

Optimisation : descente de gradient

David Loiseaux et Fanny Simões

3 Mars 2022

Table of Contents

Étude de fonction

Convexité

Différentiabilité

Optimisation différentiable, convexe et sans contrainte

Méthode de descente de gradient

Algorithme → Notebook

Garanties théoriques et variantes (très) succinctement

Application

Deep learning

Machine learning

Table of Contents

Étude de fonction

Convexité

Différentiabilité

Optimisation différentiable, convexe et sans contrainte

Méthode de descente de gradient

Algorithme → Notebook

Garanties théoriques et variantes (très) succinctement

Application

Deep learning

Machine learning

Convexité d'un ensemble

Définition

Un ensemble C est dit convexe si, pour tout $w, y \in C$,

$$\alpha \in [0, 1] \implies z = \alpha w + (1 - \alpha)y \in C.$$

De manière équivalente, C est convexe si pour tout $x, y \in C$

$$[x, y] \in C.$$

Convexité d'un ensemble

Définition

Un ensemble C est dit convexe si, pour tout $w, y \in C$,

$$\alpha \in [0, 1] \implies z = \alpha w + (1 - \alpha)y \in C.$$

De manière équivalente, C est convexe si pour tout $x, y \in C$

$$[x, y] \in C.$$

Example (de convexes)

Les polyèdres, les boules, les ellipses, les cônes, les espaces vectoriels,...

Convexité d'un ensemble

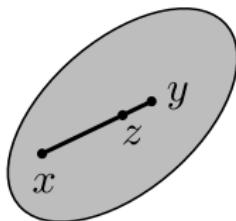
Définition

Un ensemble C est dit convexe si, pour tout $w, y \in C$,

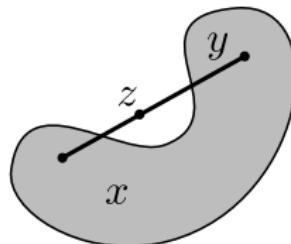
$$\alpha \in [0, 1] \implies z = \alpha w + (1 - \alpha)y \in C.$$

De manière équivalente, C est convexe si pour tout $x, y \in C$

$$[x, y] \in C.$$



Convexe



Pas convexe

Convexité

Définition

Une fonction f est dite convexe si son graphe se situe en dessous de ses cordes, c'est à dire si $\forall x, y \in \mathbb{R}^d$, et $0 \leq \alpha \leq 1$,

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

Convexité

Définition

Une fonction f est dite convexe si son graphe se situe en dessous de ses cordes, c'est à dire si $\forall x, y \in \mathbb{R}^d$, et $0 \leq \alpha \leq 1$,

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

strictement convexe si

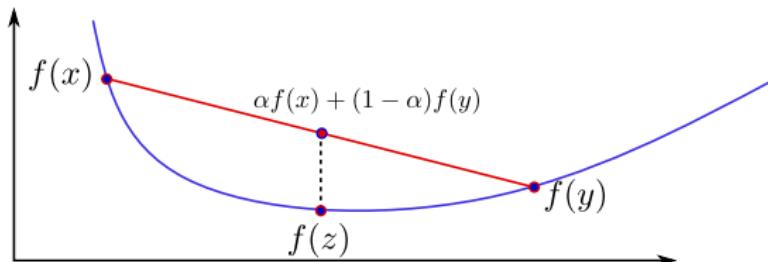
$$f(\alpha x + (1 - \alpha)y) < \alpha f(x) + (1 - \alpha)f(y).$$

Convexité

Définition

Une fonction f est dite convexe si son graphe se situe en dessous de ses cordes, c'est à dire si $\forall x, y \in \mathbb{R}^d$, et $0 \leq \alpha \leq 1$,

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

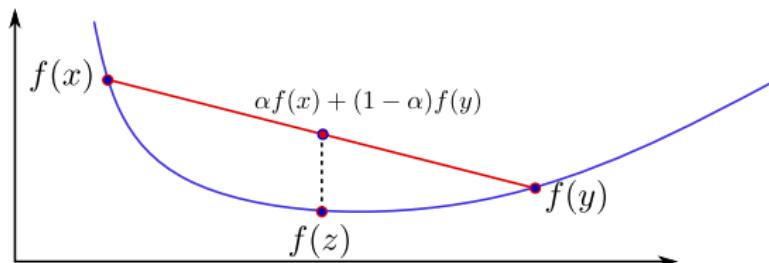


Convexité

Définition

Une fonction f est dite convexe si son graphe se situe en dessous de ses cordes, c'est à dire si $\forall x, y \in \mathbb{R}^d$, et $0 \leq \alpha \leq 1$,

