# Lecture 1. Introductory examples

## 1.1 Introduction

What is optimization? Roughly speaking, optimization is to minimize or maximize a function (which is called the *objective function*) under some constraints.

For example, we some several ways to return the campus from Hongqiao Station: by taxi (Didi / Gaode), by metro or by bus (Hongqiao 4 Line / Min-Hong 2 Line), etc. We would like to minimize the time, but our money is limited. This is an optimization problem.

Formally, an optimization problem can be defined by

$$\begin{array}{rl} \text{min / max} & f(x) \\ \text{subject to} & x \in \Omega \end{array}$$

where $f$ is called the *objective function* and $\Omega$ is called the *feasible set*, usually specified by *constraint functions*

$$\Omega = \{x \mid g_1(x) \leq 0, g_2(x) \leq 0, \ldots, g_m(x) \leq 0\} .$$

The *optimal solution* is usually denoted by

$$x^* = \arg\min_{x \in \Omega} f(x) .$$

In this course, we consider *continuous* optimization problem, where the *objective function* and the *constraints* are continuous functions. We now give some more examples.

## 1.2 Knapsack problem

**Example**

Suppose there are $n$ types of items. The $i$-th type has volume $a_i$, weight $b_i$ and value $c_i$. We have a knapsack to bring some items. However the capacity of this knapsack is $A$ and the load-bearing is $B$. That is, the total volume of the items in the knapsack can not exceed $A$ and the total weight of the items in the knapsack can not exceed $B$. What is the maximum value we can bring?

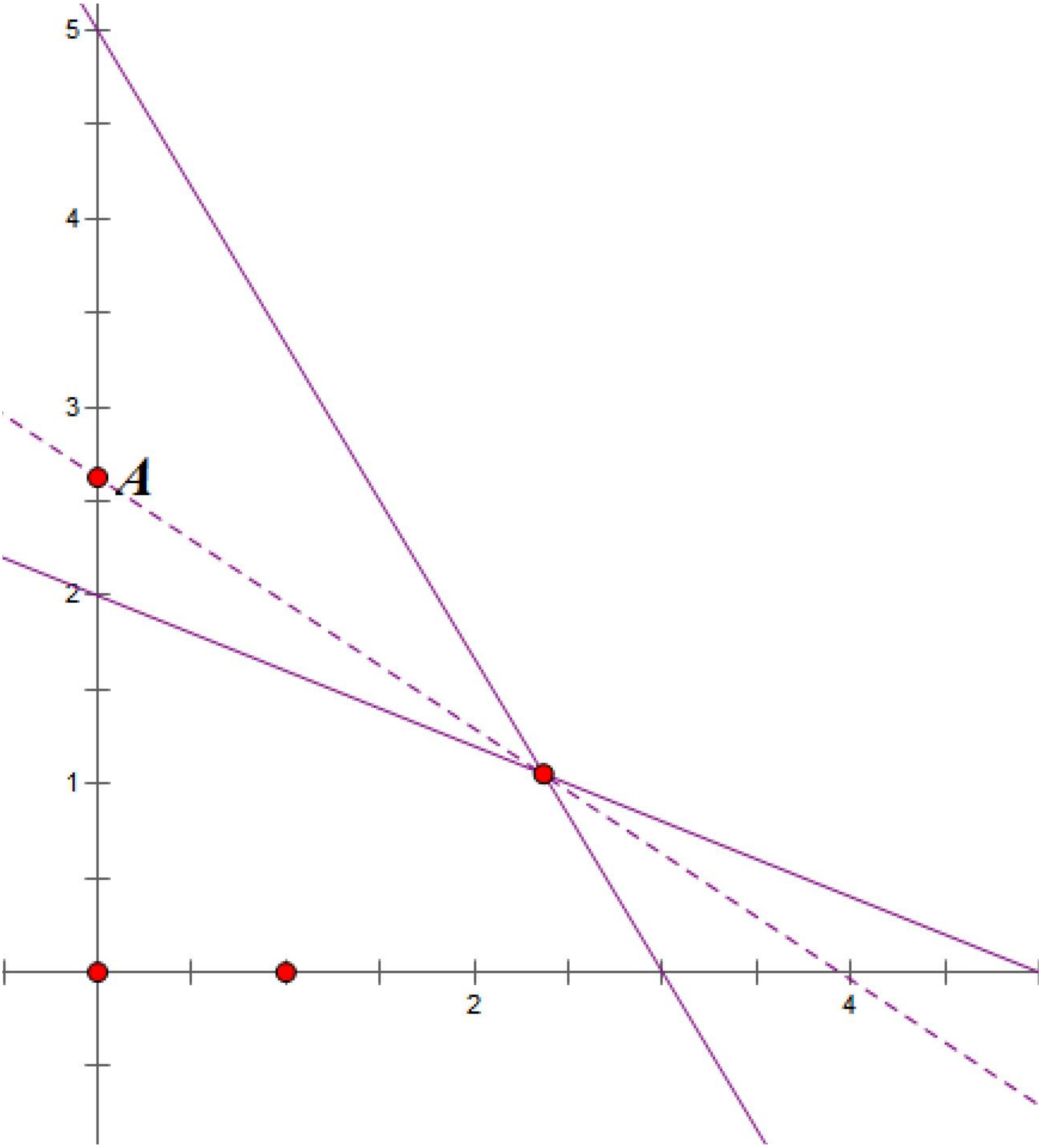
For each $i$, define a variable $x_i$ to denote the number of carried items of the $i$-th type. Then we can formalize the problem as

