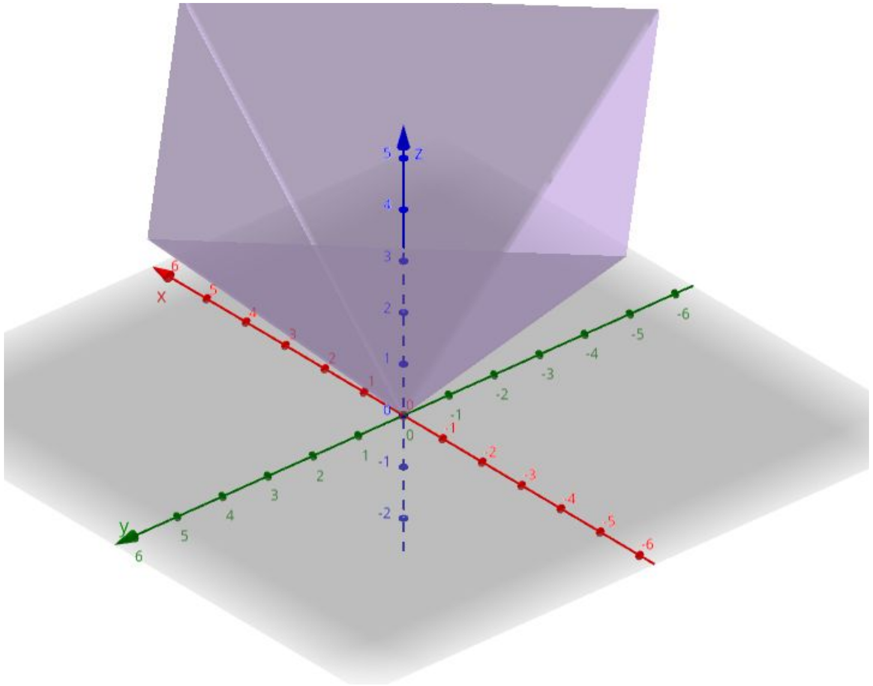
Limitations : Ces méthodes nécessitent que la fonction objectif soit différentiable (lisse).

Exemple : Minimiser la norme *L1*

Problème : Minimisation de la Norme L1

$$f(\mathbf{x}, \mathbf{y}) = |x| + |y|$$

- Les valeurs absolues créent des "arêtes" dans le graphe de la fonction



Aperçu des méthodes d'optimisation non différentiable

Méthodes de sous-gradient :

- Généralisent la descente de gradient en utilisant un sous-gradient au lieu du gradient.
- Permettent de minimiser des fonctions convexes non différentiables.

Méthodes de faisceaux :

- Construisent une approximation linéaire par morceaux de la fonction objectif.
- Plus complexes mais souvent plus efficaces que les méthodes de sous-gradient.

Autres méthodes :

- Méthodes proximales
- Méthodes de point intérieur

Méthode	Principe	Avantages	Inconvénients
Sous-gradient	Généralisation de la descente de gradient	Simple à implémenter, peu coûteuse en calcul.	Convergence lente, sensible au choix du pas.
Faisceaux	Approximation linéaire par morceaux de la fonction objectif.	Bonne convergence, efficace pour les problèmes convexes.	Plus complexe à implémenter, coûteuse en mémoire.
Méthodes proximales	Utilise un opérateur proximal.	Efficace pour les fonctions mixte différentiable/non différentiable	Nécessite le calcul de l'opérateur proximal, qui peut être coûteux.
Méthodes du point intérieur	Extension des méthodes de point intérieur pour l'optimisation linéaire.	Bonne convergence pour les problèmes convexes avec contraintes.	Complexe à implémenter, coûteuse en calcul.

