Introduction to Machine Learning (67577)

Recitation 3
Convex Optimization

Second Semester, 2021

Contents

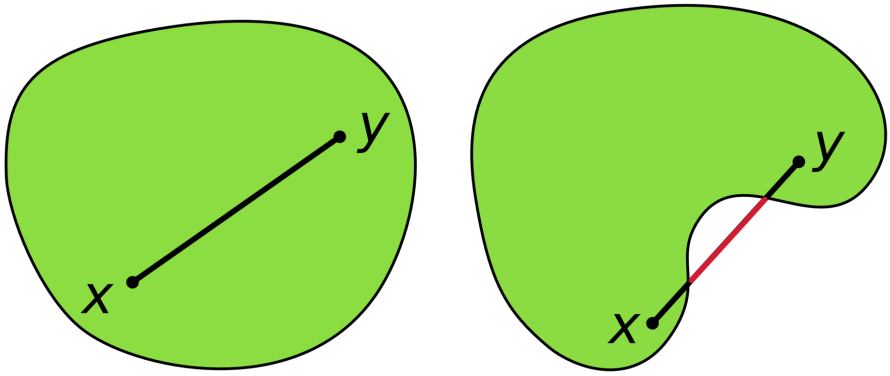
1 Motivation For Convex Analysis

Many important learning tasks can be formulated as 'convex learning problems'. For such problems, the rich and the general theory of convex optimization will help us to derive general and efficient learning algorithms. The aim of this Tirgul is to review and discuss additional fundamental concepts both in convex analysis and convex optimization.
The theory of convex analysis unifies and relates important ideas from calculus, topology and geometry to the field of mathematical optimization. In this Tirgul we give a brief introduction to this subject and state some of the important results.

2 Convex Sets

Definition 2.1 Let V be a vector space. A set $C \subseteq V$ is called *convex* if for every two vectors $u, v \in C$, and every scalar $\alpha \in [0,1]$, $\alpha u + (1-\alpha)v \in C$.

Geometrically, a set C is convex iff the line segment joining any two points u and v in C is contained in C (see figure).



■ **Example 2.1**

1. **Linear Subspaces:** Every linear subspace $U \subseteq V$ is convex. Indeed, for every $u, v \in U$, and $\alpha \in [0,1]$, $\alpha u + (1-\alpha)v$ is a linear combination of vectors in U, and thus belongs to U.
2. **Unit balls:** The unit ball $B = \{v \in V : \|v\| \leq 1\}$ is convex. Indeed, if $u, v \in B$, and $\alpha \in [0,1]$, the triangle inequality implies that

$$\begin{aligned} \|\alpha u + (1-\alpha v)\| &\leq \|\alpha u\| + \|(1-\alpha)v\| \\ &= \alpha\|u\| + (1-\alpha)\|v\| \\ &\leq \alpha + 1 - \alpha \\ &= 1 \end{aligned}$$

3. **Closed Halfspaces:** A closed halfspace is a set of the form $W = \{v : \langle w, v\rangle \leq b\}$, where w is nonzero vector in V, and $b \in \mathbb{R}$. Closed halfspaces are convex. Indeed, if $u, v \in W$, and

$\alpha \in [0,1]$, we have

$$\begin{aligned}\langle w, \alpha u + (1-\alpha)v\rangle &= \alpha \langle w,u\rangle + (1-\alpha)\langle w,v\rangle \\ &\leq \alpha b + (1-\alpha)b \\ &= b\end{aligned}$$

■

Exercise 2.1 Let V be the set of symmetric (real) $d \times d$ matrices.

1. Show that V is a vector space (equipped with matrix addition).
2. A matrix $A \in V$ is called *positive-definite (PD)* if for every $v \in \mathbb{R}^d$, $v^\top Av > 0$ (in this case we write: $A \succ 0$). Prove that the set of PD matrices is convex.

Solution:

1. Easy.
2. Let M,N be PD matrices, and let $\alpha \in [0,1]$:

$$x^\intercal(\alpha M + (1-\alpha)N)x = \alpha x^\intercal Mx + (1-\alpha)x^\intercal Nx > 0$$

■

The following theorem lists some operations that preserve convexity of sets.

Theorem 2.1

1. The intersection $C = \bigcap_{i\in I} C_i$ of any collection $\{C_i : i \in I\}$ of convex sets is convex.
2. The set $\lambda C = \{\lambda c : c \in C\}$ is convex, for any convex set C, and every scalar λ.

