# Introduction à l'Optimisation

## Cours - 4BIM

Carole FRINDEL[1]

[1]Laboratoire CREATIS
Bâtiment L. De Vinci, Campus INSA
carole.frindel@creatis.insa-lyon.fr

Octobre 2016

# Plan du cours

# Plan du cours

# Définitions

- *Optimum* : État, degré de développement d'un système jugé le plus favorable selon des critères donnés
- Si le système à optimiser est modélisable (mis sous forme d'équation), les outils d'optimisation sont utilisables
- Le processus d'optimisation comprend plusieurs étapes d'identification
  - variables de décision (paramètres)
  - influence des variables sur le système (modèle)
  - la fonction de coût
  - les contraintes

# Définitions

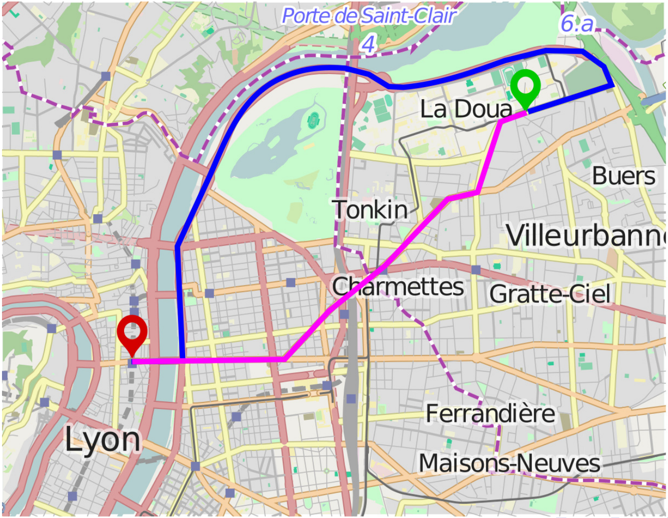
- *Optimum* : État, degré de développement d'un système jugé le plus favorable selon des critères donnés
- Si le système à optimiser est modélisable (mis sous forme d'équation), les outils d'optimisation sont utilisables
- Le processus d'optimisation comprend plusieurs étapes d'identification
    - variables de décision (paramètres)
    - influence des variables sur le système (modèle)
    - la fonction de coût
    - les contraintes

# Définitions

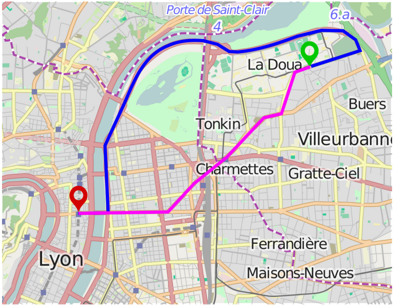
- *Optimum* : État, degré de développement d'un système jugé le plus favorable selon des critères donnés
- Si le système à optimiser est modélisable (mis sous forme d'équation), les outils d'optimisation sont utilisables
- Le processus d'optimisation comprend plusieurs étapes d'identification
  - variables de décision (paramètres)
  - influence des variables sur le système (modèle)
  - la fonction de coût
  - les contraintes

# Calculateur de trajet

# Calculateur de trajet



# Calculateur de trajet




- Quels sont les paramètres de décision ?
- Les fonctions de coût possibles ?
- Les contraintes ?

# Variables de décision

- Composantes du système sur lesquelles il est possible d'agir afin d'améliorer son état
- Généralement mises sous forme d'un vecteur noté $x = (x_1, x_2, \ldots, x_n)^T$

# La fonction objectif/coût

- Fonction qui permet d'évaluer l'état du système pour un jeu de paramètres donné
- Généralement notée $f(x) : \mathbb{R}^n \to \mathbb{R}$
- **Les paramètres sont optimaux quand $f$ est minimale**

# Les contraintes

- Contraintes précisant les valeurs que les variables peuvent prendre
- Elles définissent le domaine de définition de la fonction de coût

# Plan du cours

# Quelques exemples

## Applications

- Pilote automatique d'avion
- Séquence IRM
- Finance - Placements boursiers

## Identification du modèle

- Paramètres ?
- Fonction de coût ?
- Contraintes ?

# Quelques exemples

- Le *voyageur de commerce* est un problème classique d'optimisation complexe
- Il doit visiter $N$ villes données en passant par chaque ville exactement une fois
- Dans quel ordre enchainer les villes pour minimiser la distance parcourue ?

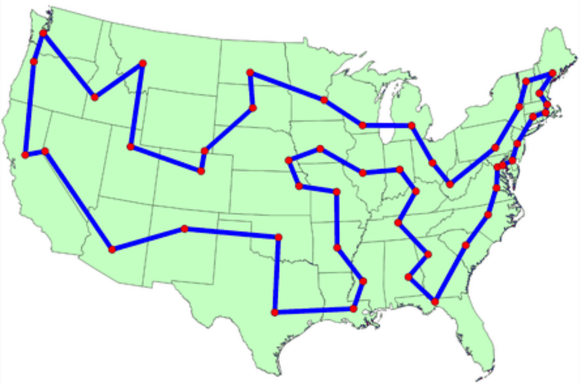
# Quelques exemples

- Le *voyageur de commerce* est un problème classique d'optimisation complexe
- Il doit visiter $N$ villes données en passant par chaque ville exactement une fois
- Dans quel ordre enchainer les villes pour minimiser la distance parcourue ?

# Quelques exemples

- Le *voyageur de commerce* est un problème classique d'optimisation complexe
- Il doit visiter $N$ villes données en passant par chaque ville exactement une fois
- Dans quel ordre enchainer les villes pour minimiser la distance parcourue ?

