(In Progress)

TC2: Optimization for Machine Learning

Master of Science in AI and Master of Science in Data Science
@ UPSaclay
2024/2025.

Faïcel Chamroukhi

chamroukhi.com

Maths concepts for the descent methods

Continuing the ingredients of (gradient) descent methods

A tour of the following aspects :

- Intuition behind descent methods
- Gradient and link to minimization
- Descent Directions
- Descent and Gradient
- Steepest/Fastest Descent
- Convergence aspects
- Convergence rates
- Line Search

Motivation of Taylor Expansion

- How to minimize a function f if we don't know much about its structure ?
- Assuming the function can be approximated by its derivatives around a point, which simplifies the problem.
- The trick is to assume the function is simpler than it really is by using Taylor's approximation, which allows us to locally approximate the function.

Taylor's Theorem :

- Let k be a natural number, $x_0 \in \mathbb{R}$, and f a function that is k-times continuously differentiable on an interval $[x_0, x]$
- Then there exists some ξ between x_0 and x such that :

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(k)}(\xi)}{k!}(x - x_0)^k.$$

Implication : Taylor's theorem allows us to approximate $f(x)$ around x_0 with increasingly accurate terms based on the derivatives at x_0.

Review

Motivation of Taylor Expansion

- How to minimize a function f if we don't know much about its structure ?
- Assuming the function can be approximated by its derivatives around a point, which simplifies the problem.
- The trick is to assume the function is simpler than it really is by using Taylor's approximation, which allows us to locally approximate the function.

Taylor's Theorem :

- Let k be a natural number, $x_0 \in \mathbb{R}$, and f a function that is k-times continuously differentiable on an interval $[x_0, x]$
- Then there exists some ξ between x_0 and x such that :

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(k)}(\xi)}{k!}(x - x_0)^k.$$

Implication : Taylor's theorem allows us to approximate $f(x)$ around x_0 with increasingly accurate terms based on the derivatives at x_0.

Taylor Approximation for $f : \mathbb{R}^n \to \mathbb{R}$:

- If f is continuously twice differentiable, then for any $x, x_0 \in \mathbb{R}^n$, we have :

$$f(x) = f(x_0) + \nabla f(x_0)^T(x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0) + R_3(x),$$

where $R_3(x)$ is the remainder term :

$$R_3(x) = O(\|x - x_0\|^3) \quad \text{which vanishes as} \quad x \to x_0.$$

- Explicitly, if f is three-times differentiable, $R_3(x)$ can be expressed as : $R_3(x) = \frac{1}{6}(x - x_0)^T \nabla^3 f(\xi)[x - x_0, x - x_0]$, where $\nabla^3 f(\xi)$ is the third-order tensor of partial derivatives evaluated at some ξ between x and x_0.
  $\nabla^3 f(\xi)[x - x_0, x - x_0]$: Multilinear application of the 3d-order derivative tensor.

Taylor Approximation for $f : \mathbb{R}^n \to \mathbb{R}$:

- If f is continuously twice differentiable, then for any $x, x_0 \in \mathbb{R}^n$, we have :

$$f(x) = f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0) + R_3(x),$$

  where $R_3(x)$ is the remainder term :

$$R_3(x) = O(\|x - x_0\|^3) \quad \text{which vanishes as} \quad x \to x_0.$$

- Explicitly, if f is three-times differentiable, $R_3(x)$ can be expressed as : $R_3(x) = \frac{1}{6}(x - x_0)^T \nabla^3 f(\xi)[x - x_0, x - x_0]$, where $\nabla^3 f(\xi)$ is the third-order tensor of partial derivatives evaluated at some ξ between x and x_0.

  $\nabla^3 f(\xi)[x - x_0, x - x_0]$: Multilinear application of the 3d-order derivative tensor.

Taylor Approximation for $f : \mathbb{R}^n \to \mathbb{R}$:

- If f is continuously twice differentiable, then for any $x, x_0 \in \mathbb{R}^n$,
  Provided that $\|x - x_0\|$ is small (i.e., x is close to x_0), we can approximate $f(x)$ by :

  $$f(x) \approx f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{(first-order approximation)}$$

  or

  $$f(x) \approx f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0) \quad \text{(second-order ap}$$

- Here, $\nabla f(x_0)$ is the gradient of f at x_0, and $\nabla^2 f(\xi)$ is the Hessian matrix
- **Comparison :** The second-order approximation is more accurate but also more computationally expensive (includes the Hessian), requiring f to be twice differentiable.
- Both approximations are valid if $\|x - x_0\|$ is small.

  Higher-Order Approximation : If f is continuously thrice differentiable, an additional error term can be expressed as $O(\|x - x_0\|^3)$.