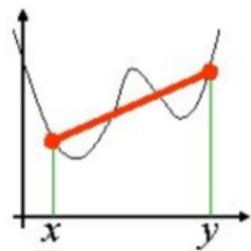
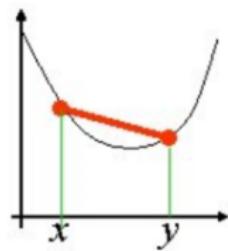
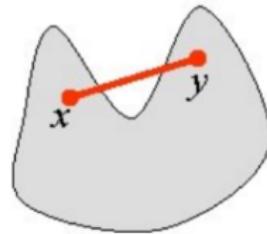
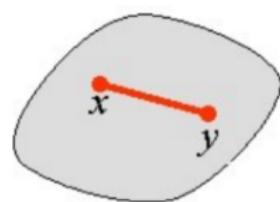
$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$



→ Épigraphe convexe.

Exemples

Ensemble convexe ensemble **non** convexe - fonction convexe fonction **non** convexe



Fonction différentiable

Définition

Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable en $x \in \mathbb{R}^n$ lorsque sa «meilleure approximation linéaire» $df_x : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est continue.

Fonction différentiable

Définition

Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable en $x \in \mathbb{R}^n$ lorsque sa «meilleure approximation linéaire» $df_x : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est continue.
Autrement dit, il existe une fonction linéaire continue df_x telle que

$$f(x + h) = f(x) + df_x(h) + o(\|h\|)$$

Fonction différentiable

Définition

Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable en $x \in \mathbb{R}^n$ lorsque sa «meilleure approximation linéaire» $df_x : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est continue.

C'est équivalent à ce que les dérivées partielles de f en x existent et soient *continues*.

Fonction différentiable

Définition

Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable en $x \in \mathbb{R}^n$ lorsque sa «meilleure approximation linéaire» $df_x : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est continue.

C'est équivalent à ce que les dérivées partielles de f en x existent et soient *continues*.

Dérivée partielle de f par rapport à x_i au point x :

$$\frac{\partial f}{\partial x_i}(x) = (x_i \mapsto f(x_1, \dots, x_i, \dots, x_n))'$$

Fonction différentiable

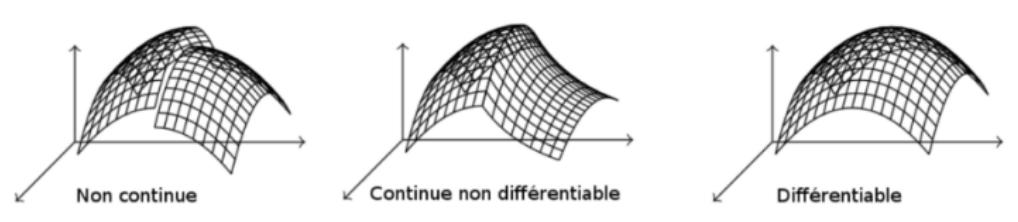
Définition

Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est différentiable en $x \in \mathbb{R}^n$ lorsque sa «meilleure approximation linéaire» $df_x : \mathbb{R}^n \rightarrow \mathbb{R}^m$ est continue.

C'est équivalent à ce que les dérivées partielles de f en x existent et soient *continues*.

Dérivée partielle de f par rapport à x_i au point x :

$$\frac{\partial f}{\partial x_i}(x) = (x_i \mapsto f(x_1, \dots, x_i, \dots, x_n))'$$



Exemple

Gradient

Définition

Le gradient $\nabla f(x)$ d'une fonction f de \mathbb{R}^n sur \mathbb{R} au point x est le vecteur dont les composantes sont les dérivées partielles de f .

$$\text{grad}f(x) = \nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x) \end{bmatrix} \quad (1)$$

Gradient

Définition

Le gradient $\nabla f(x)$ d'une fonction f de \mathbb{R}^n sur \mathbb{R} au point x est le vecteur dont les composantes sont les dérivées partielles de f .

$$\text{grad}f(x) = \nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x) \end{bmatrix} \quad (1)$$

