Mathematics of Data: From Theory to Computation

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 2: The role of computation

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-556 (Fall 2020)

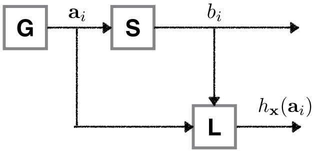
lions@epfl HASLERSTIFTUNG Google AI SDSC ZEISS FNSNF Fonds National Suisse Schweizerischer Nationalfonds Fondo Nazionale Svizzero Swiss National Science Foundation erc EPFL

License Information for Mathematics of Data Slides

Outline

- This lecture
    1. Principles of iterative descent methods
    2. Gradient descent for smooth convex problems
    3. Gradient descent for smooth non-convex problems

Recall: Learning machines result in optimization problems



$$(\mathbf{a}_i, b_i)_{i=1}^n \xrightarrow[\text{parameter } \mathbf{x}]{\text{modeling}} P(b_i|\mathbf{a}_i, \mathbf{x}) \xrightarrow[\text{identical dist.}]{\text{independency}} \mathsf{p}_{\mathbf{x}}(\mathbf{b}) := \prod_{i=1}^n P(b_i|\mathbf{a}_i, \mathbf{x})$$

Definition (Maximum-likelihood estimator)

The maximum-likelihood (ML) estimator is given by

$$\mathbf{x}^\star_{\mathsf{ML}} \in \arg\min_{\mathbf{x}\in\mathcal{X}} \left\{ L(h_{\mathbf{x}}(\mathbf{a}), \mathbf{b}) := -\log \mathsf{p}_{\mathbf{x}}(\mathbf{b}) \right\},$$

where $\mathsf{p}_{\mathbf{x}}(\cdot)$ denotes the probability density function or probability mass function of $\mathbb{P}_{\mathbf{x}}$, for $\mathbf{x} \in \mathcal{X}$.

M-Estimators

Roughly speaking, estimators can be formulated as optimization problems of the following form:

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathcal{X}} \left\{ F(\mathbf{x}) \right\},$$

with some constraints $\mathcal{X} \subseteq \mathbb{R}^p$. The term "$M$-estimator" denotes "maximum-likelihood-type estimator" [2].

Unconstrained minimization

Problem (**Mathematical formulation**)

How can we find an optimal solution to the following optimization problem?

$$\boxed{F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x}) := f(\mathbf{x})\}} \tag{1}$$

Note that (1) *is unconstrained.*

Definition (Optimal solutions and solution set)

- $\mathbf{x}^\star \in \mathbb{R}^p$ is a solution to (1) if $\boxed{F(\mathbf{x}^\star) = F^\star}$.
- $\boxed{\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathbb{R}^p \ : \ F(\mathbf{x}^\star) = F^\star\}}$ is the solution set of (1).
- (1) has solution if $\mathcal{S}^\star$ is non-empty.

Approximate vs. exact optimality

Is it possible to solve an optimization problem?

"In general, optimization problems are **unsolvable***"* - Y. Nesterov [4]

Observations:
- ◦ Even when a closed-form solution exists, numerical accuracy may still be an issue.
- ◦ We must be content with **approximately** optimal solutions.

Definition

We say that $\mathbf{x}^\star_\epsilon$ is ϵ-optimal in **objective value** if

$$f(\mathbf{x}^\star_\epsilon) - f^\star \leq \epsilon\ .$$

Definition

We say that $\mathbf{x}^\star_\epsilon$ is ϵ-optimal in **sequence** if, for some norm $\|\cdot\|$,

$$\|\mathbf{x}^\star_\epsilon - \mathbf{x}^\star\| \leq \epsilon\ ,$$

◦ The latter approximation guarantee is considered stronger.

A basic *iterative* strategy

General idea of an optimization algorithm

Guess a solution, and then *refine* it based on *oracle information*.

Repeat the procedure until the result is *good enough*.

Basic principles of descent methods

Template for iterative descent methods

1. Let $\mathbf{x}^0 \in \mathrm{dom}(f)$ be a starting point.
2. Generate a sequence of vectors $\mathbf{x}^1, \mathbf{x}^2, \cdots \in \mathrm{dom}(f)$ so that we have descent:

$$f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k), \quad \text{for all } k = 0, 1, \ldots$$

   until $\mathbf{x}^k$ is ϵ-optimal.

Such a sequence $\left\{\mathbf{x}^k\right\}_{k \geq 0}$ can be generated as:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$$

where $\mathbf{p}^k$ is a descent direction and $\alpha_k > 0$ a step-size.

Remarks:
- Iterative algorithms can use various **oracle** information in the optimization problem
- The type of oracle information used becomes a defining characteristic of the algorithm
- Example oracles: Objective value, gradient, and Hessian result in 0-th, 1-st, 2-nd order methods
- The oracle choices determine α_k and $\mathbf{p}^k$ as well as the overall convergence rate and complexity

Basic principles of descent methods

A condition for local descent directions

The iterates are given as:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$$

For a differentiable f, we have by Taylor's theorem

$$f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k) + \alpha_k \langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle + \mathcal{O}(\alpha_k^2 \left\|\mathbf{p}\right\|_2^2).$$

For α_k small enough, the term $\alpha_k \langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle$ dominates $\mathcal{O}(\alpha_k^2)$ for a fixed $\mathbf{p}^k$.

Therefore, in order to have $f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$, we require

$$\langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle < 0$$

Basic principles of descent methods

Local steepest descent direction

Since

$$\langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle = \|\nabla f(\mathbf{x}^k)\| \|\mathbf{p}^k\| \cos\theta \ ,$$

where θ is the angle between $\nabla f(\mathbf{x}^k)$ and $\mathbf{p}^k$, we have

$$\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$$

as the local *steepest descent* direction.

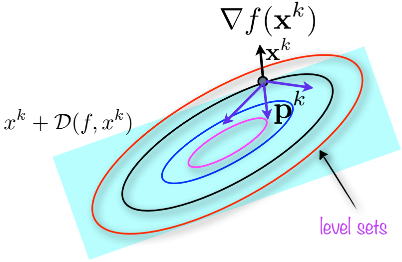
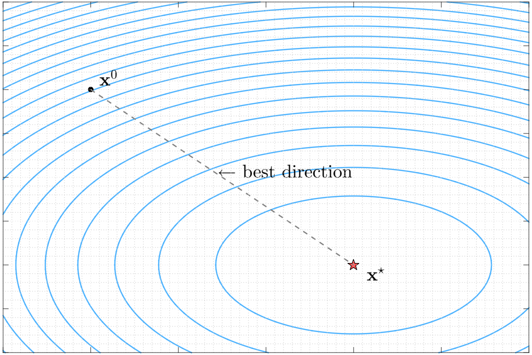


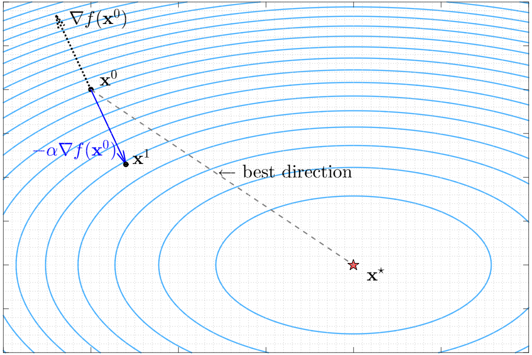
Figure: Descent directions in 2D should be an element of the cone of descent directions $\mathcal{D}(f, \cdot)$.

A simple iterative algorithm: Gradient descent



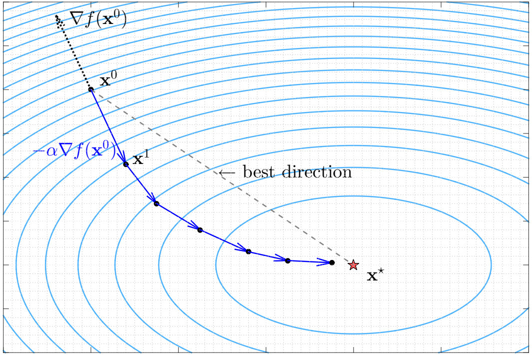
- Choose initial point: x^0.

A simple iterative algorithm: Gradient descent



- Choose initial point: x^0.
- Take a step in the negative gradient direction with a step size $\alpha > 0$: $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha\nabla f(\mathbf{x}^k)$.

A simple iterative algorithm: Gradient descent



- Choose initial point: x^0.
- Take a step in the negative gradient direction with a step size $\alpha > 0$: $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha\nabla f(\mathbf{x}^k)$.
- Repeat this procedure until x^k is accurate enough.

Recall the statistical estimation context

Observations:

- Denote $\mathbf{x}^\natural$ is the unknown true parameter
- The estimator $\mathbf{x}^\star$'s performance, e.g., $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ depends on the data size n.
- Evaluating $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ is not enough for evaluating the performance of a Learning Machine
  - We can only *numerically approximate* the solution of

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x})\}.$$

- We use algorithms to *numerically approximate* $\mathbf{x}^\star$.

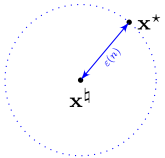
Practical performance

Denote the numerical approximation by an algorithm at time t by $\mathbf{x}^t$.

The practical performance at time t using n data samples is determined by

$$\underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\natural\right\|_2}_{\bar{\varepsilon}(t,n)} \leq \underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\star\right\|_2}_{\epsilon(t)} + \underbrace{\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2}_{\varepsilon(n)},$$

where $\varepsilon(n)$ denotes the statistical error, $\epsilon(t)$ is the numerical error, and $\bar{\varepsilon}(t,n)$ denotes the total error of the Learning Machine.