Proof.

1. Let $u,v \in C$, and let $\alpha \in [0,1]$. Then, for every $i \in I$, $u,v \in C_i$. For each $i \in I$, the convexity of C_i implies that $\alpha u + (1-\alpha)v \in C_i$. Thus, $\alpha u + (1-\alpha)v \in \bigcap_{i\in I} C_i = C$.
2. Let $u,v \in \lambda C$, and let $\alpha \in [0,1]$. Then, $u = \lambda c_1$ and $v = \lambda c_2$ for some $c_1, c_2 \in C$. Then,

$$\alpha u + (1-\alpha)v = \lambda(\alpha c_1 + (1-\alpha)c_2) \in \lambda C.$$

■

Exercise 2.2 A hyperplane is a set of the form $W = \{v \in V : \langle w,v\rangle = b\}$, where $w \in V$, and $b \in \mathbb{R}$. Show that every hyperplane is convex. ■

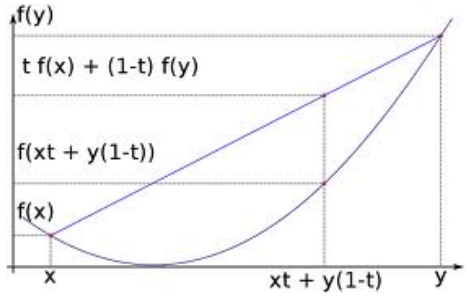
3 Convex Functions

Definition 3.1 Let V be a vector space and let C be a convex set. A function $f : C \to \mathbb{R}$ is called convex if for every $u,v \in C$ and every $0 \leq \lambda \leq 1$,

$$f(\lambda u + (1-\lambda)v) \leq \lambda f(u) + (1-\lambda)f(v)\ .$$

The function f is *strictly convex* if the above inequality holds strictly for every $u \neq v \in C$ and

$0 < \lambda < 1$.



Geometrically, a function over a convex set is convex if the graph of the function lies below the line segment joining any two points of the graph.

■ **Example 3.1**

1. The norm function is convex. By the triangle inequality, we obtain for every $u, v \in V$ and $\alpha \in [0,1]$,

$$\|\alpha u + (1-\alpha)v\| \leq \|\alpha u\| + \|(1-\alpha)v\| = \alpha\|u\| + (1-\alpha)\|v\|$$

2. Every real-valued affine function is convex; let $w \in V$ and let $b \in \mathbb{R}$. Define $f : V \to \mathbb{R}$ by

$$f(u) = \langle w, u \rangle + b.$$

Let $u, v \in V$ and let $\alpha \in [0,1]$. Then,

$$\begin{aligned} f(\alpha u + (1-\alpha)v) &= \langle w, \alpha u + (1-\alpha)v \rangle + b \\ &= \alpha(\langle w, u \rangle + b) + (1-\alpha)(\langle w, v \rangle + b) \\ &= \alpha f(u) + (1-\alpha) f(v) \ . \end{aligned}$$

■

Starting with some known convex functions, we can generate other convex functions by using some common algebraic operations.

Theorem 3.1 The following propositions hold:

1. Let $f_i : V \to \mathbb{R}$ for $i = 1, \ldots, m$ be given functions and let $\gamma_1, \ldots, \gamma_m$ be positive scalars. Consider the function $g : V \to \mathbb{R}$ given by

$$g(u) = \sum_{i=1}^{m} \gamma_i f_i(u).$$

If $f_1, \ldots, f_m$ are convex, then g is also convex.

2. Let $f : \mathbb{R}^m \to \mathbb{R}$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Define $g : \mathbb{R}^n \to \mathbb{R}$ by

$$g(u) = f(Au + b).$$

If f is convex, then g is also convex.

3. Let $f_i : V \to \mathbb{R}$, $i \in I$. Let $f : V \to \mathbb{R}$ given by

$$g(u) = \sup_{i \in I} f_i(u).$$

If f_i are convex for every $i \in I$, then g is also convex.