Taylor Approximation for $f : \mathbb{R}^n \to \mathbb{R}$:

- If f is continuously twice differentiable, then for any $x, x_0 \in \mathbb{R}^n$, Provided that $\|x - x_0\|$ is small (i.e., x is close to x_0), we can approximate $f(x)$ by :

$$f(x) \approx f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{(first-order approximation)}$$

or

$$f(x) \approx f(x_0) + \nabla f(x_0)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \nabla^2 f(x_0)(x - x_0) \quad \text{(second-order ap}$$

- Here, $\nabla f(x_0)$ is the gradient of f at x_0, and $\nabla^2 f(\xi)$ is the Hessian matrix
- **Comparison :** The second-order approximation is more accurate but also more computationally expensive (includes the Hessian), requiring f to be twice differentiable.
- Both approximations are valid if $\|x - x_0\|$ is small.

  Higher-Order Approximation : If f is continuously thrice differentiable, an additional error term can be expressed as $O(\|x - x_0\|^3)$.

Example : What is the of first-order Taylor approximation of $f(x) = x^2 + 3x$ around $x_0 = 1$.

- Compute $f(1)$, $f'(1)$, and apply the first-order Taylor approximation.
- $f(1) = 1^2 + 3 \times 1 = 4$.
- $f'(x) = 2x + 3$, so $f'(1) = 2 \times 1 + 3 = 5$.
- First-order Taylor approximation around $x_0 = 1$:

$$f(x) \approx f(1) + f'(1) \cdot (x - 1) = 4 + 5(x - 1).$$

- This linear approximation provides a close estimate of $f(x)$ near $x = 1$, which we can use to analyze the behavior of $f(x)$.

Example : What is the of first-order Taylor approximation of $f(x) = x^2 + 3x$ around $x_0 = 1$.

- Compute $f(1)$, $f'(1)$, and apply the first-order Taylor approximation.
- $f(1) = 1^2 + 3 \times 1 = 4$.
- $f'(x) = 2x + 3$, so $f'(1) = 2 \times 1 + 3 = 5$.
- First-order Taylor approximation around $x_0 = 1$:

$$f(x) \approx f(1) + f'(1) \cdot (x - 1) = 4 + 5(x - 1).$$

- This linear approximation provides a close estimate of $f(x)$ near $x = 1$, which we can use to analyze the behavior of $f(x)$.

Descent Directions

Continuing the preparation of the ingredients of the gradient descent algorithm

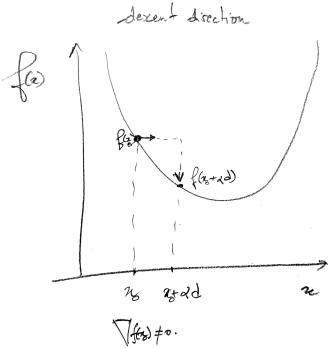
Definition (Descent Direction) :

- The concept of descent direction allows us to identify directions d in which the function f decreases locally.
- Let x be a point in the domain of f such that $\nabla f(x) \neq 0$, meaning x is not a critical point of f.
- A **descent direction** for f at x is a nonzero vector $d \in \mathbb{R}^n$ such that there exists $\bar{\alpha} > 0$ with the property :

$$f(x + \alpha d) < f(x) \quad \text{for all } \alpha,\ 0 < \alpha < \bar{\alpha}.$$

- Means f strictly decreases along the half-line $\{x + \alpha d : \alpha > 0\}$ for sufficiently small step sizes $\alpha > 0$.

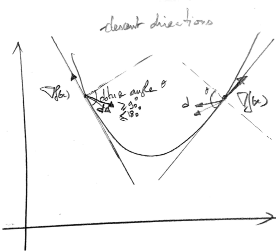
Descent Directions



Conditions for a Descent Direction

Lemma : Let x be a noncritical point of f (ie.$\nabla f(x) \neq 0$), and $d \in \mathbb{R}^n$ a nonzero vector. If $\nabla f(x)^T d < 0$, then d is a descent direction for f at x.

- **Interpretation :** $\nabla f(x)^T d \leq 0$ means d forms an obtuse angle with the gradient $\nabla f(x)$), $\implies$ A vector d that forms an obtuse angle with the gradient $\nabla f(x)$ ensures f decreases along d.
- Conversely, if d is a descent direction for f at x, then $\nabla f(x)^T d \leq 0$.

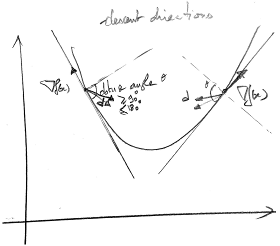


- **Implication of Descent Directions :** choosing d in a direction opposite to $\nabla f(x)$ guarantees descent. (proof for the opposite case to lead to the steepest descent will be proved later)

Conditions for a Descent Direction

Lemma : Let x be a noncritical point of f (ie.$\nabla f(x) \neq 0$), and $d \in \mathbb{R}^n$ a nonzero vector. If $\nabla f(x)^T d < 0$, then d is a descent direction for f at x.

