Optimization Methods for Machine Learning

BIMSA Open Course - SP2024

Yi-Shuai Niu

BIMSA

北京雁栖湖应用数学研究院
Beijing Institute of Mathematical Sciences and Applications

https://bimsa.net:10000/activity/OptMetforMacLea/

Table of Contents

Part 1: Course Organization

北京雁栖湖
应用数学研究院
Beijing Institute
of Mathematical
Sciences and Applications

About Me

- **Name:** Yi-Shuai Niu
- **Position:** Associate Professor of Mathematics
- **Office:** BIMSA A6-301
- **Website:** https://www.bimsa.cn/newsinfo/872707.html
- **Email:** niuyishuai@bimsa.cn
- **Brief Academic CV:**
  - **Work Experience:**
    - Since 2023: Beijing Institute of Mathematical Sciences and Applications (BIMSA), Associate Professor
    - 2021 - 2022: Hong Kong Polytechnique University, Senior Research Fellow
    - 2014 - 2021: Shanghai Jiao Tong University, Associate Professor
    - 2013 - 2014: University of Paris 6, Postdoc
    - 2010 - 2012: French National Center for Scientific Research (CNRS), Junior Researcher
    - 2007 - 2010: French National Institute of Applied Sciences of Rouen (INSA-Rouen), Lecturer
  - **Education:**
    - 2006 - 2010: INSA-Rouen, PhD in Mathematics - Optimization
    - 2005 - 2006: INSA-Rouen, Ms in Fundamental and Applied Mathematics
    - 2001 - 2006: INSA-Rouen, Bs and Ms in Genie Mathematics
- **Research:** optimization theory and algorithms, high-performance computing and optimization software development. **Applications:** machine learning, finance, image processing, turbulent combustion, quantum computing, and plasma physics, etc.

Brief Course Description

- **Lecture Time:** 2024-04-16 to 2024-06-11, Tue and Thu, 19:10-21:35 (Beijing Time)
- **Teaching Hours:** 48TH, twice a week, 3TH per lecture
- **ZOOM:** 435 529 7909 (password: BIMSA)
- **Venue:** BIMSA A3-2-303
- **Audience:** Advanced Undergraduate, Graduate, Postdoc, Researcher
- **Wechat Groups:**

群聊: 24SP Opt Methods for
ML - 2群

该二维码7天内 (4月17日前) 有效，重新进入将更新

- **Website:** https://bimsa.net:10000/activity/OptMetforMacLea/
- **Email:** niuyishuai@bimsa.cn

Brief Course Description

- **Course Title: Optimization Methods for Machine Learning**
- **Abstract:** **Stochastic Gradient Descent (SGD)**, in one form or another, serves as the workhorse method for **training modern machine learning models**. Amidst its myriad variations, the SGD domain is both extensive and burgeoning, presenting a significant challenge for both practitioners and even experts to understand its landscape and inhabitants. This course offers a mathematically rigorous and comprehensive introduction to the field, drawing upon the most recent advancements and insights. It meticulously constructs a theory of **convergence and complexity for SGD's serial, parallel, and distributed variants across strongly convex, convex, and nonconvex settings**, incorporating advanced techniques such as **sampling, mini-batching, acceleration, variance reduction, compression, quantization, sketching, dithering, sparsification**, as well as their combinations. This comprehensive exploration aims to equip learners with a deep understanding of SGD's intricate landscape, fostering the ability to adeptly apply and innovate upon these methods in their work.

Goals and Objectives

- Detailed understanding of the **key variants of SGD** for training **supervised machine learning models** and their differences
- Understanding the underlying **mathematical theory** and of the insights the theory offers for practice
- Ability to **apply the methodologies** to selected applications in machine learning
- Preparation for **original theoretical and applied research** in the field of randomized methods for optimization and machine learning

Knowledge Required

- **Linear algebra:** Abstract vector spaces, linear independence, basis, linear operators, quadratic forms, Euclidean spaces, inner product, norm, matrices, determinants, singular values, matrix decompositions
- **Multivariate calculus:** gradient, Hessian, Taylor approximation, chain rule
- **Probability theory:** probability spaces, expectation, law of large numbers, tower property of expectation
- **Convex analysis:** convex sets, convex functions, strong convexity, conjugation, Jensen's inequality, subdifferential, optimality conditions

