

# Contents

<b>1</b>	<b>Linear Programming</b>	<b>2</b>
1.1	Linear Programming . . . . .	2
1.2	Convex Sets and Functions . . . . .	3
1.3	Geometry of Linear Programming . . . . .	4

# Linear Programming

## 1.1 Linear Programming

Recall that in general, an optimisation problem can be formulated as

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \\ \text{s.t. } \mathbf{x} \in P, \end{aligned}$$

where  $P \subseteq \mathbb{R}^n$  is called the *feasible set* (or *feasible region*).

### Definition 1.1.1 ► Linear Programming Problem

A **linear programming** (LP) problem is an optimisation problem where the objective function  $f$  is linear and the feasible set  $P$  is a polyhedron.

We can therefore formulate a linear programming problem as

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \mathbf{a}_i^T \mathbf{x} \leq b_i \quad \text{for } i = 1, 2, \dots, p \\ \mathbf{a}_j^T \mathbf{x} = b_j \quad \text{for } i = 1, 2, \dots, m, \end{aligned}$$

where  $\mathbf{c} \in \mathbb{R}^n$  is called the *cost* or *profit* vector,  $\mathbf{a}_i^T \mathbf{x}$  and  $\mathbf{a}_j^T \mathbf{x}$  are called the *constraints* and  $\mathbf{x}$  is known as *decision variables*.

In particular, the following is known as the *standard form* of a linear program:

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \mathbf{c}^T \mathbf{x} \\ \text{s.t. } \mathbf{Ax} = \mathbf{b} \\ x_i \geq 0, \quad \text{for } i = 1, 2, \dots, m. \end{aligned}$$

One should realise that a linear program in the standard form can be more easily solved by using linear algebra to find the optimal solution. Note that not every optimisation problem is given in the standard form. Fortunately, we can always convert a linear program into the standard form.

For example, consider the constraint  $a_i x_i \leq b_i$ . Notice that this is essentially equivalent to  $a_i x_i + s_i = b_i$  for some  $s_i \geq 0$  known as the *slack variable*. Similarly,  $a_j x_j \geq b_j$  can be re-written as  $a_j x_j - s_j = b_j$  for some  $s_j \geq 0$ .

Note that some of the  $x_i$ 's may be free variables. In this case, we can convert it to  $x_i^+ - x_i^-$  for some  $x_i^+, x_i^- \geq 0$ . For instance, we can take  $x_i^+ = 0$  and  $x_i^- > 0$  whenever  $x_i < 0$  and vice versa for  $x_i > 0$ . Note that this correspondence is not unique.

## 1.2 Convex Sets and Functions

### Proposition 1.2.1

Let  $f_1, f_2, \dots, f_m : \mathbb{R}^n \rightarrow \mathbb{R}$  be convex functions, then the function

$$f(\mathbf{x}) := \max_{i=1,2,\dots,m} f_i(\mathbf{x})$$

is convex.

*Proof.* Take any  $\mathbf{x} \neq \mathbf{y} \in \mathbb{R}^n$  and consider  $\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}$  for some  $\lambda \in [0, 1]$ . Note that for each of the  $f_i$ 's, we have

$$f_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \lambda f_i(\mathbf{x}) + (1 - \lambda) f_i(\mathbf{y}),$$

and so

$$\max_{i=1,2,\dots,m} f_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \leq \max_{i=1,2,\dots,m} [\lambda f_i(\mathbf{x}) + (1 - \lambda) f_i(\mathbf{y})].$$

Therefore,

$$\begin{aligned} f(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) &= \max_{i=1,2,\dots,m} f_i(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) \\ &\leq \max_{i=1,2,\dots,m} [\lambda f_i(\mathbf{x}) + (1 - \lambda) f_i(\mathbf{y})] \\ &= \lambda \max_{i=1,2,\dots,m} f_i(\mathbf{x}) + (1 - \lambda) \max_{i=1,2,\dots,m} f_i(\mathbf{y}) \\ &= \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y}). \end{aligned}$$

□

## 1.3 Geometry of Linear Programming

### Definition 1.3.1 ► Polyhedron

A **polyhedron** is defined as the set

$$P := \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$$

where  $\mathbf{A} \in \mathbb{R}^{m \times n}$  and  $\mathbf{b} \in \mathbb{R}^m$ .

Geometrically, a polyhedral set  $P$  can be defined alternatively as a finite intersection of half-planes:

$$P := \bigcap_{i=1}^m \{\mathbf{x} \in \mathbb{R}^n : \mathbf{a}_i^T \mathbf{x} \leq b_i\}.$$