Recall the statistical estimation context

Observations:

- Denote $\mathbf{x}^\natural$ is the unknown true parameter
- The estimator $\mathbf{x}^\star$'s performance, e.g., $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ depends on the data size n.
- Evaluating $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ is not enough for evaluating the performance of a Learning Machine
  - We can only *numerically approximate* the solution of

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x})\}.$$

- We use algorithms to *numerically approximate* $\mathbf{x}^\star$.

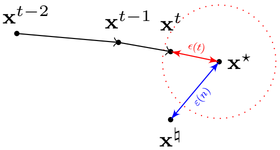
Practical performance

Denote the numerical approximation by an algorithm at time t by $\mathbf{x}^t$.

The practical performance at time t using n data samples is determined by

$$\underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\natural\right\|_2}_{\bar{\varepsilon}(t,n)} \leq \underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\star\right\|_2}_{\epsilon(t)} + \underbrace{\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2}_{\varepsilon(n)},$$

where $\varepsilon(n)$ denotes the statistical error, $\epsilon(t)$ is the numerical error, and $\bar{\varepsilon}(t,n)$ denotes the total error of the Learning Machine.



Recall the statistical estimation context

Observations:

- ∘ Denote $\mathbf{x}^\natural$ is the unknown true parameter
- ∘ The estimator $\mathbf{x}^\star$'s performance, e.g., $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ depends on the data size n.
- ∘ Evaluating $\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2^2$ is not enough for evaluating the performance of a Learning Machine
  - ▸ We can only *numerically approximate* the solution of

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x})\}.$$

- ∘ We use algorithms to *numerically approximate* $\mathbf{x}^\star$.

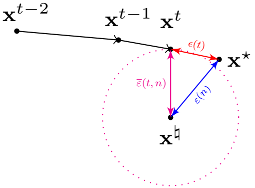
Practical performance

Denote the numerical approximation by an algorithm at time t by $\mathbf{x}^t$.

The practical performance at time t using n data samples is determined by

$$\underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\natural\right\|_2}_{\bar{\varepsilon}(t,n)} \leq \underbrace{\left\|\mathbf{x}^t - \mathbf{x}^\star\right\|_2}_{\epsilon(t)} + \underbrace{\left\|\mathbf{x}^\star - \mathbf{x}^\natural\right\|_2}_{\varepsilon(n)},$$

where $\varepsilon(n)$ denotes the statistical error, $\epsilon(t)$ is the numerical error, and $\bar{\varepsilon}(t,n)$ denotes the total error of the Learning Machine.

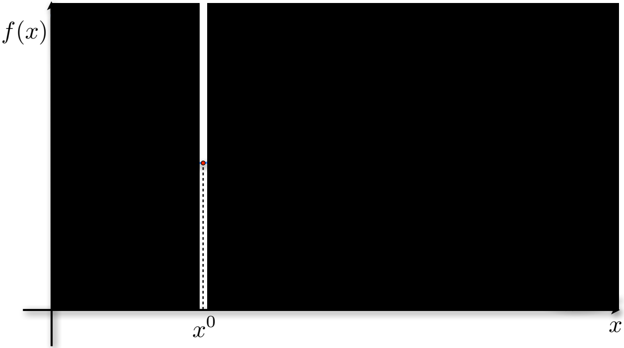


Challenges for an iterative optimization algorithm

Problem

Find the minimum $x^\star$ of $f(x)$, given starting point x^0 based on only local information.

- **Fog of war**

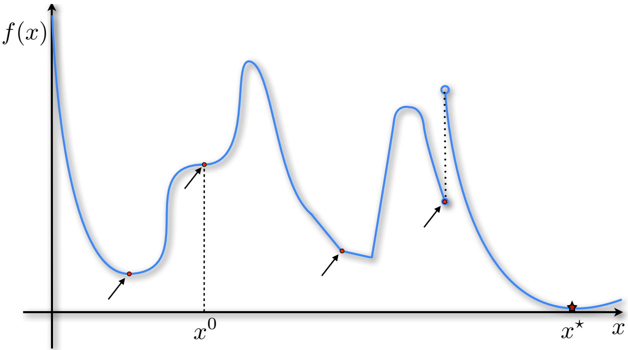


Challenges for an iterative optimization algorithm

Problem

Find the minimum $x^\star$ of $f(x)$, given starting point x^0 based on only local information.

- Fog of war, non-differentiability, discontinuities, local minima, stationary points...



A notion of convergence: Stationarity

- Let $f : \mathbb{R}^p \to \mathbb{R}$ be twice-differentiable and $\mathbf{x}^\star = \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x})$

Gradient method

Choose a starting point $\mathbf{x}^0$ and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k)$$

where $\alpha > 0$ is a step-size to be chosen so that $\mathbf{x}^k$ converges to $\mathbf{x}^\star$.

Definition (First order stationary point (FOSP))

A point $\bar{\mathbf{x}}$ is a first order stationary point of a twice differentiable function f if

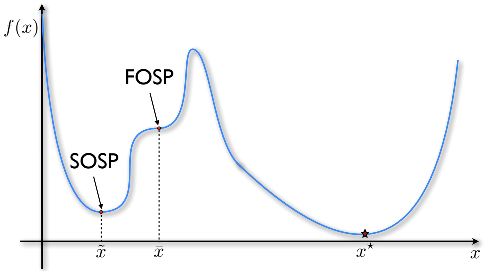
$$\nabla f(\bar{\mathbf{x}}) = \mathbf{0}.$$

Fixed-point characterization

Multiply by -1 and add $\bar{\mathbf{x}}$ to both sides to obtain the fixed point condition:

$$\bar{\mathbf{x}} = \bar{\mathbf{x}} - \alpha \nabla f(\bar{\mathbf{x}}) \qquad \text{for all } \alpha \in \mathbb{R}.$$

Geometric interpretation of stationarity



Observation: ○ Neither $\bar{\mathbf{x}}$, nor $\tilde{\mathbf{x}}$ is **necessarily** equal to $\mathbf{x}^\star$!!

Proposition (*Local minima, maxima, and saddle points)

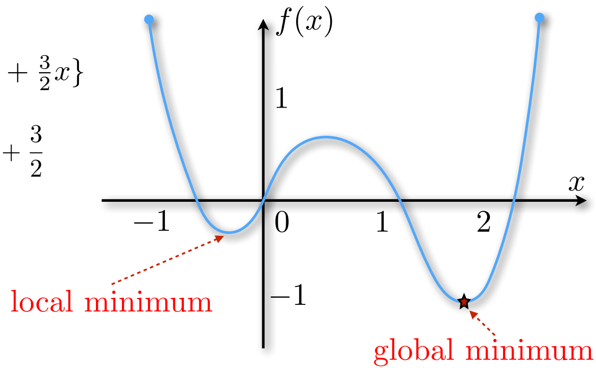
Let $\bar{\mathbf{x}}$ be a stationary point of a twice differentiable function f.

- *If $\nabla^2 f(\bar{\mathbf{x}}) \succ 0$, then the point $\bar{\mathbf{x}}$ is called a local minimum or a second order stationary point (SOSP).*
- *If $\nabla^2 f(\bar{\mathbf{x}}) \prec 0$, then the point $\bar{\mathbf{x}}$ is called a local maximum.*
- *If $\nabla^2 f(\bar{\mathbf{x}}) = 0$, then the point $\bar{\mathbf{x}}$ can be a saddle point, a local minimum, or a local maximum.*

Local minima

$$\min_{x \in \mathbb{R}} \{x^4 - 3x^3 + x^2 + \tfrac{3}{2}x\}$$

$$\frac{df}{dx} = 4x^3 - 9x^2 + 2x + \frac{3}{2}$$



Choose $x^0 = 0$ and $\alpha = \frac{1}{6}$

$x^1 = x^0 - \alpha \frac{df}{dx}\Big|_{x=x^0} = 0 - \frac{1}{6}\frac{3}{2} = -\frac{1}{4}$

$x^2 = -\frac{5}{16}$

$\ldots$

x^k converges to a **local minimum**!

From local to global optimality

Definition (Local minimum)

Given $f: \mathbb{R}^p \to \mathbb{R} \cup \{+\infty\}$, a vector $\mathbf{x}^\star \in \mathbb{R}^p$ is called a *local minimum* of f if there exists $\epsilon > 0$ s.t.

$$f(\mathbf{x}^\star) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^p \quad \text{with} \quad \|\mathbf{x} - \mathbf{x}^\star\| \leq \epsilon.$$

Theorem

If $\mathcal{Q} \subset \mathbb{R}^p$ is a convex set and $f: \mathbb{R}^p \to (-\infty, +\infty]$ is a proper convex function, then a local minimum of f over $\mathcal{Q}$ is also a global minimum of f over $\mathcal{Q}$.

Proof.