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} c_i \cdot x_i \\ \text{subject to} \quad & \sum_{i=1}^{n} a_i \cdot x_i \leq A, \\ & \sum_{i=1}^{n} b_i \cdot x_i \leq B, \\ & x_i \geq 0, \ \forall i \in [n]. \end{aligned}$$

How can we solve this problem? For simplicity, we assume that there are only two types: Cola and potato chips. Each Cola has volume 2, weight 5 and value 10; each potato chips has volume 5, weight 3 and value 15. Our knapsack has capacity 10 and load-bearing 15. Now the problem is

$$\begin{aligned} \text{maximize} \quad & 10x_1 + 15x_2 \\ \text{subject to} \quad & 2x_1 + 5x_2 \leq 10, \\ & 5x_1 + 3x_2 \leq 15, \\ & x_1 \geq 0, x_2 \geq 0. \end{aligned}$$

Actually, we can solve this problem by drawing a graph:



**Question**

1. What if we require integer $x_1, x_2$?
2. What if there are more types?

## 1.3 Data fitting

**Example**

Consider the *free falling motion*. The height and the time of a free fall follow the law $h = gt^2/2$. However, the practical data may not exhibit the perfect law.
Suppose we have the following data and we would like to use $h = gt^2/2$ to fit the data. Which value of coefficients $g$ should we choose?

| $h$ | 10 | 20 | 30 | 40 |
|---|---|---|---|---|
| $t^2$ | 1.011 | 2.019 | 3.032 | 4.041 |

However, before solving this problem, we should first ask the following question: if we choose a certain value of $g$, how can we measure the difference between the theoretical values of $h$ and the practical data?

### Question

Generally, we have the following question. Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$, $y \in \mathbb{R} = \boldsymbol{w}^\mathsf{T}\boldsymbol{x} + b = b + w_1x_1 + \cdots + w_nx_n$, where $\boldsymbol{w} = (w_1, w_2, \ldots, w_n) \in \mathbb{R}^n$. Given a set of data $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)\}$ and we guess the values of $\boldsymbol{w}$ and $b$ , how can we measure the difference between $(y_1, \ldots, y_m)$ and $(\hat{y}_1, \ldots, \hat{y}_m)$, where $\hat{y}_i = \boldsymbol{w}^\mathsf{T}\boldsymbol{x}_i + b$?