# Le voyageur de commerce



# Détermination du chemin optimal

- Nombre de permutations possibles : $\frac{(N-1)!}{2}$

| N | Nb de permutations | Temps de calcul |
|---|---|---|
| 5 | 12 | 12 $\mu$s |
| 10 | 181 440 | 0.18 s |
| 15 | $43x10^9$ | 12 h |
| 20 | $60e^{15}$ | 1928 ans |

- Primordial de trouver des méthodes (algorithmes) qui convergent vers la solution optimale sans évaluer l'ensemble des combinaisons

# Détermination du chemin optimal

- Nombre de permutations possibles : $\frac{(N-1)!}{2}$

| N | Nb de permutations | Temps de calcul |
|---|---|---|
| 5 | 12 | 12 $\mu$s |
| 10 | 181 440 | 0.18 s |
| 15 | $43x10^9$ | 12 h |
| 20 | $60e^{15}$ | 1928 ans |

- Primordial de trouver des méthodes (algorithmes) qui convergent vers la solution optimale sans évaluer l'ensemble des combinaisons

# Modélisation mathématique

- Soit la matrice $D_{ij}$ de taille $N \times N$ représentant la distance entre les villes $i$ et $j$

$$\mathbf{D} = \begin{array}{c} \\ Paris \\ Lyon \\ Grenoble \end{array} \begin{array}{c} \begin{array}{ccc} Paris & Lyon & Grenoble \end{array} \\ \begin{pmatrix} 0 & 477 & 572 \\ 477 & 0 & 111 \\ 572 & 111 & 0 \end{pmatrix} \end{array}$$

- Trouvons le vecteur de permutation $p \in \mathbb{N}^N$ qui minimise la fonction de coût $C$ suivante :

$$C : \mathbb{N}^N \to \mathbb{N}$$
$$C : p \mapsto C(p)$$

$$C(p) = \sum_{i=1}^{i=N-1} D_{p(i)p(i+1)}$$

# Modélisation mathématique

- Soit la matrice $D_{ij}$ de taille $N \times N$ représentant la distance entre les villes $i$ et $j$

$$\mathbf{D} = \begin{array}{c} \\ \textit{Paris} \\ \textit{Lyon} \\ \textit{Grenoble} \end{array} \begin{array}{c} \begin{array}{ccc} \textit{Paris} & \textit{Lyon} & \textit{Grenoble} \end{array} \\ \begin{pmatrix} 0 & 477 & 572 \\ 477 & 0 & 111 \\ 572 & 111 & 0 \end{pmatrix} \end{array}$$

- Trouvons le vecteur de permutation $p \in \mathbb{N}^N$ qui minimise la fonction de coût $C$ suivante :

$$\begin{aligned} C &: \mathbb{N}^N \to \mathbb{N} \\ C &: p \mapsto C(p) \end{aligned}$$

$$C(p) = \sum_{i=1}^{i=N-1} D_{p(i)p(i+1)}$$

# Exemple pour $N = 6$

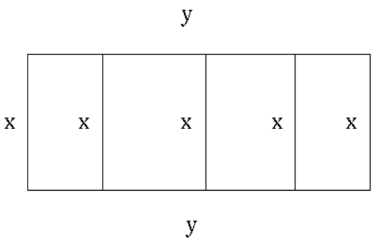
- Prenons $p = [6, 2, 4, 1, 5, 3]$
- $C(p) = D_{6,2} + D_{2,4} + D_{4,1} + D_{1,5} + D_{5,3}$
- L'enchainement des villes ($p$) est optimal si $C(p)$ est minimale ($\forall p$ dans l'espace des $p$ recevables)

# Exercices

- **Problème 1** : On veut construire un enclos rectangulaire avec trois cloisons parallèles à l'aide de 500m de clôture. Quelles sont les dimensions qui permettront de maximiser la superficie totale de l'enclos ?
- **Problème 2** : Une boîte rectangulaire ouverte à base carrée doit être faite à partir de 48 $cm^2$ de matériau. Quelles sont les dimensions qui permettront de maximiser le volume total de la boîte ?

Modéliser le problème sous la forme d'une fonction à minimiser ou à maximiser. Posez : la fonction de coût, les paramètres, les contraintes. Trouvez les paramètres optimaux grâce à la contrainte.

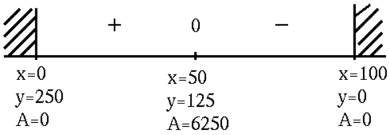
# Problème 1



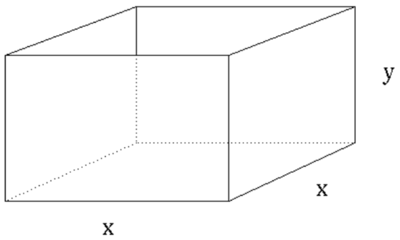

- La quantité de clôture nécessaire : 500 = 5x + 2y
- D'où y = 250 - (5/2) x
- Maximiser la surface totale : A = x y = 250x-(5/2)$x^2$

# Problème 1

- Maximiser la surface totale : f(x) = 250x-(5/2)$x^2$
- Dérivée : f'(x) = 250 - 5x = 5 (50 - x )
- D'où f'(x)=0 ssi x=50 et y= 125
- Notez que étant donné qu'il y a 5 longueurs de x dans cette construction et 500m de clôture, il en résulte que $0 \leq x \leq 100$



# Problème 2




- La superficie totale de la boîte : 48 = $x^2$ + 4(xy)
- D'où y = $\frac{12}{x} - \frac{x}{4}$
- Maximiser le volume total de la boîte : V = x x y = 12x-(1/4)$x^3$