Suppose $\mathbf{x}^\star$ is a local minimum but not global, i.e. there exist $\mathbf{x} \in \mathbb{R}^p$ s.t. $f(\mathbf{x}) < f(\mathbf{x}^\star)$. By convexity,

$$f(\alpha\mathbf{x}^\star + (1-\alpha)\mathbf{x}) \leq \alpha f(\mathbf{x}^\star) + (1-\alpha) f(\mathbf{x}) < f(\mathbf{x}^\star), \forall \alpha \in [0, 1]$$

which contradicts the local minimality of $\mathbf{x}^\star$. □

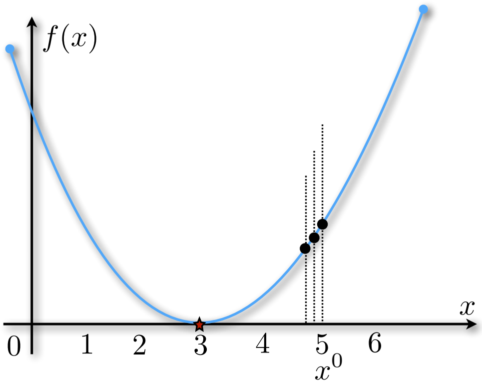
Theorem

Let $f: \mathbb{R}^p \to \mathbb{R}$ be a convex differentiable function. Then any stationary point of f is a global minimum.

Effect of very small step-size α...

$$\min_{x \in \mathbb{R}} \frac{1}{2}(x-3)^2$$

$$\frac{df}{dx} = x - 3$$



Choose $x^0 = 5$ and $\boldsymbol{\alpha = \frac{1}{10}}$

$x^1 = x^0 - \alpha \frac{df}{dx}\Big|_{x=x^0} = 5 - \frac{1}{10}2 = 4.8$

$x^2 = x^1 - \alpha \frac{df}{dx}\Big|_{x=x^1} = 4.8 - \frac{1}{10}1.8 = 4.62$

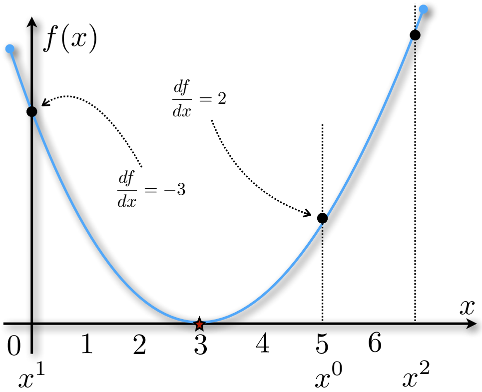
...

x^k converges **very slowly**.

Effect of very large step-size α...

$$\min_{x\in\mathbb{R}} \frac{1}{2}(x-3)^2$$

$$\frac{df}{dx} = x - 3$$



Choose $x^0 = 5$ and $\boldsymbol{\alpha = \frac{5}{2}}$

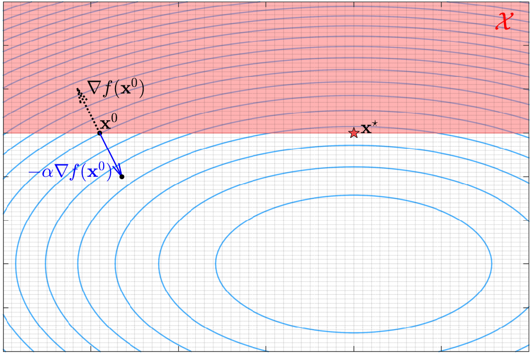
$x^1 = x^0 - \alpha \frac{df}{dx}\Big|_{x=x^0} = 5 - \frac{5}{2}2 = 0$

$x^2 = x^1 - \alpha \frac{df}{dx}\Big|_{x=x^1} = 0 - \frac{5}{2}(-3) = \frac{15}{2}$

...

x^k **diverges**.

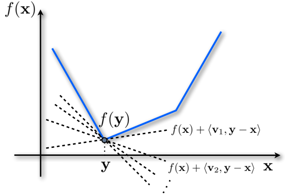
Discontinuities

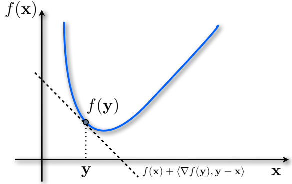


In many practical problems,
we need to **minimize** the cost **under some constraints**.

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p}\left\{f(\mathbf{x}) : \mathbf{x}\in\mathcal{X}\right\}$$

Nonsmooth functions





Definition (Subdifferential)

The subdifferential of f at x, denoted $\partial f(x)$, is the set of all vectors v satisfying

$$f(y) \geq f(x) + \langle v, y - x \rangle + o(\|y - x\|) \quad \text{as } y \to x$$

If the function f is differentiable, then its subdifferential contains only the gradient.

Subgradient method

Choose a starting point $\mathbf{x}^0$, receive a subgradient from the (set of) subdifferential, and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \partial f(\mathbf{x}^k)$$

where $\alpha_k > 0$ is a step-size procedure to be chosen so that $\mathbf{x}^k$ converges to a stationary point.

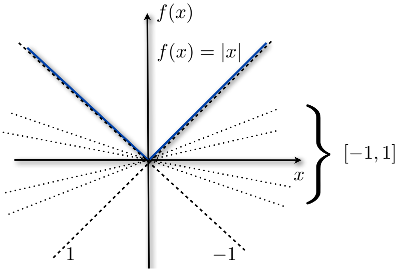
Subdifferentials and (sub)gradients

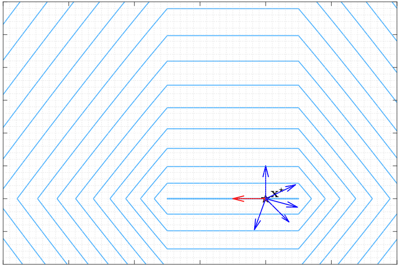
Subgradient method

Choose a starting point $\mathbf{x}^0$, receive a subgradient from the (set of) subdifferential, and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \partial f(\mathbf{x}^k)$$

where $\alpha_k > 0$ is a step-size procedure to be chosen so that $\mathbf{x}^k$ converges to a stationary point.





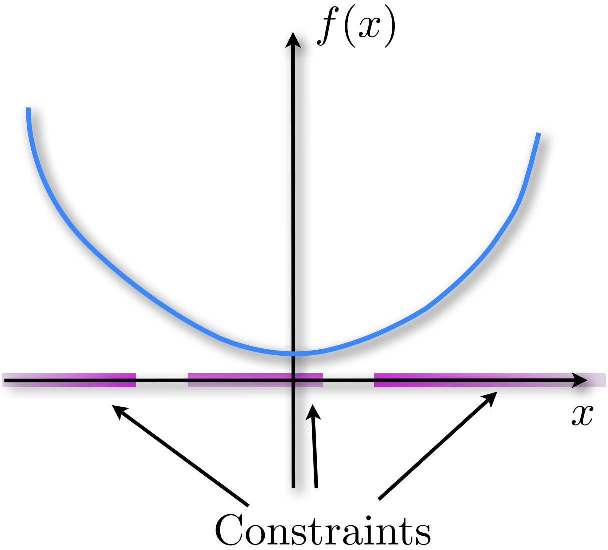
Example

$\partial|x| = \{\text{sgn}(x)\}$, if $x \neq 0$, but $[-1, 1]$, if $x = 0$.

Remark:

The step-size α_k often needs to decrease with k.

Is convexity of f enough for an iterative optimization algorithm?



Smooth unconstrained convex minimization

Problem (Mathematical formulation)

The unconstrained convex minimization problem is defined as:

$$f^\star := \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x})$$

- f *is a convex function that is*
  - *proper* : $\forall \mathbf{x} \in \mathbb{R}^p$, $-\infty < f(\mathbf{x})$ *and there exists* $\mathbf{x} \in \mathbb{R}^p$ *such that* $f(\mathbf{x}) < +\infty$.
  - *closed* : *The epigraph epi*$f = \{(\mathbf{x}, t) \in \mathbb{R}^{p+1}, f(\mathbf{x}) \leq t\}$ *is closed.*
  - *smooth* : f *is differentiable and its gradient* ∇f *is* L*-Lipschitz.*
- *The solution set* $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}\,(f) : f(\mathbf{x}^\star) = f^\star\}$ *is nonempty.*

Example: Maximum likelihood estimation and M-estimators

Problem

Let $\mathbf{x}^\natural \in \mathbb{R}^p$ *be unknown and* $b_1, ..., b_n$ *be i.i.d. samples of a random variable* B *with p.d.f.* $p_{\mathbf{x}^\natural}(b) \in \mathcal{P} := \{p_{\mathbf{x}}(b) : \mathbf{x} \in \mathbb{R}^p\}$. ***Goal:*** *Estimate* $\mathbf{x}^\natural$ *from* $b_1, \ldots, b_n$.

Optimization formulation (ML estimator)

$$\mathbf{x}^\star_{\mathsf{ML}} := \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ -\frac{1}{n}\sum_{i=1}^{n} \ln\left[\mathrm{p}_{\mathbf{x}}(b_i)\right] \right\} = \arg\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$$

Theorem (Performance of the ML estimator [3, 6])

The random variable $\hat{\mathbf{x}}_{ML}$ *satisfies*

$$\lim_{n\to\infty} \sqrt{n}\,\mathbf{J}^{-1/2}\left(\hat{\mathbf{x}}_{ML} - \mathbf{x}^\natural\right) \overset{d}{=} Z \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

where $\mathbf{J} := -\mathbb{E}\left[\nabla^2_{\mathbf{x}} \ln\left[p_{\mathbf{x}}(B)\right]\right]\Big|_{\mathbf{x}=\mathbf{x}^\natural}$ *is the Fisher information matrix associated with one sample. Roughly speaking,*

$$\left\| \sqrt{n}\,\mathbf{J}^{-1/2}\left(\hat{\mathbf{x}}_{ML} - \mathbf{x}^\natural\right)\right\|_2^2 \sim \mathrm{Tr}\left(\mathbf{I}\right) = p \quad \Rightarrow \quad \boxed{\left\|\hat{\mathbf{x}}_{ML} - \mathbf{x}^\natural\right\|_2^2 = \mathcal{O}(p/n)}.$$

Gradient descent methods

Definition

Gradient descent (GD) Starting from $\mathbf{x}^0 \in \mathrm{dom}(f)$, update $\{\mathbf{x}^k\}_{k\geq 0}$ as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k) = \mathbf{x}^k + \alpha_k \mathbf{p}^k.$$

Notice that $\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$ is the steepest descent (anti-gradient) search direction.

