Introduction to Linear Programming

The material covered in this lecture is based on the notes on linear programming due to Michel Goemans.

1 Introduction

We define the linear programming (LP) problem of minimizing a linear function subject to linear inequality constraints. The general, standard, and canonical forms of the linear programming problem are given in summation and matrix form. A formal definition is given for a vertex of a polytope or polyhedron, and it is proved that an LP always attains its optimum at a vertex. Then it is proved that the set of vectors corresponding to the current basis of an LP are linearly independent if and only if the basic feasible solution is a vertex point.

2 Equivalent Forms

2.1 General Form

This is the general form of the linear programming problem. The c_i's can be interpreted as costs. In this case, the objective is to minimize the total cost subject to the linear constraints.

$$\text{Minimize } \sum_{i=1}^{n} c_i x_i \qquad \text{(objective function)} \tag{1}$$

$$\begin{array}{rrcll} \text{subject to} & \sum_{j=1}^{n} a_{ij}x_j & \geq & b_i & ,i = 1,\ldots,m_1 & \text{(inequality)} \\ & \sum_{j=1}^{n} a_{ij}x_j & = & b_i & ,i = m_1+1,\ldots,m_1+m_2 & \text{(equality)} \\ & x_j & \geq & 0 & ,j = 1,\ldots,n_1 & \text{(non} - \text{negativity)} \\ & x_j & \lesseqgtr & 0 & ,j = n_1+1,\ldots,n & \text{(unconstrained)} \end{array}$$

The general form of the LP can be written more compactly in matrix notation.

$$\text{Minimize } \mathbf{c}^T\mathbf{x} \tag{2}$$

$$\begin{array}{rrcl} \text{subject to} & \mathbf{A}\mathbf{x}_1 & \geq & \mathbf{b} \\ & \mathbf{A}'\mathbf{x}_2 & = & \mathbf{b}' \\ & \mathbf{x}_1 & \geq & 0 \\ & \mathbf{x}_2 & <=> & 0 \end{array}$$

where

c and **x** are n x 1 vectors,

$\mathbf{A}$ is an m_1 x n matrix,

$\mathbf{A}'$ is an m_2 x n matrix,

$\mathbf{b}$ is an m_1 x 1 vector,

$\mathbf{b}'$ is an m_2 x 1 vector,

$\mathbf{x}_1$ is an n_1 x 1 vector,

$\mathbf{x}_2$ is an n_2 x 1 vector, and

$\mathbf{x} = \left\{ \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \end{array} \right\}.$

2.2 Standard Form

This is the standard form of the linear programming problem. Here the constraints take the form of linear equality constraints, plus non-negativity constraints on the independent variables.

$$\text{Minimize} \sum_{i=1}^{n} c_i x_i \qquad \text{(objective function)} \tag{3}$$

$$\begin{array}{lrcll} \text{subject to} & \sum_{j=1}^{n} a_{ij} x_j & = & b_i & , i = 1, \ldots, m \quad \text{(equality)} \\ & x_j & \geq & 0 & , j = 1, \ldots, n \quad (\text{non} - \text{negativity}) \end{array}$$

The standard form of the LP can also be written more compactly in matrix notation.

$$\text{Minimize } \mathbf{c}^T \mathbf{x} \tag{4}$$

$$\begin{array}{lrcl} \text{subject to} & \mathbf{Ax} & = & \mathbf{b} \\ & \mathbf{x} & \geq & 0 \end{array}$$

where

$\mathbf{c}$ and $\mathbf{x}$ are n x 1 vectors,

$\mathbf{A}$ is an m x n matrix, and

$\mathbf{b}$ is an m x 1 vector.

Definition 1. If x satisfies $\mathbf{Ax} = \mathbf{b},\ \mathbf{x} \geq \mathbf{0}$, then x is **feasible**.