# Plan du cours

# Modélisation, en bref...

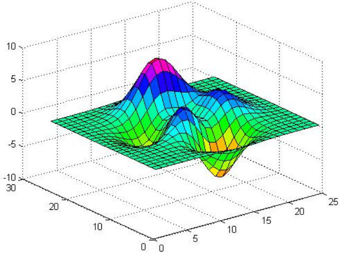
- Une bonne modélisation est fondamentale pour optimiser un système (variables, fonction de coût, contraintes)
- Dans ce cours, nous étudierons des problèmes d'optimisation
  - sans contraintes
  - avec des fonctions de coût différentiables
- **Optimiser un système revient à trouver l'état qui minimise la fonction de coût**

# Modélisation, en bref...

- Une bonne modélisation est fondamentale pour optimiser un système (variables, fonction de coût, contraintes)
- Dans ce cours, nous étudierons des problèmes d'optimisation
  - sans contraintes
  - avec des fonctions de coût différentiables
- **Optimiser un système revient à trouver l'état qui minimise la fonction de coût**

# Fonction de coût à minimiser

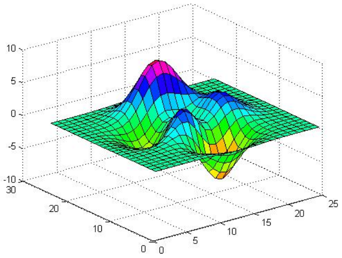
Que ce soit le trajet minimal du voyageur de commerce, le volume d'une boîte *etc.*, nous étudions alors la fonction de coût comme une fonction $\mathbb{R}^n \to \mathbb{R}$, avec $n$ le nombre de paramètres.




Visualisation avec $n = 2$

# Fonction de coût à minimiser

Que ce soit le trajet minimal du voyageur de commerce, le volume d'une boîte *etc.*, nous étudions alors la fonction de coût comme une fonction $\mathbb{R}^n \to \mathbb{R}$, avec $n$ le nombre de paramètres.



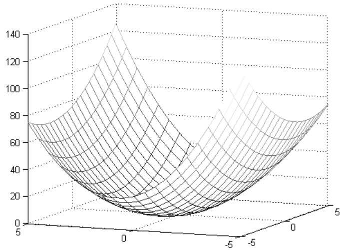

Visualisation avec $n = 2$

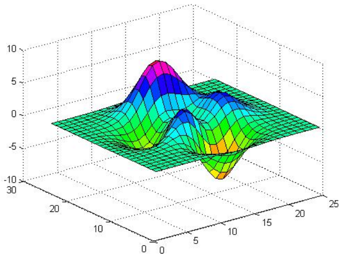
# Minimisation de $f$

- Minimiser une fonction de coût peut ne pas être trivial !
- Certaines fonctions (non convexes) présentent plusieurs minimas
- Potentiellement non dérivables
- Généralement impossible de calculer le coût sur l'ensemble de définition de $f$

# Champ d'étude

Note : Les outils présentés ci-après sont nécessaires mais insuffisants pour étudier le cas de fonctions à plusieurs minima locaux. Nous étudierons pour l'instant les cas de fonctions avec un seul minimum. Les cas des fonctions à minima locaux sera étudié en TD.



# Plan du cours

# Outils n-D

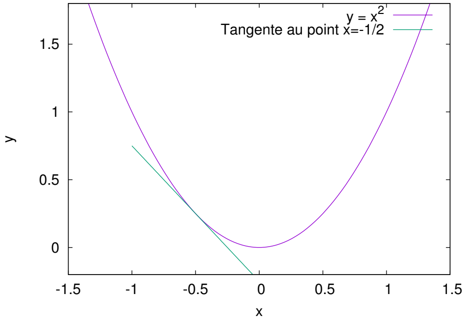
## Le gradient

Soit $f$ une fonction continue. La fonction notée $\nabla f(x) : \mathbb{R}^n \to \mathbb{R}^n$ est appelée gradient de $f$ et est définie par :

$$\nabla f(x) = \left( \frac{\partial f(x)}{\partial x_1}, \ldots, \frac{\partial f(x)}{\partial x_n} \right)^T \tag{1}$$

# Outils n-D

- En 1D le gradient est équivalent à la dérivée
- La valeur du gradient correspond à la pente de la tangente



# Outils n-D

## Dérivée directionnelle

Soit $f : \mathbb{R}^n \to \mathbb{R}$ admettant un gradient $\nabla f(x)$, et $d \in \mathbb{R}^n$. La dérivée directionnelle est le produit scalaire entre le gradient de $f$ et la direction $d$ :

$$\nabla f(x)^T d \tag{2}$$

- Comme en 1D, la dérivée directionnelle donne des indications sur la pente de $f$ dans la direction $d$:
  - $\nabla f(x)^T d < 0 \Leftrightarrow$ f est décroissante dans la direction $d$
  - $\nabla f(x)^T d > 0 \Leftrightarrow$ f est croissante dans la direction $d$