Key question: how to choose α_k to have descent/contraction?

Gradient descent methods

Definition

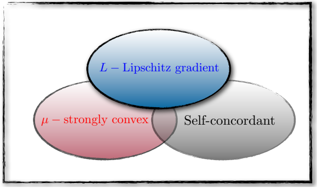
Gradient descent (GD) Starting from $\mathbf{x}^0 \in \mathrm{dom}(f)$, update $\{\mathbf{x}^k\}_{k\geq 0}$ as

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k) = \mathbf{x}^k + \alpha_k \mathbf{p}^k.$$

Notice that $\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$ is the steepest descent (anti-gradient) search direction.

Key question: how to choose α_k to have descent/contraction?

Next few slides: structural assumptions



L-smooth, μ-strongly convex functions

Definition (Recall Recitation 2)

Let $f : \mathcal{Q} \to \mathbb{R}, \mathcal{Q} \subseteq \mathbb{R}^p$ be a continuously differentiable function. Then, f μ-strongly convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{Q}$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|_2^2.$$

The function f is L-smooth if for any $\mathbf{x}, \mathbf{y} \in \mathcal{Q}$,

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|_2^2.$$

If f is twice differentiable, an equivalent characterization of f being L-smooth and μ-strongly convex is

$$\mu \mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq L \mathbf{I}.$$

L-smooth, μ-strongly convex functions

Definition (Recall Recitation 2)

Let $f : \mathcal{Q} \to \mathbb{R}, \mathcal{Q} \subseteq \mathbb{R}^p$ be a continuously differentiable function. Then, f μ-strongly convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{Q}$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|_2^2.$$

The function f is L-smooth if for any $\mathbf{x}, \mathbf{y} \in \mathcal{Q}$,

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2.$$

If f is twice differentiable, an equivalent characterization of f being L-smooth and μ-strongly convex is

$$\mu \mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq L\mathbf{I}.$$

Observations:
- ◦ Both μ and L show up in convergence rate characterization of algorithms
- ◦ **Unfortunately, μ, L are usually not known a priori...**
- ◦ When they are known, they can help significantly (even in stopping algorithms)

Example: Least-squares estimation

Problem

Let $\mathbf{x}^\natural \in \mathbb{R}^p$ *and* $\mathbf{A} \in \mathbb{R}^{n \times p}$ *(full column rank). Goal: estimate* $\mathbf{x}^\natural$*, given* $\mathbf{A}$ *and*

$$\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w},$$

where $\mathbf{w}$ *denotes unknown noise.*

Optimization formulation (Least-squares estimator)

$$\min_{\mathbf{x} \in \mathbb{R}^p} \underbrace{\frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2}_{f(\mathbf{x})} .$$

Structural properties

- $\nabla f(\mathbf{x}) = \mathbf{A}^T(\mathbf{A}\mathbf{x} - \mathbf{b})$, and $\nabla^2 f(\mathbf{x}) = \mathbf{A}^T\mathbf{A}$.
- $\lambda_p \mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq \lambda_1 \mathbf{I}$, where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$ are the eigenvalues of $\mathbf{A}^T\mathbf{A}$.
- It follows that $L = \lambda_1$ and $\mu = \lambda_p$. If $\lambda_p > 0$, then f is L-smooth and μ-strongly convex, otherwise f is just L-smooth.
- Since $\mathrm{rank}(\mathbf{A}^T\mathbf{A}) \leq \min\{n, p\}$, if $n < p$, then $\lambda_p = 0$.

Back to gradient descent methods

Gradient descent (GD) algorithm

Starting from $\mathbf{x}^0 \in \mathrm{dom}(f)$, produce the sequence $\mathbf{x}^1, ..., \mathbf{x}^k, ...$ according to

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k) = \mathbf{x}^k + \alpha_k \mathbf{p}^k.$$

Notice that $\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$ is the steepest descent (anti-gradient) direction.
Key question: how do we choose α_k to have descent/contraction?

Back to gradient descent methods

Gradient descent (GD) algorithm

Starting from $\mathbf{x}^0 \in \mathrm{dom}(f)$, produce the sequence $\mathbf{x}^1, ..., \mathbf{x}^k, ...$ according to

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k) = \mathbf{x}^k + \alpha_k \mathbf{p}^k.$$

Notice that $\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$ is the steepest descent (anti-gradient) direction.
Key question: how do we choose α_k to have descent/contraction?

Step-size selection

Case 1: If f is L-smooth, then:

- We can choose $0 < \alpha_k < \frac{2}{L}$. The optimal choice is $\alpha_k := \frac{1}{L}$.
- α_k can be determined by a line-search procedure:
  1. **Exact line search**: $\alpha_k := \arg\min_{\alpha > 0} f(\mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k))$.
  2. **Back-tracking line search** with Armijo-Goldstein's condition:

$$f(\mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k)) \leq f(\mathbf{x}^k) - c\alpha \|\nabla f(\mathbf{x}^k)\|^2, \quad c \in (0, 1/2].$$

Case 2: If in addition to being L-smooth, f is μ-strongly convex, then:

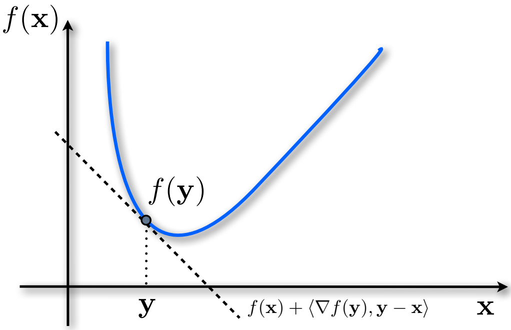
- We can choose $0 < \alpha_k \leq \frac{2}{L+\mu}$. The optimal choice is $\alpha_k := \frac{2}{L+\mu}$.

Towards a geometric interpretation I

Recall:

- Let f be L-smooth with gradient $\nabla f(\mathbf{x})$ and Hessian $\nabla^2 f(\mathbf{x})$.
- First-order Taylor approximation of f at $\mathbf{y}$:

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$$



- Convex functions: 1^{st}-order Taylor approximation is a global lower surrogate.

An equivalent characterization of smoothness

Lemma

Let f be a continuously differentiable convex function :

$$f \text{ is } L\text{-Lipschitz gradient} \implies f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2$$

Proof:

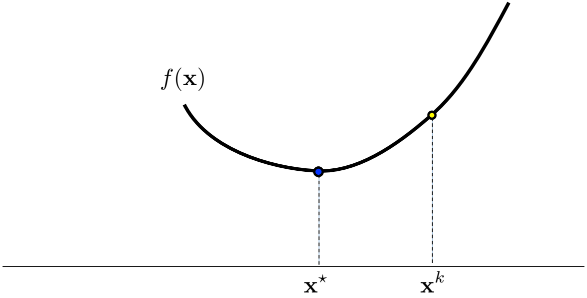
- By Taylor's theorem:

$$f(\mathbf{y}) = f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \langle \nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle d\tau.$$

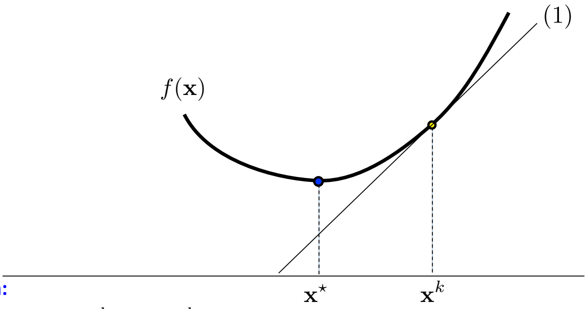
Therefore,

$$\begin{aligned} f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle &\leq \int_0^1 \|\nabla f(\mathbf{x} + \tau(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x})\|^* \cdot \|\mathbf{y} - \mathbf{x}\| d\tau \\ &\leq L\|\mathbf{y} - \mathbf{x}\|_2^2 \int_0^1 \tau d\tau = \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2 \end{aligned}$$

Gradient descent methods: geometrical intuition



Gradient descent methods: geometrical intuition



Structure in optimization:

$$(1) \qquad f(\mathbf{x}) \geq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$$

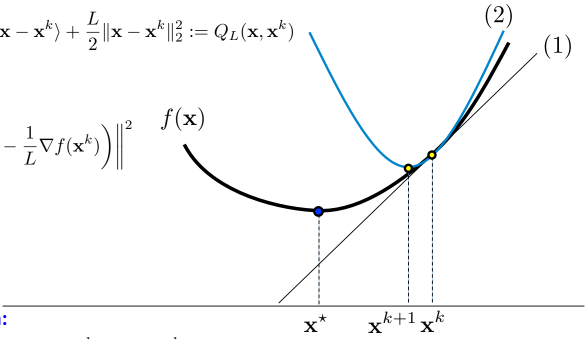
Gradient descent methods: geometrical intuition