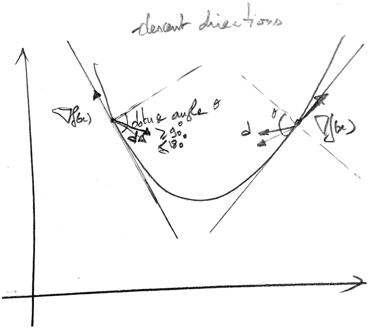
- **Interpretation :** $\nabla f(x)^T d \leq 0$ means d forms an obtuse angle with the gradient $\nabla f(x)$), $\implies$ A vector d that forms an obtuse angle with the gradient $\nabla f(x)$ ensures f decreases along d.
- Conversely, if d is a descent direction for f at x, then $\nabla f(x)^T d \leq 0$.



- **Implication of Descent Directions :** choosing d in a direction opposite to $\nabla f(x)$ guarantees descent. (proof for the opposite case to lead to the steepest descent will be proved later)

graphic illustration of descent directions

- **Interpretation :** $\nabla f(x)^T d \leq 0$: A vector d that forms an obtuse angle with the gradient $\nabla f(x)$ ensures f decreases along d.



- **Implication of Descent Directions :** choosing d in a direction opposite to $\nabla f(x)$ guarantees descent. (proof for the opposite case to lead to the steepest descent will be proved later)

Proof of the lemma :

- Since f is differentiable, then by first-order Taylor expansion's theorem we can approximate $f(x + \alpha d)$ for small $\alpha > 0$ as :

$$f(\alpha d + x) = f(x) + \alpha \nabla f(x)^T d + o(\alpha),$$

  where $o(\alpha)$ represents higher-order terms that vanish as $\alpha \to 0$.

- If $\nabla f(x)^T d < 0$, then for small $\alpha > 0$, the term $\alpha \nabla f(x)^T d$ is negative, implying $f(x + \alpha d) < f(x)$.
- Therefore, d is a descent direction for f at x.

The Steepest-Descent Direction

what is the best (fastest) descent we can achieve ? $\hookrightarrow$ We saw that :

- by first-order Taylor approximation we have :

$$f(\alpha d + x) = f(x) + \alpha \nabla f(x)^T d + o(\alpha),$$

$$f(x + \alpha d) \approx f(x) + \alpha \nabla f(x)^T d \quad \text{for small } \alpha > 0,$$

- if $d \neq \mathbf{0}$ is such that $\nabla f(x)^T d \leq 0$, then it is a descent direction for f at x
- $\hookrightarrow$ to achieve the maximum decrease in $f(x)$ for a small $\alpha > 0$, we should minimize $\nabla f(x)^T d$ over all directions $d \in \mathbb{R}^n$ with $\|d\| = 1$.

Derivation :

- $\nabla f(x)^T d = \|\nabla f(x)\| \|d\| \cos(\theta)$, where θ is the angle between $\nabla f(x)$ and d
- The minimum occurs when $\cos(\theta) = -1$. This indicates that the two vectors $\nabla f(x)$ and d are pointing in exactly opposite directions.
- Thus, we choose $\nabla f(x)^T d = -\|\nabla f(x)\| \|d\|$, which leads to $d = \frac{-\nabla f(x)}{\|\nabla f(x)\|}$.

- The (unnormalized) direction $d = -\nabla f(x)$ (anti-gradient) is called the **steepest-descent direction** of f at x, as it yields the greatest decrease in f

The Steepest-Descent Direction

what is the best (fastest) descent we can achieve ? $\hookrightarrow$ We saw that :

- by first-order Taylor approximation we have :

$$f(\alpha d + x) = f(x) + \alpha \nabla f(x)^T d + o(\alpha),$$

$$f(x + \alpha d) \approx f(x) + \alpha \nabla f(x)^T d \quad \text{for small } \alpha > 0,$$

- if $d \neq \mathbf{0}$ is such that $\nabla f(x)^T d \leq 0$, then it is a descent direction for f at x
- $\hookrightarrow$ to achieve the maximum decrease in $f(x)$ for a small $\alpha > 0$, we should minimize $\nabla f(x)^T d$ over all directions $d \in \mathbb{R}^n$ with $\|d\| = 1$.

Derivation :

- $\nabla f(x)^T d = \|\nabla f(x)\| \|d\| \cos(\theta)$, where θ is the angle between $\nabla f(x)$ and d
- The minimum occurs when $\cos(\theta) = -1$. This indicates that the two vectors $\nabla f(x)$ and d are pointing in exactly opposite directions.
- Thus, we choose $\nabla f(x)^T d = -\|\nabla f(x)\| \|d\|$, which leads to $d = \frac{-\nabla f(x)}{\|\nabla f(x)\|}$.
- The (unnormalized) direction $d = -\nabla f(x)$ (anti-gradient) is called the **steepest-descent direction** of f at x, as it yields the greatest decrease in f

Key Idea :

Thes ingredients form the basis idea of descent methods in optimization : take iterative steps in descent directions to reduce the value of f and guide the search towards a minimum.

