Chapter 4

Linear programming

4.1 Introduction to linear programming

Linear programming is a branch of optimisation, where both the objective function and the constraints are given by linear equations, or linear inequalities. The word “programming” does not refer to coding in the context of computer science. Linear programming was originally studied in a military context around the time of the Second World War. It was being applied to optimize military supply-chain problems, and supply schedules were called “programs” by the military.

Efficient methods for solving linear programming problems became available in the late 1930s. In 1939, Kantorovich presented a number of solutions to some problems related to production and transportation planning. During World War II, Koopmans contributed significantly to the solution of transportation problems. Kantorovich and Koopmans were awarded a Nobel Prize in economics in 1975 for their work on the theory of optimal allocation of resources. In 1947, Dantzig developed a efficient and elegant method for solving linear programs, known today as the simplex method, which will be seen later in the chapter.

The following basic example, which will be our main example, is based on an example in *Calvert and Voxman's* Linear Programming [2]. A machine shop produces two products each requiring manufacturing time on 3 machines. Each product has a fixed profit associated with it. The total available time on each of the three machines is limited. The situation is represented in a table:

hours per machine per item			
machine type	product I	product II	total hours available per machine per week
A	2	1	70
B	1	1	40
C	1	3	90
profit per item	\$40	\$60	

Let x_1, x_2 represent the number of units of products I and II respectively produced each week. The objective function to optimise is the weekly profit,

$$z = 40x_1 + 60x_2.$$

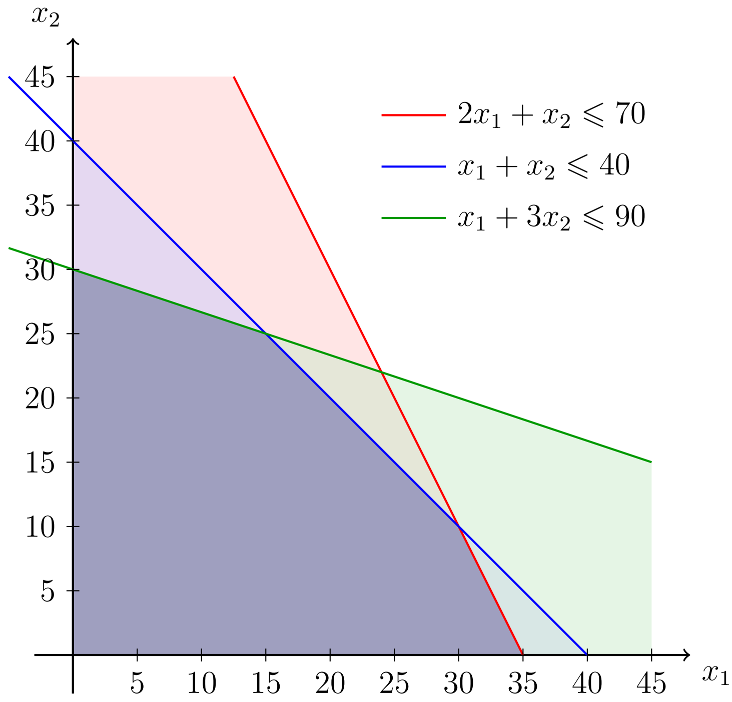
Then, for example, the total time needed by machine A is $2x_1 + x_2$, which is capped at 70 hours per week resulting in the constraint $2x_1 + x_2 \leqslant 70$. The other constraints are obtained similarly, and are all linear:

$$\begin{aligned}
2x_1 + x_2 &\leqslant 70 && \text{available time on machine A} \\
x_1 + x_2 &\leqslant 40 && \text{available time on machine B} \\
x_1 + 3x_2 &\leqslant 90 && \text{available time on machine C} \\
x_1 &\geqslant 0 \\
x_2 &\geqslant 0.
\end{aligned}$$

