40.002 OPTIMIZATION

An Introduction to Convex Optimization

Michael Hoon

March 26, 2024

Contents

1 Introduction to Linear Programming

An optimization problem is defined by:

- **Decision variables**: elements under the control of the decision maker
- **A (single) objective function**: a function of the decision variables that we want to optimize, corresponding to a criterion for measuring maximize
- **Constraints**: restrictions that define which values of the decision variables are allowed.

We want to find the **minimum** or **maximum** of a function of one or many variables subject to a set of **constraints**:

$$\begin{aligned} &\min \ f(x_1, \dots x_n) \\ &\ni (x_1, \dots x_n) \in \chi \subseteq \mathbb{R}^n \end{aligned} \tag{1}$$

where the decision variables are vectors $x_1, \dots x_n$, the objective function is $f(x_1, \dots x_n)$ and the constraints are defined by the set $\chi \subseteq \mathbb{R}^n$. A vector $\mathbf{x}^*$ is called *optimal*, or a *solution* of the problem, if it has the **smallest objective value** among all vectors that satisfy the constraints.

1.1 Standard Form

A linear program is a class of optimisation problem in which the objective and all constraint functions are **linear**. For a minimisation problem,

$$\begin{aligned} &\min \ \mathbf{c}^\top \mathbf{x} \\ &\ni \mathbf{A}\mathbf{x} \geq \mathbf{b}, \text{ and } \mathbf{x} \geq 0 \end{aligned} \tag{2}$$

and for maximisation problems,

$$\begin{aligned} &\max \ \mathbf{c}^\top \mathbf{x} \\ &\ni \mathbf{A}\mathbf{x} \leq \mathbf{b}, \text{ and } \mathbf{x} \geq 0 \end{aligned} \tag{3}$$

where the decision vector is $\mathbf{x}$ (n variables), linear objective function: $f(\mathbf{x}) = \mathbf{c}^\top \mathbf{x} = \sum_{i=1}^n c_i x_i$, and the linear constraints are $\chi = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}\}$ (m constraints) [1]. Note that matrix $\mathbf{A}_{(m \times n)}$ is of $m \times n$ dimension.

1.1.1 Inequality Transformations

We have matrix $\mathbf{A}$ given by:

$$\mathbf{A} = \begin{pmatrix} - & \mathbf{a_1}^\top & - \\ \vdots & \vdots & \vdots \\ - & \mathbf{a_m}^\top & - \end{pmatrix}$$

- An equality constraint $\mathbf{a_i}^\top \mathbf{x} = b_i$ is equivalent to two equality constraints $\mathbf{a_i}^\top \mathbf{x} \leq b_i$ and $\mathbf{a_i}^\top \mathbf{x} \geq b_i$
- An inequality constraint $\mathbf{a_i}^\top \mathbf{x} \leq b_i$ is equivalent to the inequality constraint $-\mathbf{a_i}^\top \mathbf{x} \geq -b_i$ (Note the negatives applied to both sides of the inequality).

[1] Note: vector inequalities are interpreted componentwise.

- Constraints such as $x_j \geq 0,\ x_j \leq 0$ can be expressed in the form $\mathbf{a_i}^\top \mathbf{x} \geq b_i$ by appropriately choosing $\mathbf{a}_i,\ b_i$.

Note that there is no simple analytic formula for the solution of a linear program, but there are a variety of effective methods for solving them, including Dantzig's simplex method, and the more recent interior-point methods. We cannot give the exact number of arithmetic operations required to solve a linear program, but we can establish rigorous bounds on the number of operations required to solve a linear program using an interior-point method (in practice, this is of the order n^2m, assuming $m \geq n$).

1.1.2 Terminology

Definition 1.1. We now introduce some terminology for geometric linear programming:

- A **linear function** $f : \mathbb{R}^n \to \mathbb{R}$ is a function of the form:

$$f(x_1, \dots, x_n) = \sum_{i=1}^{n} a_i x_i, \quad a_i \in \mathbb{R}$$

- A **hyperplane** in $\mathbb{R}^n$ is the set of points satisfying a single linear **equation**:

$$a_1 x_1 + \cdots + a_n x_n = b, \quad a_n \in \mathbb{R}$$

- A **halfspace** in $\mathbb{R}^n$ is the set of points satisfying a single linear **constraint**:

$$a_1 x_1 + \cdots + a_n x_n \geq b, \quad a_n, b \in \mathbb{R}$$

  A halfspace is a **convex set**.

- An LP is **bounded** if there is some value Z such that $\mathbf{c}^\top \mathbf{x} \leq Z$.

- A **polyhedron** is a set that can be described by a finite number of halfspaces. A **polytope** is a **bounded** polyhedron. The polytope of an LP is **convex**, since it is the intersection of halfspaces (which are convex).

- An assignment of values to the decision variables is a **feasible solution** if it **satisfies all the constraints** (infeasible otherwise). The set of all feasible solutions is the **feasible region.**

- An **optimal solution** is a feasible solution that achieves the **best possible objective function value**. For a minimisation problem, x^* is optimal **iff** $\mathbf{c}^\top \mathbf{x}^\top \leq \mathbf{c}^\top \mathbf{x}$ for all feasible $\mathbf{x}$.

- We call $\mathbf{c}^\top x^*$ the **optimal objective value**.

- $\forall K \in \mathbb{R}$ we can find a feasible solution $\mathbf{x}$ such that $\mathbf{c}^\top \mathbf{x} \leq K$, then the linear program in **minimisation** form has **unbounded** cost. The optimum cost is then $-\infty$. In this case, we can find a feasible $\mathbf{x}$ and direction $\mathbf{d}$ such that $\mathbf{x} + t\mathbf{d}$ is feasible $\forall t \geq 0$ and $\mathbf{c}^\top d < 0$.

For every linear program, we know that one of the following cases must hold:

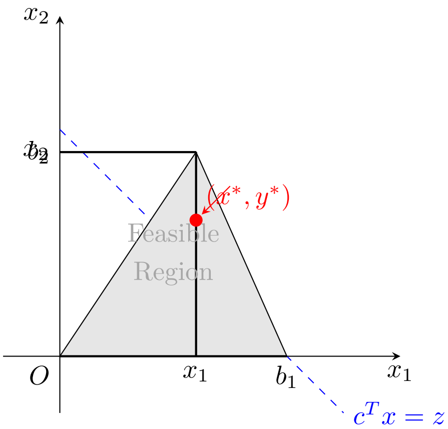
1. The LP is infeasible. There is no value of x that satisfies the constraints.
2. The LP has an optimal solution
3. The LP is unbounded.

Mathematically, this follows from the fact that if the LP is **feasible and bounded**, then it is a closed and bounded subset of $\mathbb{R}^n$ and hence has a maximum point.

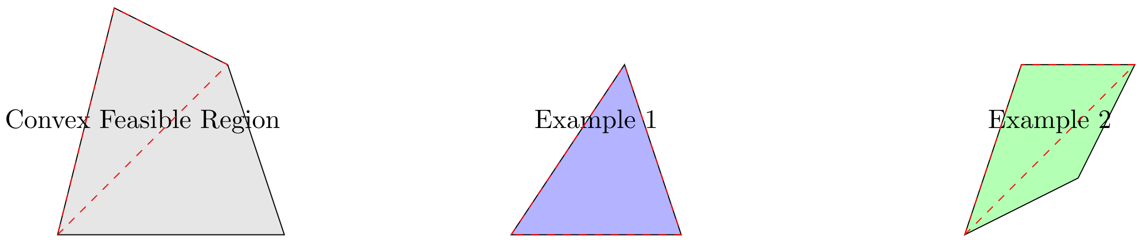
1.2 Geometric Definition

In a simple two-dimensional space with the equation $x_1 + x_2 = z$, this function can be represented by a line. The decision variables are x_1 ad x_2, and this line represents all possible combinations of x_1 and x_2 that yield the same objective value z.