Descent Methods in Optimization

To minimize a differentiable function f, The **Gradient Descent** algorithm operates the folliwng sequence of iterates :

- **Initialization :** Start with an initial point $x^{(0)}$.
- **Iteration :** For $k = 1, 2, \dots$:

$$x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)},$$

  - $d^{(k)} = -\nabla f(x^{(k)})$: the descent direction (negative gradient).
  - $\alpha^{(k)}$: the step size (learning rate).
- until a stopping criterion is reached.

Why it works : By moving in the direction opposite to the gradient, the algorithm ensures f decreases at each step for a properly chosen step size $\alpha^{(k)}$.

Does this converge ?

Theorem : Convergence to a Critical Point

- Let f satisfy smoothness and convexity conditions (detailed later)
- Let d_k satisfy the condition of a descent direction (i.e., the angle between the gradient $\nabla f(x_k)$ and and d_k is an obtuse angle (between 90 and 180 degrees, or equivalently, the angle θ_k between the anti-gradient $-\nabla f(x_k)$ and d_k is positive and less than 90 degrees), so that we ensure we are indeed moving in a decreasing direction.
- Let $\{x_k\}_{k=0}^{\infty}$ be the sequence of vectors generated by a descent method :

  $$x_{k+1} = x_k + \alpha_k d_k,$$

  where the step size α_k is properly chosen (a critical question !) (eg., by **line search**, like the Armijo rule its parameters s (initial step size), β (reduction factor), and σ (sufficient decrease condition)). [Will be seen later]
- If the sequence $\{x_k\}_{k=0}^{\infty}$ has a limit point $x^* = \lim_{i\to\infty} x_{k_i}$, **then** x^* is a critical point of f, i.e., $\nabla f(x^*) = 0$.

Assumptions :

- $x^* = \lim_{i\to\infty} x_{k_i}$ is a limit point of the sequence $\{x_k\}_{k=0}^{\infty}$.
- By definition of a limit point, the subsequence $\{x_{k_i}\}$ converges to x^*, i.e., $x_{k_i} \to x^*$ as $i \to \infty$.

Since :

- d_k is a descent direction, ensuring $f(x_k)$ decreases at each step unless $\nabla f(x_k) = 0$.

  This implies that near a limit point x^*, gradient $\nabla f(x_k)$ must approach 0.
- By continuity of the gradient $\nabla f(x)$, as $x_k \to x^*$, the gradient satisfies :

$$\nabla f(x^*) = \lim_{k\to\infty} \nabla f(x_k) = 0.$$

Then :

- The sequence $\{x_k\}$ converges to x^*, and at x^*, we have $\nabla f(x^*) = 0$.
- Therefore, x^* is a critical point of f, as required.

Convergence Rates

Essentials (convexity, Smoothness, ..) for analyzing convergence rates of optimization algorithms.

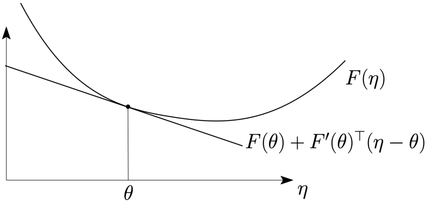
Definition of a Convex Function

Definition (Convex Function) :

- A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be **convex** iff $\forall x, \theta \in \mathbb{R}^d$,

$$f(x) \geq f(\theta) + f'(\theta)^\top (x - \theta).$$

- The inequality implies that f is always above its linear approximation at θ.



- **Consequence :** This implies : $f(\theta) - f(x) \leq f'(\theta)^\top (\theta - x)$, $\forall x, \theta \in \mathbb{R}^d$.

Consequence for Optimization :

- A key property we will use frequently in the analysis of GD and SGD is :

$$f(x^*) \geq f(\theta) + f'(\theta)^\top (x^* - \theta),$$

which implies :

$$f(\theta) - f(x^*) \leq f'(\theta)^\top (\theta - x^*),$$

for all $\theta \in \mathbb{R}^d$, where x^* is the minimizer of f.

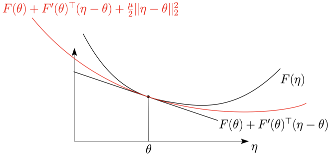
$\rightarrow$ an upper bound for the function value gap at any point

Definition of Strong Convexity

Definition (Strong Convexity) :

- A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be μ-**strongly convex** if, for all $x, \theta \in \mathbb{R}^d$,

$$f(x) \geq f(\theta) + f'(\theta)^\top (x - \theta) + \frac{\mu}{2}\|x - \theta\|^2.$$



- Strong convexity ensures that $f(x)$ is "curved" everywhere, and μ quantifies the lower bound on this curvature.
- Consequence in Optimization : At a critical point, (by taking $\theta = x^*$), Strong convexity implies :

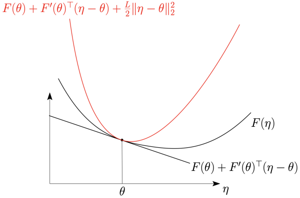
$$f(x) - f(x^*) \geq \frac{\mu}{2}\|x - x^*\|^2. \quad \textbf{NB}$$

Smoothness

Definition (L-Smoothness) :

- A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be L-**smooth** if and only if :

$$|f(x) - f(\theta) - f'(\theta)^T(x - \theta)| \leq \frac{L}{2}\|\theta - x\|^2, \quad \forall \theta, x \in \mathbb{R}^d.$$



- This is equivalent to **Smoothness (Lipschitz Continuity of Gradient) :**
  - A function f is L-smooth if its gradient is L-Lipschitz continuous, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ for all $x, y \in \mathbb{R}^d$.