Majorize:

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_L(\mathbf{x}, \mathbf{x}^k)$$

Minimize:

$$\begin{aligned}
\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{x}^k) \\
&= \arg\min_{\mathbf{x}} \left\| \mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k) \right) \right\|^2 \\
&= \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k)
\end{aligned}$$



Structure in optimization:

(1) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

(2) $\quad f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

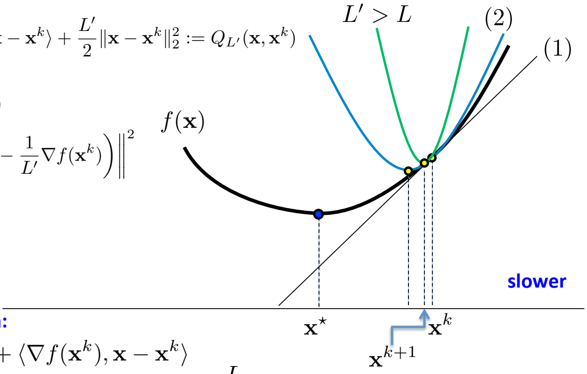
Gradient descent methods: geometrical intuition

Majorize:

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L'}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_{L'}(\mathbf{x}, \mathbf{x}^k)$$

Minimize:

$$\begin{aligned}
\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_{L'}(\mathbf{x}, \mathbf{x}^k) \\
&= \arg\min_{\mathbf{x}} \left\| \mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L'}\nabla f(\mathbf{x}^k) \right) \right\|^2 \\
&= \mathbf{x}^k - \frac{1}{L'}\nabla f(\mathbf{x}^k)
\end{aligned}$$



Structure in optimization:

(1) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

(2) $\quad f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

Convergence rate of gradient descent

Theorem

Let f be a twice-differentiable convex function, if

$$
\begin{array}{llllll}
f \text{ is } L\text{-smooth,} & \alpha = \frac{1}{L}: & f(\mathbf{x}^k) - f(\mathbf{x}^\star) & \leq & \frac{2L}{k+4} & \|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2 \\
f \text{ is } L\text{-smooth and } \mu\text{-strongly convex,} & \alpha = \frac{2}{L+\mu}: & \|\mathbf{x}^k - \mathbf{x}^\star\|_2 & \leq & \left(\frac{L-\mu}{L+\mu}\right)^k & \|\mathbf{x}^0 - \mathbf{x}^\star\|_2 \\
f \text{ is } L\text{-smooth and } \mu\text{-strongly convex,} & \alpha = \frac{1}{L}: & \|\mathbf{x}^k - \mathbf{x}^\star\|_2 & \leq & \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} & \|\mathbf{x}^0 - \mathbf{x}^\star\|_2
\end{array}
$$

Note that $\frac{L-\mu}{L+\mu} = \frac{\kappa-1}{\kappa+1}$, where $\kappa := \frac{L}{\mu}$ is the condition number of $\nabla^2 f$.

Convergence rate of gradient descent

Theorem

Let f be a twice-differentiable convex function, if

$$
\begin{aligned}
&f \text{ is } L\text{-smooth}, && \alpha = \frac{1}{L}: && f(\mathbf{x}^k) - f(\mathbf{x}^\star) && \leq \frac{2L}{k+4} && \|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2 \\
&f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, && \alpha = \frac{2}{L+\mu}: && \|\mathbf{x}^k - \mathbf{x}^\star\|_2 && \leq \left(\frac{L-\mu}{L+\mu}\right)^k && \|\mathbf{x}^0 - \mathbf{x}^\star\|_2 \\
&f \text{ is } L\text{-smooth and } \mu\text{-strongly convex}, && \alpha = \frac{1}{L}: && \|\mathbf{x}^k - \mathbf{x}^\star\|_2 && \leq \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} && \|\mathbf{x}^0 - \mathbf{x}^\star\|_2
\end{aligned}
$$

Note that $\frac{L-\mu}{L+\mu} = \frac{\kappa-1}{\kappa+1}$, where $\kappa := \frac{L}{\mu}$ is the condition number of $\nabla^2 f$.

Remarks

- **Assumption:** Lipschitz gradient. **Result:** convergence rate in **objective values**.
- **Assumption:** Strong convexity. **Result:** convergence rate in **sequence** of the iterates and in **objective values**.
- Note that the suboptimal step-size choice $\alpha = \frac{1}{L}$ adapts to the strongly convex case (i.e., it features a linear rate vs. the standard sublinear rate).

Example: Ridge regression

Optimization formulation

- Let $\mathbf{A} \in \mathbb{R}^{n \times p}$ and $\mathbf{b} \in \mathbb{R}^n$ given by $\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w} \in \mathbb{R}^n$ is some noise.
- A classical estimator of $\mathbf{x}^\natural$, known as ridge regression, is

$$\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) := \frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \frac{\rho}{2}\|\mathbf{x}\|_2^2.$$

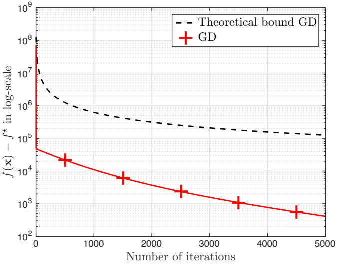
where $\rho \geq 0$ is a regularization parameter

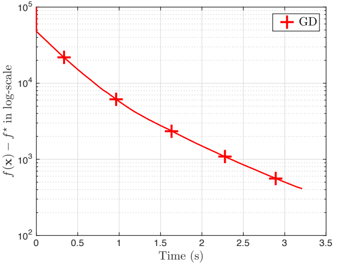
Remarks

- f is L-smooth and μ-strongly convex with:
  - $L = \lambda_1(\mathbf{A}^T\mathbf{A}) + \rho$;
  - $\mu = \lambda_p(\mathbf{A}^T\mathbf{A}) + \rho$;
  - where $\lambda_1 \geq \ldots \geq \lambda_p$ are the eigenvalues of $\mathbf{A}^T\mathbf{A}$.
- The ratio $\kappa = \frac{L}{\mu}$ decreases as ρ increases, leading to faster linear convergence.
- Note that if $n < p$ and $\rho = 0$, we have $\mu = 0$, hence f is only L-smooth and we can expect only $\mathcal{O}(1/k)$ convergence from the gradient descent method.

Example: Ridge regression

Case 1: $n = 500, p = 2000, \rho = 0$

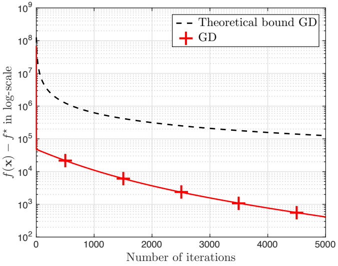


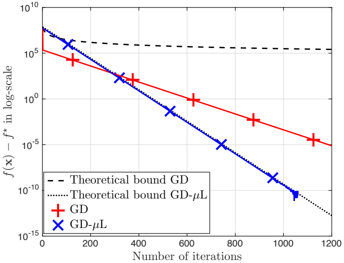


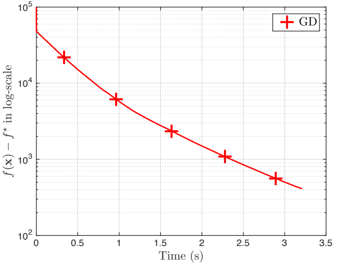
Example: Ridge regression

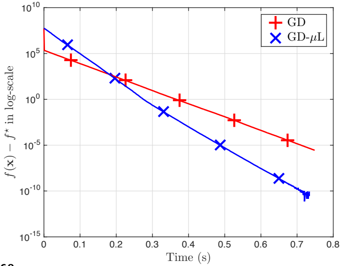
Case 1: $n = 500, p = 2000, \rho = 0$

Case 2: $n = 500, p = 2000, \rho = 0.01\lambda_p(\mathbf{A}^T\mathbf{A})$









Smooth unconstrained non-convex minimization

Problem (Mathematical formulation)

Let us consider the following problem formulation:

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$$

- *f is a smooth and possibly non-convex function.*
- *Recall that finding the global minimizer, i.e., $f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$, is NP-hard*

Example: Image classification using neural networks

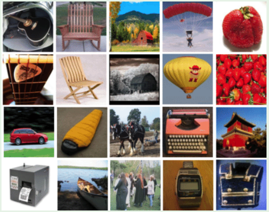
Neural network formulation

- $(\mathbf{a}_i, b_i)$: sample points, $\sigma(\cdot)$: non-linear activation function
- the function class $\mathcal{H}$ is given by $\mathcal{H} := \left\{h_{\mathbf{x}}(\mathbf{a}), \mathbf{x} \in \mathbb{R}^d\right\}$, where

$$\mathbf{x} = (\mathbf{W}_1, \boldsymbol{\mu}_1, \mathbf{W}_2, \boldsymbol{\mu}_2, \ldots, \mathbf{W}_k, \boldsymbol{\mu}_k), \quad \mathbf{W}_i \in \mathbb{R}^{d_i \times d_{i-1}}, \quad \boldsymbol{\mu}_i \in \mathbb{R}^{d_i},$$
$$h_{\mathbf{x}}(\mathbf{a}) = \sigma\left(\mathbf{W}_k \sigma\left(\cdots \sigma\left(\mathbf{W}_2 \sigma\left(\mathbf{W}_1 \mathbf{a} + \boldsymbol{\mu}_1\right) + \boldsymbol{\mu}_2\right) \cdots\right) + \boldsymbol{\mu}_k\right)$$

- the loss function is given by $L(h_{\mathbf{x}}(\mathbf{a}), b) := (b - h_{\mathbf{x}}(\mathbf{a}))^2$.