Each constraint is a linear inequality, which creates a boundary in the solution space. The feasible region is now the polygon formed by the intersection of all these constraint boundaries.



The feasible region is often convex, meaning that if points A and B are inside the region, the line segment connecting A and B are also inside the region. Some examples are shown below:



1.3 Graphical Approach

Solving the LP via a graphical approach involves drawing the halfspaces defined by the constraints, as well as the iso-lines defined by the optimisation problem.

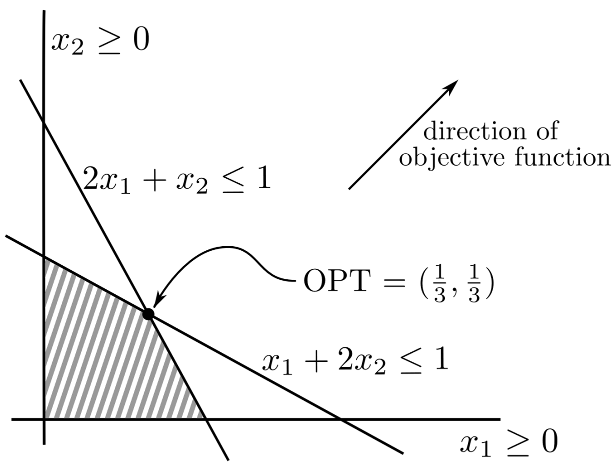


Figure 1: A linear Program in 2 dimensions

After determining the bounded-ness of the LP problem, shade the feasible region (polytope) defined by the constraints. Points within this feasible region satisfy all constraints, and is a **convex polygon**.

Note that the feasible region for an LP may be *empty, a single point, or infinite (unbounded)*. Our goal now is to find a point in the feasible region that optimises this objective. Since there are an infinite number of possible points within the polytope, we need to reduce this space.

Theorem 1.2. The maximum point for an LP is always achieved at one of the vertices of a polytope. In general, if there are n dimensions (variables), a vertex may occur wherever n (linearly independent) hyperplanes (i.e. constraints) intersect.

Recall in linear algebra that if you have n linearly independent equations and n variables, there is only one optimal solution (and if they are not linearly independent - you have infinitely many solutions). So in a system with m constraints and n variables, there are $\binom{m}{n} = O(m^n)$ vertices.

Now, to determine which of the vertices gives the maximum objective value, we can substitute the variables into the objective function and compare the final values [2]

1.3.1 Geometric Intuition

The objective function gives the optimisation direction, and the goal is to find the feasible point that is furthest in this direction.

1.4 Convexity

When optimisation is concerned, we equate "convex" with "nice", and "non-convex" with "nasty".

1.4.1 Convex Sets

Theorem 1.3. The feasible region (Polytope) of an LP is **convex**.

Intuitively, a subset $C \subseteq \mathbb{R}^n$ is convex if it is "filled in", meaning that it contains all line segments between its points (if you draw a line segment between two points of the region, the line segment itself must be in the region).

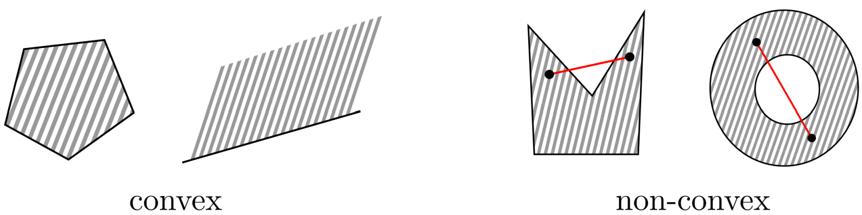


Figure 2: Examples of convex and non-convex sets

Mathematically, a set $\chi \subseteq \mathbb{R}^n$ is convex if

$$\lambda \mathbf{x} + (1-\lambda)\mathbf{y} \in \chi, \ \forall \mathbf{x}, \mathbf{y} \in \chi, \ \forall \lambda \in [0, 1]$$

As λ ranges from 0 to 1, it traces out the line segment from $\mathbf{y}$ to $\mathbf{x}$.

[2]Note that at each vertex, if the iso-line falls within the polytope, then it is not a maximum.

1.4.2 Convex Functions

We define a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to be *convex* if and only if the region above its graph is a convex set.

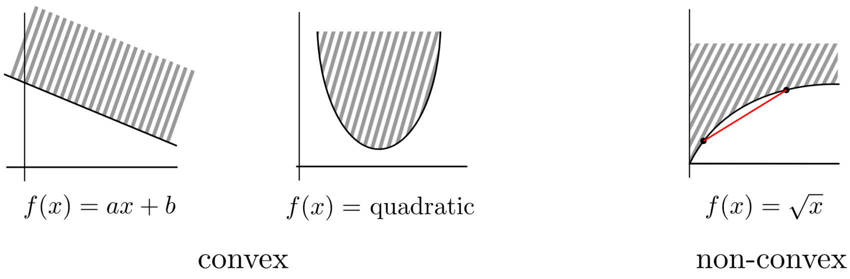


Figure 3: Examples of convex and non-convex functions

Equivalently, a convex function is one where all "chords" of its graph lie above the graph. Mathematically,

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y}), \ \forall\, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \ \forall \lambda \in [0,1] \tag{4}$$

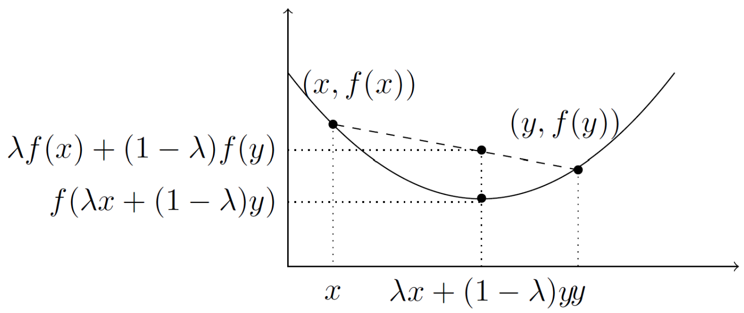


Figure 4: Visualisation of a convex function

That is, for points $\mathbf{x}$ and $\mathbf{y}$, if you take the average of $\mathbf{x}$ and $\mathbf{y}$, and then apply f, you'll get a smaller number than if you first apply f to $\mathbf{x}$ and $\mathbf{y}$, and then average the results. Likewise, a function f is **concave** if $-f$ is convex.

1.4.3 Why Convexity Helps

Consider the case where the feasible region or the objective function is not convex. With a non-convex feasible region, there can be "locally optimal" feasible points that are not globally optimal, even with a linear objective function. The same problem arises with a non-convex objective function, even when the feasible region is just the real line. When both the objective function and feasible region are convex, this cannot happen - **all local optima are also global optima** (which makes optimisation easier).

Theorem 1.4. Let χ be a convex set and $f(x)$ be a convex function. Then:

$$\begin{aligned} &\min \ f(x) \\ &\text{s.t.} \ \ x \in \chi \subseteq \mathbb{R}^n \end{aligned}$$

is a convex optimisation problem. A key property of such a convex optimisation problem that is a **local minimum is always the global minimum**.