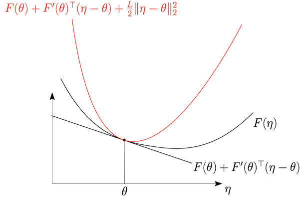
$\hookrightarrow$ This means the gradient of $f(x)$ cannot change arbitrarily fast, and L represents the upper bound on this rate of change.

Smoothness

Definition (L-Smoothness) :

- A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be L-**smooth** if and only if :

$$|f(x) - f(\theta) - f'(\theta)^T(x - \theta)| \leq \frac{L}{2}\|\theta - x\|^2, \quad \forall \theta, x \in \mathbb{R}^d.$$



- This is equivalent to **Smoothness (Lipschitz Continuity of Gradient) :**
  - A function f is L-smooth if its gradient is L-Lipschitz continuous, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ for all $x, y \in \mathbb{R}^d$.

$\hookrightarrow$ This means the gradient of $f(x)$ cannot change arbitrarily fast, and L represents the upper bound on this rate of change.

Equivalent Conditions for Twice Differentiable Functions

For a twice differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, convexity, strong convexity and smoothness can be expressed in terms of the Hessian matrix $f''(x)$.

- **Equivalent Condition for Convexity :** convexity is equivalent to requiring :

  $$f''(x) \succeq 0, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f positive

- **Eq. Condition for Strong Convexity :** f is μ-strongly convex iff :

  $$f''(x) \succeq \mu I, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f are larger than μ

- **Equivalent Condition for Smoothness :** L-smoothness is equivalent to :

  $$-LI \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are at most equal to L

- **Equivalent Condition for Strong Convexity and Smoothness :** f is μ-strongly convex and L-smooth is equivalent to :

  $$\mu I \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are larger than μ and are at most equal to L

Equivalent Conditions for Twice Differentiable Functions

For a twice differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, convexity, strong convexity and smoothness can be expressed in terms of the Hessian matrix $f''(x)$.

- **Equivalent Condition for Convexity :** convexity is equivalent to requiring :

$$f''(x) \succeq 0, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f positive

- **Eq. Condition for Strong Convexity :** f is μ-strongly convex iff :

$$f''(x) \succeq \mu I, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f are larger than μ

- **Equivalent Condition for Smoothness :** L-smoothness is equivalent to :

$$-LI \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are at most equal to L

- **Equivalent Condition for Strong Convexity and Smoothness :** f is μ-strongly convex and L-smooth is equivalent to :

$$\mu I \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are larger than μ and are at most equal to L

For a twice differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, convexity, strong convexity and smoothness can be expressed in terms of the Hessian matrix $f''(x)$.

- **Equivalent Condition for Convexity :** convexity is equivalent to requiring :

$$f''(x) \succeq 0, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f positive

- **Eq. Condition for Strong Convexity :** f is μ-strongly convex iff :

$$f''(x) \succeq \mu I, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f are larger than μ

- **Equivalent Condition for Smoothness :** L-smoothness is equivalent to :

$$-LI \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are at most equal to L

- **Equivalent Condition for Strong Convexity and Smoothness :** f is μ-strongly convex and L-smooth is equivalent to :

$$\mu I \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are larger than μ and are at most equal to L

Equivalent Conditions for Twice Differentiable Functions

For a twice differentiable function $f : \mathbb{R}^d \to \mathbb{R}$, convexity, strong convexity and smoothness can be expressed in terms of the Hessian matrix $f''(x)$.

- **Equivalent Condition for Convexity :** convexity is equivalent to requiring :

$$f''(x) \succeq 0, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f positive

- **Eq. Condition for Strong Convexity :** f is μ-strongly convex iff :

$$f''(x) \succeq \mu I, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of f are larger than μ

- **Equivalent Condition for Smoothness :** L-smoothness is equivalent to :

$$-LI \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are at most equal to L

- **Equivalent Condition for Strong Convexity and Smoothness :** f is μ-strongly convex and L-smooth is equivalent to :

$$\mu I \preceq f''(x) \preceq LI, \quad \forall x \in \mathbb{R}^d.$$

  all the eigenvalues of the Hessian of$f$ are larger than μ and are at most equal to L

The **condition Number** κ measures how "well-conditioned" the optimization pblm is :

- When a function $f : \mathbb{R}^n \to \mathbb{R}$ is both L-smooth and μ-strongly convex, we define its **condition number** κ as :

$$\kappa = \frac{L}{\mu} \geq 1,$$

  where L is the smoothness constant and μ is the strong convexity constant.