Concepts Fondamentaux

Différentiabilité et non-différentiabilité

Différentiabilité :

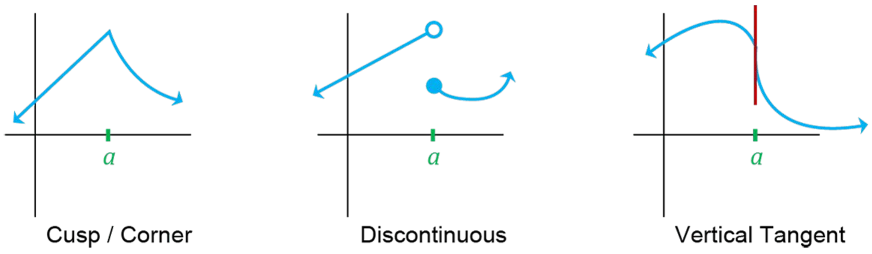
- Intuitivement : une fonction est différentiable si elle n'a pas de "cassure" ou de "point anguleux".

- Formellement : une fonction est différentiable en un point si elle admet une dérivée en ce point.

Non-différentiabilité :

- Une fonction est non différentiable si elle n'est pas différentiable en au moins un point.

- Exemples : f(x) = |x|, g(x) = max(0,x) pour x réel



Mehdi MUNIM, mehdi.munim.int@groupe-gema.com

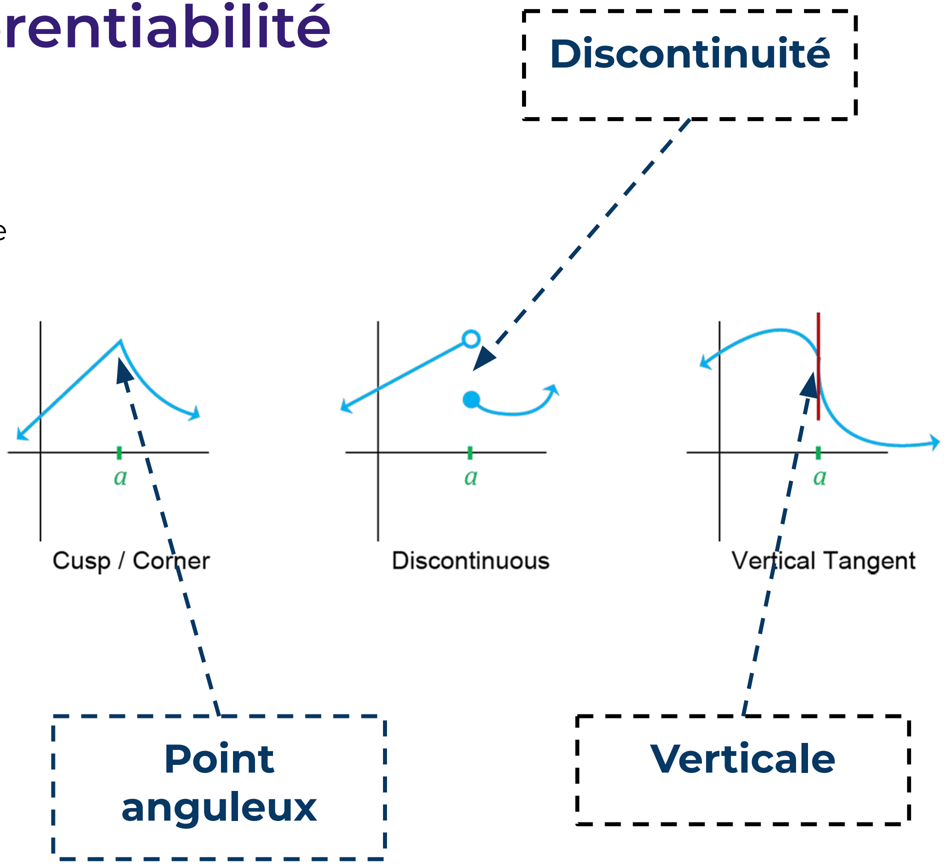
Différentiabilité et non-différentiabilité

Différentiabilité :

- Intuitivement : une fonction est différentiable si elle n'a pas de "cassure" ou de "point anguleux".
- Formellement : une fonction est différentiable en un point si elle admet une dérivée en ce point.