If we only have two numbers $y$ and $\hat{y}$, it is natural to use the *absolute value* $|y - \hat{y}|$ to measure the difference. Moreover, it is clear that 3 is closer to 1 than 2.
However, if we have two vectors, how can we measure the difference? Is $(2, 1)$ closer to $(1, 1)$ than $(1, 2)$?

We need to extend the concept of the absolute value to measure the distances between vectors in $\mathbb{R}^n$.

### Definition (*Norm*)

Given a vector space $V$ over a field $F$ (usually $V = \mathbb{R}^n$), a norm $\|\cdot\| : V \to \mathbb{R}$ is a function having the following properties:
1. (*Nonnegativity*) $\forall\, v \in V$, $\|v\| \geq 0$.
2. (*Positive definiteness*) $\|v\| = 0$ iff $v = \mathbf{0}$.
3. (*Absolute homogeneity*) $\forall\, r \in \mathbb{R}$ and $v \in V$, $\|r \cdot v\| = |r| \cdot \|v\|$.
4. (*Triangle inequality*) $\forall\, u, v \in V$, $\|u + v\| \leq \|u\| + \|v\|$.

This definition is not constructive. That means any function $\|\cdot\| : V \to \mathbb{R}$ satisfying above properties can reasonably measure the distance between two vectors. We now see some specific examples.

### Example ($L^p$ *norm*)

- $L^p$ norm defined on $\mathbb{R}^n$: $\|x\|_p = \left(|x_1|^p + |x_2|^p + \cdots + |x_n|^p\right)^{1/p}$, where $p \geq 1$. In particular,
  - $L^1$ norm: $\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n|$.
  - $L^2$ norm: $\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$, which is the most common norm in $\mathbb{R}^n$.
  - $L^\infty$ norm: $\|x\|_\infty = \lim_{p\to\infty} \|x\|_p = \max\{|x_1|, |x_2|, \ldots, |x_n|\}$.
- Sometimes we will see the so-called $L^0$ norm, given by $\|x\|_0 = |x_1|^0 + \cdots + |x_n|^0$, that is, the number of nonzero entries. Note that it is not a *norm* indeed. We call it $L^0$ norm just for convenience.

### Question

Why do $L^p$ norms satisfy the triangle inequality?

### Tip

The triangle inequality follows the so-called *Minkowski inequality*, which we will prove several weeks later.

Another example is called the *canonical norm*, which is induced by the *inner product*. Usually, the inner product of two vectors $x = (x_1, \ldots, x_n)^\mathsf{T}$ and $y = (y_1, \ldots, y_n)^\mathsf{T}$ is define by their *dot product*

$$\langle x, y \rangle \triangleq x^\mathsf{T} y = x_1 y_1 + \cdots + x_n y_n \,.$$

However, in fact, the inner product can be defined more general.

### Definition (*Inner product*)

An inner product for a vector space $V$ is a function $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{F}$ (we assume $\mathbb{F} = \mathbb{R}$ in our course) satisfying

1. (*Nonnegetivity*) $\forall\, v \in V, \langle v, v\rangle \geq 0$.
2. (*Positive definiteness*) $\langle v, v\rangle = 0$ iff $v = \mathbf{0}$.
3. (*Symmetry*) $\forall\, u, v \in V, \langle u, v\rangle = \langle v, u\rangle$.
4. (*Linearity*) $\forall\, r \in \mathbb{F}, u, v, w \in V, \langle ru + v, w\rangle = r\langle u, w\rangle + \langle v, w\rangle$.

Given a vector space with an inner product, the *canonical norm* is given by $\|x\| = \sqrt{\langle x, x\rangle}$.

**Example (*Euclidean space* $\mathbb{R}^n$)**

The inner product is given by

$$\langle x, y\rangle = x^\mathsf{T} y = \sum_{i=1}^{n} x_i y_i \,,$$

and thus the canonical norm is the $L^2$ norm.