- μ : Describes the **minimum curvature** (strong convexity of $f(x)$).

  μ : Ensures $f(x)$ is not too "flat" (sufficient curvature everywhere).
- L : Describes the **maximum curvature** (smoothness of $f(x)$).

  L : Prevents $f(x)$ from being too "steep" (gradient does not grow arbitrarily fast).
- Since μ is the sharpest lower bound on curvature and L is the broadest upper bound, then $L \geq \mu \quad \Longrightarrow \quad \kappa = \frac{L}{\mu} \geq 1$.

  The ratio $\frac{L}{\mu}$ measures the disparity between the "steepest" and "flattest" directions

- **Perfect Case :** When $L = \mu$: The function is perfectly conditioned ($\kappa = 1$, e.g., quadratic with spherical level sets).
- When $L \gg \mu$: $\kappa \gg 1$, indicating worse conditioning.

The **condition Number** κ measures how "well-conditioned" the optimization pblm is :

- When a function $f : \mathbb{R}^n \to \mathbb{R}$ is both L-smooth and μ-strongly convex, we define its **condition number** κ as :
$$\kappa = \frac{L}{\mu} \geq 1,$$
where L is the smoothness constant and μ is the strong convexity constant.

- μ : Describes the **minimum curvature** (strong convexity of $f(x)$).
  μ : Ensures $f(x)$ is not too "flat" (sufficient curvature everywhere).
- L : Describes the **maximum curvature** (smoothness of $f(x)$).
  L : Prevents $f(x)$ from being too "steep" (gradient does not grow arbitrarily fast).
- Since μ is the sharpest lower bound on curvature and L is the broadest upper bound, then $L \geq \mu \quad \Longrightarrow \quad \kappa = \frac{L}{\mu} \geq 1$.
  The ratio $\frac{L}{\mu}$ measures the disparity between the "steepest" and "flattest" directions

- **Perfect Case :** When $L = \mu$: The function is perfectly conditioned ($\kappa = 1$, e.g., quadratic with spherical level sets).
- When $L \gg \mu$: $\kappa \gg 1$, indicating worse conditioning.

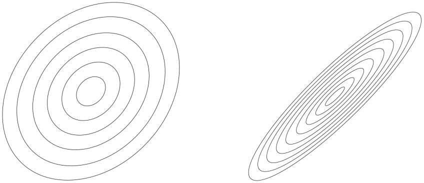
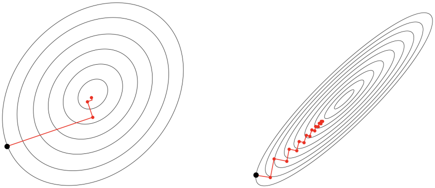


FIGURE – Level sets (Contours) : small κ vs large κ

Level Set Definition : Given a function $f : \mathbb{R}^n \to \mathbb{R}$, the *level set* of f corresponding to a scalar $c \in \mathbb{R}$ is the set of all points $x \in \mathbb{R}^n$ such that : $\mathcal{L}_c = \{x \in \mathbb{R}^n \mid f(x) = c\}$.

Condition Number κ and Gradient Descent :

- The performance of gradient descent is influenced by the condition number $\kappa = \frac{L}{\mu}$.
- A **small condition number** $\kappa \approx 1$ (function with level sets that are nearly circular), results in **fast convergence**.
- A **large condition number** $\kappa \gg 1$ leads to **slow convergence and oscillations** (zigzag).



(small $\kappa = L/\mu$) (large $\kappa = L/\mu$)

FIGURE – small κ : fast convergence, vs large κ oscillations

Convergence Rates

Theorem (Convergence Rate of Gradient Descent for Strongly Convex Functions) :

- Assume f is L-smooth and μ-strongly convex.
- For gradient descent with a fixed step size $\alpha_k = \frac{1}{L}$, the iterates $(x_k)_{k\geq 0}$ satisfy :

$$f(x_t) - f(x^*) \leq \exp\left(-\frac{k\mu}{L}\right)\left(f(x_0) - f(x^*)\right),$$

  where :
  - x^* is the minimizer of f,
  - $\frac{\mu}{L}$ determines the rate of convergence and depends on the condition number $\kappa = \frac{L}{\mu}$.
- Gradient descent therefore achieves exponential (**linear** in log-scale) convergence rate for strongly convex functions.