Reference Text

- **Books:** There is no book that covers exactly the material contained in this course. But there are some related books worth reading:
  - Lectures on Convex Optimization – Y. Nesterov
  - Learning Theory from First Principles – F. Bach
  - First-Order Methods in Optimization – A. Beck
  - Large-Scale Convex Optimization: Algorithms and Analyses via Monotone Operators – E.K. Ryu and W.T. Yin
  - First-order and Stochastic Optimization Methods for Machine Learning – G.H. Lan
  - Accelerated Optimization for Machine Learning: First-Order Algorithms – Z.C. Lin, H. Li, C. Fang
- **Slides:**
  - Slides will include all relevant material (e.g., explanations, theorems, algorithms, and proofs).
  - Slides will be made available in PDF and uploaded to the course's webpage after each lecture.
- **Papers:** Relevant research papers will be referred to and recommended in lectures.

Part 2: Introduction to Supervised Machine Learning

北京雁栖湖
应用数学研究院
Beijing Institute
of Mathematical
Sciences and Applications

Notation

- $\mathbb{R}^d$: Euclidean space of d-dimensional real vectors $x = (x_1, \ldots, x_d)$
- $\langle x, y \rangle \stackrel{\text{def}}{=} \sum_i x_i y_i$ is the **standard Euclidean inner product**
- $\|x\| \stackrel{\text{def}}{=} \sqrt{\langle x, x \rangle}$ is the **standard Euclidean norm**
- $[k] \stackrel{\text{def}}{=} \{1, 2, \ldots, k\}$ for any positive integer k

Supervised Machine Learning

The Goal of Supervised Machine Learning

We wish to **learn** an approximation $h : \mathcal{A} \to \mathcal{B}$ of a function $h^* : \mathcal{A} \to \mathcal{B}$ mapping inputs from a **domain space** $\mathcal{A}$ to outputs from a **label space** $\mathcal{B}$. The function h is called a **predictor**, a **classifier**, or a **hypothesis**.

In other words, we wish to **learn a predictor** that will be able to predict the label $h^*(a)$ of an unseen/new input $a \in \mathcal{A}$.

Set of "natural" objects $\mathcal{A}$	**Set of labels $\mathcal{B}$**	**Prediction task**
Images	Image category (finite set)	Multi-class classification
Articles	Article category (finite set)	Multi-class classification
E-mails	Spam/not-spam $\{-1, 1\}$	Binary classification
Surveillance videos	Probability of a threat [0, 1]	Regression
Sequences of texts/videos	Next texts/videos	Sequence generation (ChatGPT/Sora)

Figure: Examples of prediction tasks.

Where Does the Data Come From?

We have access to **unbiased samples** of input-output pairs

$$(a, b) = (a, h^*(a)) \in \mathcal{A} \times \mathcal{B}$$

following some distribution $\mathcal{D}$.

- We do not assume (even in theory) to know $\mathcal{D}$. Instead, we can learn about $\mathcal{D}$ by repeatedly sampling input-output pairs. All theory and methods should be distribution agnostic.
- Typically, we sample n (where n is large enough) input-output pairs, and call it the **training dataset**. We then use this dataset to learn a "good" approximation of h^*.

What Function are we Learning?

We choose some **parametric class** of functions/hypotheses/models

$$h_x : \mathcal{A} \to \mathcal{B},$$

where the parameter x is described by d features; i.e., $x \in \mathcal{X} \subseteq \mathbb{R}^d$.

- The choice of the hypothesis class is crucial. If we choose a class which does not contain any function close to h^*, we can't do well.
- Decisions about the hypothesis class are often "art"/"black magic" in practice. **Prior knowledge** about h^* is encoded in the selection of the hypothesis class.