Example: Image classification



Imagenet: 1000 object classes.
1.2M/100K train/test images
Below human level error rates!

Example: Phase retrieval for fourier ptychography

Definition (Phase retrieval)

Given a set of measurements of the amplitude of a signal, phase retrieval is the task of finding the phase for the original signal that satisfies certain constraints/properties.

Definition (Fourier ptychography)

Fourier ptychography is the task of reconstructing high-resolution images from low resolution samples, based on optical microscopy. It is a special case of phase retrieval problem.

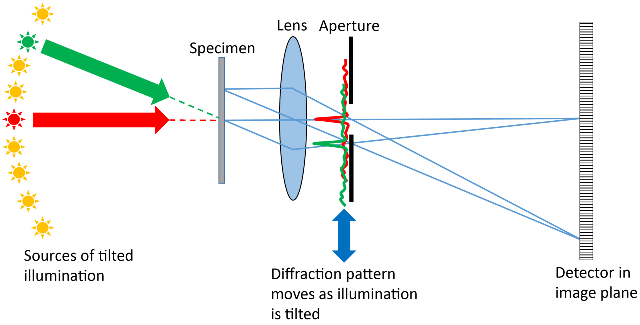
Example: Phase retrieval for fourier ptychography

Definition (Phase retrieval)

Given a set of measurements of the amplitude of a signal, phase retrieval is the task of finding the phase for the original signal that satisfies certain constraints/properties.

Definition (Fourier ptychography)

Fourier ptychography is the task of reconstructing high-resolution images from low resolution samples, based on optical microscopy. It is a special case of phase retrieval problem.



The necessity of non-convex optimization

Why non-convex?

- Inherent properties of optimization problem, e.g., phase retrieval
- Robustness or better estimation, e.g., binary classification with non-convex losses

Optimization Formulation: Phase Retrieval

$$\min_{\mathbf{x}} \||\mathbf{A}\mathbf{x}|^2 - \mathbf{b}\|_2^2$$

where $\mathbf{x} \in \mathbb{C}^p$ is a complex signal and $|\mathbf{A}\mathbf{x}|$ is the component-wise magnitude of the measurement $\mathbf{A}\mathbf{x}$.

Optimization Formulation: Binary Classification

$$\min_{x} \left\{ \frac{1}{n} \sum_{i=1}^{n} (b_i - g(\mathbf{a}_i, \mathbf{x}))^2 \right\}$$

where $g(\cdot, \cdot)$ is non-linear, and hence, the loss function is non-convex.

Notion of convergence: Stationarity

◦ Let $f : \mathbb{R}^d \to \mathbb{R}$ be twice-differentiable and $\mathbf{x}^\star \in \arg\min_{x \in \mathbb{R}^d} f(\mathbf{x})$

Definition (Recall - First order stationary point)

A point $\bar{\mathbf{x}}$ is a first order stationary point of a twice differentiable function $f(\mathbf{x})$ if

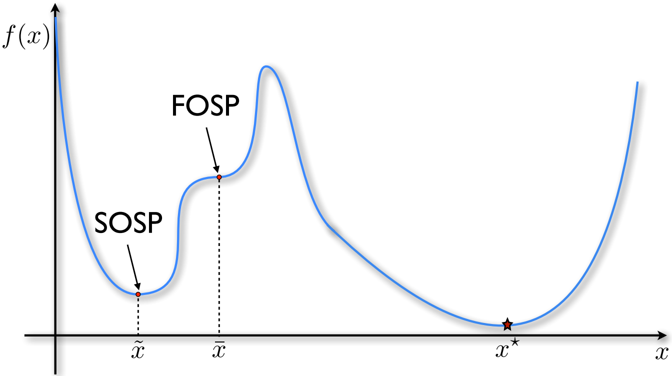
$$\nabla f(\bar{\mathbf{x}}) = \mathbf{0}.$$

Definition (Recall - Second order stationary point)

A point $\tilde{\mathbf{x}}$ is a second order stationary point of a twice differentiable function $f(\mathbf{x})$ if

$$\nabla f(\tilde{\mathbf{x}}) = \mathbf{0} \quad \text{and} \quad \nabla^2 f(\tilde{\mathbf{x}}) \succeq \mathbf{0}.$$

Geometric interpretation of stationarity



∘ Note that neither $\bar{\mathbf{x}}$, nor $\tilde{\mathbf{x}}$ is **not necessarily** equal to $\mathbf{x}^\star$!!

Assumptions and the gradient method

Assumption: Smoothness

Let f be a twice differentiable function that is L-Lipschitz gradient with respect to ℓ_2-norm, such that,

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L\|\mathbf{x} - \mathbf{y}\|_2$$

Gradient descent

Let $\alpha \leq \frac{1}{L}$ be the constant step size and $\mathbf{x}^0 \in \mathrm{dom}(f)$ be the initial point. Then, gradient method produces iterates using the following iterative update,

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k)$$

Convergence rate and iteration complexity

Theorem

Let f be a twice differentiable L-Lipschitz gradient function, and $\alpha \leq \frac{1}{L}$. Then, gradient method converges to the FOSP with the following properties:

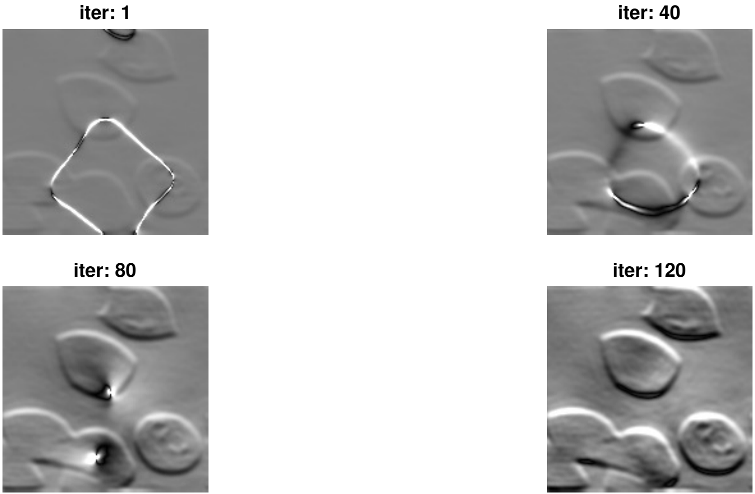
Convergence rate to an ϵ-FOSP:

$$\|\nabla f(\mathbf{x}^k)\| = O\left(\frac{1}{\sqrt{k}}\right)$$

Iteration complexity to reach an ϵ-FOSP:

$$O\left(\frac{1}{\epsilon^2}\right)$$

Example: Malaria infection detection



*Proof of convergence rates of gradient descent in the convex case

- We first need to prove a basic result about convex L-Lipschitz gradient functions.

Lemma

Let f be a convex differentiable L-Lipschitz gradient function. Then it holds that

$$\frac{1}{L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq \langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle \tag{2}$$

Proof.

First, recall the following result about convex Lipschitz gradient functions h

$$h(\mathbf{x}) \leq h(\mathbf{y}) + \langle \nabla h(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|_2^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathrm{dom} h \tag{3}$$

To prove the result, take ϕ to be the convex function $\phi(\mathbf{y}) := f(\mathbf{y}) - \langle \nabla f(\mathbf{x}), \mathbf{y}\rangle$, with $\nabla \phi(\mathbf{y}) = \nabla f(\mathbf{y}) - \nabla f(\mathbf{x})$. Using the first order characterization of convexity of f, we can show that for all y, $\phi(y) - \phi(x) \geq 0$. Therefore ϕ attains its minimum value at $\mathbf{y}^\star = \mathbf{x}$. By applying (3) with $h = \phi$ and $\mathbf{x} = \mathbf{y} - \frac{1}{L}\nabla\phi(\mathbf{y})$, we get

$$\phi(\mathbf{x}) \leq \phi\left(\mathbf{y} - \frac{1}{L}\nabla \phi(\mathbf{y})\right) \leq \phi(\mathbf{y}) - \frac{1}{2L}\|\nabla \phi(\mathbf{y})\|_2^2.$$

Plugging the definition of ϕ back in the left and right hand sides gives

$$f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq f(\mathbf{y}) \tag{4}$$

By adding two copies of (4) with each other $\mathbf{x}$ and $\mathbf{y}$ swapped, we obtain (2).

*The proof of convergence rates in the convex case- part I

Theorem

If f is twice differentiable, convex, L-Lipschitz gradient, with the choice $\alpha = \frac{1}{L}$, the iterates of GD satisfy

$$\boxed{f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2L}{k+4}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2} \tag{5}$$

Proof

- Consider the constant step-size iteration $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha\nabla f(\mathbf{x}^k)$.
- Let $r_k := \|\mathbf{x}^k - \mathbf{x}^\star\|$. Show $\boxed{r_k \leq r_0}$.

$$\begin{aligned} r_{k+1}^2 &:= \|\mathbf{x}^{k+1} - \mathbf{x}^\star\|^2 = \|\mathbf{x}^k - \mathbf{x}^\star - \alpha\nabla f(\mathbf{x}^k)\|^2 \\ &= \|\mathbf{x}^k - \mathbf{x}^\star\|^2 - 2\alpha\langle\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star), \mathbf{x}^k - \mathbf{x}^\star\rangle + \alpha^2\|\nabla f(\mathbf{x}^k)\|^2 \\ &\leq r_k^2 - \alpha(2/L - \alpha)\|\nabla f(\mathbf{x}^k)\|^2 \quad \text{(by (2))} \\ &< r_k^2, \quad \forall\alpha < 2/L. \end{aligned}$$

  Hence, the gradient iterations are contractive when $\alpha < 2/L$ for all $k \geq 0$.