→ si f est différentiable en x

$$\begin{aligned} f(x+h) &= f(x) + \nabla f(x) \cdot h + o(h) \\ &= f(x) + \sum_{i=1}^n \frac{\partial f}{\partial x_i}(x) h_i + o(h) \end{aligned}$$

Exemple : calcul de gradient

- ▶ Fonction : $f(x, y) = (x + y)(y + 1) = xy + x + y^2 + y$

Exemple : calcul de gradient

- ▶ Fonction : $f(x, y) = (x + y)(y + 1) = xy + x + y^2 + y$
- ▶ Dérivées partielles :
 $\frac{\partial}{\partial x}f(x, y) = y + 1$ et $\frac{\partial}{\partial y}f(x, y) = x + 2y + 1$

Exemple : calcul de gradient

- ▶ Fonction : $f(x, y) = (x + y)(y + 1) = xy + x + y^2 + y$
- ▶ Dérivées partielles :
 $\frac{\partial}{\partial x}f(x, y) = y + 1$ et $\frac{\partial}{\partial y}f(x, y) = x + 2y + 1$
- ▶ Gradient : $\nabla f(x, y) = [y + 1, x + 2y + 1]^T$

Exemple : calcul de gradient

- ▶ Fonction : $f(x, y) = (x + y)(y + 1) = xy + x + y^2 + y$
- ▶ Dérivées partielles :
 $\frac{\partial}{\partial x}f(x, y) = y + 1$ et $\frac{\partial}{\partial y}f(x, y) = x + 2y + 1$
- ▶ Gradient : $\nabla f(x, y) = [y + 1, x + 2y + 1]^T$
- ▶ Gradient au point (3,5) : $\nabla f(3, 5) = [6, 14]^T$

Points critiques

Un point x est un *point critique* si $df_x = 0$, ou encore $\nabla f(x) = 0$.

Types de points critiques :

- ▶ minimum (local)

Points critiques

Un point x est un *point critique* si $df_x = 0$, ou encore $\nabla f(x) = 0$.

Types de points critiques :

- ▶ minimum (local)
- ▶ maximum (local)

Points critiques

Un point x est un *point critique* si $df_x = 0$, ou encore $\nabla f(x) = 0$.

Types de points critiques :

- ▶ minimum (local)
- ▶ maximum (local)
- ▶ point de selle.

Points critiques

Un point x est un *point critique* si $df_x = 0$, ou encore $\nabla f(x) = 0$.

Types de points critiques :

- ▶ minimum (local)
- ▶ maximum (local)
- ▶ point de selle.

→ Comment on calcule ça ?

Convexité et minimum

f strictement convexe \implies au plus un minumum.

Convexité et minimum

f strictement convexe \implies au plus un minumum.
—> s'il y en a deux : par exemple x et y

$$f(0.5x + 0.5y) < 0.5f(x) + 0.5f(y) = \min f$$

Convexité et minimum

f strictement convexe \implies au plus un minimum.
→ s'il y en a deux : par exemple x et y

$$f(0.5x + 0.5y) < 0.5f(x) + 0.5f(y) = \min f$$

Un point critique x d'une fonction convexe f est un minimum.

Convexité et minimum

f strictement convexe \implies au plus un minumum.

→ s'il y en a deux : par exemple x et y

$$f(0.5x + 0.5y) < 0.5f(x) + 0.5f(y) = \min f$$

Un point critique x d'une fonction convexe f est un minimum.

→ (En dimension 1) si c est un point critique la convexité implique

$$f(x) \geq f(c) + f'(c)(x - c) = f(c)$$

Différentielle d'ordre 2.

Une fonction f est 2 fois différentiable en un point x si elle est différentiable sur un voisinage de x et si $x \mapsto df_x$ est différentiable en x .

Différentielle d'ordre 2.

Une fonction f est 2 fois différentiable en un point x si ses dérivées partielles d'ordre 2 en x :

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$$

existent et sont continues.

Différentielle d'ordre 2.

Une fonction f est 2 fois différentiable en un point x si ses dérivées partielles d'ordre 2 en x :

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$$

existent et sont continues.