Example 1 (Hypothesis classes)

- **linear model:** $h_x(a) = x^T a$
- **linear model with feature map:** $h_x(a) = x^T \phi(a)$, where $\phi : \mathcal{A} \to \mathbb{R}^d$ is a **feature map**
- **neural network:** $h_x(a) = x_l^T \phi_{x_1,\dots,x_{l-1}}(a)$, where $x = (x_1, \dots, x_{l-1}, x_l)$ and $\phi_{x_1,\dots,x_{l-1}} : \mathcal{A} \to \mathbb{R}^{dim(x_l)}$ is a **learnable feature map** of the structure $\phi_{x_1,\dots,x_{l-1}}(a) = \sigma(x_{l-1}^T \dots \sigma(x_2^T \sigma(x_1^T a)))$, where σ is a **nonlinear activation function**

How do we Know a Prediction is Good?

We choose a **loss function**

$$\ell : \mathcal{B} \times \mathcal{B} \to \mathbb{R},$$

where $\ell(b', b)$ measures the loss of predicting b' when in fact the real output is b.

Example 2 (Loss functions)

Assume $\mathcal{B} \subseteq \mathbb{R}$.

- **square loss:** $\ell(b', b) = \frac{1}{2}(b' - b)^2$, $b \in \mathbb{R}$ (used for regression)
- **absolute loss:** $\ell(b', b) = |b' - b|$, $b \in \mathbb{R}$ (used for regression)
- **hinge loss:** $\ell(b', b) = \max\{0, 1 - b'b\}$, $b \in \{-1, 1\}$ (used for classification)
- **logistic loss:** $\ell(b', b) = \log(1 + e^{-b'b})$, $b \in \{-1, 1\}$ (used for classification)

What Parameters are Good?

We wish to find $x \in \mathbb{R}^d$ which solves the following **stochastic optimization problem**:

$$\min_{x \in \mathbb{R}^d} \mathbb{E}_{(a,b)\sim\mathcal{D}} \left[f_{a,b}(x)\right] = \mathbb{E}_{(a,b)\sim\mathcal{D}} \left[\ell(h_x(a), b)\right]. \quad (1)$$

- The function $\mathbb{E}_{(a,b)\sim\mathcal{D}}[f_{a,b}(x)]$ is also called population risk or true risk.
- Problem (1) is called the (population) risk minimization problem, and the value $\mathbb{E}_{(a,b)\sim\mathcal{D}}[\ell(h_x(a), b)]$ is the generalization loss (i.e., expected prediction loss) of model h_x.
- In problem (1), **we seek to find the parameters $x \in \mathbb{R}^d$ describing a model h_x which minimizes the population risk.**
- A key problem with (1) is that since we do not have access to $\mathcal{D}$, we may not be able to solve (1).

How to Solve Problem (1)?

- **Collect training data**. First collect a (finite) training dataset consisting of n input-output pairs sampled from $\mathcal{D}$:

$$S_n \stackrel{\text{def}}{=} \{(a_1, b_1), (a_2, b_2), \ldots, (a_n, b_n)\}.$$

- **Work with empirical risk instead of the true risk**. If n is "large enough", the empirical risk, defined below, is a good approximation of the true risk

$$f(x) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} f_{a_i,b_i}(x) \approx \mathbb{E}_{(a,b)\sim\mathcal{D}}[f_{a,b}(x)].$$

Note that f is a random function of the training data S_n, and that f is an unbiased estimator of the true risk. As n increases, its variance decreases.

Empirical Risk Minimization (ERM)

- Solving the **empirical risk minimization (ERM)** problem:

$$\min_{x \in \mathbb{R}^d} f(x) \tag{2}$$

where

$$f(x) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_{a_i,b_i}(x)$$

with $f_{a_i,b_i}(x) \stackrel{\text{def}}{=} \ell(h_x(a_i), b_i) = f_i(x)$.

  - Any solution x^{ERM} of the ERM problem (2) is depending on the training dataset S_n.
  - The value $f(x^{ERM})$ is the **training loss** associated with the ERM solution x^{ERM}.
  - The value $\mathbb{E}_{(a,b)\sim\mathcal{D}}[f_{a,b}(x^{ERM})]$ is the **generalization loss** associated with the ERM solution x^{ERM}.