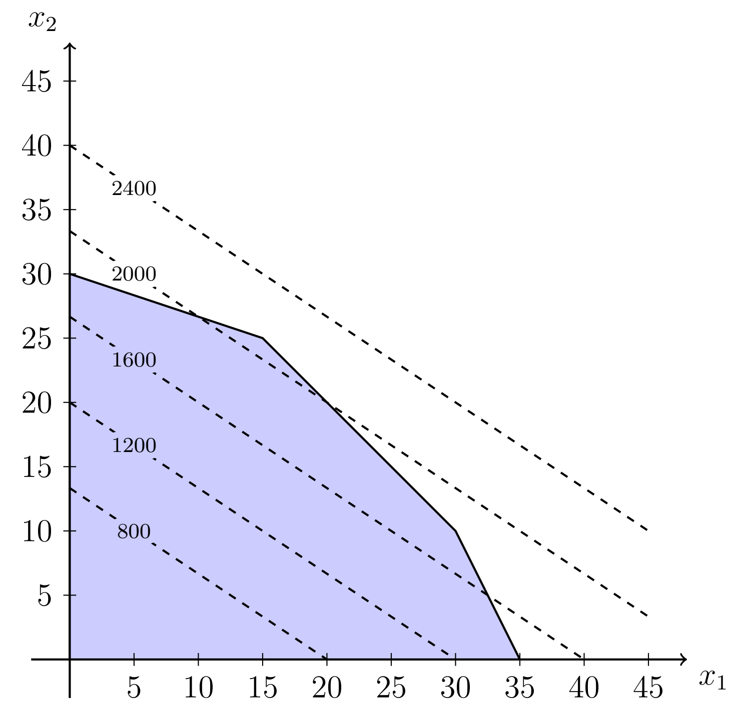
Using matrices, we could express the problem as follows:

$$\begin{aligned}
\text{maximise} \quad & z = \begin{pmatrix} 40 & 60 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\
\text{subject to} \quad & \begin{pmatrix} 2 & 1 \\ 1 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leqslant \begin{pmatrix} 70 \\ 40 \\ 90 \end{pmatrix} \\
& \begin{pmatrix} x_1 & x_2 \end{pmatrix} \geqslant \mathbf{0}.
\end{aligned}$$

In this example we can obtain the solution by graphical means. Each constraint defines a half-plane in $\mathbb{R}^2$. The region defined by each constraint is shown in the diagram below.



Taking the intersection of these regions yields the feasible set, plotted below. In addition, five level sets of the objective function are shown.



Notice that as z increases, the level set of z sweep across the feasible region and then "leave" the feasible region at one of the corner points. This suggests that z will take its maximum value at one of these corner points. In the above example, the maximum is clearly attained at the point $P = (15, 25)^T$. The maximum value is thus

$$z = 40 \times 15 + 60 \times 25 = 2100.$$

So the solution is that 15 units per week of product 1 and 25 units per week of product 2 gives a maximum weekly profit of \$2,100.

In general, a **linear program** is an optimisation problem of the following form:

$$\begin{aligned} \text{maximise} \quad & z = \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}_1\mathbf{x} \leqslant \mathbf{b}_1 \\ & \mathbf{A}_2\mathbf{x} \geqslant \mathbf{b}_2 \\ & \mathbf{A}_3\mathbf{x} = \mathbf{b}_3 \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned}$$

where each $\mathbf{b}_i \geqslant 0$. The stipulation that each $\mathbf{b}_i \geqslant 0$ is a necessary feature for the linear programming algorithms to work, but since a constraint of the form $a_1x_1 + \ldots + a_nx_n \leqslant -b$ can be written equivalently as $-a_1x_1 - \ldots - a_nx_n \geqslant b$, this does not cause any issues.

In linear programming, some mathematical objects have non-mathematical names because of the subject's background in economics. The coefficients, $\mathbf{c}$, in the objective function are called **cost coefficients**, and the constants in each $\mathbf{b}_i$ are called **resource values**. The objective function is sometimes called the **profit**. For the same reason, it is typical to treat maximisation problems when studying linear programming, which we will do herein.

4.2 The corner point theorem

A full understanding of the solutions to a linear program depends highly on the underlying geometry. An important result, which is also dubbed the **fundamental theorem of linear programming**, is the **corner point theorem** given below. This theorem narrows down the search space for optimal solutions by identifying that they exclusively reside at the corner points of the feasible region.