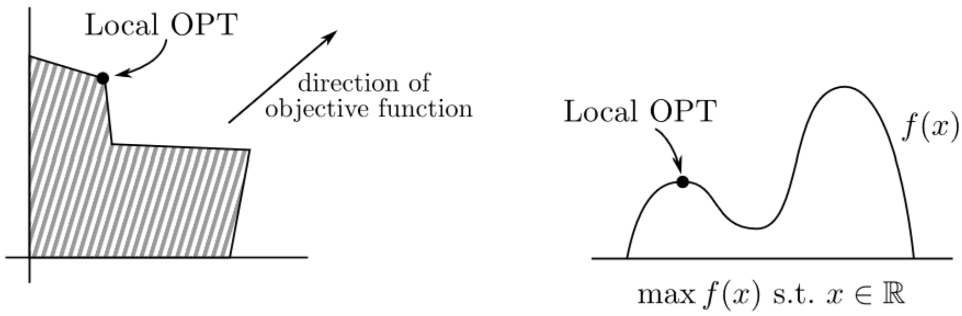


Figure 5: Non-convexity and local optima. (Left) A linear (i.e. convex) objective function with a non-convex feasible region. (Right) A non-convex objective function over a convex feasible region (the real line).

1.4.4 First-Order Characterisation

Suppose a function $f : \mathbb{R}^n \to \mathbb{R}$ is differentiable [3]. Then f is convex if and only if $D(f)$ is a convex set and for all $x, y \in D(f)$:

$$f(y) \geq f(x) + \nabla_x f(x)^\top (y - x) \tag{5}$$

The function on the right is called the **first order approximation** to f at the point x.

Theorem 1.5. The first order condition for convexity says that f is convex if and only if the tangent line is a global underestimator of the function f. i.e. if we draw a tangent line at any point (refer to Figure **??**), then every point on this line will lie below the corresponding point on f.

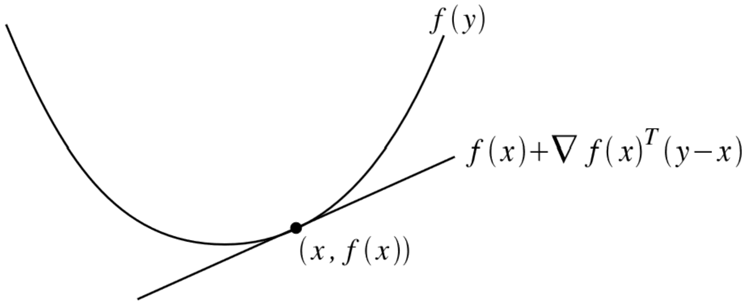


Figure 6: Illustration of first-order condition for convexity

1.4.5 Second-Order Characterisation

Suppose that a function $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable (i.e. the Hessian $\nabla_x^n f(x)$ is defined for all x in the domain of f). Then f is convex if and only if $D(f)$ is a convex set and its Hessian is positive semidefinite (PSD): i.e. for any $x \in D(f)$,

$$\nabla_x^2 f(x) \succeq 0 \tag{6}$$

where $\succeq$ denotes PSD-ness. In one-dimension, this is equivalent to the condition that the second derivative $f''(x)$ always be non-negative.

The Hessian is defined as:

[3] the gradient $\nabla_x f(x)$ exists at all points x in the domain of f. It is defined as $\nabla_x f(x) \in \mathbb{R}^n$, $(\nabla_x f(x))_i = \frac{\partial f(x)}{\partial x_i}$

$$\nabla_x^2 f(x) \in \mathbb{R}^{n\times n}, \quad (\nabla_x^2 f(x))_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j} \tag{7}$$

Theorem 1.6. We can see that f is **strictly convex** if its Hessian is **Positive Definite**, concave if it is **Negative Semidefinite**, and **strictly concave** if it is **Negative Definite**.

1.4.6 Solving Convexity-Related Questions

1.5 Some Matrix Calculus

1.5.1 The Gradient

Definition 1.7. Suppose that $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ is a function that takes as input a matrix A of size $m \times n$ and returns a real value. Then the **gradient** of f (with respect to $A \in \mathbb{R}^{m\times n}$) is the matrix of **partial derivatives**:

$$\nabla_A f(A) \in \mathbb{R}^{m\times n} = \begin{bmatrix} \frac{\partial f(A)}{\partial A_{11}} & \frac{\partial f(A)}{\partial A_{12}} & \cdots & \frac{\partial f(A)}{\partial A_{1n}} \\ \frac{\partial f(A)}{\partial A_{21}} & \frac{\partial f(A)}{\partial A_{22}} & \cdots & \frac{\partial f(A)}{\partial A_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(A)}{\partial A_{m1}} & \frac{\partial f(A)}{\partial A_{m2}} & \cdots & \frac{\partial f(A)}{\partial A_{mn}} \end{bmatrix} \tag{8}$$

Note that the gradient of a function [4] is only defined if the function is real-valued, i.e. if it returns a scalar value.

Some equivalent properties of the gradient from partial derivatives:

1. $\nabla_x(f(x) + g(x)) = \nabla_x f(x) + \nabla_x g(x)$
2. For $t \in \mathbb{R}$, $\nabla_x(tf(x)) = t\nabla_x f(x)$

1.5.2 The Hessian

Definition 1.8. Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is a function that takes a vector in $\mathbb{R}^n$ and returns a real number. Then the **Hessian** with respect to x, written $\nabla_x^2 f(x)$ is the $n \times n$ matrix of partial derivatives:

$$\nabla_x^2 f(x) \in \mathbb{R}^{n\times n} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial f(x)}{\partial x_2 \partial x_1} & \frac{\partial f(x)}{\partial x_2^2} & \cdots & \frac{\partial f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(x)}{\partial x_n \partial x_1} & \frac{\partial f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial f(x)}{\partial x_n^2} \end{bmatrix} \tag{9}$$

Note that the Hessian is always symmetric, since

$$\frac{\partial^2 f(x)}{\partial x_i \partial x_j} = \frac{\partial^2}{\partial x_j \partial x_i}$$

The Hessian is defined only when $f(x)$ is real-valued. Note that for functions of a vector, the gradient of the function is a vector, and we cannot take the gradient of a vector. **Therefore, it is not the case that the Hessian is the Gradient of the Gradient**.

[4] Gradients are a natural extension of partial derivatives to functions of multiple variables.

1.5.3 Definite Matrix & Eigenvalues

Definition 1.9. Let M be an $n \times n$ Hermitian Matrix [a](including Symmetric Matrices [b]). All eigenvalues of M are real, and their sign characterise its definite-ness:

1. M is **Positive Definite** if and only if all of its eigenvalues are **positive**.
2. M is **Positive Semi-Definite** if and only if all its eigenvalues are **non-negative**.
3. M is **Negative Definite** if and only if all of its eigenvalue are **negative**.
4. M is **Negative Semi-Definite** if and only if all if its eigenvalues are **non-positive**.
5. M is **indefinite** if and only if it has both positive and negative eigenvalues.

[a] A complex square matrix that is equal to its own conjugate transpose: $A = \overline{A^\top}$.
[b] A square matrix that is equal to its transpose.

