Contents

1 Linear Programming

1.1 Linear Programming

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

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

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

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

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

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

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

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 ome $s_j \geq 0$.

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 $f_1, f_2, \cdots, f_m : \mathbb{R}^n \to \mathbb{R}$ be convex functions, then the function

$$f(\boldsymbol{x}) \coloneqq \max_{i=1,2,\cdots,m} f_i(\boldsymbol{x})$$

is convex.

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

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

and so

$$\max_{i=1,2,\cdots,m} f_i\big(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{y}\big) \leq \max_{i=1,2,\cdots,m} \left[\lambda f_i(\boldsymbol{x}) + (1-\lambda) f_i(\boldsymbol{y})\right].$$

Therefore,

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

□

2

The Simplex Method

2.1 Geometry of Linear Programming

Definition 2.1.1 ▸ Polyhedron

A **polyhedron** is defined as the set

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

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

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

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

Theorem 2.1.2 ▸ Basic Solution Characterisation

A vector $\boldsymbol{x}^* \in \mathbb{R}^n$ is a basic solution if and only if

- $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}$, *and*
- *There exists an index set $B \subseteq \{1, 2, \cdots, n\}$ such that the set*

$$\{\boldsymbol{A}_i : i \in B\}$$

is linearly independent and $x_j^* = 0$ for all $j \notin B$, where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 & \cdots & \boldsymbol{A}_n \end{bmatrix}.$$

Proof. Suppose $B = \{B(1), B(2), \cdots, B(m)\}$ and let $N = \{1, 2, \cdots, n\} - B$. For each $i \in N$, since $x_i^* = 0$, we have $\boldsymbol{e}_i^{\mathsf{T}}\boldsymbol{x}^* = 0$. Therefore, the matrix representation for the active constraints is

$$\begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{e}_{N(1)}^{\mathrm{T}} \\ \boldsymbol{e}_{N(2)}^{\mathrm{T}} \\ \vdots \\ \boldsymbol{e}_{N(n-m)}^{\mathrm{T}} \end{bmatrix}.$$

Re-arranging the columns, the above matrix can be re-written as

$$\begin{bmatrix} \boldsymbol{A}_B & \boldsymbol{A}_N \\ \boldsymbol{0} & \boldsymbol{I}_N \end{bmatrix}.$$

Note that the columns of $\boldsymbol{A}_B$ is linearly independent, so $\det(\boldsymbol{A}_B) \neq 0$. Therefore,

$$\begin{vmatrix} \boldsymbol{A}_B & \boldsymbol{A}_N \\ \boldsymbol{0} & \boldsymbol{I}_N \end{vmatrix} = \det(\boldsymbol{A}_B)\det(\boldsymbol{I}_N) \neq 0,$$

and so the matrix is invertible. Therefore, the rows of the matrix are linearly independent. This means that the set

$$\{\nabla h_i : \ i = 1, 2, \cdots, n\}$$

is linearly independent. Therefore, $\boldsymbol{x}^*$ is a basic feasible solution.

Suppose conversely that $\boldsymbol{x}^*$ is a basic feasible solution, then clearly $\boldsymbol{A}\boldsymbol{x}^* = \boldsymbol{b}$. Suppose there are m equality constraints, then we must have $(n-m)$ active active inequality constraints at $\boldsymbol{x}^*$, indexed by $N = \{N(1), N(2), \cdots, N(n-m)\}$, such that the constraints are linearly independent. Therefore, the matrix

$$\begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{e}_{N(1)}^{\mathrm{T}} \\ \boldsymbol{e}_{N(2)}^{\mathrm{T}} \\ \vdots \\ \boldsymbol{e}_{N(n-m)}^{\mathrm{T}} \end{bmatrix}$$

is invertible. Let $B \coloneqq \{B(1), B(2), \cdots, B(m)\} = \{1, 2, \cdots, n\} - N$ be an index set, then the above matrix can be re-arranged as

$$\begin{bmatrix} \boldsymbol{A}_B & \boldsymbol{A}_N \\ \boldsymbol{0} & \boldsymbol{I}_N \end{bmatrix},$$

which is invertible. Therefore, $\{\boldsymbol{A}_{B(1)}, \boldsymbol{A}_{B(2)}, \cdots, \boldsymbol{A}_{B(m)}\}$ is linearly independent. Note that $x_i^* = 0$ for all $i \in N$. □

Theorem 2.1.3 ▸ Equivalent Conditions for the Existence of Basic Feasible Solution

Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ where $m \geq n$ and

$$P = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}\} \neq \varnothing.$$

The following statements are equivalent:

1. *P does not contain any straight line.*
2. *P has a basic feasible solution.*
3. *P has n linearly independent constraints.*

2.2 The Simplex Method Algorithm

Definition 2.2.1 ▸ Feasible Direction

Let P be a polyhedron and $\boldsymbol{x} \in P$ be a feasible point. A vector $\boldsymbol{d}$ is a **feasible direction** if $\boldsymbol{x} + \lambda\boldsymbol{d} \in P$ for some $\lambda > 0$.