→ (Théorème de Schwartz) Dans ce cas

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(x) = \frac{\partial^2 f}{\partial x_j \partial x_i}$$

On a :

$$f(x + h) = f(x) + \nabla f(x) \cdot h + \frac{1}{2} \sum_{1 \leq i, j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j + o(h^2)$$

Hessienne

On veut simplifier

$$\frac{1}{2} \sum_{1 \leq i,j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j$$

Hessienne

On veut simplifier

$$\frac{1}{2} \sum_{1 \leq i,j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j$$

→ on adore les matrices, donc on définit

$$\text{Hess } f(x) = \begin{pmatrix} \partial_{x_1} \partial_{x_1} f(x) & \cdots & \partial_{x_1} \partial_{x_n} f(x) \\ \vdots & \ddots & \vdots \\ \partial_{x_n} \partial_{x_1} f(x) & \cdots & \partial_{x_n} \partial_{x_n} f(x) \end{pmatrix}$$

Hessienne

On veut simplifier

$$\frac{1}{2} \sum_{1 \leq i, j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j$$

→ on adore les matrices, donc on définit

$$\text{Hess } f(x) = \begin{pmatrix} \partial_{x_1} \partial_{x_1} f(x) & \cdots & \partial_{x_1} \partial_{x_n} f(x) \\ \vdots & \ddots & \vdots \\ \partial_{x_n} \partial_{x_1} f(x) & \cdots & \partial_{x_n} \partial_{x_n} f(x) \end{pmatrix}$$

et on remarque :

$$\frac{1}{2} \sum_{1 \leq i, j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j = \frac{1}{2} h^T \text{Hess } f(x) h.$$

Hessienne

On veut simplifier

$$\frac{1}{2} \sum_{1 \leq i, j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j$$

→ on adore les matrices, donc on définit

$$\text{Hess } f(x) = \begin{pmatrix} \partial_{x_1} \partial_{x_1} f(x) & \cdots & \partial_{x_1} \partial_{x_n} f(x) \\ \vdots & \ddots & \vdots \\ \partial_{x_n} \partial_{x_1} f(x) & \cdots & \partial_{x_n} \partial_{x_n} f(x) \end{pmatrix}$$

et on remarque :

$$\frac{1}{2} \sum_{1 \leq i, j \leq n} \frac{\partial^2 f}{\partial x_i \partial x_j}(x) h_i h_j = \frac{1}{2} h^T \text{Hess } f(x) h.$$

Finalement,

$$f(x + h) = f(x) + \nabla f(x) \cdot h + \frac{1}{2} h^T \text{Hess } f(x) h + o(h^2)$$

Types de points critiques

Si x est un point critique de f ,

$$f(x + h) = f(x) + \nabla f(x) \cdot h + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Types de points critiques

Si x est un point critique de f ,

$$f(x + h) = f(x) + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Types de points critiques

Si x est un point critique de f ,

$$f(x + h) = f(x) + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Donc si pour tout h

$$h^T \text{Hess}f(x)h \geq 0$$

Alors x est un minimum local

Types de points critiques

Si x est un point critique de f ,

$$f(x + h) = f(x) + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Donc si pour tout h

$$h^T \text{Hess}f(x)h \leq 0$$

Alors x est un maximum local

Types de points critiques

Si x est un point critique de f ,

$$f(x + h) = f(x) + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Types de points critiques

Si x est un point critique de f ,

$$f(x+h) = f(x) + \frac{1}{2} h^T \text{Hess}f(x)h + o(h^2)$$

Sinon, il faut dériver plus ...

Exemple :

$$f : x \longmapsto \pm x^4$$

ou

$$f : x, y \mapsto x^4 - y^4$$

Matrices Positives (J'avais oublié ça au premier cours)

On a vu que ça nous arrangeait bien si la matrice $\text{Hess}f(x)$ vérifiait pour tout $h \in \mathbb{R}^n$

$$h^T \text{Hess}f(x)h \geq 0$$

Matrices Positives (J'avais oublié ça au premier cours)

On a vu que ça nous arrangeait bien si la matrice $\text{Hess}f(x)$ vérifiait pour tout $h \in \mathbb{R}^n$

$$h^T \text{Hess}f(x)h \geq 0$$

En général si $x^T Ax \geq 0$ pour tout x on dit que la matrice A est *positive*.

Matrices Positives (J'avais oublié ça au premier cours)

On a vu que ça nous arrangeait bien si la matrice $\text{Hess}f(x)$ vérifiait pour tout $h \in \mathbb{R}^n$

$$h^T \text{Hess}f(x)h \geq 0$$

En général si $x^T Ax \geq 0$ pour tout x on dit que la matrice A est *positive*.