# Outils n-D

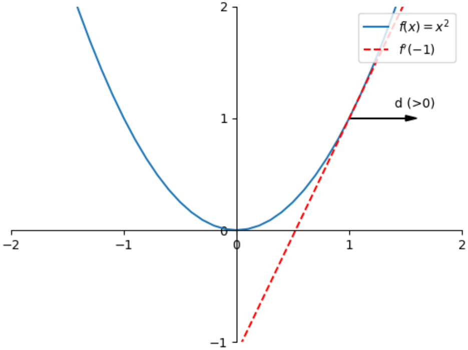
## Dérivée directionnelle

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ admettant un gradient $\nabla f(x)$, et $d \in \mathbb{R}^n$. La dérivée directionnelle est le produit scalaire entre le gradient de $f$ et la direction $d$ :

$$\nabla f(x)^T d \tag{2}$$

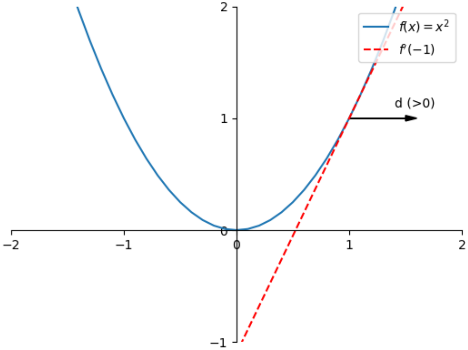
- Comme en 1D, la dérivée directionnelle donne des indications sur la pente de $f$ dans la direction $d$:
  - $\nabla f(x)^T d < 0 \Leftrightarrow$ f est décroissante dans la direction $d$
  - $\nabla f(x)^T d > 0 \Leftrightarrow$ f est croissante dans la direction $d$

# Direction de descente - illustration dans $\mathbb{R}^1$



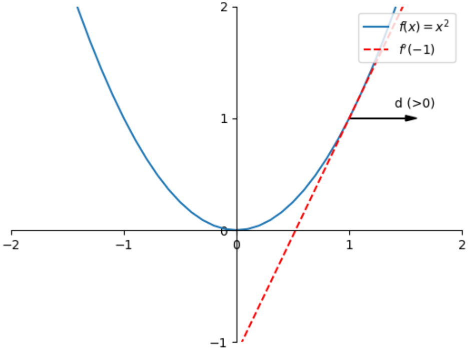

- $\nabla f(x)^T > 0$
- $d > 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$



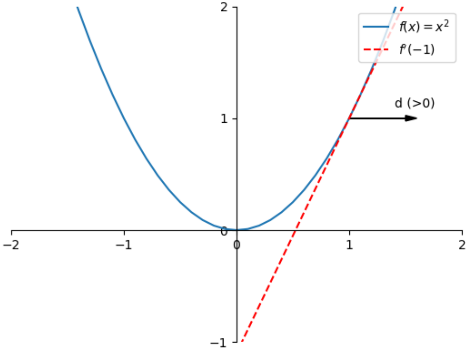

- $\nabla f(x)^T > 0$
- $d > 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$



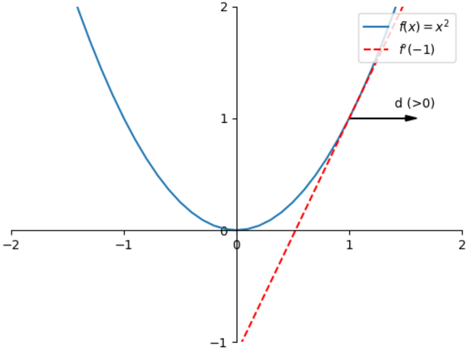

- $\nabla f(x)^T > 0$
- $d > 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$



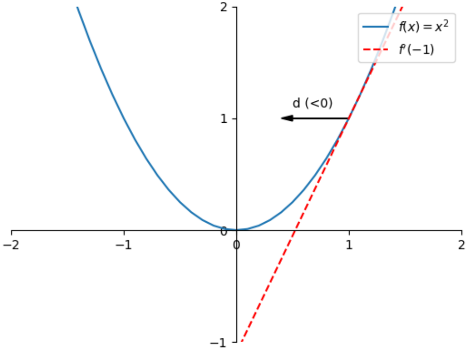

- $\nabla f(x)^T > 0$
- $d > 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$



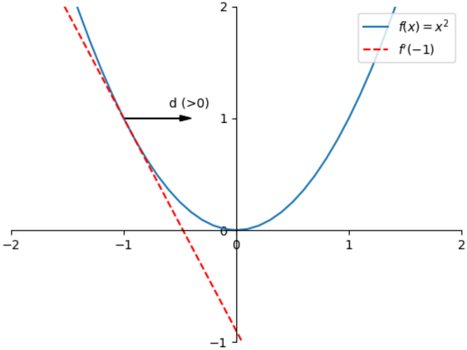

- $\nabla f(x)^T > 0$
- $d > 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$



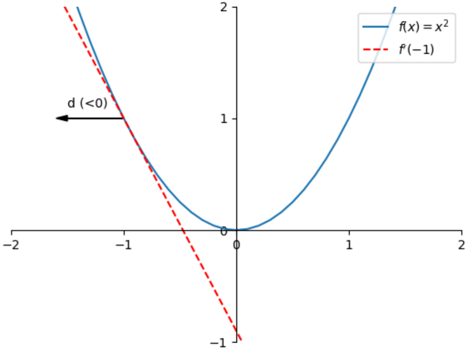

- $\nabla f(x)^T > 0$
- $d < 0$
- $\nabla f(x)^T d < 0$
- En allant dans cette direction, on "descend" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$




- $\nabla f(x)^T < 0$
- $d > 0$
- $\nabla f(x)^T d < 0$
- En allant dans cette direction, on "descend" la fonction.

# Direction de descente - illustration dans $\mathbb{R}^1$




- $\nabla f(x)^T < 0$
- $d < 0$
- $\nabla f(x)^T d > 0$
- En allant dans cette direction, on "remonte" la fonction.