Proof. Here we will prove the second part.
Let $u, v \in \mathbb{R}^n$ and let $\alpha \in [0,1]$. Then,

$$\begin{aligned} g(\alpha u + (1-\alpha)v) &= f(A(\alpha u + (1-\alpha)v) + b) \\ &= f(\alpha(Au+b) + (1-\alpha)(Av+b) \\ &\leq \alpha f(Au+b) + (1-\alpha)(f(Av+b)) \\ &= \alpha g(u) + (1-\alpha) g(v) \ . \end{aligned}$$

■

■ **Example 3.2** Recall that the squared loss w.r.t. $(x, y) \in \mathbb{R}^d \times \mathbb{R}$, is the function $f : \mathbb{R}^d \to \mathbb{R}$ which is defined by

$$f(w) = (\langle w, x \rangle - y)^2 \ .$$

Since the function $w \mapsto \langle w, x \rangle - y$ is affine, and the scalar function $a \mapsto a^2$ is convex (3.1), we obtain from 3.1 that f is convex. ■

4 Subgradients

The notion of subgradients is fundamental in convex analysis and convex optimization. It allows us to deal with non-differentiable convex functions.

Definition 4.1 Let $f : V \to \mathbb{R}$ be a function. Then, $g \in V$ is a *subgradient* of f at $u \in V$ if for every $v \in V$, one has

$$f(v) \geq f(u) + \langle g, v - u \rangle \ .$$

The set of subgradients at u is denoted $\partial f(u)$.

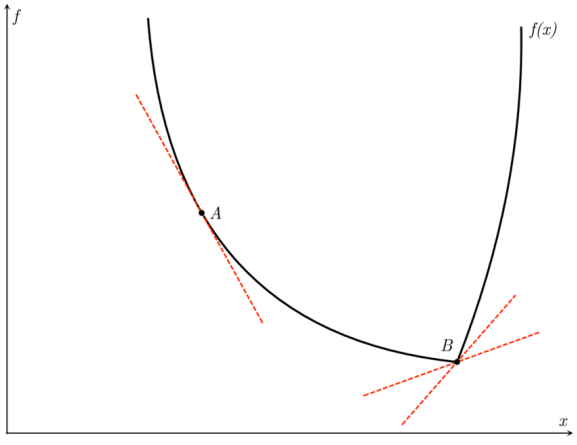


Figure 1: *Subgradient*

Theorem 4.1 If f is convex and differentiable at x, then $\partial f(x) = \{\nabla f(x)\}$.

The next lemma will be useful for computing subgradients.

Proposition 4.2 — member in the subgradient of max. For each $i = 1, \ldots, n$, let $f_i : V \to \mathbb{R}$ be convex functions. Define $f : V \to \mathbb{R}$ by $f(u) = \max_{i \in [n]} f_i(u)$. Given some $u \in V$, let $j \in argmax_i f_i(u)$. Then, $\partial f_j(u) \subseteq \partial f(u)$.

Proof. Let $g \in \partial f_j(u)$. By the choice of j, the definition of f and the definition of the subgradient, we obtain that for every $v \in V$,

$$f(v) \geq f_j(v) \geq f_j(u) + \langle g, v - u \rangle = f(u) + \langle g, v - u \rangle \ .$$

Thus, g is a subgradient of f at u. ■

■ **Example 4.1** Note that $f(x) = |x|$ can be described by $f(x) = \max\{+x, -x\}$. Proposition 4.2 (together with 4.1) can be used to show that $\partial f(x) = \{1\}$ for $x > 0$, $\partial f(x) = \{-1\}$ for $x < 0$ and $\partial f(0) \supseteq \{-1, 1\}$ ■

Corollary 4.3 Let $f : V \to \mathbb{R}$ be a convex function. Assume that for some point $\bar{w} \in V$, $0 \in \partial f(\bar{w})$. Then, $\bar{w}$ is a global minimizer of f.

Proof. For every $w \in V$, we have

$$f(w) \geq f(\bar{w}) + \langle 0, w - \bar{w} \rangle = f(\bar{w}) \ .$$

■

5 High Order Conditions For Convexity

Theorem 5.1 — First order condition. A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if the following inequality holds for all points x, y in its domain:

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x)$$

■ **Example 5.1** Let $c \in \mathbb{R}^n$ and define $f : \mathbb{R}^n \to \mathbb{R}$ by: $f(x) = \langle c, x \rangle$. f is convex, since for all $x \in \mathbb{R}^n$ we have: $\nabla f(x) = c$, so:

$$f(y) = c^\top y = c^\top x + c^\top (y - x) = f(x) + \nabla f(x)^\top (y - x)$$

■

Theorem 5.2 — Second order condition. A twice differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if for any point x in its domain, the Hessian evaluated at x is PSD:

$$\nabla^2 f(x) \succeq 0$$

■ **Example 5.2** Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and define $f(x) = x^\top Ax$. Let us compute the first and second derivations of f.