Non-différentiabilité :

- Une fonction est non différentiable si elle n'est pas différentiable en au moins un point.
- Exemples : f(x) = |x|, g(x) = max(0,x) pour x réel



Sous-gradient

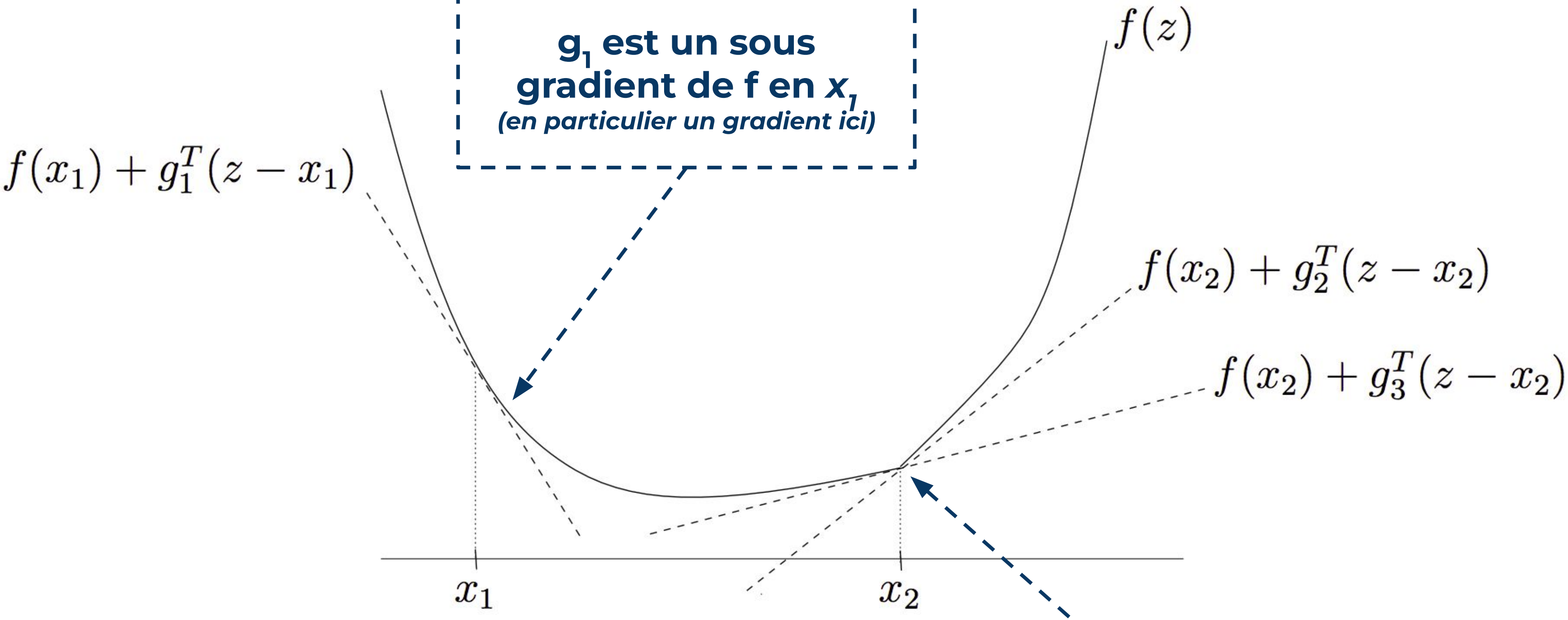
Motivation : Comment généraliser la notion de gradient pour les fonctions non différentiables ?

Sous-gradient : Un vecteur qui "sous-estime" la fonction en un point donné.