2 Simplex Method

2.1 The True Standard Form

Previously, we know that a linear program can take either a maximisation of minimisation form, depending on the context. The constraints thus can either be inequalities or equalities. Some variables might be unrestricted in sign, while others might be restricted to be non-negative.

A linear program is said to be in *standard form* if the following hold:

1. It is a minimisation program.
2. There are only equalities (no inequalities) and
3. All variables are restricted to be non-negative.

In matrix form, we have

$$\begin{aligned} \min \quad & c^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{aligned} \tag{10}$$

where n is the number of decision variables and m is the number of equality constraints and $\mathbf{0}$ is a vector of zeros.

2.1.1 Transformation Tricks

We have a few transformation tricks to convert any LP into the true standard form:

1. Eliminate non-positive and free variables (unrestricted in sign):

$$\begin{aligned} & x_j \leq 0 \rightarrow \text{Let } \hat{x}_j = -x_j \text{ and set } \hat{x}_j \geq 0 \\ & x_j \text{is free } \rightarrow \text{Let } x_j = x_j^+ - x_j^- \text{ and set } x_j^+, x_j^- \geq 0 \end{aligned}$$

2. If we have an inequality constraint $a_{i1}x_1 + \cdots + a_{in}x_n \leq b$, then we can transform it into an equality constraint by adding a *slack* variable s_i, restricted to be non-negative:

$$a_{i_1}x_1 + \cdots + a_{in}x_n + s_i = b_i, \quad s_i \geq 0$$

3. Similarly, if we have an inequality constraint $a_{i_1}x_1 + \cdots + a_{in}x_n \geq b_i$ then we can transform it into an equality constraint by adding a *surplus* variable, s_i, restricted to be non-negative:

$$a_{i_1}x_1 + \cdots + a_{in}x_n - s_i = b_i, \quad s_i \geq 0$$

2.2 Active Constraints

1. A constraint is said to be **active** at a point $\mathbf{x}$ if the constraint is satisfied at equality at that point.

2. A set of linear constraints are linearly independent if their coefficient vectors are linearly independent; else they are linearly dependent.

2.3 Optimality Test of LP in Inequality Form

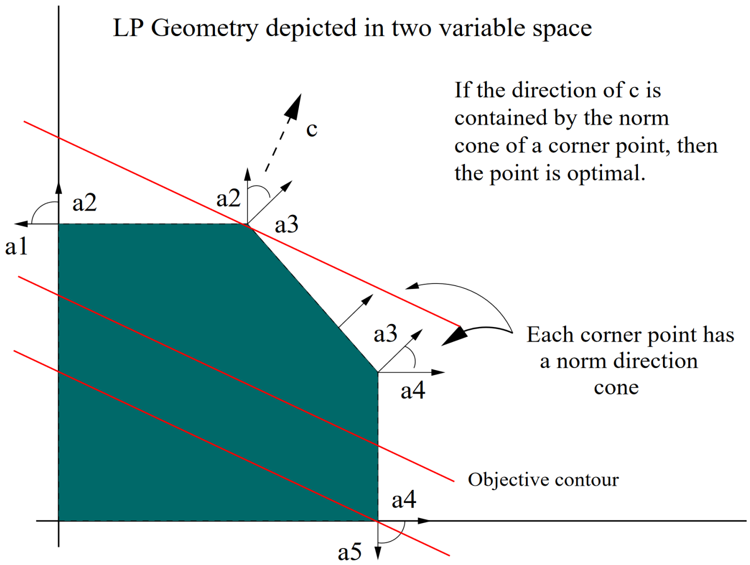


Figure 7: Feasible Region with Objective Contours

Consider an LP with m variables and n linear inequality constraints.

1. A **Corner Point** is an intersection point of the hyperplanes of m linearly-independent inequality constraints.

2. These constraints are called active constraints at the corner solution.

3. Two corner solutions are adjacent if they differ by one active constraint.

Theorem 2.1. For an LP in the standard form, a **Corner Point** is maximal if and only if the objective vector is a conic combination of the normal direction vectors of the m hyperplanes. Refer to Figure 7.

2.4 Some Polyhedron Definitions

Let $P \subseteq \mathbb{R}^n$ be a non-empty polyhedron.

1. A vector $\curvearrowleft \in P$ is an **extreme point** of P if we cannot find two vectors $\mathbf{y}, \mathbf{z} \in P$, both different from $\mathbf{x}$, and a scalar $\lambda \in [0, 1]$, such that $\mathbf{x} = \lambda\mathbf{y} + (1 - \lambda)\mathbf{z}$ (Geometric Definition).

2. A vector $\mathbf{x} \in P$ is a **vertex** of P if there exists some $\mathbf{c}$ such that $\mathbf{x}$ is the unique minimizer of the LP $\min\{\mathbf{c}^\top\mathbf{y} \mid \mathbf{y} \in P\}$ (Optimisation definition)

3. A vector $\mathbf{x} \in P$ is a **basic feasible solution** (BFS) if there exists n linearly independent constraints that are active at $\mathbf{x}$ (including both inequality and equality constraints) (Algebraic definition).

Theorem 2.2. Let $P \subseteq \mathbb{R}^n$ be a nonempty polyhedron and let $x^* \in P$. Then:

$$x^* \text{ is an extreme point } \iff x^* \text{ is a vertex } \iff x^* \text{ is a BFS.}$$

Notably, these 3 are the same definitions but in different form.

2.5 Finding a Basic Feasible Solution

We can "eyeball" a BFS by the following steps:

Consider an LP where:

1. LP is of standard form
2. The right hand side vector $\mathbf{b}$ is non-negative
3. The constraint matrix contains the identity matrix I as a **submatrix** (or a permutation of the identity matrix).

Then, we can easily identify a BFS:

1. Variables associated with I (or permutations of I) are **basic variables**. THey take the value of the right hand side vector. Other variables are nonbasic variables; they take values of zero.
2. Columns of the constraint matrix corresponding to the basic variables form a **basis**; we also refer to the set of basic variables sometimes as a basis.

2.6 The Simplex Method

To use the Simplex Method on an LP, we shall assume the following:

1. The LP is in standard form
2. $b \geq 0$
3. There exists a collection B of m variables called a *basis* such that:
   (a) the submatrix A_B of A consisting of the columns of A corresponding to the variables in B is the $m \times m$ identity matrix
   (b) the cost coefficients corresponding to the variables in B are all equal to 0.

2.7 Slack Variables and Pivoting

Simplex method is an iterative procedure which corresponds, geometrically, to moving from one feasible corner to another until optimal feasible point is located. Slack variables are introduced to ensure corner points are *feasible*, not outside solution region. Algebraically, hopping from one feasible corner point to another corresponds to repeatedly identifying *pivot column*, *pivot row*, and consequently, *pivot element*, in a succession of matrix tableaus. Having identified pivot element, a new tableau is created by *pivoting* (Gauss-Jordan method) on this element.

2.8 Algorithmic Approach

The steps of the simplex algorithm can be carried out using a simplex tableau. We have the LP in standard form:

$$\begin{aligned}
\min \quad & -3x_1 - 2x_2 \\
\text{s.t.} \quad & x_1 - x_2 + x_3 = 1 \\
& 2x_1 + x_2 \qquad + x_4 = 4 \\
& \qquad x_2 \qquad\qquad + x_5 = 2 \\
& x_1, x_2, x_3, x_4, x_5 \geq 0
\end{aligned}$$

Row			x_1	x_2	x_3	x_4	x_5
R_0		0	-3	-2	0	0	0
R_1	x_3	1	1	-1	1	0	0
R_2	x_4	4	2	1	0	1	0
R_3	x_5	2	0	1	0	0	1

Table 1: Simple Tableau

In this example, $B = \{x_4, x_5, x_6\}$. The variables in B are called basic variables while the others are *nonbasic*. The set of nonbasic variables is denoted by $N = \{x_1, x_2, x_3\}$. We now have a matrix $A_B = I$, where it is easier to quickly infer the values of the basic variables given the values of the nonbasic variables[5].