Denote $g(x) = Ax$ and $h(x) = x$ so we can write f as $f \equiv h^\top g$. Using the product rule then:

$$\frac{\partial f(x)}{\partial x} = \frac{\partial h(x)^\top}{\partial x} \cdot g(x) + \frac{\partial g(x)}{\partial x} \cdot h(x)$$

As we have seen in the previous recitation, deriving g by x equals to A and therefore:

$$\nabla f(x) = A^\top x + Ax = \left(A + A^\top\right) x$$

As A is symmetric then $\nabla f(x) = 2Ax$. Next let us compute the second derivative:

$$\nabla^2 f(x) = \frac{\partial^2 f}{\partial^2 x} = \frac{\partial 2Ax}{\partial x} = 2A$$

Using the theorem above we learn that f is convex iff $A \succeq 0$. That is iff A is a PSD matrix.

■

Note, that in the scalar case, i.e. $f : \mathbb{R} \to \mathbb{R}$, the condition in theorem 5.2 reduces to the condition $f'' \geq 0$.

6 Convex Optimization

Many problems we will encounter throughout the course is part of a wide family of **convex optimization problems**.

Definition 6.1 — Optimization Problem. An optimization problem over $\mathbb{R}^d$ has the general form:

$$\begin{array}{ll} minimize & f_0(\mathbf{x}) \\ subject\,to & f_i(\mathbf{x}) \leq b_i \quad i = 1,\ldots,n \end{array}$$

where $\mathbf{x}$ is the optimization variable, $f_0 : \mathbb{R}^d \to \mathbb{R}$ is the objective function and $f_i : \mathbb{R}^d \to \mathbb{R}$ are the constraint functions. It is implicitly implied that the optimization problem happens over $dom(f_0) \subset \mathbb{R}^d$, the domain of f_0.

Then, naturally, a convex optimization problem is an optimization problem as above in which $f_0, f_1, \ldots, f_n$ are all convex functions. When these functions are all linear, this is a linear programming problem (as you seen in Algorithmic course).

Definition 6.2 — Linear Program. An optimization problem is called a Linear Program (LP) if it can be written in the following form:

$$\begin{array}{ll} \underset{\mathbf{x}\in\mathbb{R}^n}{min} & c^\top \mathbf{x} \\ such\,that & A\mathbf{x} \leq \mathbf{b} \end{array}$$

where $A \in \mathbb{R}^{m\times n}, \mathbf{c} \in \mathbb{R}^n, \mathbf{b} \in \mathbb{R}^m$ are fixed vectors and matrices.

Definition 6.3 — Quadratic Program. An optimization problem is called a Quadratic Program (QP) if it can be written in the following form:

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^n} \quad & \tfrac{1}{2}\mathbf{w}^\top Q\mathbf{w} + \mathbf{a}^\top \mathbf{w} \\ such that \quad & A\mathbf{w} \leq \mathbf{d} \end{aligned}$$

where $Q \in \mathbb{R}^{n\times n}, A \in \mathbb{R}^{m\times n}, \mathbf{a} \in \mathbb{R}^n, \mathbf{d} \in \mathbb{R}^m$ are fixed vectors and matrices.

In general, optimization problems are hard to solve computationally. We take special interest in **convex optimization** problems since they have a unique solution, and that solution can be found in computationally tractable ways. A great deal is known about **convex optimization algorithms**, which are iterative numerical algorithms that converge to the solution of a convex optimization problem. There are general solvers, which will solve a convex problem in the general form above, and there are specialized solvers for specific types, or families, of convex optimization problems. A specialized solver is typically preferred, as it leverages some particular structure of the problem to solve it more efficiently, using less space, etc. One example for specialized solvers you've seen in Algorithms course are specialized solvers for linear programs.

Why is convex optimization interesting for machine learning? In supervised learning, we would like to choose a hypothesis $h \in \mathcal{H}$ from our selected hypothesis class, based on some learning principle (such as ERM). Many learning principles are formulated as optimization problems, namely, the h our learning algorithm chooses is given as the minimizer of some quantity (such as empirical risk). So implementation of the learning algorithm needs to solve an optimization problem.