Définition :

g est un sous-gradient de **f** en **x** si pour tout **y**, on a :

$$f(y) \geq f(x) + g^T(y - x)$$



g_1 est un sous gradient de f en x_1
(en particulier un gradient ici)

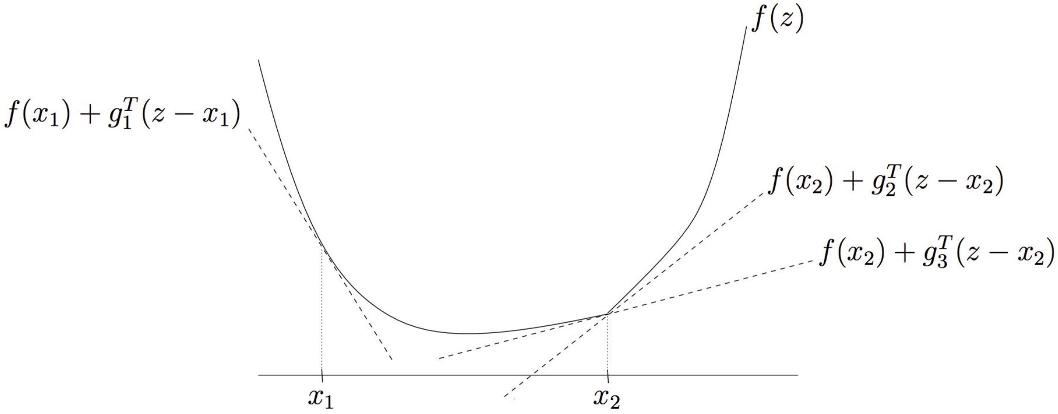
g_2 et g_3 sous-gradients de f en x_2

Sous-différentiel

Sous-différentiel : L'ensemble de tous les sous-gradients de f en x noté $\partial f(x)$

Pour les fonctions convexes, la sous-différentiel n'est jamais vide (il y a toujours un sous-gradient)

- $\partial f(x_1) = g_1$
- $\partial f(x_2) = g_2, g_3$

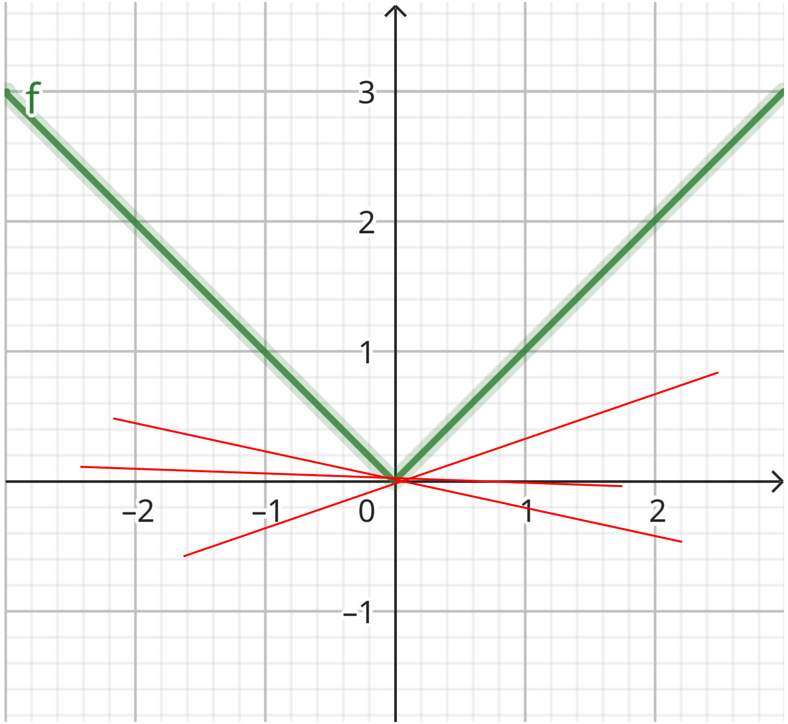


Exemple 1 - Fonction valeur absolue

Fonction : $f(x) = |x|$

Sous-différentiel :