Background: convex sets and corner points

A **hyperplane** is the solution set to a linear equation

$$\mathcal{H} = \{\mathbf{x} : \mathbf{u}^T\mathbf{x} = b\}$$

where $\mathbf{0} \neq \mathbf{u} \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are given. If $\mathbf{x}$ and $\mathbf{y}$ are distinct elements of $\mathcal{H}$, then the difference $\mathbf{x}-\mathbf{y}$ is orthogonal to the vector $\mathbf{u}$, so $\mathbf{u}$ is called the **normal** to $\mathcal{H}$. The hyperplane $\mathcal{H}$ divides $\mathbb{R}^n$ into two **closed half-spaces**, the **positive half-space** $\mathcal{H}_+ = \{\mathbf{x} : \mathbf{u}^T\mathbf{x} \geqslant b\}$ and the **negative half-space** $\mathcal{H}_- = \{\mathbf{x} : \mathbf{u}^T\mathbf{x} \leqslant b\}$. An **open half-space** uses strict inequalities instead. For a given $S \subseteq \mathbb{R}^n$, we say that a hyperplane $\mathcal{H}$ is a **supporting hyperplane** (of S) if $\mathcal{H}$ contains at least one point of S, and S is contained in one of the two half-spaces $\mathcal{H}_+$ or $\mathcal{H}_-$.

In $\mathbb{R}^n$ the **line segment** between two points $\mathbf{x}$ and $\mathbf{y}$ is the set

$$\mathcal{S} = \{\alpha\mathbf{x} + (1-\alpha)\mathbf{y} : \alpha \in [0,1]\}.$$

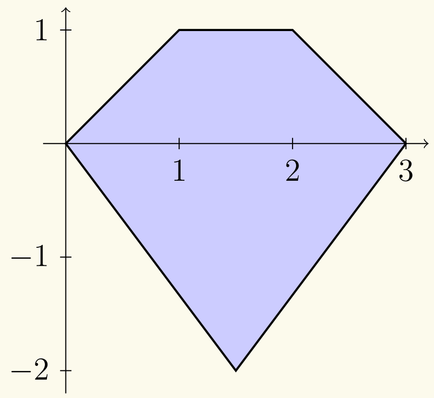
A set $\mathcal{U} \subset \mathbb{R}^n$ is **convex** if, for all $\mathbf{x}, \mathbf{y} \in \mathcal{U}$, the line segment between $\mathbf{x}$ and $\mathbf{y}$ is a subset of $\mathcal{U}$. Examples of convex sets are: the empty set; a set containing exactly one point; a line segment; a hyperplane; a half-space; $\mathbb{R}^n$. The intersection of any collection of convex sets is also convex. A point $\mathbf{x}$ in a convex set $\mathcal{U}$ is an **extreme point** (or **corner point**) if the set $\mathcal{U}\backslash\{\mathbf{x}\}$, obtained from $\mathcal{U}$ by removing $\mathbf{x}$, is still convex.

Example: convex sets and corner points

Let R_1 be the region obtained as the intersection of the half-spaces defined by the following inequalities:

$$y \leqslant x, \quad y \leqslant 1, \quad y \leqslant (3-x), \quad y \geqslant -\tfrac{4}{3}x, \quad y \geqslant -\tfrac{4}{3}(3-x).$$

It is plotted below.

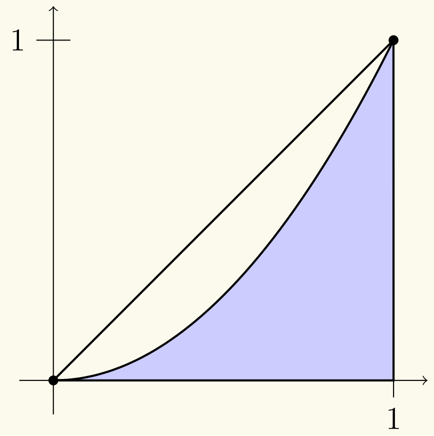


Since half-spaces are convex, and R_1 is the intersection of a collection of half-spaces, it follows that R_1 is convex. The corner points, which can be obtained by computing the intersection of the boundary lines, are $(0,0)$, $(1,1)$, $(2,1)$, $(3,0)$ and $(\frac{3}{2}, -\frac{3}{2})$.

Now consider the region R_2 defined by

$$R_2 = \{(x, y) \in \mathbb{R}^2 : 0 \leqslant y \leqslant 1, x \leqslant 1, y \leqslant x^2\}.$$

As convexity is defined by the ability to draw line segments between arbitrary points, a single counterexample is enough to disqualify a set from being convex. The points $(0, 0)$ and $(1, 1)$ are in R_2, but the line segment from $(0, 0)$ to $(1, 1)$ is evidently not contained in R_2, as the following diagram demonstrates.



