Linear programming

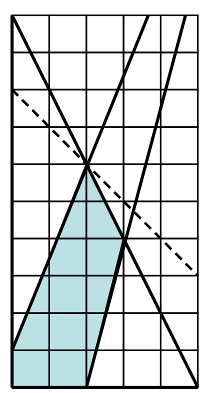
Saad Mneimneh

1 Introduction

Consider the following problem:

$$\begin{array}{rl} \text{maximize} & x_1 + x_2 \\ \text{subject to} & 4x_1 - x_2 \leq 8 \\ & 2x_1 + x_2 \leq 10 \\ & 5x_1 - 2x_2 \geq -2 \\ & x_1, x_2 \geq 0 \end{array}$$

The feasible solution is a point (x_1, x_2) that lies within the region defined by the lines $4x_1 - x_2 = 8$, $2x_1 + x_2 = 10$, $5x_1 - 2x_2 = -2$, $x_1 = 0$, and $x_2 = 0$. We need such a point that will maximize $x_1 + x_2 = k$. Note that $x_2 = k - x_1$ is a line with slope -1 that intersects $x_1 = 0$ at k. Therefore, as illustrated below, the point $(2, 6)$ is the optimal solution.



In general, it is not possible to solve geometrically, especially for higher dimensions. Instead, we will develop a systematic way for finding the optimal solution algebraically. Assume, in general, we are given the following Linear Program.

$$\begin{array}{rl} \text{maximize} & c^T x \\ \text{subject to} & Ax \leq b \\ & x \geq 0 \end{array}$$

where A is an $m \times n$ matrix, and b and c are both vectors of size m and n respectively. The matrix notation represents m constraints, for $i = 1 \ldots m$, of the form:

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i$$

The goal is to find a vector x of size n that satisfies the constraints and maximizes the objective function $c^T x$. Note that it is always possible to convert a minimization to a maximization, and a $\ge$ constraint to a $\le$ constraint. Furthermore, if for some x_i, the constraint $x_i \ge 0$ is not present, we can replace x_i by the difference $x'_i - x''_i$, where $x'_i > 0$ and $x''_i > 0$.

2 The slack form

We will transform the above linear program into a more useful form, known as the slack form. We replace the i^{th} constraint:

$$\sum_{j} a_{ij} x_j \le b_i$$

by

$$\sum_{j} a_{ij} x_j + s_i = b_i$$

$$s_i \ge 0$$

where $s_i \ge 0$ is a newly introduced slack variable. After introducing all m slack variables, we make sure that $b \ge 0$, i.e. $b_i \ge 0$ for all i. This can be achieved, if $b_i < 0$, by multiplying the i^{th} constraint by -1 on both sides.

$$\sum_{j} -a_{ij} x_j - s_i = -b_i$$

Our slack form now becomes:

$$\begin{array}{rl} \text{maximize} & c^T x \\ \text{subject to} & Ax = b \quad (b > 0) \\ & x \ge 0 \end{array}$$

where A is an $m \times n$ matrix ($n \ge m$), $b \ge 0$ is a vector of size m, and c is a vector of size n. We seek a solution x.

Assume that A can be divided into two parts, a square $m \times m$ matrix B, and another $(n-m) \times m$ matrix N. Therefore, $A = [B, N]$. Assume further that B has an inverse B^{-1}. Accordingly, let x^T be $[x_B^T, x_N^T]$ with m basic variable x_B and $n - m$ non-basic variables x_N. Set $x_N = 0$ and $x_B = B^{-1}b$. Therefore, x is a feasible solution because $Ax = [B, N]x = Bx_B + Nx_N = BB^{-1}b + 0 = b$. Using this initial feasible (but not necessarily optimal) solution, we introduce the simplex algorithm below.

3 The simplex algorithm

Let A_j be the j^{th} column of A (the column corresponding to variable x_j).

Simplex
while $\exists x_j \in x_N$ s.t. $c_j - c_B^T B^{-1} A_j > 0$
  increase x_j until some $x_i \in x_B$ is zero
  switch the roles of x_i and x_j
  update B (N, x_B, and x_N too)

Note that increasing $x_j \in x_N$ changes x_B too.

$$Bx_B + A_j x_j = b \Rightarrow x_B = B^{-1}(b - A_j x_j)$$

The new objective function becomes:

$$c^T x = c_B^T x_B + c_j x_j = c_B^T B^{-1}(b - A_j x_j) + c_j x_j = c_B^T B^{-1} b + (c_j - c_B^T B^{-1} A_j) x_j$$

Now observe that $c_j - c_B^T B^{-1} A_j$ is the coefficient of x_j in the objective function, and $c_B^T B^{-1} b$ is the value of the objective function prior to increasing x_j. Therefore, since $c_j - c_B^T B^{-1} A_j > 0$, increasing x_j increases the objective. But how much can we increase x_j? We still require $x_B \geq 0$; therefore, $B^{-1}(b - A_j x_j) \geq 0 \Rightarrow B^{-1} b - B^{-1} A_j x_j \geq 0$. If $B^{-1} A_j \leq 0$ (all components of $B^{-1} A_j$ are negative or zero), then we can increase x_j as much as we want. In this case, the linear program in unbounded. Otherwise, x_j can increase until some element in x_B becomes 0.

When the condition of the while loop is false, i.e. $c_j - c_B^T B^{-1} A_j \leq 0$ for all $x_j \in x_N$, the solution is optimal. We will later prove this fact using the concept of duality.