**Theorem (*Cauchy-Schwarz inequality*)**

For any vector space with any inner product and the canonical norm, it holds that

$$|\langle x, y\rangle| \leq \|x\| \cdot \|y\| \,.$$

## Least square method

We now return to the linear regression problem. A famous and well-applied method is the *least square method*, which use the $L^2$-norm as the objective function.

Given $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_m)^\mathsf{T} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{y} = (y_1, \dots, y_m) \in \mathbb{R}^m$, assume the value of coefficients are $\boldsymbol{w} = \hat{\boldsymbol{w}} = (\hat{w}_1, \dots, \hat{w}_n)^\mathsf{T}$, then the value of $\boldsymbol{y}$ should be (using $\boldsymbol{X}$ and our $\hat{\boldsymbol{w}}$) $\hat{\boldsymbol{y}} = (\hat{y}_1, \dots, \hat{y}_n)$ where

$$\hat{y}_i = \hat{w}_1 x_1 + \cdots + \hat{w}_n x_n + b \,,$$

and our goal is to minimize $\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2 = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$. Using the form of matrix multiplication, this is to solve

$$\min_{\boldsymbol{w}\in\mathbb{R}^n, b\in\mathbb{R}} \|\mathbf{X}\boldsymbol{w} + b\mathbf{1} - \boldsymbol{y}\|_2^2$$

Note that the term $b\mathbf{1}$ is not necessary. Let $\mathbf{x}'_i = (x_{i1}, \ldots, x_{in}, 1), \boldsymbol{w}' = (w_1, \ldots, w_n, b)$, the above problem is converted into the following form:

$$\min_{\boldsymbol{w} \in \mathbb{R}^{n+1}} \|\mathbf{X}'\boldsymbol{w} - \boldsymbol{y}\|_2^2$$

We now consider how to solve the above problem. $\boldsymbol{Xw}$ is the column space of the matrix $\boldsymbol{X}$, denoted by $\mathcal{R}(X)$ or $\operatorname{im}(X)$. The above optimization problem is to ask the minimum distance from $\boldsymbol{y}$ to the subspace $\mathcal{R}(X)$. The answer is the distance from $\boldsymbol{y}$ to the orthogonal projection of $\boldsymbol{y}$ onto the subspace.

Assume $\boldsymbol{y}'$ is the orthogonal projection, and let $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{y}'$. Then we have that $\boldsymbol{e}$ is orthogonal to the subspace $\mathcal{R}(X)$, i.e., for any $\boldsymbol{w} \in \mathbb{R}^n$, $\boldsymbol{e}^\top \boldsymbol{Xw} = 0$. It yields that $\boldsymbol{X}^\top \boldsymbol{e} = \boldsymbol{0}$. Also there exists $\hat{\boldsymbol{w}}$ such that $\boldsymbol{y}' = \boldsymbol{X}\hat{\boldsymbol{w}}$ (actually $\hat{\boldsymbol{w}}$ is the desired $\boldsymbol{w}$ in the above optimization problem). So we have

$$\begin{cases} \boldsymbol{X}^\top \boldsymbol{e} = \boldsymbol{0}\,, \\ \exists\, \hat{\boldsymbol{w}} \text{ such that } \boldsymbol{y} - \boldsymbol{e} = \boldsymbol{X}\hat{\boldsymbol{w}}\,. \end{cases}$$

Thus, we have

$$\boldsymbol{X}^\top (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}}) = \boldsymbol{0}\,,$$

which implies

$$\boldsymbol{X}^\top \boldsymbol{y} = \boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{w}}\,.$$

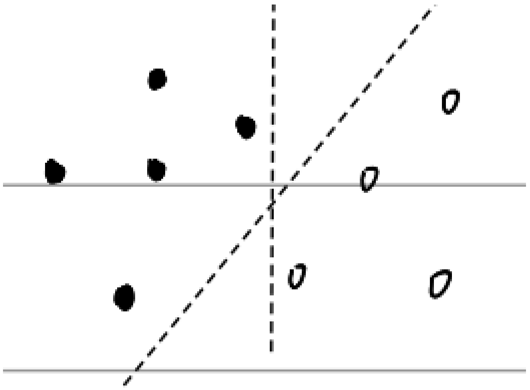
So $\hat{\boldsymbol{w}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$ if $\boldsymbol{X}^\top \boldsymbol{X}$ is invertible. Then $\operatorname{rank}(X) = n$ suffices. We will revisit this topic later.