Regularized ERM

Often in practice, we add a **regularizer** $R(x)$ to the empirical risk and solve the **regularized ERM** problem:

$$\min_{x \in \mathbb{R}^d} \left[\frac{1}{n} \sum_{i=1}^{n} f_i(x) + R(x) \right]. \tag{3}$$

- There are **learning theoretic reasons to add a regularizer**. We will not talk about them. Refer to the book "Understanding Machine Learning: From Theory to Algorithms" by Shai Shalev-Shwartz and Shai Ben-David for explanation.
- **Regularizers can be used to encode "prior knowledge"** about the model. For instance, the L_1 regularizer,

$$R(x) = \|x\|_1 = \sum_{i=1}^{d} |x_i|,$$

encourages sparsity in x.

Examples of Regularized ERM Problems

Regularized ERM Problem	Loss Function $\ell(b', b)$	Regularizer $R(x)$
Least Squares (Linear Regression)	$\frac{1}{2}(b' - b)^2$	0
L1 Regression	$\lvert b' - b \rvert$	0
Ridge Regression (L2 regularized Least Squares)	$\frac{1}{2}(b' - b)^2$	$\frac{\lambda}{2}\lVert x \rVert^2$
LASSO (L1 regularized Least Squares)	$\frac{1}{2}(b' - b)^2$	$\lambda \lVert x \rVert_1$
Nonnegative Least Squares	$\frac{1}{2}(b' - b)^2$	$R(x) = \begin{cases} 0 & x \geq 0 \\ +\infty & \text{otherwise} \end{cases}$
Support Vector Machine (SVM)	$\max\{0, 1 - b'b\}$	$\frac{\lambda}{2}\lVert x \rVert^2$
Logistic Regression	$\log\left(1 + e^{-b'b}\right)$	$\frac{\lambda}{2}\lVert x \rVert^2$
Best Approximation	$\ell(b', b) = \begin{cases} 0 & b' = b \\ +\infty & b' \neq b \end{cases}$	$\lVert x - x^0 \rVert$

Optimization Problems Arising in Supervised ML

Optimization Problems Arising in Machine Learning

In this course, we are interested in the optimization problem

$$\min_{x\in\mathbb{R}^d} f(x) + R(x) \tag{4}$$

- **Infinite sum:**

$$f(x) = \mathbb{E}_{\xi\sim\mathcal{D}}[f_\xi(x)], \tag{5}$$

- **Finite sum:**

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{6}$$

  - **Finite Sum of Finite Sums:**

$$f_i(x) = \frac{1}{m_i}\sum_{j=1}^{m_i} f_{ij}(x) \tag{7}$$

  - **Finite Sum of Infinite Sums:**

$$f_i(x) = \mathbb{E}_{\xi_i\sim\mathcal{D}_i}[f_{i\xi_i}(x)] \tag{8}$$

These problems are of key importance in **supervised learning theory and practice.**
Common feature: It is prohibitively expensive to compute the gradient of f, while an unbiased estimator of the gradient can be computed efficiently/cheaply.

Distributed & Federated Training

In distributed training of supervised models, one considers the finite sum problem (6), with n being the number of machines/devices, and each f_i

- also having a **finite sum structure**, i.e.,

$$f_i(x) = \frac{1}{m_i} \sum_{j=1}^{m_i} f_{ij}(x), \tag{9}$$

  where m_i **corresponds to the number of training examples stored on machine** i.

- or an **infinite-sum structure**, i.e.,

$$f_i(x) = \mathbb{E}_{\xi_i \sim \mathcal{D}_i}[f_{i\xi_i}(x)], \tag{10}$$

  where $\mathcal{D}_i$ **is the distribution of data stored on machine** i**.**

Part 3: Basic Tools from Convex Analysis, Optimization and Probability

Differentiable Functions

Fundamental Theorem of Calculus

We will use the following theorem repeatedly in various disguises.

Theorem 1

Let $\phi : \mathbb{R} \to \mathbb{R}$ be continuously differentiable on an open interval containing points a and b. Then

$$\phi(b) - \phi(a) = \int_a^b \phi'(t)\, dt. \tag{11}$$

Corollary 2

Let $f : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable. Then for any $x, y \in \mathbb{R}^d$ we have

$$f(y) - f(x) = \int_0^1 \langle \nabla f(x + t(y - x)), y - x \rangle\, dt. \tag{12}$$

Proof.