2.8.1 Basic Feasible Solutions (BFS)

Furthermore, we don't need to know the values of the basic variables to evaluate the cost of the solution (there is no guarantee however that the solution is feasible). By setting all nonbasic variables to be 0, the values of the basic variables are just given by the right-hand-sides of the constraints. The simplex method is an iterative method that generates a sequence of BFS (corresponding to different bases) and eventually stops when it has found an optimal basic feasible solution.

At this point, to move to a better solution, we can use the ratio test:

> **Theorem 2.3. Minimum Ratio Test**: When introducing variable x_s into the basis, identify the row that gives the **minimum ratio of left-hand side values** in the tableau to the corresponding x_s coefficient in the chosen column. Compute these ratios only for constraints that have positive coefficients for x_s.

In this case, we have

$$t^* = \min\left(\frac{1}{1}, \frac{4}{2}\right) = 1$$

In particular, x_1 enters the basis and x_3 leaves the basis. Do row operations using this identified row (leaving variable) such that for the chosen column, (entering variable), we **obtain a 1 in the entry where they intersect and 0 otherwise**. Now, after pivoting ($R_0 \to R_0 + 3R_1,\ R_2 \to R_2 - 2R_1$):

[5] Only if the constraint matrix contains the identity matrix as a submatrix (or a permutation of it). Then, the basic variables take the value of the right hand side vector, while the others (nonbasic) are zero.

Row			x_1	x_2	x_3	x_4	x_5
R_0		3	0	-5	3	0	0
R_1	x_1	1	1	-1	0	0	0
R_2	x_4	2	0	3	-2	1	0
R_3	x_5	2	0	1	0	0	1

Table 2: Next Pivot

At this point, we now see:

1. BFS is $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 0, 2, 2)$, identified by the variables on the left of the tableau (x_1, x_4, x_5).
2. Objective function value is -3 (negative of value of top left entry).
3. However, x_2 has negative cost in the top row, so the solution is still non-optimal.
4. Note that the element that we are pivoting on must always end up being 1.

We now apply the minimum ratio test again:

$$t^* = \min\left(\frac{2}{3}, \frac{2}{1}\right) = \frac{2}{3}$$

where x_4 leaves, and x_2 enters. We now perform the row operations ($R_2 \to \frac{R_2}{3},\ R_0 \to R_0 + 5R_2,\ R_1 \to R_1 + R_2,\ R_3 \to R_3 - R_2$). We obtain the next tableau:

Row			x_1	x_2	x_3	x_4	x_5
R_0		19/3	0	0	-1/3	5/3	0
R_1	x_1	5/3	1	0	1/3	1/3	0
R_2	x_4	2/3	0	1	-2/3	1/3	0
R_3	x_5	4/3	0	0	2/3	-1/3	1

Table 3: Simplex Tableau after pivoting again

Applying the minimum ratio test again,

$$t^* = \min\left(\frac{5/3}{1/3}, \frac{4/3}{2/3}\right) = 2$$

Since we still have negative cost in x_3, x_5 leaves the basis, while x_3 enters. We perform the row operations: ($R_0 \to R_0 + \frac{R_3}{2},\ R_1 \to R_1 - \frac{R_3}{2},\ R_2 \to R_2 + R_3,\ R_3 \to \frac{R_3}{2/3}$).

Row			x_1	x_2	x_3	x_4	x_5
R_0		7	0	0	0	3/2	1/2
R_1	x_1	1	1	0	0	1/2	-1/2
R_2	x_4	2	0	1	0	0	1
R_3	x_5	2	0	0	1	-1/2	3/2

Table 4: Final Simplex Tableau

At this point, all entries are nonnegative in R_0 for columns corresponding to x_1 to x_5. **The simplex algorithm terminates**. The optimal solution is now $x_1 = 1, x_2 = 2$, with optimal objective value being $z = -7$.

2.9 Simplex Method: Matrix Form

Consider an LP in standard form, with n variables and m equality constraints, and the rows of $\mathbf{A}$ are linearly independent ($m \leq n$). Index the $m \times n$ matrix $\mathbf{A}$ using column vectors:

$$\mathbf{A} = \begin{pmatrix} | & & | \\ \mathbf{A}_1 & \dots & \mathbf{A}_n \\ | & & | \end{pmatrix}$$

We start with a basis matrix $\mathbf{B}$ where $B_1, \dots, B_m$ are the indices of the basic variables and $\mathbf{B} = (\mathbf{A}_{B_1}, \dots, \mathbf{A}_{B_m})$. $\mathbf{B}$ is of size $m \times m$ and invertible, partitioning the matrix as $\mathbf{A} = \begin{pmatrix} \mathbf{B} & \mathbf{N} \end{pmatrix}$ (basic and nonbasic), the vector $\mathbf{x}$ to $\mathbf{x}_B$ and $\mathbf{x}_N$ and the cost vector $\mathbf{c}$ to $\mathbf{c}_B$ and $\mathbf{c}_N$ (the cost vector is the coefficients of the objective function).

To find an associated solution, we set $\mathbf{x}_N = \mathbf{0}$, and $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}$. If $\mathbf{B}^{-1}\mathbf{b} \geq 0$, this is a BFS. The objective value at the BFS is $z = \mathbf{c}_B^\top \mathbf{B}^{-1}\mathbf{b}$. See that we now have

$$\mathbf{A} = \begin{pmatrix} \mathbf{B} & \mathbf{N} \end{pmatrix}$$

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} \begin{matrix} m \\ \\ n-m \end{matrix} \qquad \mathbf{c} = \begin{pmatrix} \mathbf{c}_B \\ \mathbf{c}_N \end{pmatrix}$$

2.9.1 Objective Value & Solution

By setting the variables $\mathbf{x}_N$ of $\mathbf{x}$ corresponding to the remaining columns of $\mathbf{A}$ equal to 0, we obtain a solution $\mathbf{x}$ of $\mathbf{A}\mathbf{x} = \mathbf{b}$:

$$\begin{pmatrix} \mathbf{B} & \mathbf{N} \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \mathbf{b} \iff \mathbf{B}\mathbf{x}_B + \mathbf{N}\cancelto{0}{\mathbf{x}_N} = \mathbf{b}$$

We now have $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}$.

Theorem 2.4. When $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b} \geq 0$, the solution is a BFS. The objective value at this BFS is:

$$\mathbf{c}^\top \mathbf{x} = \begin{pmatrix} \mathbf{c}_B^\top & \mathbf{c}_N^\top \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \mathbf{b} \implies \mathbf{c}_B^\top \mathbf{x}_B + \mathbf{c}_N^\top \cancelto{0}{\mathbf{x}_N} = \mathbf{c}_B^\top \mathbf{B}^{-1}\mathbf{b}$$

Suppose that we want to move from $\mathbf{x}$ to a new vector $\mathbf{x} + t\mathbf{d}$ by selecting a nonbasic variable x_j (at zero level) and increasing it while keeping all other nonbasic variables at 0. The vector $\mathbf{d}$ is the direction vector.