## 1.4 Classification and the support vector machine

Given a data set $\{\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})^\top\}_{i=1,2,\ldots m}$, a *support-vector machine* is to classify (separate) these data using a $(n-1)$-dimensional hyperplane. Associate $y_i \in \{-1, +1\}$ to each $\boldsymbol{x}_i$. We would like to divide the group of $\boldsymbol{x}_i$ for which $y_i = -1$ from the group of $\boldsymbol{x}_j$ for which $y_j = +1$. Then a hyperplane $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ is a desired one if for all $i = 1, 2, \ldots, m$,

$$y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) > 0\,.$$

However, there are infinite many hyperplanes satisfying our requirements. For example, we would like to classify black points and white points in the following picture, and two dot lines are satisfied ones. Which one is better? A reasonable choice is to find the "maximum-margin" hyperplane, that is, to make the minimum distance from points to the hyperplane as large as possible.



## Distance to a hyperplane

We first consider the problem of computing the distance from a point to a hyperplane.
Assume the hyperplane is $P : \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x} + b = 0$ and the point is $\boldsymbol{y}$. Again (similar to the least square method), consider the orthogonal projection of $\boldsymbol{y}$ onto $P$. Suppose the orthogonal projection is $\boldsymbol{y}'$. It is clear that $\boldsymbol{w} \perp P$. So $\exists\, r \in \mathbb{R}$ such that $\boldsymbol{y} - \boldsymbol{y}' = r\boldsymbol{w}$ . Also, $\boldsymbol{w}^{\mathsf{T}}\boldsymbol{y}' + b = 0$ since $\boldsymbol{y}' \in P$.
Now we have

$$\boldsymbol{w}^{\mathsf{T}}(\boldsymbol{y} - r\boldsymbol{w}) + b = 0\,,$$

which yields

$$r = \frac{\boldsymbol{w}^{\mathsf{T}}\boldsymbol{y} + b}{\|\boldsymbol{w}\|_2^2}\,.$$

The distance from $\boldsymbol{y}$ to $P$ is

$$\|r\boldsymbol{w}\| = \frac{|\boldsymbol{w}^{\mathsf{T}}\boldsymbol{y} + b|}{\|\boldsymbol{w}\|_2}\,.$$

Back to the problem of classification. Now our goal is to solve the following optimization:

$$\begin{aligned}\max_{\boldsymbol{w}\in\mathbb{R}^n,\, b\in\mathbb{R}}\ \min_i\ & \frac{|\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|} \\ \text{subject to}\quad & y_i(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_i + b) > 0\,.\end{aligned}$$

But this form is too complicated to solve. We would like to simplify it.
Note that $|\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_i + b| = y_i(\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_i + b)$ and we could choose proper $\boldsymbol{w}$ and $b$ so that

$\min y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = 1$. So the optimization is equivalent to

$$\begin{aligned} \max_{\boldsymbol{w}\in\mathbb{R}^n,\, b\in\mathbb{R}} \quad & \frac{1}{\|\boldsymbol{w}\|} \\ \text{subject to} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1\,, \end{aligned}$$

which is further equivalent to

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^n,\, b\in\mathbb{R}} \quad & \|\boldsymbol{w}\|^2 \\ \text{subject to} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1\,. \end{aligned}$$

The last form is easy to solve (since it is actually a *convex optimization*).

**Remark**

The constraints $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$ are equivalent to our assumption $\min y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = 1$, because our goal is to *minimize* the norm of $\boldsymbol{w}$. If $\min y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) > 1$, the corresponding $\boldsymbol{w}$ cannot be the optimal solution.