- si $x > 0$ $\partial f(x) = 1$
- si $x = 0$ $\partial f(x) = [-1, 1]$
- si $x < 0$ $\partial f(x) = -1$

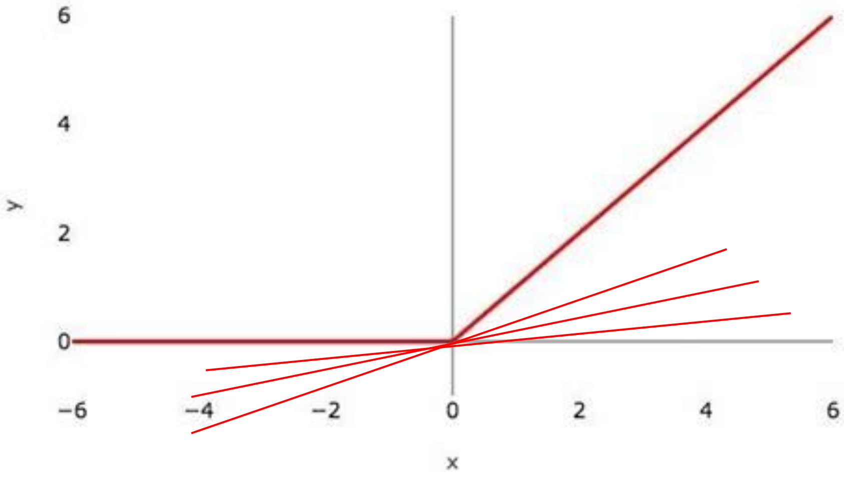


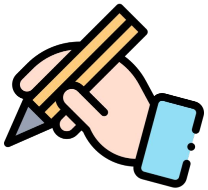
Exemple 2 - Fonction maximum

Fonction : $f(x) = \max(0, x)$

Sous-différentiel :

- si $x < 0$ $\partial f(x) = 0$
- si $x = 0$ $\partial f(x) = [0, 1]$
- si $x > 0$ $\partial f(x) = 1$





Place à la pratique

Rendez-vous au TD 2, exercices partie I

Jour 2 :
Partie 2

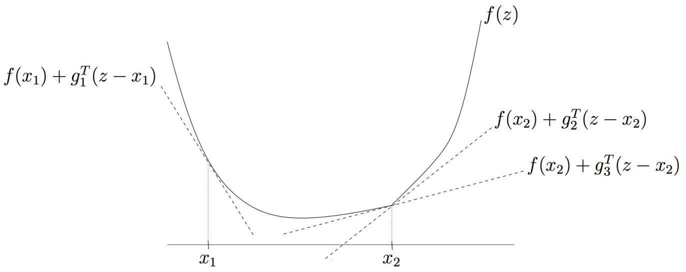
Méthode des sous-gradients

Principe de la méthode des sous-gradients

C’est une généralisation de la descente de gradient :

- Rappel : la descente de gradient utilise le gradient pour trouver la direction de descente.
- Problème : le gradient n'est pas défini pour les fonctions non différentiables.
- Solution : utiliser un **sous-gradient** à la place du gradient.

Idée intuitive : Le sous-gradient donne une direction de "descente approximative".



Méthode des sous-gradients

Algorithme :

1. Initialiser x_0
2. Pour $k = 0, 1, 2, ...$ répéter jusqu'à convergence :
    - Calculer **un sous-gradient** . $g_k \in \partial f(x_k)$
    - Choisir un pas de descente α_k
    - Mettre à jour : $x_{k+1} = x_k - \alpha_k g_k$

Choix du pas de “descente”

Importance du pas : Influence la vitesse de convergence et la stabilité de l'algorithme.