# Direction de descente

## Direction de descente

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction différentiable. Soient $x, d \in \mathbb{R}^n$. La direction $d$ est une direction de descente en $x$ si :

$$\nabla f(x)^T d < 0 \tag{3}$$

## Direction de plus forte descente (inégalité de CS)

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction différentiable. Soient $x \in \mathbb{R}^n$, et $d^* = -\nabla f(x)$. Alors, $\forall d \in \mathbb{R}^n$ tel que $\|d\| = \|\nabla f(x)\|$, on a :

$$d^T \nabla f(x) \geq {d^*}^T \nabla f(x) = -\nabla f(x)^T \nabla f(x) \tag{4}$$

# Direction de descente

## Direction de descente

Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction différentiable. Soient $x$, $d \in \mathbb{R}^n$. La direction $d$ est une direction de descente en $x$ si :

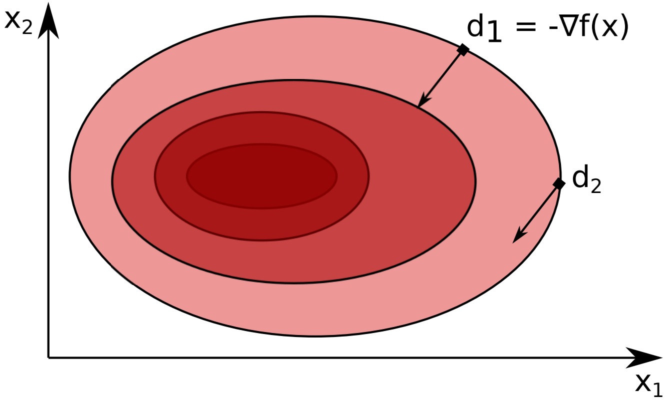
$$\nabla f(x)^T d < 0 \tag{3}$$

## Direction de plus forte descente (inégalité de CS)

Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction différentiable. Soient $x \in \mathbb{R}^n$, et $d^* = -\nabla f(x)$. Alors, $\forall d \in \mathbb{R}^n$ tel que $\|d\| = \|\nabla f(x)\|$, on a :

$$d^T \nabla f(x) \geq {d^*}^T \nabla f(x) = -\nabla f(x)^T \nabla f(x) \tag{4}$$

# Illustration de 2 directions de descente



# Plan du cours

# Hessienne

## Matrice Hessienne

Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction deux fois différentiable. La fonction notée $\nabla^2 f(x) : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ est appelée matrice hessienne de $f$, est toujours symétrique, et définie par :

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

# Convexité

- La hessienne donne des informations sur la convexité de $f$
- Si $\nabla^2 f(x)$ est définie positive (valeurs propres $> 0$) alors $f$ est strictement convexe.
- Dans ce cas, $f$ admet un minimum unique
- Exemple : $f(x, y) = x^2 + y^2$ est-elle convexe ?
- Et $f(x, y) = x^2 - y^2$ ?

# Convexité

- La hessienne donne des informations sur la convexité de $f$
- Si $\nabla^2 f(x)$ est définie positive (valeurs propres $> 0$) alors $f$ est strictement convexe.
- Dans ce cas, $f$ admet un minimum unique
- Exemple : $f(x, y) = x^2 + y^2$ est-elle convexe ?
- Et $f(x, y) = x^2 - y^2$ ?

# Convexité

- La hessienne donne des informations sur la convexité de $f$
- Si $\nabla^2 f(x)$ est définie positive (valeurs propres $> 0$) alors $f$ est strictement convexe.
- Dans ce cas, $f$ admet un minimum unique
- Exemple : $f(x, y) = x^2 + y^2$ est-elle convexe ?
- Et $f(x, y) = x^2 - y^2$ ?

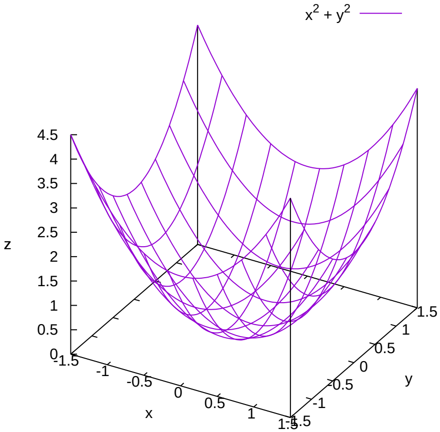
# Convexité

- La hessienne donne des informations sur la convexité de $f$
- Si $\nabla^2 f(x)$ est définie positive (valeurs propres $> 0$) alors $f$ est strictement convexe.
- Dans ce cas, $f$ admet un minimum unique
- Exemple : $f(x, y) = x^2 + y^2$ est-elle convexe ?
- Et $f(x, y) = x^2 - y^2$ ?

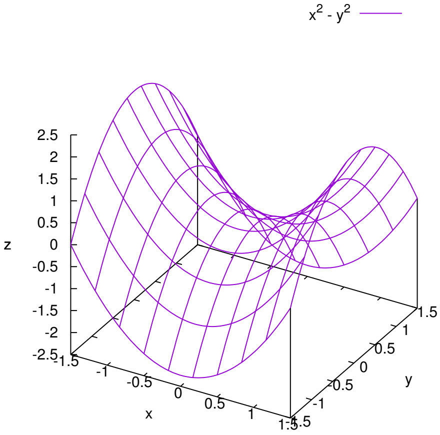
# Convexité (information globale)

- La fonction $f(x,y) = x^2 + y^2$ a pour valeurs propres 2 et 2 :
$$\nabla^2 f(x) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$
- Elle est convexe $\rightarrow$ minimum unique
- La fonction $f(x,y) = x^2 - y^2$ a pour valeurs propres -2 et 2:
$$\nabla^2 f(x) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$
- Elle n'est pas convexe $\rightarrow$ pas de minimum unique

# Illustration de la fonction $f(x, y) = x^2 + y^2$



# Illustration du point de selle pour $f(x, y) = x^2 - y^2$



# Courbure (information locale)

## Courbure

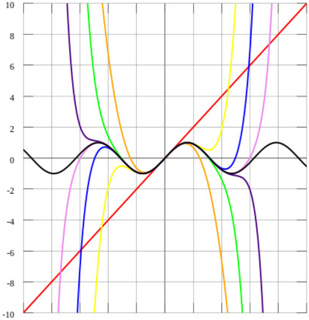
Soit $f : \mathbb{R}^n \to \mathbb{R}$ deux fois différentiable. Soient $x$ et $d \in \mathbb{R}^n$. La quantité :

$$\frac{d^T \nabla^2 f(x) d}{d^T d}$$

représente la courbure de la fonction $f$ en $x$ dans la direction $d$.