1. **Gradient Descent Update Rule :** $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$.
2. Substituting $\alpha_k = \frac{1}{L}$: $x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$.
3. **Strong Convexity Inequality :** For μ-strongly convex f, we have :

$$f(x) \geq f(y) + \nabla f(y)^T(x-y) + \frac{\mu}{2}\|x-y\|^2.$$

Substituting $y = x^*$, where $\nabla f(x^*) = 0$, gives :

$$f(x_k) - f(x^*) \leq -\nabla f(x_k)^T(x_k - x^*) - \frac{\mu}{2}\|x_k - x^*\|^2.$$

4. **Smoothness Inequality :** For L-smooth f :

$$f(x_{k+1}) \leq f(x_k) + \nabla f(x_k)^T(x_{k+1} - x_k) + \frac{L}{2}\|x_{k+1} - x_k\|^2,$$

Using $x_{k+1} - x_k = -\frac{1}{L}\nabla f(x_k)$, gives

$$f(x_{k+1}) \leq f(x_k) - \frac{1}{L}\|\nabla f(x_k)\|^2 + \frac{1}{2L}\|\nabla f(x_k)\|^2 = f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|^2$$

$$f(x_{k+1}) \leq f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|^2.$$ **NB**

Proof II

5 **Combining Inequalities :** From strong convexity (see proof separataley) :

$$\|\nabla f(x_k)\|^2 \geq 2\mu \left(f(x_k) - f(x^*)\right). \quad \textbf{NB}$$

Substituting into the smoothness inequality :

$$f(x_{k+1}) - f(x^*) \leq \left(f(x_k) - f(x^*)\right) - \frac{1}{2L} 2\mu \left(f(x_k) - f(x^*)\right).$$

Simplifying :

$$f(x_{k+1}) - f(x^*) \leq \left(1 - \frac{\mu}{L}\right) \left(f(x_k) - f(x^*)\right).$$

6 **Exponential Convergence :** By induction (simple) :

$$f(x_k) - f(x^*) \leq \left(1 - \frac{\mu}{L}\right)^k \left(f(x_0) - f(x^*)\right).$$

Using $1 - x \leq e^{-x}$:

$$f(x_k) - f(x^*) \leq \exp\left(-\frac{k\mu}{L}\right) \left(f(x_0) - f(x^*)\right).$$

CQFD

Goal : Derive the inequality : $\|\nabla f(x_k)\|^2 \geq 2\mu\left(f(x_k) - f(x^*)\right).$

1. **Strong Convexity :** $f(y) \geq f(x) + \nabla f(x)^T(y-x) + \frac{\mu}{2}\|y-x\|^2, \quad \forall x, y.$
   Substitute $y = x^*$: $f(x^*) \geq f(x_k) + \nabla f(x_k)^T(x^* - x_k) + \frac{\mu}{2}\|x^* - x_k\|^2.$
   Rearrange : $f(x_k) - f(x^*) \leq -\nabla f(x_k)^T(x^* - x_k) - \frac{\mu}{2}\|x^* - x_k\|^2.$
2. **Cauchy-Schwarz Inequality :** Using
   $-\nabla f(x_k)^T(x^* - x_k) \leq \|\nabla f(x_k)\| \cdot \|x^* - x_k\|$:
   $$f(x_k) - f(x^*) \leq \|\nabla f(x_k)\| \cdot \|x^* - x_k\| - \frac{\mu}{2}\|x^* - x_k\|^2.$$
3. **Minimize the r.h.s w.r.t** $\|x^* - x_k\|$ leads to $\|x^* - x_k\| = \frac{\|\nabla f(x_k)\|}{\mu}.$
   Note : We minimize the r.h.s. to express the inequality solely in terms of the gradient norm $\|\nabla f(x_k)\|$ and the function value gap $f(x_k) - f(x^*)$. This also ensures the sharpest possible lower bound (worst case) on $\|\nabla f(x_k)\|^2$
   Substitute : $f(x_k) - f(x^*) \leq \frac{\|\nabla f(x_k)\|^2}{2\mu}.$
   Rearrange : $\|\nabla f(x_k)\|^2 \geq 2\mu\left(f(x_k) - f(x^*)\right).$
   Rk : This inequality relates the gradient norm $\|\nabla f(x_k)\|$ to the function value gap $(f(x_k) - f(x^*))$ and provides a lower bound

Convergence of Gradient Descent for Smooth and Convex Functions

Theorem : For a convex and L-smooth function f, gradient descent with a step size $\alpha = \frac{1}{L}$ satisfies :

$$f(x_k) - f(x^*) = O\left(\frac{1}{k}\right),$$

where x^* is the minimizer of f.

If f is only assumed to be smooth and convex, gradient descent with a constant step size $\alpha = \frac{1}{L}$ still converges, but at a slower rate (sublinear rate).

Rather than $O\left(e^{-\frac{k\mu}{L}}\right)$ for μ-strong convex and L-smooth functions universite PARIS-SACLAY

Proof :

1. **Smoothness Inequality :** We saw $f(x_{k+1}) \leq f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|^2$(relating function decrease togradient norm).
2. **Convexity Inequality :** From convexity, $f(x_k) - f(x^*) \leq \|\nabla f(x_k)\| \cdot \|x_k - x^*\|$, bounding the gap.
3. **Combining both :** Substituting convexity bound into smoothness inequality :

$$\underbrace{f(x_{k+1}) - f(x^*)}_{\text{function gap at iteration } k+1} \leq \underbrace{f(x_k) - f(x^*)}_{\text{function gap at iteration } k} - \frac{1}{2L}\frac{(f(x_k) - f(x^*))^2}{\|x_k - x^*\|^2}.$$

   NB This shows that the function value gap $f(x_k) - f(x^*)$ decreases iteratively, but the amount of decrease depends on the current gap squared $(f(x_k) - f(x^*))^2$, scaled by $\|x_k - x^*\|^2$ the distance to the minimizer x^*.