We want to ensure that $\mathbf{x} + t\mathbf{d}$ is feasible for some $t > 0$. Thus, $\mathbf{A}(\mathbf{x} + t\mathbf{d}) = \mathbf{d} \implies \mathbf{A}\mathbf{d} = 0$.

$$\begin{aligned} \mathbf{A}\mathbf{d} = \mathbf{0} &\iff \mathbf{B}\mathbf{d}_B + \mathbf{N}\mathbf{d}_N = \mathbf{0} \\ &\iff \mathbf{d}_B = -\mathbf{B}^{-1}\mathbf{N}\mathbf{d}_N \end{aligned}$$

When $\mathbf{d}_N$ has 1 for the jth entry and 0 otherwise, this gives us $\mathbf{d}_B = -\mathbf{B}^{-1}\mathbf{A}_j$, which is the direction vector for the basic variables.

2.9.2 Minimum Ratio Test

We want to ensure that the direction value is in the **positive (largest) direction** (since we want to move to the optimum point). To set a bound on the value of t in $\mathbf{x} + t\mathbf{d} \geq 0$:

1. If $\mathbf{d} \geq 0$, then $\mathbf{x} + t\mathbf{d} \geq 0 \ \forall t \geq 0$, and we can choose $t^* = \infty$ (objective is unbounded).
2. If $d_i < 0$ for some i, then $x_i + td_i \geq \mathbf{0}$ gives $t \leq -\frac{x_1}{d_i}$

Definition 2.5. We have the Minimum Ratio Test (MRT)

$$t^* = \min_{\{i | d_i < 0\}} \left\{ -\frac{x_i}{d_i} \right\}$$

We only need to consider the basic variables while performing the ratio test, as for nonbasic variables we have $\mathbf{x}_N = \mathbf{0}$.

2.9.3 Reduced Cost

After moving in the specific direction of the largest change in cost, we want to calculate this rate of change in cost. This is given by:

$$\begin{aligned}
&\frac{\mathbf{c}^\top(\mathbf{x} + t\mathbf{d} - \mathbf{c}^\top\mathbf{x})}{t} = \mathbf{c}^\top\mathbf{d} \\
\implies &\mathbf{c}^\top\mathbf{d} = \mathbf{c}_B^\top\mathbf{d}_B + c_j(1) = c_j - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{A}_j
\end{aligned}$$

Definition 2.6. In LP, reduced cost (opportunity cost) is the amount by which an objective function coefficient would have to improve before it would be possible for a corresponding variable to assume a positive value in the optimal solution (for the variables to become non-basic and enter the solution with a positive value). We define the reduced cost of each variable x_j as:

$$\bar{c}_j = c_j - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{A}_j$$

and the vector of reduced costs as:

$$\bar{\mathbf{c}}^\top = \mathbf{c}^\top - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{A}$$

For $\mathbf{A} = \begin{pmatrix}\mathbf{B} & \mathbf{N}\end{pmatrix}$, we can write $\mathbf{c} = \begin{pmatrix}\mathbf{c}_B & \mathbf{c}_N\end{pmatrix}$, then we have

$$\bar{\mathbf{c}}^\top = \mathbf{c}_B^\top - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{B} = \mathbf{0}$$

Geometrically, the reduced cost vector represents the direction along which the objective function **increases most rapidly** at the current vertex of the feasible region. When the reduced cost is zero, the **current vertex is the most optimal**, any further movements will only keep the objective value constant or decrease it.

Theorem 2.7. Consider a BFS $\mathbf{x}$ associated with a basis $\mathbf{B}$ and reduced cost vector $\bar{\mathbf{c}}$. If $\bar{\mathbf{c}}^\top \geq 0$, then $\mathbf{x}$ is optimal.

Proof.
Consider any feasible solution $\mathbf{y}$. Then $\mathbf{A}\mathbf{y} = \mathbf{b},\ \mathbf{y} \geq \mathbf{0}$. Consider the direction vector $\mathbf{d} = \mathbf{y} - \mathbf{x}$. The change in cost from $\mathbf{x}$ to $\mathbf{y}$ is given by:

$$\begin{aligned}
\mathbf{c}^\top(\mathbf{y} - \mathbf{x}) &= \mathbf{c}^\top\mathbf{d} \\
&= \mathbf{c}_B^\top\mathbf{d}_B + \mathbf{c}_N^\top\mathbf{d}_N \\
&= \mathbf{c}_B^\top(-\mathbf{B}^{-1}\mathbf{N}\mathbf{d}_N) + \mathbf{c}^\top\mathbf{d}_N \\
&= \underbrace{(\mathbf{c}_N^\top - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{N})}_{\text{nonbasic reduced cost vector}}\mathbf{d}_N \\
&= \sum_{j\in\mathbf{N}} \bar{\mathbf{c}}_j^\top\mathbf{d}_j
\end{aligned}$$

For any nonbasic variable, $x_j = 0$ and since $y_j \geq 0$, we have $d_j = y_j - x_j \geq 0$. Since $\bar{c}_i \geq 0$ by assumption, then $\bar{c}_j dJ \geq 0$ for each nonbasic variable. Hence, we have from above, $\sum_{j\in\mathbf{N}} \bar{c}_j d_j \geq 0 \implies \mathbf{c}^\top(\mathbf{y} - \mathbf{x}) \geq 0 \iff \mathbf{c}^\top\mathbf{y} \geq \mathbf{c}^\top\mathbf{x}$. Since $\mathbf{y}$ was an arbitrary feasible solution, then $\mathbf{x}$ is optimal. □

Theorem 2.8. A basis matrix $\mathbf{B}$ is optimal if:

1. $\mathbf{B}^{-1}\mathbf{b} \geq 0$
2. $\bar{\mathbf{c}}^\top = \mathbf{c}^\top - \mathbf{c}_B^\top\mathbf{B}^{-1}\mathbf{A} \geq \mathbf{0}^\top$

The Simplex Tableau at each step is given in Table 5 below.

$-\mathbf{c}_B^\top \mathbf{B}^{-1}\mathbf{b}$	$\mathbf{c}^\top - \mathbf{c}_B^\top \mathbf{B}^{-1}\mathbf{A}$
$\mathbf{B}^{-1}\mathbf{b}$	$\mathbf{B}^{-1}\mathbf{A}$

Table 5: Simplex Tableau in Matrix Form

Similar to Section 2.8, we perform the following steps:

1. Choose variable with negative reduced cost to enter the basis
2. Perform the MRT to choose the leaving variable
3. Perform row operations to make the corresponding vector a unit vector and make the reduced cost for the variable to be zero.
4. Repeat until all reduced costs are zero.

3 Degeneracy

At a basic solution, by definition, we must have n linearly independent active constraints. In the case that there are more than n active constraints, we have a **degenerate** basic solution.

Definition 3.1. A basic solution $\mathbf{x} \in \mathbb{R}^n$ is said to be **degenerate** if more than n of the constraints are active at $\mathbf{x}$.

Geometrically, in 2 dimensions, a degenerate basic solution is at the intersection of three or more lines, in 3 dimensions, it is the intersection of 4 or more planes, refer to the Figure below.