Hence, R_2 is not convex.

Theorem 10

The feasible region for a linear program is convex.

This is because each linear inequality constraint forms a half-space, and each equality constraint forms a hyperplane, which are all convex, and the intersection of any collection of convex sets is convex.

Theorem 11 (The corner point theorem)

If the objective function for a linear program attains a maximum on its feasible region Ω, then it occurs at a corner point of Ω. Moreover, if Ω is bounded, then the maximum is attained.

Notice that the maximizer may or may not be a unique maximiser, but, if the feasible region is bounded, then at least one maximiser is guaranteed. Also, if the feasible region is unbounded, then an optimal solution may not exist; however, if one does exist, then the optimal solution is attained at a corner point of the feasible region.

A proof of the corner point theorem is not examinable, but nice enough to be included here. Assume that $f(\mathbf{x}) = \mathbf{c}^T\mathbf{x}$ has a maximiser $\mathbf{p} \in \Omega$, and let H be the level set of $f(\mathbf{p})$; i.e.,

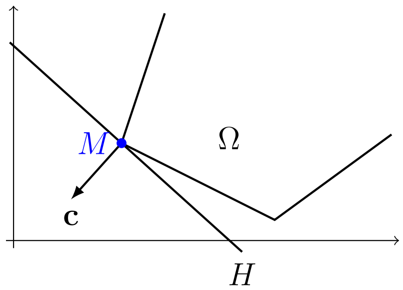
$$H = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{c}^T\mathbf{x} = f(\mathbf{p})\}.$$

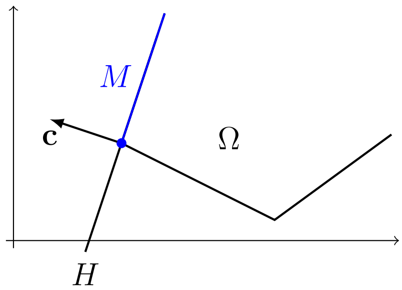
Let $M = H \cap \Omega$. We claim that M contains at least one corner point of Ω.

Note that H is a supporting hyperplane of Ω, because $\mathbf{p} \in \Omega \cap H$ and, as $\mathbf{p}$ is a maximiser of f, the feasible region Ω is contained in the negative half-space

$$H_- = \{\mathbf{x} : \mathbf{c}^T\mathbf{x} \leqslant f(\mathbf{p})\}.$$

The open half-space $H_- \backslash H$ is convex, so for any two points $\mathbf{p}_1, \mathbf{p}_2 \in \Omega \backslash H$, the line segment joining $\mathbf{p}_1$ and $\mathbf{p}_2$ does not intersect H. Hence, if M contains exactly one point $\mathbf{m}$, then $\Omega \backslash \{\mathbf{m}\}$ is convex and it follows by definition that $\mathbf{m}$ is a corner point. Otherwise, there is a face of Ω which is parallel to H. The constraint $\mathbf{x} \geqslant 0$ ensures that the face has corner points, in which case the corner points of that face are contained in M. This is depicted below in two dimensions.





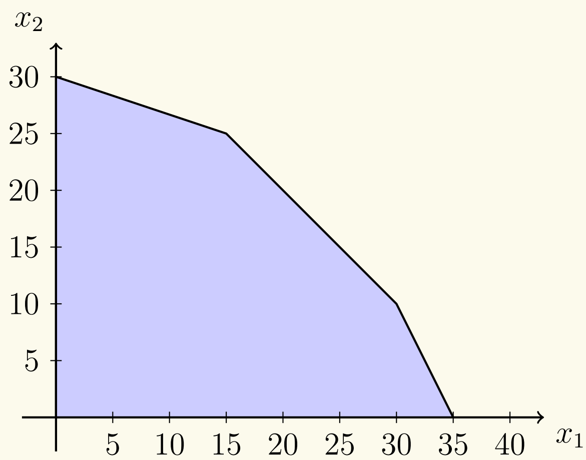
In general, Ω is closed, so if Ω is bounded, then by Weierstrass (Theorem 4), there exists a maximiser of f over Ω. Hence, the maximum is attained in that case.