→ Pour vérifier ça en pratique on utilise

A positive \Leftrightarrow Toutes les valeurs propres de A sont positives.

Matrices Positives (J'avais oublié ça au premier cours)

On a vu que ça nous arrangeait bien si la matrice $\text{Hess}f(x)$ vérifiait pour tout $h \in \mathbb{R}^n$

$$h^T \text{Hess}f(x)h \geq 0$$

En général si $x^T Ax \geq 0$ pour tout x on dit que la matrice A est *positive*.

→ Pour vérifier ça en pratique on utilise

A positive \Leftrightarrow Toutes les valeurs propres de A sont positives.

Remarque : f (strictement) convexe sur $\mathbb{R}^n \Leftrightarrow \text{Hess}f(x)$ est (strictement) positive pour tout $x \in \mathbb{R}^n$.

Formes quadratique – pas très important

Si la hessienne $\text{Hess } f(x)$ de f est diagonale et que x est un point critique de f ,

$$f(x + h) = f(x) + \sum_i \lambda_i h_i^2 + o(h)$$

Table of Contents

Étude de fonction

Convexité

Différentiabilité

Optimisation différentiable, convexe et sans contrainte

Méthode de descente de gradient

Algorithme → Notebook

Garanties théoriques et variantes (très) succinctement

Application

Deep learning

Machine learning

Descente de gradient

Minimiser en pratique ?

Descente de gradient

Minimiser en pratique ?

On part de n'importe quel point $x \in \mathbb{R}^n$; et on aimerait le faire converger vers un point $x^* \in \mathbb{R}^n$ tel que

$$f(x^*) = \min_{\mathbb{R}^n} f$$

Descente de gradient

Minimiser en pratique ?

On part de n'importe quel point $x \in \mathbb{R}^n$; et on aimerait le faire converger vers un point $x^* \in \mathbb{R}^n$ tel que

$$f(x^*) = \min_{\mathbb{R}^n} f$$

Remarque : Si on veut trouver le minimum sur un sous espace de $\mathbb{R}^n \rightarrow$ il suffit de se restreindre.

Descente de gradient

Minimiser en pratique ?

On part de n'importe quel point $x \in \mathbb{R}^n$; et on aimerait le faire converger vers un point $x^* \in \mathbb{R}^n$ tel que

$$f(x^*) = \min_{\mathbb{R}^n} f$$

Remarque : Si on veut trouver le minimum sur un sous espace de $\mathbb{R}^n \rightarrow$ il suffit de se restreindre. **Idée :** on regarde la dérivée (ou le gradient) et on déplace x itérativement vers là où ça descend.

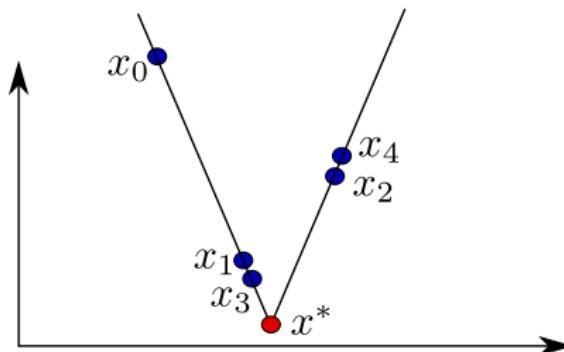
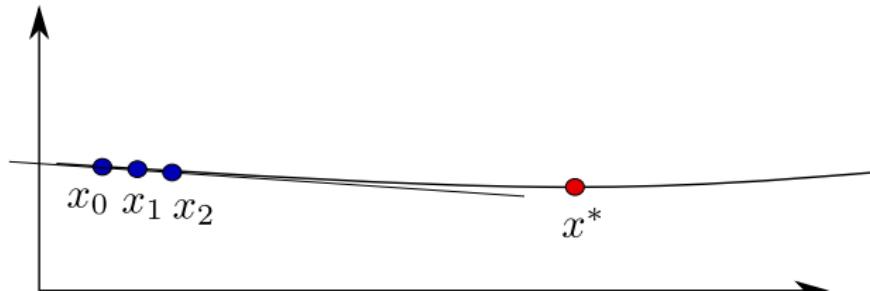
$$\begin{cases} \text{On choisit } x_0 \in \mathbb{R}^n \\ x_{t+1} := x_t - \rho \nabla f(x_t) \end{cases}$$

pour un pas $\rho > 0$ à choisir au préalable.