Let $\phi(t) = f(x + t(y - x))$. Note that $\phi(0) = f(x)$, $\phi(1) = f(y)$ and $\phi'(t) = \langle \nabla f(x + t(y - x)), y - x \rangle$. The rest follows by applying Theorem 1. $\square$

Fundamental Theorem of Calculus

Corollary 3

Let $f : \mathbb{R}^d \to \mathbb{R}$ be twice continuously differentiable. Then for any $x, y, s \in \mathbb{R}^d$ we have

$$\langle \nabla f(y) - \nabla f(x), s \rangle = \int_0^1 \left\langle \nabla^2 f(x + t(y - x))(y - x), s \right\rangle dt. \tag{13}$$

Proof.

Let $\phi(t) = \langle \nabla f(x + t(y - x)), s \rangle$. Note that $\phi(0) = \langle \nabla f(x), s \rangle$, $\phi(1) = \langle \nabla f(y), s \rangle$ and $\phi'(t) = \langle \nabla^2 f(x + t(y - x))(y - x), s \rangle$. The rest follows by applying Theorem 1. $\square$

Bregman Divergence

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function.

Definition 1 (Bregman Divergence)

Bregman divergence of f is the mapping $D_f : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ defined by:

$$D_f(x, y) \stackrel{\text{def}}{=} f(x) - f(y) - \langle \nabla f(y), x - y \rangle$$

- Note that D_f is not necessarily symmetric, i.e., in general $D_f(x, y) \neq D_f(y, x)$.
- This can be fixed by defining a symmetric version of Bregman divergence.

Definition 2 (Symmetrized Bregman Divergence)

Symmetrized Bregman divergence of f is defined by:

$$D_f(x, y) + D_f(y, x) = \langle \nabla f(x) - \nabla f(y), x - y \rangle$$

Convex Functions

Convex Functions: Definition

Definition 3 (Convex function)

A function $f : \mathbb{R}^d \to \mathbb{R}$ is **convex** if for all $x, y \in \mathbb{R}^d$ and $\lambda \in [0, 1]$:

$$0 \leq \lambda f(x) + (1-\lambda) f(y) - f(\lambda x + (1-\lambda) y) \stackrel{\text{def}}{=} C_f^{\lambda(x,y)}$$

Some facts:

- If f_1, f_2 are convex and $\alpha_1, \alpha_2 \geq 0$, then $\alpha_1 f_1 + \alpha_2 f_2$ is convex.
- If $f : \mathbb{R}^n \to \mathbb{R}$ is convex and $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$, then the function $x \mapsto f(Ax + b)$ is convex.
- If f_1, f_2 are convex, then so is $\max\{f_1, f_2\}$.

Convex Functions: Characterization

Theorem 4 (Convexity and gradient)

Let $f : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable. Then the following statements are equivalent:

1. *f is convex (**convexity of the function**)*
2. *$0 \leq D_f(x, y)$ for all $x, y \in \mathbb{R}^d$ (**nonnegativity of the Bregman divergence**)*
3. *$0 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle$ for all $x, y \in \mathbb{R}^d$ (**monotonicity of the gradient**)*

Theorem 5 (Convexity and Hessian)

Let $f : \mathbb{R}^d \to \mathbb{R}$ be **twice** *continuously differentiable. Then f is convex if and only if*

$$\nabla^2 f(x) \succeq 0 \quad \text{for all} \quad x \in \mathbb{R}^d.$$

*(**positive semi-definiteness of the Hessian**)*

Convex Functions: Examples

Example 3

- Linear function $f(x) = \langle b, x\rangle + c$ is convex.
- Quadratic function $f(x) = \frac{1}{2}\langle Ax, x\rangle + \langle b, x\rangle + c$ is convex if and only if $A \succeq 0$.
- Exponential function $f(t) = e^t$ is convex.
- The Fenchel conjugate $f^*(x)$ defined by

$$f^*(x) \stackrel{\text{def}}{=} \sup_{y\in\mathbb{R}^d} \langle x, y\rangle - f(y)$$

of any function $f : \mathbb{R}^d \to \mathbb{R}$ is convex.