Example: the corner point theorem

To illustrate the corner point theorem, let us return to our basic example,

$$\begin{aligned} \text{maximise} \quad & z = 40x_1 + 60x_2 \\ \text{subject to} \quad & 2x_1 + x_2 \leqslant 70 \\ & x_1 + x_2 \leqslant 40 \\ & x_1 + 3x_2 \leqslant 90 \\ & \mathbf{x} \geqslant 0. \end{aligned}$$

The feasible region is depicted below.



We can see that Ω is bounded, so we know the optimal value exists, and that it occurs at one the corners. The corner points and corresponding values of f are:

$$\begin{aligned} (x_1, x_2) &= (0, 0), & f(0, 0) &= 0 \\ (x_1, x_2) &= (0, 30), & f(0, 30) &= 1800, \\ (x_1, x_2) &= (35, 0), & f(35, 0) &= 1400, \\ (x_1, x_2) &= (15, 25), & f(15, 25) &= 2100, \\ (x_1, x_2) &= (30, 10), & f(30, 10) &= 1800. \end{aligned}$$

Hence the optimal value is $z = 2100$, which occurs at the point $(15, 25)$.

4.3 Basic solutions

In this section, we will see that matrix methods like those used in Section 2.2 can be applied to linear programming problems. A system of linear equations with m equations and n variables:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ \vdots \qquad\quad \vdots \qquad\quad &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n &= b_m, \end{aligned}$$

(or the corresponding augmented matrix) is said to be in **canonical form** if there are m distinguished variables called **basic variables** such that

(a) each basic variable x_i has coefficient 1 in exactly one equation of the system, and coefficient zero in each of the remaining $(m-1)$ equations, and

(b) each equation contains exactly one basic variable.

Note that canonical form is **not** the same as reduced row echelon form. First of all, a matrix in canonical form does not have zero rows. Secondly, the order of the basic variables may be different. In a row reduced echelon form matrix the row leading variables are the basic variables; they are the first non-zero variables and appear in the rows in increasing order. For a canonical matrix neither of these are required.

Another way of expressing when a system is in canonical form uses the matrix of coefficients of the system

$$\mathbf{A} = \begin{pmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \ldots & \mathbf{a}_n \end{pmatrix}$$

whose i-th column is $\mathbf{a}_i = (a_{1i}, a_{2i}, \ldots, a_{mi})^T \in \mathbb{R}^m$. The system is in canonical form if a subset of m columns of $\mathbf{A}$ is equal to the standard basis for $\mathbb{R}^m$, that is the set of vectors (in any order)

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \ \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \ \ldots, \ \mathbf{e}_m = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}.$$

Example: canonical form

For an augmented matrix with n rows, there must be n basic variables, corresponding to the basis vectors $\mathbf{e}_1$, $\mathbf{e}_2$, …, $\mathbf{e}_n$. For example, if an augmented matrix has 2 rows, then the following columns must appear for it to be in canonical form:

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \ \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The following system is in canonical form but not in reduced row echelon form:

$$\left(\begin{array}{cccc|c} 2 & 0 & 1 & 1 & 4 \\ 0 & 1 & 2 & 0 & -2 \end{array}\right).$$

Columns 2 and 4 correspond to $\mathbf{e}_2$ and $\mathbf{e}_1$, respectively, so the basic variables are x_2 and x_4. Note that x_2 appears exactly once, namely in equation 2, and x_4 appears exactly once, namely in equation 1.

The following system is not in canonical form:

$$\left(\begin{array}{cccc|c} 1 & 1 & 2 & 1 & 4 \\ 0 & 2 & 1 & 0 & -2 \end{array}\right).$$

The column $\mathbf{e}_1$ appears twice, but $\mathbf{e}_2$ does not appear at all.

The following system is in canonical form:

$$\left(\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array}\right).$$

The column $\mathbf{e}_1$ appears twice, and $\mathbf{e}_2$ appears once. We can choose the distinguished basic variables to be either x_1 and x_3, or x_2 and x_3.

Similarly, if an augmented matrix has 3 rows, then the following columns must appear for it to be in canonical form:

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \ \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \ \mathbf{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

The following augmented matrix is in canonical form:

$$\left(\begin{array}{cccc|c} 1 & 0 & 3 & 0 & -4 \\ 0 & 0 & 3 & 1 & -2 \\ 0 & 1 & 0 & 0 & 4 \end{array}\right),$$

with basic variables x_1, x_2 and x_4 (corresponding to columns 1, 4 and 2, respectively).