Notebook

Forte (μ -)convexité et L -régularité

Pas d'hypothèses = pas de garanties de convergence :



Forte (μ -)convexité et L -régularité

Pas d'hypothèses = pas de garanties de convergence :

On doit par exemple supposer que f est μ -fortement convexe
i.e. les valeurs propres de $\text{Hess}(f)$ ne tombent jamais en dessous de $\mu > 0$.

Forte (μ -)convexité et L -régularité

Pas d'hypothèses = pas de garanties de convergence :

On doit par exemple supposer que f est μ -fortement convexe
i.e. les valeurs propres de $\text{Hess}(f)$ ne tombent jamais en dessous de $\mu > 0$.

Et / ou f est L -régulière *i.e.* les valeurs propres de $\text{Hess}(f)$ ne dépassent jamais L .

Garanties

Cas le plus gentil :

On suppose f L -régulière et μ -fortement convexe ;

Garanties

Cas le plus gentil :

On suppose f L -régulière et μ -fortement convexe ; Alors, en choisissant le pas constant $\rho = 1/L$:

$$f(x_t) - f(x^*) \leq e^{-t\frac{\mu}{L}}(f(x_0) - f(x^*))$$

Garanties

Cas le plus gentil :

On suppose f L -régulière et μ -fortement convexe ; Alors, en choisissant le pas constant $\rho = 1/L$:

$$f(x_t) - f(x^*) \leq e^{-t\frac{\mu}{L}}(f(x_0) - f(x^*))$$

Cas un peu moins gentil :

On suppose f L -régulière et convexe ;

Garanties

Cas le plus gentil :

On suppose f L -régulière et μ -fortement convexe ; Alors, en choisissant le pas constant $\rho = 1/L$:

$$f(x_t) - f(x^*) \leq e^{-t\frac{\mu}{L}}(f(x_0) - f(x^*))$$

Cas un peu moins gentil :

On suppose f L -régulière et convexe ; Alors, en choisissant le pas constant $\rho = 1/L$:

$$f(x_t) - f(x^*) \leq \frac{L}{2t} \|x_0 - x^*\|_2^2$$

Variantes

- ▶ Changer la taille du pas en fonction du temps t ,

Variantes

- ▶ Changer la taille du pas en fonction du temps t ,
- ▶ Rajouter de l'inertie à la descente de gradient pour garder l'information précédente en compte (Nesterov's trick)

Variantes

- ▶ Changer la taille du pas en fonction du temps t ,
- ▶ Rajouter de l'inertie à la descente de gradient pour garder l'information précédente en compte (Nesterov's trick)
- ▶ Descente de gradient stochastique (direction de descente aléatoire)

Variantes

- ▶ Changer la taille du pas en fonction du temps t ,
- ▶ Rajouter de l'inertie à la descente de gradient pour garder l'information précédente en compte (Nesterov's trick)
- ▶ Descente de gradient stochastique (direction de descente aléatoire)
- ▶ Méthodes de boosting (réduction de variance)

Variantes

- ▶ Changer la taille du pas en fonction du temps t ,
- ▶ Rajouter de l'inertie à la descente de gradient pour garder l'information précédente en compte (Nesterov's trick)
- ▶ Descente de gradient stochastique (direction de descente aléatoire)
- ▶ Méthodes de boosting (réduction de variance)
- ▶ ... → Regarder les slides de la star nationale du domaine (Francis Bach).