Proof of Theorem 4 (Convexity and gradient)

- (i) $\Rightarrow$ (ii) Choose any $x, y \in \mathbb{R}^d$ and $\lambda \in [0,1]$. Define $z = \lambda x + (1-\lambda)y$. Then:

$$\begin{aligned} f(y) &\geq \frac{1}{1-\lambda}(f(z) - \lambda f(x)) \\ &= f(x) + \frac{1}{1-\lambda}(f(z) - f(x)) \\ &= f(x) + \frac{1}{1-\lambda}(f(\lambda x + (1-\lambda)(y-x)) - f(x)). \end{aligned}$$

  By taking limit $\lambda \to 1$, we get:

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

  and hence (ii) holds.

- (ii) $\Rightarrow$ (i) Choose any $x, y \in \mathbb{R}^d$ and $\lambda \in [0,1]$ and define $z = \lambda x + (1-\lambda)y$. Since $D_f(y,z) \geq 0$ and $D_f(x,z) \geq 0$, we have:

$$f(z) \leq f(y) + \langle \nabla f(y), z - y \rangle = f(y) + \lambda \langle \nabla f(z), x - y \rangle$$

  and

$$f(z) \leq f(x) + \langle \nabla f(z), z - x \rangle = f(x) - (1-\lambda) \langle \nabla f(z), x - y \rangle$$

  Multiplying the first inequality by $1-\lambda$, the second by λ, and adding the resulting inequalities, we get:

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

Proof of Theorem 4 (Convexity and gradient)

- (ii) $\Rightarrow$ (iii) Choose any $x, y \in \mathbb{R}^d$. Adding the inequalities $D_f(x, y) \geq 0$ and $D_f(y, x) \geq 0$, we get (iii).
- (iii) $\Rightarrow$ (ii) Choose any $x, y \in \mathbb{R}^d$ and define $z = x + t(y - x)$. Then by the fundamental theorem of calculus, we have:

$$\begin{aligned} f(y) &= f(x) + \int_0^1 \langle \nabla f(x + t(y - x)), y - x \rangle \, dt \\ &= f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \langle \nabla f(z) - \nabla f(x), y - x \rangle \, dt \\ &= f(x) + \langle \nabla f(x), y - x \rangle + \int_0^1 \frac{1}{t} \langle \nabla f(z) - \nabla f(x), z - x \rangle \, dt \\ &\geq f(x) + \langle \nabla f(x), y - x \rangle, \end{aligned}$$

where the inequality follows from the bound $\langle \nabla f(z) - \nabla f(x), z - x \rangle \geq 0$, which holds due to (iii).

Proof of Theorem 5 (Convexity and Hessian)

- Assume f is convex. Choose any $x, y' \in \mathbb{R}^d$ and $\theta > 0$. Then in view of Theorem 4(iii), we have

$$0 \le \frac{1}{\theta^2} \langle \nabla f(x + \theta(y' - x)) - \nabla f(x), (x + \theta(y' - x)) - x \rangle$$
$$= \underbrace{\frac{1}{\theta} \langle \nabla f(x + \theta(y' - x)) - \nabla f(x), y' - x \rangle,}_{\stackrel{\text{def}}{=} A_\theta}$$

and hence $A_\theta \ge 0$ for all $\theta > 0$. Applying the fundamental theorem of calculus (Corollary 3) with $x \leftarrow x, y \leftarrow x + \theta(y' - x)$ and $s \leftarrow y' - x$ gives

$$A_\theta = \frac{1}{\theta} \int_0^1 \langle \nabla^2 f(x + t(y - x))(y - x), y' - x \rangle \, dt$$
$$= \int_0^1 \langle \nabla^2 f(x + \theta t(y' - x))(y' - x), y' - x \rangle \, dt.$$

Proof of Theorem 5 (Convexity and Hessian)

- Finally, taking limit $\theta \to 0^+$, we get

$$
\begin{aligned}
0 &\leq \lim_{\theta \to 0^+} A_\theta \\
&= \lim_{\theta \to 0^+} \int_0^1 \left\langle \nabla^2 f(x + \theta t(y' - x))(y' - x), y' - x \right\rangle \, dt \\
&= \int_0^1 \lim_{\theta \to 0^+} \left\langle \nabla^2 f(x + \theta t(y' - x))(y' - x), y' - x \right\rangle \, dt \\
&= \int_0^1 \left\langle \nabla^2 f(x)(y' - x), y' - x \right\rangle \, dt \\
&= \left\langle \nabla^2 f(x)(y' - x), y' - x \right\rangle .
\end{aligned}
$$

  Since y' was arbitrary, this means that $\nabla^2 f(x)$ is positive semi-definite.
- Proof of the reverse implication is left as an exercise.