4. **Gradient Descent Reduction :** Gradient descent reduces $f(x_k) - f(x^*)$ iteratively. By iteratively applying the inequality, it can be shown that :

$$f(x_k) - f(x^*) \leq \frac{C}{k}, \quad \text{(eg. Exercice)}$$

   where $C > 0$ is a constant depending on the initial function value gap $f(x_0) - f(x^*)$ and the smoothness parameter L.

Line Search

Purpose of the Armijo Rule :

- The Armijo rule is used to select a step size α_k in descent methods, ensuring that each step decreases the objective function $f(x)$ by a sufficient amount.
- It prevents steps that are too small (which slow down convergence) or too large (which may cause divergence).

Armijo Condition :

- For a given descent direction d_k at x_k, the Armijo rule requires that α_k **satisfies** :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma \alpha_k \nabla f(x_k)^T d_k,$$

where $0 < \sigma < 1$ is a parameter that controls the "sufficient decrease" in $f(x)$. as by convexity $f(\theta) - f(x_k) \leq f'(\theta)^\top(\theta - x_k), \forall x_k, \theta \in \mathbb{R}^d$, by taking $\theta = x_k + \alpha_k d_k$

Procedure :

- Start with an initial step size s (often $s = 1$).
- If the Armijo condition is not met, reduce α_k by multiplying it with a factor β (with $0 < \beta < 1$), and repeat until the condition holds.

Example of Parameters :

- $\sigma = 10^{-4}$: Controls the level of decrease.

Purpose of the Armijo Rule :

- The Armijo rule is used to select a step size α_k in descent methods, ensuring that each step decreases the objective function $f(x)$ by a sufficient amount.
- It prevents steps that are too small (which slow down convergence) or too large (which may cause divergence).

Armijo Condition :

- For a given descent direction d_k at x_k, the Armijo rule requires that α_k **satisfies** :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma \alpha_k \nabla f(x_k)^T d_k,$$

where $0 < \sigma < 1$ is a parameter that controls the "sufficient decrease" in $f(x)$.
as by convexity $f(\theta) - f(x_k) \leq f'(\theta)^\top(\theta - x_k), \forall x_k, \theta \in \mathbb{R}^d$, by taking $\theta = x_k + \alpha_k d_k$

Procedure :

- Start with an initial step size s (often $s = 1$).
- If the Armijo condition is not met, reduce α_k by multiplying it with a factor β (with $0 < \beta < 1$), and repeat until the condition holds.

Example of Parameters :

- $\sigma = 10^{-4}$: Controls the level of decrease.

Purpose of the Armijo Rule :

- The Armijo rule is used to select a step size α_k in descent methods, ensuring that each step decreases the objective function $f(x)$ by a sufficient amount.
- It prevents steps that are too small (which slow down convergence) or too large (which may cause divergence).

Armijo Condition :

- For a given descent direction d_k at x_k, the Armijo rule requires that α_k **satisfies** :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma \alpha_k \nabla f(x_k)^T d_k,$$

  where $0 < \sigma < 1$ is a parameter that controls the "sufficient decrease" in $f(x)$.
  as by convexity $f(\theta) - f(x_k) \leq f'(\theta)^\top (\theta - x_k), \forall x_k, \theta \in \mathbb{R}^d$, by taking $\theta = x_k + \alpha_k d_k$

Procedure :

- Start with an initial step size s (often $s = 1$).
- If the Armijo condition is not met, reduce α_k by multiplying it with a factor β (with $0 < \beta < 1$), and repeat until the condition holds.

Example of Parameters :

- $\sigma = 10^{-4}$: Controls the level of decrease.

universite PARIS-SACLAY

Algorithm 1 Pseudo Code for GD with linear search.

(S0) Choose $x^0 \in \mathbb{R}^n$, $\sigma, \beta \in (0, 1)$, and put $k := 0$.

(S1) If Convergence Criterion Reached. STOP.

(S2) Put $d^k := -\nabla f(x^k)$.

(S3) Determine $\alpha_k > 0$ by

$$\alpha_k := \max_{l \in \mathbb{N}_0} \beta^{(l)} \quad \text{s.t.} \quad f(x^k + \beta^{(l)} d^k) \leq f(x^k) + \beta^{(l)} \sigma \nabla f(x^k)^T d^k.$$

(S4) Update $x^{k+1} := x^k + \alpha_k d^k$

(S4) $k \leftarrow k + 1$ and go to (S1).