A system can be brought into canonical form by using the elementary row operations mentioned in Section 2.2. It can subsequently be solved by expressing the basic variables in terms of the non-basic variables. The solution will have the non-basic variables as parameters; we identify a special solution called the **basic solution** which is obtained by setting each non-basic variable equal to zero.

Example: basic solutions

Consider the following system:

$$\left(\begin{array}{cccc|c} 2 & 0 & 1 & 1 & 4 \\ 0 & 1 & 2 & 0 & -2 \end{array}\right).$$

The basic variables are x_2 and x_4, so we set the non-basic variables x_1 and x_3 to zero. This gives

$$\begin{aligned} 2x_1 + x_3 + x_4 = 4 &\implies x_4 = 4, \\ x_2 + 2x_3 = -2 &\implies x_2 = -2. \end{aligned}$$

The basic solution is then $(x_1, x_2, x_3, x_4) = (0, -2, 0, 4)$.

For the system

$$\left(\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array}\right),$$

the basic solution depends on the choice of basic variables. If x_1, x_3 are basic, then the basic solution is $(x_1, x_2, x_3) = (1, 0, 1)$. If x_2, x_3 are basic, then the basic solution is $(x_1, x_2, x_3) = (0, 1, 1)$.

For the system

$$\left(\begin{array}{cccc|c} 1 & 0 & 3 & 0 & -4 \\ 0 & 0 & 3 & 1 & -2 \\ 0 & 1 & 0 & 0 & 4 \end{array}\right),$$

the basic solution is $(x_1, x_2, x_3, x_4) = (-4, 4, 0, -2)$.

A linear program of the form

$$\begin{aligned} \text{maximise} \quad & z = \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned}$$

with $\mathbf{b} \geqslant 0$, is said to be in **standard form**.

A linear program with linear inequalities can be transformed into a linear program in standard by introducing a **slack variable** to each inequality constraint. A slack variable s is introduced by replacing a linear inequality $a_1x_1 + \ldots + a_nx_n \leqslant b$ with the linear equality $a_1x_1 + \ldots + a_nx_n + s = b$, and adding the constraint $s \geqslant 0$.

After adding slack variables, the optimal solution to the new linear program will have the same optimal solution as the original, but with each slack variable representing the amount by which the left-hand side of the corresponding inequality falls short of the right-hand side of the inequality.

Example: slack variables

Note carefully that slack variables are defined relative to inequalities of the form $a_1x_1+\ldots+a_nx_n \leqslant b$. Consider the following linear program:

$$\begin{aligned} \text{maximise} \quad & 5x_1 + 2x_2 - 3x_3 \\ \text{subject to} \quad & 2x_1 + 3x_2 + x_3 \leqslant 10, \\ & x_1 - 2x_2 + 3x_3 \geqslant 5, \\ & 3x_1 + 4x_2 - x_3 \leqslant 12, \\ & \mathbf{x} \geqslant \mathbf{0}. \end{aligned}$$

The first inequality, $2x_1 + 3x_2 + x_3 \leqslant 10$, is in a right form. We introduce the slack variable x_4, so the corresponding equation is $2x_1 + 3x_2 + x_3 + x_4 = 10$.

An equivalent expression for the second inequality, in the right form, is $-x_1 + 2x_2 - 3x_3 \leqslant -5$. We introduce the slack variable x_5, so the corresponding equation is $-x_1 + 2x_2 - 3x_3 + x_5 = -5$. As we proceed, it will seen that it is necessary for the right-hand side to be positive, so the equivalent equation is $x_1 - 2x_2 + 3x_3 - x_5 = 5$. Notice that the slack variable has a negative coefficient here.

The third inequality, $3x_1 + 4x_2 - x_3 \leqslant 12$, is in the right form. We introduce the slack variable x_6, so the corresponding equation is $3x_1 + 4x_2 - x_3 + x_6 = 12$.

The resulting linear program in standard form is

$$\begin{aligned} \text{maximise} \quad & 5x_1 + 2x_2 - 3x_3 \\ \text{subject to} \quad & 2x_1 + 3x_2 + x_3 + x_4 = 10, \\ & x_1 - 2x_2 + 3x_3 - x_5 = 5, \\ & 3x_1 + 4x_2 - x_3 + x_6 = 12, \\ & \mathbf{x} \geqslant \mathbf{0}. \end{aligned}$$

Note that this is *not* in canonical form (as the second equation has no basic variable).