Jensen's Inequality

Theorem 6 (Jensen's Inequality)

If f is convex and $X \in \mathbb{R}^d$ is a random vector, then

$$0 \leq \mathbb{E}[f(X)] - f(\mathbb{E}[X]).$$

Strongly Convex Functions

Strongly Convex Functions: Definition

Definition 4 (Strongly convex function)

Let $\mu \geq 0$. A function $f : \mathbb{R}^d \to \mathbb{R}$ is **μ-strongly convex** (**μ-convex** for short) if the function

$$x \mapsto f(x) - \frac{\mu}{2}\|x\|^2$$

is convex. That is, if

$$\frac{\mu}{2}\lambda(1-\lambda)\|x-y\|^2 \leq C_f^\lambda(x,y)$$

for all $x, y \in \mathbb{R}^d$ and $\lambda \in [0,1]$.

- Note that the $\mu = 0$ case reduces to convexity.

Example 4

Let $\mu \geq 0$. $f(x) = \frac{\mu}{2}\|x\|^2$ is μ-convex since $f(x) - \frac{\mu}{2}\|x\|^2 = 0$ is convex.

Strongly Convex Functions: Characterization

Theorem 7 (Strong convexity and gradient)

Let $f : \mathbb{R}^d \to \mathbb{R}$ be continuously differentiable. Then the following statements are equivalent:

1. f *is* μ*-convex*
2. $\mu\|x - y\|^2 \leq 2D_f(x, y)$ *for all* $x, y \in \mathbb{R}^d$
3. $\mu\|x - y\|^2 \leq \langle \nabla f(x) - \nabla f(y), x - y \rangle$ *for all* $x, y \in \mathbb{R}^d$

Theorem 8 (Strong convexity and gradient II)

If $f : \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable and μ-convex, then[a]

- 2' $2D_f(x, y) \leq \frac{1}{\mu}\|\nabla f(x) - \nabla f(y)\|^2$ *for all* $x, y \in \mathbb{R}^d$
- 3' $\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq \frac{1}{\mu}\|\nabla f(x) - \nabla f(y)\|^2$ *for all* $x, y \in \mathbb{R}^d$

[a] Note: If $\mu = 0$, we interpret $\frac{1}{\mu}$ as $+\infty$, and the bounds hold trivially.

Theorem 9 (Strong convexity and Hessian)

If $f : \mathbb{R}^d \to \mathbb{R}$ is twice continuously differentiable, then it is μ-convex if and only if

$$\mu I \preceq \nabla^2 f(x) \quad \textit{for all} \quad x \in \mathbb{R}^d.$$

Strong Jensen's Inequality

Theorem 10 (Strong Jensen's Inequality)

If f is μ-convex and $X \in \mathbb{R}^d$ is a random vector, then

$$\frac{\mu}{2}\mathrm{Var}[X] \leq \mathbb{E}[f(X)] - f(\mathbb{E}[X]).$$

Proof.

Let us apply Jensen's inequality $F(\mathbb{E}[X]) \leq \mathbb{E}[F(X)]$ to $F(x) \overset{\text{def}}{=} f(x) - \frac{\mu}{2}\|x\|^2$, which is convex by definition. We get

$$f(\mathbb{E}[X]) - \frac{\mu}{2}\mathbb{E}[\|X\|^2] \leq \mathbb{E}[f(X)] - \mathbb{E}\left[\frac{\mu}{2}\|X\|^2\right].$$

It remains to rearrange the inequality and use the identity

$$\mathrm{Var}[X] = \mathbb{E}[\|X\|^2] - \|\mathbb{E}[X]\|^2.$$

□