Types de pas :

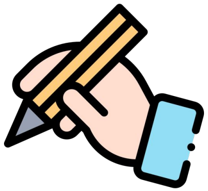
- **Pas constant :** (même valeur à chaque itération) $\alpha_k = \alpha$
- **Pas décroissant :** (diminue progressivement)
  - Exemples : $\alpha_k \to 0$ $\quad \alpha_k = \frac{1}{\sqrt{k}}$ $\quad \alpha_k = \frac{1}{k}$
- **Pas adaptatif :** Ajusté en fonction du comportement de l'algorithme.

Convergence de la méthode

Conditions de convergence :

- La fonction objectif doit être convexe (mais la convergence est assurée)
- Le pas de descente doit satisfaire certaines conditions (e.g., être suffisamment petit).

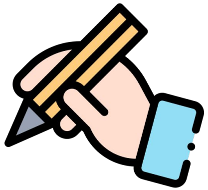
Vitesse de convergence : Généralement plus lente que la descente de gradient classique.



Quelques exercices sur la méthode

Rendez-vous au [TD 2], exercices partie II

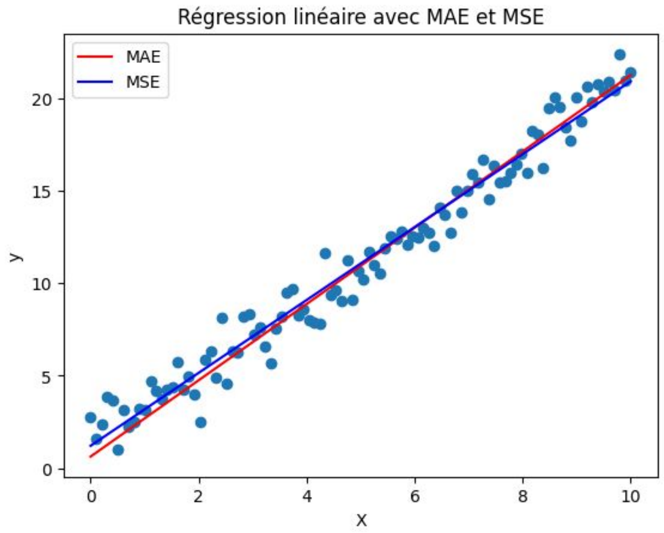
TP 1 - Régression avec MAE



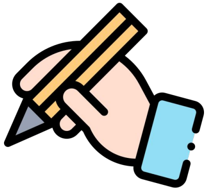
La MAE (*Mean Absolute Error)* est une fonction non différentiable.

Utilisons la MAE pour ajuster un modèle de régression

Créez une copie puis répondez aux questions du [notebook suivant]()

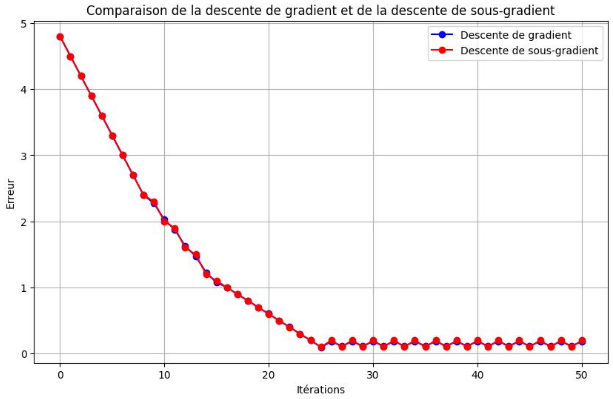


TP 2 - Norme L1



Nous allons voir différentes méthodes pour chercher le minimum de la norme L1 :

- Par une descente du gradient naïve
- Par un lissage de la fonction
- Par la méthode des sous-gradients



Rendez-vous au [notebook suivant].

Créez une copie puis répondez aux questions par des cellules de code et markdown

TP 3 - Débruitage

Reference Image

Noisy Image

Denoised Image

Rendez-vous dans le notebook suivant (créez une copie, puis suivez les questions)

https://colab.research.google.com/drive/16vGEETIT1SgV6K11urhoH1uRtiywLqF_?usp=sharing