For the main example, we introduce slack variables x_3, x_4 and x_5:

$$\begin{aligned} 2x_1 + x_2 &\leqslant 70 & \qquad & 2x_1 + x_2 + x_3 = 70 \\ x_1 + x_2 &\leqslant 40 & \longrightarrow \quad & x_1 + x_2 + x_4 = 40 \\ x_1 + 3x_2 &\leqslant 90 & & x_1 + 3x_2 + x_5 = 90. \end{aligned}$$

Notice that the inequalities on the left are equivalent to the inequalities: $x_3 \geqslant 0, x_4 \geqslant 0, x_5 \geqslant 0$. So we have $x_i \geqslant 0$ for all $i = 1, 2, 3, 4, 5$.

The resulting system of equations is now in canonical form, with augmented matrix

$$\left(\begin{array}{ccccc|c} 2 & 1 & 1 & 0 & 0 & 70 \\ 1 & 1 & 0 & 1 & 0 & 40 \\ 1 & 3 & 0 & 0 & 1 & 90 \end{array}\right).$$

The corresponding basic solution is: $x_1 = x_2 = 0$ (the non-basic variables) and $x_3 = 70,\ x_4 = 40,\ x_5 = 90$ (the basic variables). Herein lies the elegance; any linear program of the following form:

$$\begin{aligned} \text{maximise} \quad & z = \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \leqslant \mathbf{b} \\ & \mathbf{x} \geqslant \mathbf{0}, \end{aligned}$$

with $\mathbf{b} \geqslant \mathbf{0}$, will automatically be in canonical form for which the basic variables are the slack variables. If the linear program has inequalities of the opposite form or equalities without basic variables, more careful consideration is needed; however, we will see in due course that these linear programs can be converted to one of the form above.

Next, consider the objective function $z = 40x_1 + 60x_2$. We would like to keep track of this in the matrix as well. We write this equation in the form:

$$z - 40x_1 - 60x_2 = 0,$$

where z is an additional variable, which we add to the beginning of the list x_1, x_2, x_3, x_4, x_5. In this way we obtain a linear system with the following augmented matrix:

$$\begin{array}{cl} \left(\begin{array}{cccccc|c} 0 & 2 & 1 & 1 & 0 & 0 & 70 \\ 0 & 1 & 1 & 0 & 1 & 0 & 40 \\ 0 & 1 & 3 & 0 & 0 & 1 & 90 \\ 1 & -40 & -60 & 0 & 0 & 0 & 0 \end{array}\right) & \leftarrow z \text{ equation} \\ \uparrow & \\ z \text{ column} & \end{array}$$

If we forget about the slack variables (and z) for a moment, the basic solution is $x_1 = 0, x_2 = 0$, which is the origin and also a corner of the feasible region. Notice that the *value* of z (which does not depend on the slack variables) at the basic solution is equal to 0 and is held in the last column of the last row.

When we solve linear systems using the Gaussian algorithm, we apply elementary row operations to the augmented matrix until we get to reduced row echelon form. The same operations are used to bring a matrix into canonical form, or, to go from one canonical form to another.

Before we do that, we have another notational convention to mention. The (objective) variable z is a basic variable. It is the main quantity we are interested in, and its value at any basic solution will be given in the last column in the last row. We will never make this variable a non-basic variable and therefore we won't be adding multiples of the last row to the other rows. Thus the first column will not change, and may as well be omitted, which we do. The augmented matrix of our main example then becomes

$$\left(\begin{array}{ccccc|c} 2 & 1 & 1 & 0 & 0 & 70 \\ 1 & 1 & 0 & 1 & 0 & 40 \\ 1 & 3 & 0 & 0 & 1 & 90 \\ -40 & -60 & 0 & 0 & 0 & 0 \end{array}\right),$$

keeping the constraints $\mathbf{x} \geqslant \mathbf{0}$ in the back of our minds. We will still say the above matrix is in canonical form, as we keep in mind the omitted first column $\mathbf{e}_m$.