2.3 Canonical Form

This is the canonical form of the linear programming problem. Here the constraints take the form of linear inequality constraints, plus non-negativity constraints on the independent variables.

$$\text{Minimize} \sum_{i=1}^{n} c_i x_i \qquad \text{(objective function)} \tag{5}$$

$$\begin{array}{rrcll} \text{subject to} & \sum_{j=1}^{n} a_{ij} x_j & \geq & b_i & , i = 1, \dots, m \quad \text{(inequality)} \\ & x_j & \geq & 0 & , j = 1, \dots, n \quad (\text{non} - \text{negativity}) \end{array}$$

Once more, the canonical form of the LP can be written more compactly in matrix notation.

$$\text{Minimize } \mathbf{c}^T \mathbf{x} \tag{6}$$

$$\begin{array}{rrcl} \text{subject to} & \mathbf{Ax} & \geq & \mathbf{b} \\ & \mathbf{x} & \geq & 0 \end{array}$$

where

c and **x** are n x 1 vectors,

A is an m x n matrix, and

b is an m x 1 vector.

2.4 Constraint Conversion

It is possible to convert between equality and inequality constraints by the following method.

1. To convert a less than or equal inequality constraint,

$$Ax \leq b$$

to an equality constraint, add the vector of slack variables, s.

$$Ax + s = b, \;\; s \geq 0$$

2. To convert a greater than or equal inequality constraint,

$$Ax \geq b$$

to an equality constraint, add the vector of surplus variables, t.

$$Ax - t = b, \;\; t \geq 0$$

3. Finally, to convert an equality constraint,

$$Ax = b$$

to an inequality constraint, add two inequality constraints.

$$Ax \leq b, \;\; -Ax \leq -b$$

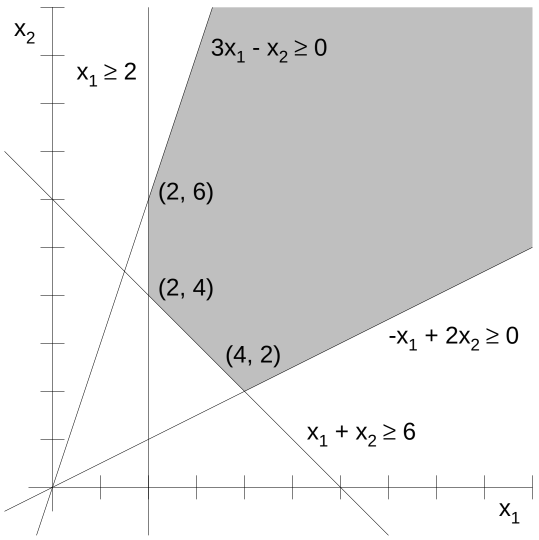

Figure 1: Feasible Region

3 Example

Example 1. Consider the following linear program:

$$
\begin{array}{lrcrcl}
\text{Minimize} & & & x_2 & & \\
\text{subject to} & x_1 & & & \geq & 2 \\
& 3x_1 & - & x_2 & \geq & 0 \\
& x_1 & + & x_2 & \geq & 6 \\
& -x_1 & + & 2x_2 & \geq & 0 \\
& x_1 & & & \geq & 0 \\
& & & x_2 & \geq & 0
\end{array}
$$

The optimal solution is (4, 2) of cost 2 (See Figure 1). If we were maximizing x_2, instead of minimizing under the same feasible region, the resulting linear program would be unbounded, since x_2 can increase arbitrarily. From this picture, the reader should be convinced that, for any objective function for which the linear program is bounded, there exists an optimal solution which is a "corner" of the feasible region. This notion will be formalized in the next section.

4 The Geometry of LP

Definition 2. A **polytope** is the analogue in n-dimensional space of point, segment, polygon, and polyhedron in spaces of dimensions 0, 1, 2, and 3. A **convex polytope** in n-space is the convex span of a finite set of points that do not all lie in the same hyperplane; thus a convex polytope is a bounded convex subset enclosed by a finite number of hyperplanes.

Definition 3. A set is **convex** if it contains the line segment joining any two of its points; in a *vector space*, a set such that $rx + (1-r)y$ is in the set for $0 < r < 1$, if x and y are in the set.

Definition 4. A point x is a **vertex** of $\mathcal{P}$ if $\nexists y \neq 0$ such that $x + y,\ x - y \in \mathcal{P}$.