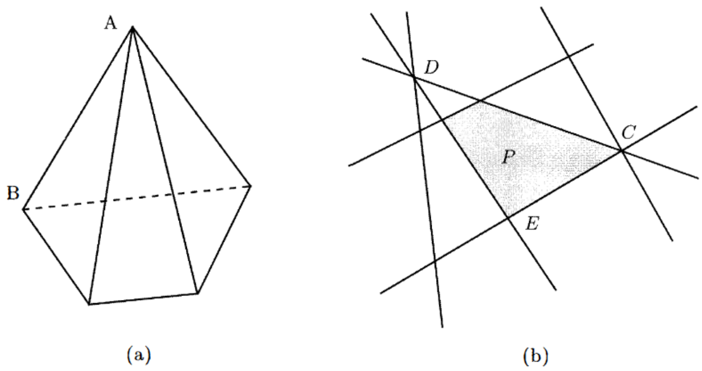


Figure 8: Points A and C are *degenerate basic feasible solutions*. Points B and E are *nondegenerate basic feasible solutions*. Point D is a *degenerate basic solution*.

3.1 Degeneracy in Standard Form Polyhedra

At a basic solution of a polyhedron in standard form, the m equality constraints are always active. Therefore, having more than n active constraints is the same as having more than $n-m$ variables at zero level.

Definition 3.2. Consider the standard form polyhedron $P = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{A}\mathbf{x} = \mathbf{b},\ x \geq \mathbf{0}\}$, and let $\mathbf{x}$ be a basic solution. Let m be the number of rows of $\mathbf{A}$. The vector $\mathbf{x}$ is a **degenerate** basic solution if more than $n-m$ of the *components of* $\mathbf{x}$ *are zero.*

3.2 Pivot Rules

The smallest subscript pivoting rule (Bland's Rule) is as follows:

Theorem 3.3.
1. Among variables with negative reduced cost, choose the one with the smallest reduced cost $\bar{c}_j < 0$ (in Row R_0)
2. Among variables with negative reduced cost, choose the one with the largest decrease in cost $t^*|\bar{c}_j|$ (Minimum Ratio Test).
3. Among variables with negative reduced cost, choose the one with the smallest index j with reduced cost $\bar{c}_j < 0$ (leftmost column).
4. Among variables eligible to exit basis (tied in MRT), choose the one with the smallest index (topmost row).

Under this pivoting rule, it is known that cycling never occurs, and the simplex method is **guaranteed to terminate** after a finite number of iterataions.

3.3 Unbounded-ness

Consider an LP in standard tableau form. If there are no coefficients which are positive (i.e. no solution for minimum ratio test), Then we have an unbounded problem.

Definition 3.4. If there exists a nonbasic variable x_j with negative reduced cost such that x_j appears with nonpositive coefficients on all the remaining rows in that column, then the problem is unbounded. i.e., we have the distance travelled (from MRT) being $t^* \in \{(-\infty, 0], +\infty\}$.

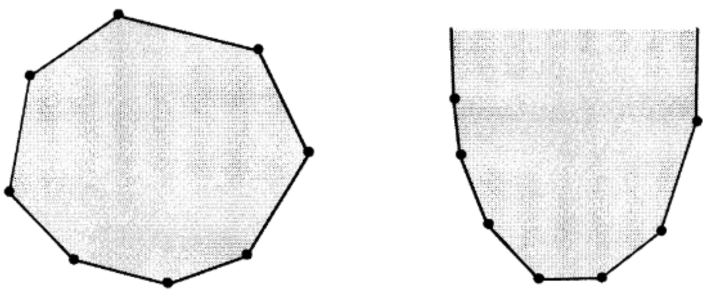


Figure 9: (a) A bounded polyhedron, (b) An unbounded polyhedron

3.3.1 Stopping Criterion and Finite Termination

Consider the solution of an LP in minimisation form with the simplex method. There are a few cases:

1. All variables have nonnegative reduced costs: the **current BFS is optimal**, and algorithm terminates.
2. There exists a variable with negative reduced cost that appears with all nonpositive coefficients in the column (MRT): **the problem is unbounded**, algorithm terminates.
3. All variables with negative reduced costs appear with at least one positive coefficient in the constraints: **the current BFS can be improved**, algorithm continues, change basis.

Theorem 3.5. Assume that no set of basic variables ever repeats (simplex tableau are different for every BFS visited by the algorithm). Then the simplex algorithm terminates after a finite number of iterations.

4 Two-Phase Simplex Method

In the even that an initial basis is not apparent and it is not immediately clear how to start the simplex algorithm, we employ the two-phase simplex algorithm.

4.1 Phase I

Phase I involves transforming a linear program into the matrix form below, and obtaining an initial basis. Suppose our constraints are of the form $\mathbf{Ax} \leq \mathbf{b}$, where $\mathbf{b} \geq 0$. We add slack variables $\mathbf{s} \geq 0$ and rewrite the constraint as $\mathbf{Ax} + \mathbf{s} = \mathbf{b}$. We set $(\mathbf{x}, \mathbf{s}) = (\mathbf{0}, \mathbf{b})$.

Given constraints of the form $\mathbf{Ax} = \mathbf{b},\ \mathbf{x} \geq 0$,

1. Ensure $\mathbf{b} \geq \mathbf{0}$ (multiplying constraints by -1 and ensure $\mathbf{x}_i \geq 0$).
2. Solve the following LP with auxiliary variables $\mathbf{y}$:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} y_i \\ \text{s.t.} \quad & \mathbf{Ax} + \mathbf{y} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0},\ \mathbf{y} \geq \mathbf{0} \end{aligned} \tag{11}$$

3. Start with the BFS: $(\mathbf{x}, \mathbf{y}) = (\mathbf{0}, \mathbf{b})$.
4. Apply the simplex method. If optimal cost is strictly positive, then original LP is infeasible.
5. Else cost is 0 and original LP is feasible. If no artificial variable is in the basis, it gives us a BFS for original problem. If there are, we drive them out by introducing original variables into the basis.

4.2 Phase II

Start with BFS from Phase I and then solve using simplex method. Consider Phase I as the *side quest* before starting on the *main quest* in Phase II.

Consider a standard form LP in phase 1 from Equation 11. For LP to be feasible, we need the $\mathbf{y}$'s to be $\mathbf{0}$, if not we will have $\mathbf{Ax} \leq \mathbf{b}$, then the original LP will be infeasible.

5 Multiple Optimal

Geometrically, if we have a convex set, there may be a case that there are multiple optimal solutions, i.e. when the solutions lie in a line segment in 2 dimensions.

6 Duality Theory

Duality theory deals with the relation between the primal and the dual problems, and is the most important and useful structural property of linear programs.

6.1 Duality in Canonical Form

Given an LP in canonical form, which we call the Primal (P) :

$$\begin{aligned} \min \quad & \mathbf{c}^\top \mathbf{x} & \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} & (m) \\ & \mathbf{x} \geq \mathbf{0} & (n) \end{aligned}$$

Let $\mathbf{x}^*$ be an optimal solution. We introduce a *relaxed* problem in which the constraint $\mathbf{Ax} = \mathbf{b}$ is replaced by a penalty $\mathbf{p}^\top(\mathbf{b} - \mathbf{Ax})$, where $\mathbf{p}$ is a price vector of the same dimension as $\mathbf{b}$ (penalised for violating equality constraints), and $\mathbf{p} \in \mathbb{R}^m$. Let $g(\mathbf{p})$ be the optimal cost for the relaxed problem, as a function of the price vector $\mathbf{p}$. The relaxed problem allows for more options than in (P), and we expect $g(\mathbf{p})$ to be no larger than the optimal primal cost $\mathbf{c}^\top\mathbf{x}^*$.

$$\begin{aligned} g(\mathbf{p}) &= \min_{\mathbf{x} \geq \mathbf{0}} \left[\mathbf{c}^\top \mathbf{x} + \mathbf{p}^\top(\mathbf{b} - \mathbf{Ax})\right] & (12) \\ &\leq \mathbf{c}^\top \mathbf{x}^* + \mathbf{p}^\top(\mathbf{b} - \mathbf{Ax}^*) \quad [\text{since } x^* \geq \mathbf{0} \text{ and is feasible in the definition of } g(\mathbf{p})] & (13) \\ &= \mathbf{c}^\top \mathbf{x}^* \quad [\text{since } \mathbf{Ax}^* = \mathbf{b}] & (14) \end{aligned}$$

Theorem 6.1. The dual of the dual is the primal.