# Séries de Taylor

- En analyse, la série de Taylor d'une fonction f (au point a) est une série entière construite à partir de f et de ses dérivées successives en a
- La série aboutit à un polynôme : plus son **degré** est élevé, meilleure est l'approximation au point a



# Analogie avec les séries de Taylor

- La série de Taylor d'une fonction $f(x)$ infiniment dérivable en $a$ s'exprime grâce à ses dérivées successives :

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$

- A l'ordre n=1, on obtient :

$$f(x) = f(a) + (x-a)f'(a) + O(x^2)$$

# Résumé

- La dérivabilité d'ordre 1 (gradient) donne des informations sur la direction de descente
- La dérivabilité d'ordre 2 (hessienne) donne des informations sur la courbure (localement) et la convexité (globalement).
- Ces notions sont primordiales dans la recherche du minimum de la fonction de coût d'un problème d'optimisation

# Plan du cours

# Généralités

- Une méthode numérique d'optimisation cherche à minimiser la fonction de coût en suivant un certain schéma de convergence
- Chaque problème d'optimisation est unique $\rightarrow$ pas de méthode idéale

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

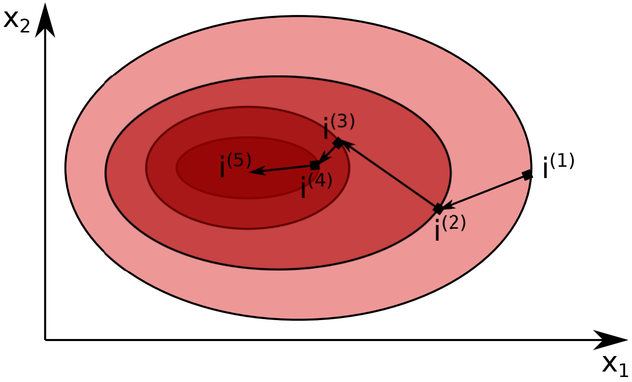
- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Principe itératif général

Le type d'algorithmes que nous étudierons le plus est un algorithme itératif.

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k$
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

# Illustration d'une suite d'itérations



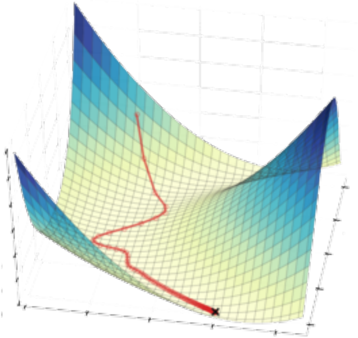
# Plan du cours

# Méthode de la plus forte pente

Consiste à choisir $d$ comme la direction de plus forte pente (orthogonale aux lignes de niveaux)

- Soit $x_0 \in \mathbb{R}^n$. Posons $k = 0$.
- Tant que [conditions d'arrêt]
  - Choisir une direction de descente $d_k^\star$ (rappel : $d\star = -\nabla f(x)^T$)
  - Choisir un pas $\alpha_k > 0$
  - Calculer $x_{k+1} = x_k + \alpha_k d_k$
  - $k = k + 1$

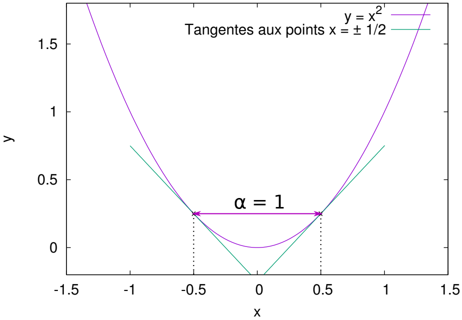
# Illustration de la plus grande pente



# Choix du pas de convergence

Algorithme itératif qui progresse à chaque étape d'une valeur pondérée par un pas : $x_{k+1} = x_k - \alpha_k \nabla f(x)$

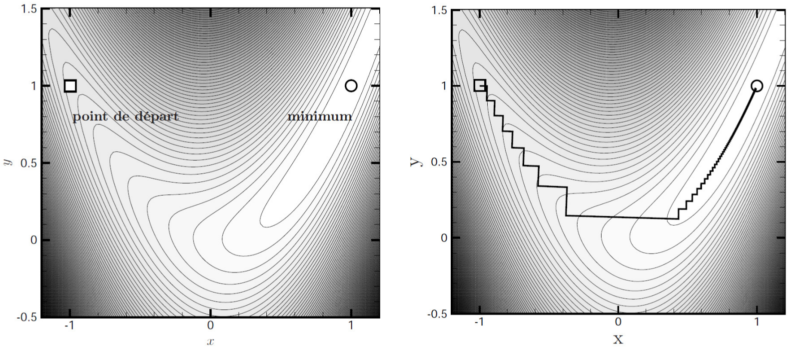
- Le choix du pas de convergence $\alpha_k$ est crucial
- Le plus simple : $\alpha_k = \alpha = cst \rightarrow$ risqué !