Theorem 1. *If the optimal solution to an LP in canonical form is bounded (from below), then given any $x \in \mathcal{P}$, $\exists$ a vertex v such that $c^T v \leq c^T x$.*

Corollary 1. *If the LP is bounded, then $\exists$ a vertex that yields the optimal solution.*

Proof: (of theorem) Given $x \in \mathcal{P}$. If x is a vertex, we are done. Otherwise, $\exists y \neq 0$ such that $x + y \in \mathcal{P}$ and $x - y \in \mathcal{P}$. By feasibility of $x + y$ and $x - y$, $A(x + y) = b,\ A(x - y) = b$. But this implies that $Ay = 0$. Without loss of generality, choose y such that $c^T y \leq 0$. Then $c^T(x + \lambda y) = c^T x + \lambda c^T y \leq c^T x$.

Case 1. $\exists j$ such that $y_j < 0$.
We need to ensure that $x + \lambda y \geq 0$ to maintain feasibility. Hence set $\lambda = \min_{\{j: y_j < 0\}} \{\frac{x_j}{-y_j}\}$. Then it is clear that $x + \lambda y \geq 0$. Then $x + \lambda y$ has at least one more zero component than x.

Case 2. $y_j \geq 0,\ \forall j$.
In this case, $c^T y < 0$. But, clearly, $x + \lambda y \geq 0$ for $\lambda > 0$. Hence, $c^T(x + \lambda y) = c^T x + \lambda c^T y$. Then λ can be chosen arbitrarily large, and the objective function is unbounded. Contradiction.

Case 1 can happen at most n times, since x has n components. By induction on the number of non-zero components of x, we obtain a vertex v. ■

Remark 1. Linear Programming was developed during World War II to solve logistics problems for military resources. The simplex method for solving linear programming problems was developed in 1947 by George Dantzig, an American mathematician. Although the simplex method performs well in practice, in 1972 Klee and Minty demonstrated an example in which the simplex method takes an exponential number of steps. In 1979, L. G. Khachian produced the ellipsoidal method, which was shown to solve any linear program in polynomial time. However, when implemented, his method was not competitive with the simplex method, and lost favor. Then, in 1984, Narinder K. Karmarkar presented a method based on the projective transformation of a simplex. Karmarkar showed his method solves any linear program in time that is a polynomial function of the data of the problem. In contrast to Khachian's method, implementations of various generalizations of Karmarkar's method are very competitive with the simplex method, and generally outperform it for large problems.

Theorem 2. *Let x be a point in $\mathcal{P}$. The submatrix $A_x = \{$ columns corresponding to $x_j > 0\}$ is linearly independent if and only if x is a vertex.*

Proof:

($\Leftarrow$) Suppose A_x has linearly dependent columns. Then $\exists y$ such that $A_x y = 0,\ y \neq 0$. Extend y to $\mathbb{R}^n$ by adding zero-valued components. Then $\exists y \in \mathbb{R}^n$ such that $Ay = 0,\ y \neq 0$. Consider $x + \lambda y$ for small $\lambda > 0$. Then we can chose λ such that $x + \lambda y,\ x - \lambda y \in \mathcal{P}$, by an argument analogous to that in Case 1 of the proof of Theorem 1, above. Hence, x is not a vertex.

($\Rightarrow$) Assume x is not a vertex. Then, by definition, $\exists y \neq 0$ such that $x+y,\ x-y \in \mathcal{P}$. Let A_y be the submatrix corresponding to the non-zero components of y. As in the proof of Theorem 1,

$$\left.\begin{array}{r} Ax + Ay = b \\ Ax - Ay = b \end{array}\right\} \Rightarrow Ay = 0.$$

Therefore, A_y has dependent columns, since $y \neq 0$. Moreover,

$$\left.\begin{array}{r} x + y \geq 0 \\ x - y \geq 0 \end{array}\right\} \Rightarrow y_j = 0 \text{ whenever } x_j = 0.$$

Therefore, A_y is a submatrix of A_x. Since A_y is a submatrix of A_x, A_x has linearly dependent columns.

$\blacksquare$