Example: setting up the matrix for a linear program

Consider the following linear program:

$$\begin{aligned} \text{maximise} \quad & z = 4x_1 + 2x_2 + 5x_3 \\ \text{subject to} \quad & 2x_1 + 3x_2 + x_3 \leqslant 10 \\ & x_1 + 2x_2 + 2x_3 \leqslant 8 \\ & 3x_1 + x_2 + 4x_3 \leqslant 16 \\ & x_1, x_2, x_3 \geqslant 0 \end{aligned}$$

To construct the initial matrix, we must introduce three slack variables, which will appear on the right hand side of the matrix as a copy of I_3. Writing the objective function as $z - 4x_1 - 2x_2 - 5x_3 = 0$, observe that the objective function is encoded by writing the coefficients of its negation in the bottom row. So, the initial matrix is

$$\left(\begin{array}{cccccc|c} 2 & 3 & 1 & 1 & 0 & 0 & 10 \\ 1 & 2 & 2 & 0 & 1 & 0 & 8 \\ 3 & 1 & 4 & 0 & 0 & 1 & 16 \\ -4 & -2 & -5 & 0 & 0 & 0 & 0 \end{array}\right)$$

Take care in the event of a minimisation problem, such as the following:

$$\begin{aligned} \text{minimise} \quad & z = 3x_1 + 2x_2 - x_3 \\ \text{subject to} \quad & 2x_1 + 3x_2 + x_3 \leqslant 10 \\ & x_1 - 2x_2 + 3x_3 \leqslant 5 \\ & x_1 + 4x_2 + 2x_3 \leqslant 12 \\ & x_1, x_2, x_3 \geqslant 0 \end{aligned}$$

The current formulation assumes the problem is a maximisation problem, so we must negate the objective function and maximise $-z = -3x_1 - 2x_2 + x_3$, resulting in the matrix

$$\left(\begin{array}{cccccc|c} 2 & 3 & 1 & 1 & 0 & 0 & 10 \\ 1 & -2 & 3 & 0 & 1 & 0 & 5 \\ 1 & 4 & 2 & 0 & 0 & 1 & 12 \\ 3 & 2 & -1 & 0 & 0 & 0 & 0 \end{array}\right).$$

The following row operations can be applied to change the basic variables to x_2, x_3, x_4:

$$\left(\begin{array}{ccccc|c} 2 & 1 & 1 & 0 & 0 & 70 \\ 1 & 1 & 0 & 1 & 0 & 40 \\ 1 & 3 & 0 & 0 & 1 & 90 \\ -40 & -60 & 0 & 0 & 0 & 0 \end{array}\right) \equiv \left(\begin{array}{ccccc|c} 2 & 1 & 1 & 0 & 0 & 70 \\ 1 & 1 & 0 & 1 & 0 & 40 \\ 1/3 & 1 & 0 & 0 & 1/3 & 30 \\ -40 & -60 & 0 & 0 & 0 & 0 \end{array}\right) \begin{array}{l} \\ \\ R_3' = R_3/3 \\ \\ \end{array}$$

$$\equiv \left(\begin{array}{ccccc|c} 5/3 & 0 & 1 & 0 & -1/3 & 40 \\ 2/3 & 0 & 0 & 1 & -1/3 & 10 \\ 1/3 & 1 & 0 & 0 & 1/3 & 30 \\ -20 & 0 & 0 & 0 & 20 & 1800 \end{array}\right) \begin{array}{l} R_1 = R_1 - R_3 \\ R_2 = R_2 - R_3 \\ \\ R_4 = R_4 + 60R_3 \end{array}$$

Setting the non-basic variables x_1, x_5 to zero we obtain the basic solution

$$x_1 = 0,\ x_2 = 30,\ x_3 = 40,\ x_4 = 10,\ x_5 = 0;\ z = 1800.$$

This basic solution corresponds to the corner point $(x_1, x_2) = (0, 30)$ of the feasible region.

As you might expect by now, basic solutions correspond to corner points (see [3, Theorem 15.2]).

Theorem 12

Let $\Omega = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geqslant \mathbf{0}\}$, where $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{A} \in \mathbb{R}^{m\times n}$, $m < n$, and $\text{Rank}(\mathbf{A}) = m$, be a convex set of feasible solutions. Then $\mathbf{x}$ is a corner point of Ω if and only if $\mathbf{x}$ is a basic solution to $\mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geqslant \mathbf{0}$.

Combining this observation with the corner point theorem, Theorem 11, we see that to solve a linear programming problem we only need to examine the basic solutions of the constraint set.

This is the basic idea of the simplex algorithm, which we cover next: start at one corner of the feasible region, and move from corner to corner until you get to the optimal solution.