# Choix du pas de convergence

- Plusieurs méthodes d'optimisation existent pour le choisir
- Principe de la méthode "line search" :
  - A l'itération $k$
  - Définir une direction de descente donnée $d_k$
  - Choisir $\alpha > 0$ pour minimiser $h(\alpha) = f(x_k + \alpha d_k)$
  - Calcul de $h'(\alpha) = 0 \rightarrow$ nouveau problème d'optimisation !
- Compromis nécessaire entre pas optimal et temps de calcul

# Méthode de la plus forte pente avec pas optimal



# Points clés

## Rapidité

- Temps de calcul de la direction de descente à chaque itération
- Nombre total d'itérations

## Seuil de convergence

- On atteint très rarement le minimum exact $\rightarrow$ définition d'une tolérance de convergence basé sur :
  - nombre max d'itérations
  - la norme du gradient
  - la norme du pas de convergence

# Critères d'arrêt

- Nombre max d'itérations
  - + Simple, évite les temps de calcul trop long
  - – Pas relié à la convergence réelle de l'algo
- Norme du gradient
  - + Assez fiable
  - – Peut échouer avec des fonctions à décroissance faible
- Norme du pas de convergence
  - + Assez fiable
  - – Dépendant de la méthode de calcul du pas
- En pratique : combinaison de plusieurs critères d'arrêt

# Résumé sur l'ordre 1

- Méthode simple à mettre en œuvre
- Minimise forcément la fonction de coût à chaque itération
- Peu de calculs par itération
- Peu efficace dans le cas de fonctions à décroissance lente

# Plan du cours

# Méthode de Newton

- Utilisation de la dérivée seconde pour le calcul de la direction $d$
- Prise en compte de l'information de courbure
- Elle se base sur une approximation quadratique de $f$ en $x$

# Modèle quadratique

## Modèle quadratique d'une fonction

Soit $f : \mathbb{R}^n \to \mathbb{R}$ deux fois différentiable. Le modèle quadratique de $f$ en $\hat{x}$ est une fonction $m_{\hat{x}} : \mathbb{R}^n \to \mathbb{R}$ définie par :

$$m_{\hat{x}}(x) = f(\hat{x}) + (x - \hat{x})^T \nabla f(\hat{x}) + \frac{1}{2}(x - \hat{x})^T \nabla^2 f(\hat{x})(x - \hat{x})$$

où $\nabla f$ est le gradient de $f$ et $\nabla^2(f)$ la hessienne. En posant $d = x - \hat{x}$, on obtient la formule équivalente :

$$m_{\hat{x}}(\hat{x} + d) = f(\hat{x}) + d^T \nabla f(\hat{x}) + \frac{1}{2} d^T \nabla^2 f(\hat{x}) d$$

# Modèle quadratique

- Le modèle quadratique est une approximation (au second ordre) de la fonction $f$ en $\hat{x}$ (séries de Taylor)
- Cherchons le minimum de $m_{\hat{x}}$ :

$$\min_{d \in \mathbb{R}^n} m_{\hat{x}}(\hat{x}+d) = f(\hat{x}) + d^T \nabla_x f(\hat{x}) + \frac{1}{2} d^T \nabla_x^2 f(\hat{x}) d$$

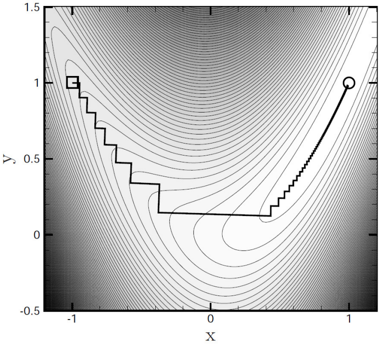
$$\Leftrightarrow \nabla_d m_{\hat{x}}(\hat{x}+d) = \nabla_x f(\hat{x}) + \nabla_x^2 f(\hat{x}) d = 0$$

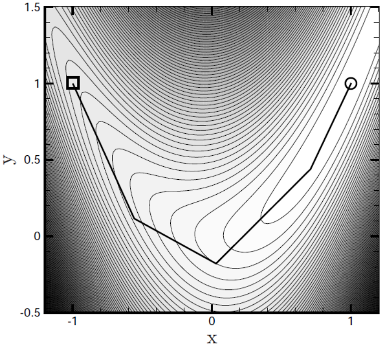
$$\Leftrightarrow d = -\frac{\nabla_x f(\hat{x})}{\nabla_x^2 f(\hat{x})}$$

# Modèle quadratique : algorithme

- Condition suffisante d'optimalité : il faut que $\nabla^2 f(\hat{x})$ soit définie positive
- Dans ce cas, on minimise **en une seule itération** le modèle quadratique
- On peut alors définir l'algorithme suivant :
- Choisir $k = 0$ et $x_0$. Tant que pas convergence :
  - Calculer le modèle quadratique en $x_k$ : $m_{x_k}(x)$
  - Calculer $d_k = \min_d [m_{x_k}(x_k + d)] = -\frac{\nabla f(x_k)}{\nabla^2 f(x_k)}$
  - $x_{k+1} = x_k + d_k$
  - $k = k + 1$

# Comparaison des deux ordres



# Modèle quadratique : exemple

- Soit la fonction $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculons $m_{x_k}(x)$ au point $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$

# Modèle quadratique : exemple

- Soit la fonction $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculons $m_{x_k}(x)$ au point $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$

# Modèle quadratique : exemple

- Soit la fonction $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculons $m_{x_k}(x)$ au point $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calcul des dérivées :

$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- D'où $m_3(x) = 7x^2 - 48x + 81$

# Modèle quadratique : exemple

- Soit la fonction $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculons $m_{x_k}(x)$ au point $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calcul des dérivées :