6.2 Weak Duality

Assume that we have a primal LP in standard form:

$$\begin{aligned} \min \quad & \mathbf{c}^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

and the dual LP:

$$\begin{aligned} \max \quad & \mathbf{p}^\top \mathbf{b} \\ \text{s.t.} \quad & \mathbf{p}^\top \mathbf{A} \leq \mathbf{c} \end{aligned}$$

Theorem 6.2. If $\mathbf{x}$ is a feasible solution to the primal problem (minimisation) and $\mathbf{p}$ is a feasible solution to the dual problem (maximisation), then

$$\mathbf{c}^\top \mathbf{x} \geq \mathbf{p}^\top \mathbf{b} \tag{15}$$

Proof. $\mathbf{p}^\top \mathbf{b} = \mathbf{p}^\top(\mathbf{Ax}) = \left(\mathbf{p}^\top \mathbf{A}\right)\mathbf{x} \leq \mathbf{c}^\top \mathbf{x}$ □

6.2.1 Implications of Weak Duality

Theorem 6.3.
1. If the optimal cost in the primal is $-\infty$ (unbounded), then the dual problem must be infeasible.
2. If the optimal cost in the dual is $+\infty$ (unbounded), then the primal problem must be infeasible.

Proof. Suppose that the optimal cost in the primal problem is $-\infty$, and that the dual has a feasible solution $\mathbf{p}$. By weak duality, $\mathbf{p}$ satisfies $\mathbf{p}^\top\mathbf{b} \leq \mathbf{c}^\top\mathbf{x}$ for every primal feasible $\mathbf{x}$. Taking the minimum over all primal feasible $\mathbf{x}$, then:

$$\mathbf{p}^\top \mathbf{b} \leq -\infty$$

This is impossible and shows that the dual cannot have a feasible solution. Similar argument for the dual. □

Theorem 6.4. Let $\mathbf{x}^*$ and $\mathbf{p}^*$ be feasible solutions to the primal and the dual respectively, and suppose that $\mathbf{c}^\top \mathbf{x}^* = \mathbf{p}^{*\top} \mathbf{b}$. **Then, $\mathbf{x}^*$ and $\mathbf{p}^*$ are optimal solutions to the primal and the dual**.

Proof. For every primal feasible solution $\mathbf{x}$, the weak duality theorem yields $\mathbf{c}^\top \mathbf{x}^* = (\mathbf{p})^\top \mathbf{b} \leq \mathbf{c}^\top \mathbf{x}$ which implies x^* is optimal. □

6.3 Strong Duality

Theorem 6.5. If an LP has an optimal solution, so does its dual, and the respective optimal solutions are equal.

Proof. Consider the standard form problem:

$$\begin{aligned} \min \quad & \mathbf{c}^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

□

6.3.1 Implications of Strong Duality

1. An optimal solution of the dual linear program can be obtained as a byproduct of using the simplex method to solve the primal linear program.
2. In linear programming, exactly one of the following three possibilities will occur:
   - There is an optimal solution
   - The problem is 'unbounded'; that is, the optimal cost is $-\infty$ (for minimisation) or $+\infty$ (for maximisation).
   - The problem is infeasible.

	Finite Optimum	**Unbounded**	**Infeasible**
Finite Optimum	Possible	Impossible	Impossible
Unbounded	Impossible	Impossible	Possible
Infeasible	Impossible	Possible	Possible

Table 6: The different possibilities for the primal and dual.

Theorem 6.6. Consider the primal ($\mathbb{P}$) and its dual ($\mathbb{D}$). THen exactly one of the following statements is true:

- Both $\mathbb{P}$ and $\mathbb{D}$ posses optimal solutions and their objective function values are equal.
- $\mathbb{P}$ is unbounded and $\mathbb{D}$ is infeasible.
- $\mathbb{D}$ is unbounded and $\mathbb{P}$ is infeasible
- Both problems are infeasible.

7 Integer Programming

7.1 Knapsack Problem

A classic problem in computer science or combinatorial optimization. It involves selecting a subset of items from a set, each with a weight and value, to maximise the total value while keeping the total weight within a certain limit.

Definition 7.1. We formulate the knapsack problem as follows:

1. Set of items with weight w_i and value v_i.
2. A knapsack with a maximum weight of W.
3. Goal is to select a combination of items to maximise the total value while ensuring the total weight does not exceed the knapsack's capacity.

To formulate the problem:

1. Let n be the number of items.
2. Let x_i be a binary variable representing whether item i is selected (1) or not (0).
3. The objective function is given by:
$$\max \sum_{i=1}^{n} v_i \cdot x_i$$
4. The constraint is that the total weight cannot exceed the knapsack capacity:
$$\sum_{i=1}^{n} w_i \cdot x_i \leq W$$
5. Integer constraint: $x_i \in \{0, 1\}$, $\forall i = 1, \ldots, n$

7.2 Facility Location Problem

Involves determining the optimal locations for facilities to serve a set of demand points, while considering factors such as facility costs, transportation costs, and demand requirements. We first introduce Boolean decision variables y_f expressing if facility f is open:

$$y_f = \begin{cases} 1, & \text{facility } f \text{is open} \\ 0, & \text{otherwise} \end{cases}$$

Next, we introduce scalar decision variables x_{if} capturing the demand of client i that is serviced by facility f.

We need constraints on the decision variable:

1. For each client i, their demand d_i should be met. We express this algebraically by the constraints
$$\sum_{f=1}^{\text{facilities}} x_{if} = d_i, \quad \text{for each client } i$$
2. Facility Capacity Constraint: Ensure that the total demand served by each facility does not exceed its capacity u_f:
$$\sum_{i=1}^{\text{clients}} x_{if} \leq u_f y_f$$

3. Binary variables constraint: Ensure that if a facility is used to serve any demand, it must be opened:

$$x_{if} \leq d_i y_f \forall i,\ f$$

Now, we formulate the objective function: to minimise the total cost, which comprises facility costs and transportation costs. Recall that we pay c_f if we choose to open facility f. Thus, the opening cost for the facilities we choose to open is

$$\sum_{f=1}^{\text{facilities}} c_f y_f$$

We also pay c_{if} for each product unit we transfer from facility f to customer i. Thus, the transportation cost is given by

$$\sum_{i=1}^{\text{clients}} \sum_{f=1}^{\text{facilities}} f_{if} x_{if}$$

8 Maximum Matching

9 Network Simplex Algorithm

10 Integer Programming

11 LP Relaxation

12 Branch-and-Bound

Hello there

$$\int_{-\infty}^{\infty} x \, \mathrm{d}x \tag{16}$$

13 Dynamic Programming

14 Travelling Salesman Problem

Appendix & Acknowledgements