Table of Contents

Étude de fonction

Convexité

Différentiabilité

Optimisation différentiable, convexe et sans contrainte

Méthode de descente de gradient

Algorithme → Notebook

Garanties théoriques et variantes (très) succinctement

Application

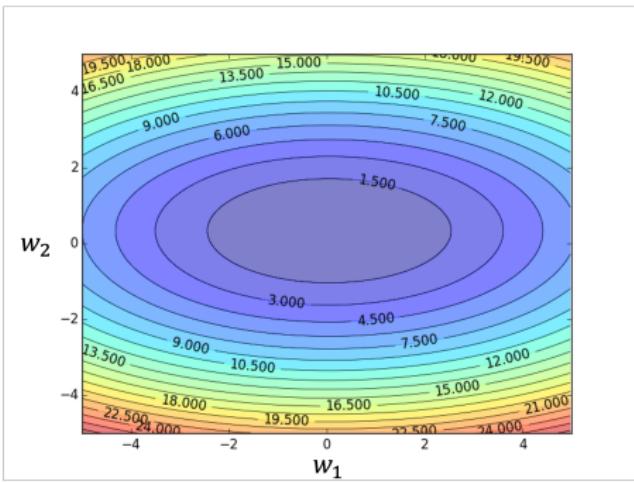
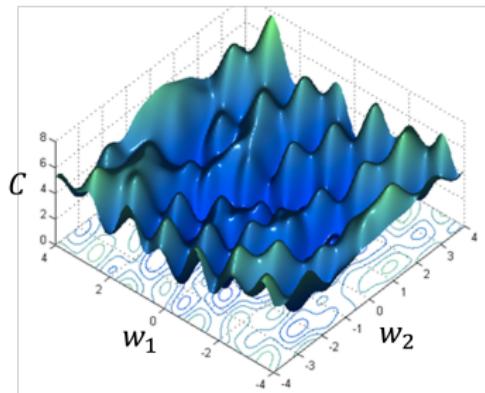
Deep learning

Machine learning

Descente de gradient en deep learning

On suppose que ce sont des fonctions convexes même si ce n'est pas le cas.

Choix du Learning rate.



Algorithmes d'optimisation les plus utilisés en deep learning

- ▶ Stochastic Gradient Descent (SGD)

Algorithmes d'optimisation les plus utilisés en deep learning

- ▶ Stochastic Gradient Descent (SGD)
- ▶ Root Mean Squared Propagation (RMSprop)

Algorithmes d'optimisation les plus utilisés en deep learning

- ▶ Stochastic Gradient Descent (SGD)
- ▶ Root Mean Squared Propagation (RMSprop)
- ▶ Adaptive moment estimation (Adam)

Algorithmes d'optimisation les plus utilisés en deep learning

- ▶ Stochastic Gradient Descent (SGD)
- ▶ Root Mean Squared Propagation (RMSprop)
- ▶ Adaptive moment estimation (Adam)
- ▶ Autres exemples : Momentum, Adagrad, Adadelta, Nadam, AdaSecant ...

Stochastic Gradient Descent (SGD)

- ▶ Calcul le gradient sur un sous ensemble des données d'entraînement plutôt que sur son ensemble.

Stochastic Gradient Descent (SGD)

- ▶ Calcul le gradient sur un sous ensemble des données d'entraînement plutôt que sur son ensemble.
- ▶ Algorithme important en deep learning.

Stochastic Gradient Descent (SGD)

- ▶ Calcul le gradient sur un sous ensemble des données d'entraînement plutôt que sur son ensemble.
- ▶ Algorithme important en deep learning.
- ▶ Permet de converger plus rapidement que la descente de gradient classique avec plus de pas le long du gradient.

Stochastic Gradient Descent (SGD)

- ▶ Calcul le gradient sur un sous ensemble des données d'entraînement plutôt que sur son ensemble.
- ▶ Algorithme important en deep learning.
- ▶ Permet de converger plus rapidement que la descente de gradient classique avec plus de pas le long du gradient.
- ▶ Très utilisé en machine learning (ex : regression logistique).

Root Mean Squared Propagation (RMSprop)

- ▶ Converge plus rapidement que SGD.

Root Mean Squared Propagation (RMSprop)

- ▶ Converge plus rapidement que SGD.
- ▶ Très utilisé dans la littérature.

Root Mean Squared Propagation (RMSprop)

- ▶ Converge plus rapidement que SGD.
- ▶ Très utilisé dans la littérature.
- ▶ Utilise un learning rate variable.

Root Mean Squared Propagation (RMSprop)

- ▶ Converge plus rapidement que SGD.
- ▶ Très utilisé dans la littérature.
- ▶ Utilise un learning rate variable.
- ▶ Normalise le gradient.

Adaptive moment estimation (Adam)

- ▶ RMSprop + Momentum.

Adaptive moment estimation (Adam)

- ▶ RMSprop + Momentum.
- ▶ Similaire à RMSprop mais fonctionne mieux dans la plupart des situations.

Algorithmes utilisés en machine learning

- ▶ Gradient Boosting

Algorithmes utilisés en machine learning

- ▶ Gradient Boosting
- ▶ Xgboost