$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- D'où $m_3(x) = 7x^2 - 48x + 81$

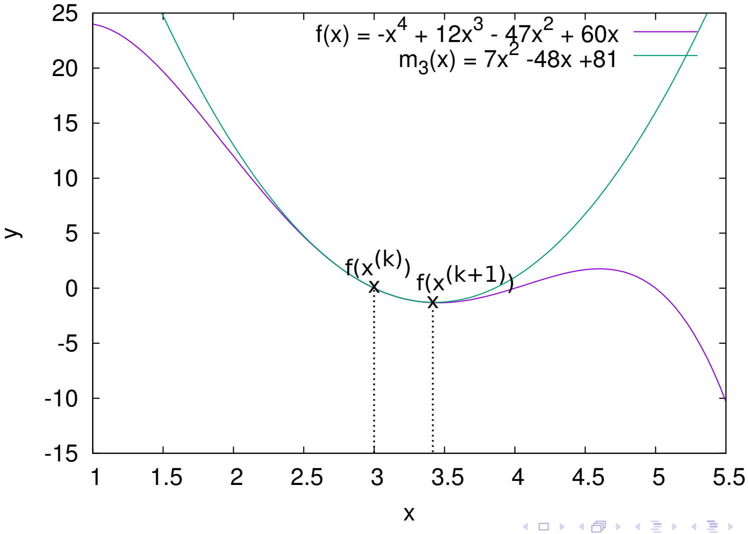
# Modèle quadratique : exemple

- Soit la fonction $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculons $m_{x_k}(x)$ au point $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calcul des dérivées :

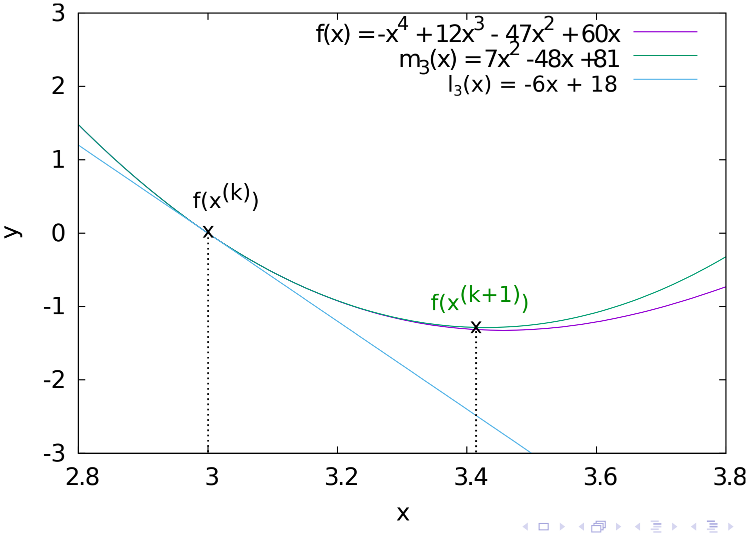
$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- D'où $m_3(x) = 7x^2 - 48x + 81$

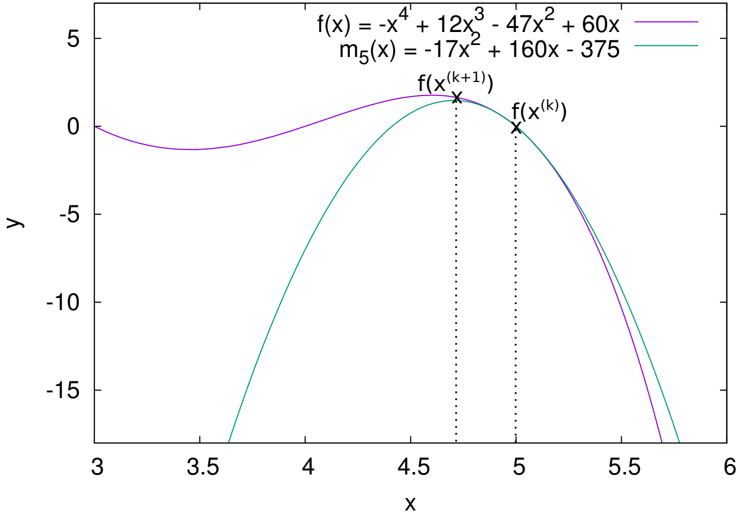
# Modèle quadratique : exemple



# Modèle quadratique : zoom



# Modèle quadratique : cas critique



# Modèle quadratique : bilan

- Avantages :
  - Généralement moins d'itérations nécessaires que la méthode de la plus forte pente
  - Optimale pour les fonctions de coût quadratiques
- Limitations :
  - Pas de distinctions minimas, maximas, point de selle
  - Calcul de la dérivée seconde à chaque itération
  - Le hessien doit être défini positif

# Extensions de la méthode de Newton

- Méthodes de Quasi-Newton
- Elles évitent le calcul (généralement long) de la Hessienne
- On peut citer les méthodes suivantes :
  - Davidon–Fletcher–Powell (DFP)
  - Symmetric rank-one (SR1)
  - Broyden–Fletcher–Goldfarb–Shanno (BFGS)

# Pour aller plus loin...

- *Introduction à l'optimisation différentiable*, Michel Bierlaire, 2006
- *Optimisation et analyse convexe*, Jean-Baptiste Hiriart-Urruty, 2012
- *Conditioning of quasi-Newton methods for function minimization*, D. F. Shanno, Math. Comp. 24 (1970), pp. 647-656
- Librairie Python : *http://docs.scipy.org/doc/scipy/reference/tutorial/optimize.html*
- Librairie C/C++ : *http://public.kitware.com/vxl/doc/release/books/core/book_6.html*