Sometimes, our hypothesis class is equivalent to a Euclidean space. When this happens, our learning principle reduces to solving an optimization problem, namely, the hypothesis we choose $h \in \mathcal{H}$ is found as a minimum over $\mathbb{R}^d$ or a subset of $\mathbb{R}^d$ of some objective function, usually a loss function. When this objective is convex, we can use convex optimization algorithms to implement our learning algorithm efficiently.

■ **Example 6.1 — Linear least squares:.** Recall the ls problem, where we are given some samples $\{x_k\}_{k=1}^n \subseteq \mathbb{R}^d$ and labels $\{y_k\}_{k=1}^n \subseteq \mathbb{R}$, we assume $\langle w, x_i\rangle + b \simeq y_i$ for some $w \in \mathbb{R}^d, b \in \mathbb{R}$ and we try to minimize the MSE loss function. Consider the following example where we are given the training samples: $x = (3,6,9)$, and $y = (4,6,8)$. Find the parameters of the model by using gradient descent.

solution:

- First of all we would like to find the gradient of the function:

$$MSE(w,b) = \frac{1}{N}\sum_{i=1}^{N}(y_i - (wx_i + b))^2$$

$$\frac{\partial MSE}{\partial w} = \frac{2}{N}\sum_{i=1}^{N} -x_i(y_i - (wx_i + b)) = \frac{2}{N}\sum_{i=1}^{N} -x_i y_i + wx_i^2 + x_i b = -80 + 84w + 12b$$

$$\frac{\partial MSE}{\partial b} = \frac{2}{N}\sum_{i=1}^{N} -(y_i - (wx_i + b)) = \frac{2}{N}\sum_{i=1}^{N} -y_i + wx_i + b = 2b - 12 + 12w$$

- Now that we have the update rule we will use it in order to iteratively update the parameters, to make it easier we will use **colab notebook** (use your HUJI account in order to open the link).

■

As we can see, the convergence of the method depends on the learning rate we define before the process. Now you may ask what conditions give us guarantees for convergence? Actually this question is one of the most important questions in learning. Some of the conditions depend on the properties of the function f. As we saw in the demo, this method can guarantee that in the next step we will get lower loss value (for small enough step size). It does not guarantee convergence to the optimal parameters since we might converge to a local minimum. In the case of a convex function we know that there is only one global minimum, and it helps us to find the optimal parameters efficiently in many cases.

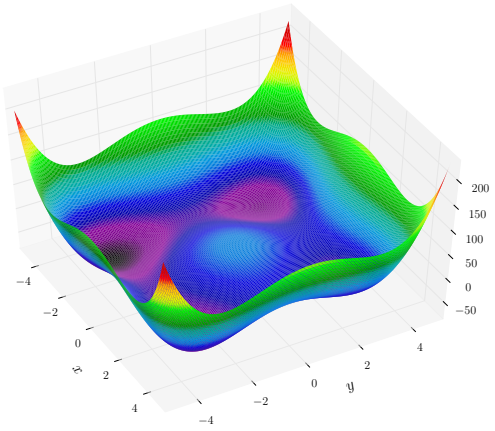


Figure 2: *Non Convex function- may have local minima*

Formally, an optimization problem is convex optimization problem if the problem is of the form:

$$\min_{x \in C} f(x)$$

Where f is a convex function and C is a convex set. In the rest of the tirgul we will learn about convex sets and functions.

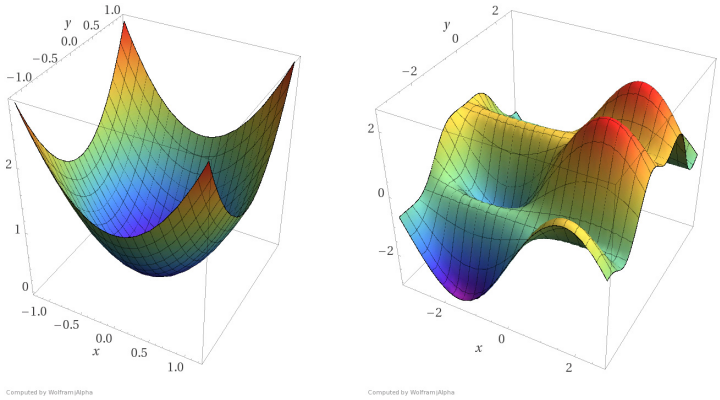


Figure 3: *Left: a convex function. Right: a non-convex function. It is much easier to find the bottom of the surface in the convex function than the non-convex surface.*