- **An auxiliary result:** Let $\Delta_k := f(\mathbf{x}^k) - f^\star$. Show $\boxed{\Delta_k \leq r_0\|\nabla f(\mathbf{x}^k)\|}$.

$$\Delta_k \leq \langle\nabla f(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}^\star\rangle \leq \|\nabla f(\mathbf{x}^k)\|\|\mathbf{x}^k - \mathbf{x}^\star\| = r_k\|\nabla f(\mathbf{x}^k)\| \leq r_0\|\nabla f(\mathbf{x}^k)\|.$$

*The proof of convergence rates in the convex case- part II

Proof (continued)

- We can establish **convergence** along with the auxiliary result above:

$$
\begin{aligned}
f(\mathbf{x}^{k+1}) &\leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \\
&\leq f(\mathbf{x}^k) - \omega_k \|\nabla f(\mathbf{x}^k)\|^2, \quad \omega_k := \alpha(1 - L\alpha/2).
\end{aligned}
$$

Subtract f^* from both sides and apply the last equation of the previous slide to get $\boxed{\Delta_{k+1} \leq \Delta_k - (\omega_k/r_0^2)\Delta_k^2}$. Thus, dividing by $\Delta_{k+1}\Delta_k$

$$
\Delta_{k+1}^{-1} \geq \Delta_k^{-1} + (\omega_k/r_0^2)\Delta_k/\Delta_{k+1} \geq \Delta_k^{-1} + (\omega_k/r_0^2).
$$

By induction, we have $\Delta_{k+1}^{-1} \geq \Delta_0^{-1} + (\omega_k/r_0^2)(k+1)$. Then, taking $(\cdot)^{-1}$ of both sides (and hence replacing $\geq$ by $\leq$) and substituting all of the definitions gives

$$
f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2(f(\mathbf{x}_0) - f(\mathbf{x}^\star))\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2}{2\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2 + k\alpha(2 - \alpha L)(f(\mathbf{x}_0) - f^\star)},
$$

- In order to choose the **optimal** step-size, we maximize the function $\phi(\alpha) = \alpha(2 - \alpha L)$. Hence, the optimal step size for the gradient method for f L-Lispchitz gradient is given by $\alpha = \frac{1}{L}$.
- Finally, since $f(\mathbf{x}_0) \leq f^* + \nabla f(\mathbf{x}^\star)^T(\mathbf{x}_0 - \mathbf{x}^\star) + (L/2)\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2 = f^* + (L/2)r_0^2$, we obtain (5).

□

*The proof of convergence rates in the convex case- part III

Theorem

If f is twice-differentiable, μ-strongly convex and L-smooth,

- *with $\alpha = \frac{2}{L+\mu}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2} \tag{6}$$

- *with $\alpha = \frac{1}{L}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2} \tag{7}$$

Before proving the convergence rate, we first need a result about μ-strongly convex and L-smooth functions.

Theorem

If f is μ-strongly convex and L-smooth, then for any $\mathbf{x}$ and $\mathbf{y}$, we have

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \frac{\mu L}{\mu + L}\|\mathbf{x} - \mathbf{y}\|^2 + \frac{1}{\mu + L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2. \tag{8}$$

*The proof of convergence rates in the convex case - part III

Proof of (6) and (7)

- Let $r_k = \|\mathbf{x}^k - \mathbf{x}^\star\|$. Then, using (8) and the fact that $\nabla f(x^*) = 0$, we have

$$\begin{aligned} r_{k+1}^2 &= \|\mathbf{x}_{k+1} - \mathbf{x}^\star - \alpha \nabla f(\mathbf{x}^k)\|^2 \\ &= r_k^2 - 2\alpha\langle \nabla f(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}^\star\rangle + \alpha^2 \|\nabla f(\mathbf{x}^k)\|^2 \\ &\leq \left(1 - \frac{2\alpha\mu L}{\mu + L}\right) r_k^2 + \alpha\left(\alpha - \frac{2}{\mu + L}\right)\|\nabla f(\mathbf{x}^k)\|^2 \end{aligned}$$

- Since $\mu \leq L$, we have $\alpha \leq \frac{2}{\mu+L}$ in both the cases $\alpha = \frac{1}{L}$ or $\alpha = \frac{2}{\mu+L}$. So the last term in the previous inequality is less than 0, and hence

$$r_{k+1}^2 \leq \left(1 - \frac{2\alpha\mu L}{\mu + L}\right)^k r_0^2$$

- Plugging $\alpha = \frac{1}{L}$ and $\alpha = \frac{2}{\mu+L}$, we obtain the rates as advertised.
- For $f \in \mathcal{F}_{L,\mu}^{1,1}$, the **optimal** step-size is given by $\alpha = \frac{2}{\mu+L}$ (i.e., it optimizes the worst case bound).

□

*From gradient descent to mirror descent

Gradient descent as a majorization-minimization scheme

- **Majorize** f at $\mathbf{x}^k$ by using L-Lipschitz gradient continuity

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q(\mathbf{x}, \mathbf{x}^k)$$

- **Minimize** $Q(\mathbf{x}, \mathbf{x}^k)$ to obtain the next iterate $\mathbf{x}^{k+1}$

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q(\mathbf{x}, \mathbf{x}^k) \Rightarrow \nabla f(\mathbf{x}^k) + L(\mathbf{x}^{k+1} - \mathbf{x}^k) = 0 \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k)\end{aligned}$$

Other majorizers

We can re-write the majorization step as

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \alpha d(\mathbf{x}, \mathbf{x}^k)$$

where $d(\mathbf{x}, \mathbf{x}^k) = \frac{1}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$ is the Euclidean distance and $\alpha = L$.

- Can we use a different function $d(\mathbf{x}, \mathbf{x}^k)$ that is better suited to minimizing f?

*Bregman divergences

Definition (Bregman divergence)

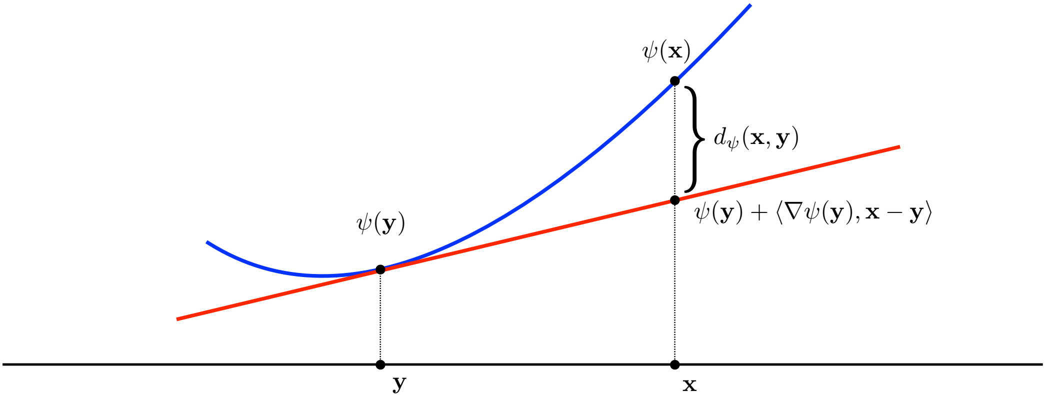
Let $\psi : \mathcal{S} \to \mathbb{R}$ be a continuously-differentiable and strictly convex function defined on a closed convex set $\mathcal{S}$. The **Bregman divergence** (d_ψ) associated with ψ for points $\mathbf{x}$ and $\mathbf{y}$ is:

$$d_\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x}) - \psi(\mathbf{y}) - \langle \nabla\psi(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$$

- $\psi(\cdot)$ is referred to as the Bregman or proximity function.
- The Bregman divergence satisfies the following properties:
  (a) $d_\psi(\mathbf{x}, \mathbf{y}) \geq 0$ for all $\mathbf{x}$ and $\mathbf{y}$ with equality if and only if $\mathbf{x} = \mathbf{y}$
  (b) Define $q(\mathbf{x}) := d_\psi(\mathbf{x}, \mathbf{y})$ for a fixed $\mathbf{y}$, then $\nabla q(\mathbf{x}) = \nabla\psi(\mathbf{x}) - \nabla\psi(\mathbf{y})$
  (c) For all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{S}$, $d_\psi(\mathbf{x}, \mathbf{y}) = d_\psi(\mathbf{x}, \mathbf{z}) + d_\psi(\mathbf{z}, \mathbf{y}) + \langle (\mathbf{x} - \mathbf{z}), \nabla\psi(\mathbf{y}) - \nabla\psi(\mathbf{z}) \rangle$
  (d) For all $\mathbf{x}, \mathbf{y} \in \mathcal{S}$, $d_\psi(\mathbf{x}, \mathbf{y}) + d_\psi(\mathbf{y}, \mathbf{x}) = \langle (\mathbf{x} - \mathbf{y}), \nabla\psi(\mathbf{x}) - \nabla\psi(\mathbf{y}) \rangle$
- The Bregman divergence becomes a Bregman distance when it is *symmetric* (i.e. $d_\psi(\mathbf{x}, \mathbf{y}) = d_\psi(\mathbf{y}, \mathbf{x})$) and satisfies the *triangle inequality*.
- "*All Bregman distances are Bregman divergences but the reverse is not true!*"

*Bregman divergences

- The Bregman divergence is the vertical distance at $\mathbf{x}$ between ψ and the tangent of ψ at $\mathbf{y}$, see figure below



- The Bregman divergence measures the strictness of convexity of $\psi(\cdot)$.

⋆Bregman divergences

Table: **Bregman functions** $\psi(\mathbf{x})$ & corresponding Bregman divergences/distances $d_\psi(\mathbf{x},\mathbf{y})$[a].

Name (or Loss)	**Domain**[b]	$\psi(\mathbf{x})$	$d_\psi(\mathbf{x},\mathbf{y})$
Squared loss	$\mathbb{R}$	x^2	$(x-y)^2$
Itakura-Saito divergence	$\mathbb{R}_{++}$	$-\log x$	$\frac{x}{y} - \log\left(\frac{x}{y}\right) - 1$
Squared Euclidean distance	$\mathbb{R}^p$	$\\|\mathbf{x}\\|_2^2$	$\\|\mathbf{x}-\mathbf{y}\\|_2^2$
Squared Mahalanobis distance	$\mathbb{R}^p$	$\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle$	$\langle (\mathbf{x}-\mathbf{y}), \mathbf{A}(\mathbf{x}-\mathbf{y})\rangle$[c]
Entropy distance	p-simplex[d]	$\sum_i x_i \log x_i$	$\sum_i x_i \log\left(\frac{x_i}{y_i}\right)$
Generalized I-divergence	$\mathbb{R}^p_+$	$\sum_i x_i \log x_i$	$\sum_i \left(\log\left(\frac{x_i}{y_i}\right) - (x_i - y_i)\right)$
von Neumann divergence	$\mathbb{S}^{p\times p}_+$	$\mathbf{X}\log\mathbf{X} - \mathbf{X}$	$\mathrm{tr}\left(\mathbf{X}(\log\mathbf{X} - \log\mathbf{Y}) - \mathbf{X} + \mathbf{Y}\right)$[e]
logdet divergence	$\mathbb{S}^{p\times p}_+$	$-\log\det\mathbf{X}$	$\mathrm{tr}\left(\mathbf{X}\mathbf{Y}^{-1}\right) - \log\det\left(\mathbf{X}\mathbf{Y}^{-1}\right) - p$

[a] $x, y \in \mathbb{R}$, $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$ and $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{p\times p}$.

[b] $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote non-negative and positive real numbers respectively.

[c] $\mathbf{A} \in \mathbb{S}^{p\times p}_+$, the set of symmetric positive semidefinite matrix.

[d] p-simplex:= $\{\mathbf{x} \in \mathbb{R}^p : \sum_{i=1}^p x_i = 1, x_i \geq 0, i = 1, \ldots, p\}$

[e] $\mathrm{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$.

*Mirror descent [1]

What happens if we use a Bregman distance d_ψ in gradient descent?

Let $\psi : \mathbb{R}^p \to \mathbb{R}$ be a μ-strongly convex and continuously differentiable function and let the associated Bregman distance be $d_\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x}) - \psi(\mathbf{y}) - \langle \mathbf{x} - \mathbf{y}, \nabla\psi(\mathbf{y})\rangle$.
Assume that the inverse mapping $\psi^\star$ of ψ is easily computable (i.e., its convex conjugate).

- **Majorize**: Find α_k such that

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k\rangle + \frac{1}{\alpha_k} d_\psi(\mathbf{x}, \mathbf{x}^k) := Q_\psi^k(\mathbf{x}, \mathbf{x}^k)$$

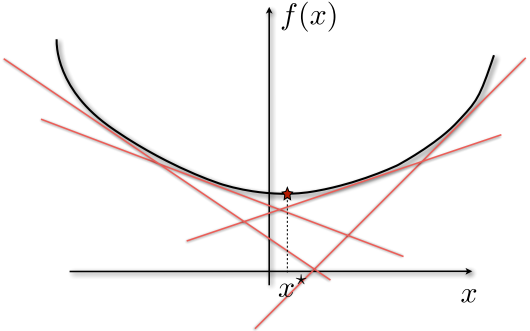
- **Minimize**

$$\begin{aligned}
\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_\psi^k(\mathbf{x}, \mathbf{x}^k) \Rightarrow \nabla f(\mathbf{x}^k) + \frac{1}{\alpha_k}\left(\nabla\psi(\mathbf{x}^{k+1}) - \nabla\psi(\mathbf{x}^k)\right) = 0 \\
\nabla\psi(\mathbf{x}^{k+1}) &= \nabla\psi(\mathbf{x}^k) - \alpha_k \nabla f(\mathbf{x}^k) \\
\mathbf{x}^{k+1} &= \nabla\psi^*(\nabla\psi(\mathbf{x}^k) - \alpha_k\nabla f(\mathbf{x}^k)) \qquad (\nabla\psi(\cdot))^{-1} = \nabla\psi^*(\cdot)[5].
\end{aligned}$$

- Mirror descent is a **generalization** of gradient descent for functions that are Lipschitz-gradient in norms other than the Euclidean.
- MD allows to deal with some **constraints** via a proper choice of ψ.

*What to keep in mind about mirror descent?

- Approximates the optimum by lower bounding the function via hyperplanes at $\mathbf{x}_t$



- The smaller the gradients, the better the approximation!

*Mirror descent example

How can we minimize a convex function over the unit simplex?

$$\min_{\mathbf{x}\in\Delta} f(\mathbf{x}),$$

where

- $\Delta := \{\mathbf{x} \in \mathbb{R}^p \ : \ \sum_{j=1}^p x_j = 1, \mathbf{x} \geq 0\}$ is the **unit simplex**;
- f is convex L_f-Lipschitz continuous with respect to some norm $\|\cdot\|$. (not necessarily *L-Lipschitz gradient*)

Entropy function

- Define the entropy function

$$\psi_e(\mathbf{x}) = \sum_{j=1}^p x_j \ln x_j \quad \text{if } \mathbf{x} \in \Delta, \quad +\infty \text{ otherwise.}$$

- ψ_e is 1-strongly convex over $\mathrm{int}\Delta$ with respect to $\|\cdot\|_1$.
- $\psi_e^\star(\mathbf{z}) = \ln \sum_{j=1}^p e^{z_j}$ and $\|\nabla\psi_e(\mathbf{x})\| \to \infty$ as $\mathbf{x} \to \tilde{\mathbf{x}} \in \Delta$.
- Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$, then $d_\psi(\mathbf{x}, \mathbf{x}^0) \leq \ln p$ for all $\mathbf{x} \in \Delta$.

⋆Entropic descent algorithm [1]

Entropic descent algorithm (EDA)

Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$ and generate the following sequence

$$x_j^{k+1} = \frac{x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}{\sum_{j=1}^{p} x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}, \quad t_k = \frac{\sqrt{2 \ln p}}{L_f} \frac{1}{\sqrt{k}},$$

where $f'(\mathbf{x}) = (f_1(\mathbf{x})', \ldots, f_p(\mathbf{x})')^T \in \partial f(\mathbf{x})$, which is the **subdifferential** of f at $\mathbf{x}$.

- This is an example of **non-smooth** and **constrained** optimization;
- The updates are multiplicative.

*Convergence of mirror descent

Problem

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \tag{9}$$

where

- $\mathcal{X}$ *is a closed convex subset of* $\mathbb{R}^p$*;*
- f *is convex* L_f*-Lipschitz continuous with respect to some norm* $\|\cdot\|$.

Theorem ([1])

Let $\{\mathbf{x}^k\}$ *be the sequence generated by mirror descent with* $\mathbf{x}^0 \in \mathrm{int}\mathcal{X}$.
If the step-sizes are chosen as

$$\alpha_k = \frac{\sqrt{2\mu d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}}{L_f}\frac{1}{\sqrt{k}}$$

the following convergence rate holds

$$\min_{0\leq s\leq k} f(\mathbf{x}^s) - f^\star \leq L_f\sqrt{\frac{2d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}{\mu}}\frac{1}{\sqrt{k}}$$

- *This convergence rate is* **optimal** *for solving* (9) *with a first-order method.*

References I

[1] Amir Beck and Marc Teboulle.
Mirror descent and nonlinear projected subgradient methods for convex optimization.
Operations Research Letters, 31(3):167–175, 2003.

[2] Peter J. Huber and Elvezio M. Ronchetti.
Robust Statistics.
John Wiley & Sons, Hoboken, NJ, 2009.

[3] Lucien Le Cam.
Asymptotic methods in Statistical Decision Theory.
Springer-Verl., New York, NY, 1986.

[4] Yu. Nesterov.
Introductory Lectures on Convex Optimization: A Basic Course.
Kluwer, Boston, MA, 2004.

[5] R.T. Rockafellar.
Convex analysis.
Princeton University Press (Princeton, NJ), 1970.

References II

[6] A. W. van der Vaart.
Asymptotic Statistics.
Cambridge Univ. Press, Cambridge, UK, 1998.