4 Example of simplex

We will show an example of using simplex. While matrix notation is concise for the purpose of illustration, we will not follow it here. Equivalently, we will choose a non-basic variable with positive coefficient in the objective function, and increase it.

$$\begin{array}{rl} \text{maximize} & 3x_1 + x_2 + 2x_3 \\ \text{subject to} & x_1 + x_2 + 3x_3 \leq 30 \\ & 2x_1 + 2x_2 + 5x_3 \leq 24 \\ & 4x_1 + x_2 + 2x_3 \leq 36 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$

We obtain the slack form:

$$\begin{array}{rl} \text{maximize} & 3x_1 + x_2 + 2x_3 \\ \text{subject to} & x_1 + x_2 + 3x_3 + x_4 = 30 \\ & 2x_1 + 2x_2 + 5x_3 + x_5 = 24 \\ & 4x_1 + x_2 + 2x_3 + x_6 = 36 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{array}$$

Consider the initial solution given by $x_B^T = [x_4, x_5, x_6]$ (the slack variables).

$$\begin{array}{rl} \text{maximize} & 3x_1 + x_2 + 2x_3 \\ \text{subject to} & x_4 = 30 - x_1 - x_2 - 3x_3 \\ & x_5 = 24 - 2x_1 - 2x_2 - 5x_3 \\ & x_6 = 36 - 4x_1 - x_2 - 2x_3 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{array}$$

Therefore, $x = [0, 0, 0, 30, 24, 36]$ is our initial feasible solution (Section 5 deals with the problem when such an initial solution cannot be found). We now repeatedly identify a variable $x_j \in x_N$ with a positive coefficient in the objective function and increase it until an $x_i \in x_B$ becomes zero. We will always rewrite the objective function in terms of non-basic variables so that this identification is easy to make.

We can increase x_1 up to 9 which will make $x_6 = 0$. We exchange x_1 and x_6.

$$\begin{array}{rl} \text{maximize} & 27 + \frac{x_2}{4} + \frac{x_3}{2} - 3\frac{x_6}{4} \\ \text{subject to} & x_4 = 21 - 3\frac{x_2}{4} - 5\frac{x_3}{2} + \frac{x_6}{4} \\ & x_5 = 6 - 3\frac{x_2}{2} - 4x_3 + \frac{x_6}{2} \\ & x_1 = 9 - \frac{x_2}{4} - \frac{x_3}{2} - \frac{x_6}{4} \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{array}$$

Next, we can choose, say, x_3. We can increase x_3 up to $3/2$ which will make $x_5 = 0$. Updating, we get:

$$\begin{array}{rl} \text{maximize} & \frac{111}{4} + \frac{x_2}{16} - \frac{x_5}{8} - 11\frac{x_6}{16} \\ \text{subject to} & x_1 = \frac{33}{4} - \frac{x_2}{16} + \frac{x_5}{8} - 5\frac{x_6}{16} \\ & x_3 = \frac{3}{2} - 3\frac{x_2}{8} - \frac{x_5}{4} + \frac{x_6}{8} \\ & x_4 = \frac{69}{4} + 3\frac{x_2}{16} + 5\frac{x_5}{8} - \frac{x_6}{16} \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{array}$$

The only choice we have now is x_2. Increasing x_2 to 4 makes $x_3 = 0$.

$$\begin{array}{rl} \text{maximize} & 28 - \frac{x_3}{6} - \frac{x_5}{6} - 2\frac{x_6}{3} \\ \text{subject to} & x_1 = 8 + \frac{x_3}{6} + \frac{x_5}{6} - \frac{x_6}{3} \\ & x_2 = 4 - 8\frac{x_3}{3} - 2\frac{x_5}{3} + \frac{x_6}{3} \\ & x_4 = 18 - \frac{x_3}{2} + \frac{x_5}{2} \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{array}$$

We stop with a value of 28 for the objective function. This is the optimal solution for this linear program since all non-basic variables have negative coefficients in the objective function. Therefore, the solution is $x^T = [x_1, x_2, x_3, x_4, x_5, x_6] = [8, 4, 0, 18, 0, 0]$.

5 Initial feasible solution

The slack variables may not offer a feasible solution as illustrated in the above example. For instance, consider the following linear program:

$$\begin{array}{rl} \text{maximize} & 2x_1 - x2 \\ \text{subject to} & 2x_1 - x_2 \leq 2 \\ & x_1 - 5x_2 \leq -4 \\ & x_1, x_2 \geq 0 \end{array}$$

Converting to slack form (and making $b \geq 0$), we have:

$$\begin{array}{rl} \text{maximize} & 2x_1 - x2 \\ \text{subject to} & 2x_1 - x_2 + x_3 = 2 \\ & -x_1 + 5x_2 - x_4 = 4 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{array}$$

The initial solution is then given by $x^T = [x_1, x_2, x_3, x_4] = [0, 0, 2, -4]$ which is not feasible (because $x_4 < 0$). Therefore, we need a general approach by which we either find an initial feasible solution, or determine that the linear program is not feasible. Consider the following auxiliary linear program:

$$\begin{array}{rl} \text{maximize} & -1^T y \\ \text{subject to} & Ax + Iy = b \\ & x, y \geq 0 \end{array}$$

where $1^T = [1, \ldots, 1]$ and I is the identity matrix.

Observation: the original linear program is feasible if and only if the auxiliary linear program has optimal solution $y = 0$. Proof: on one hand, if the original linear program is feasible, then there exists an x such that $Ax = b$. Therefore, $y = 0$ is feasible for the auxiliary linear program, and since the objective is to maximize $-1^T y$ and $y \geq 0$, $y = 0$ is optimal. On the other hand, if the $y = 0$ is the optimal solution for the auxiliary linear program, then we have found an x such that $Ax = b$, which is feasible for the original linear program.

Therefore, we solve for the auxiliary linear program. If the optimal solution is $y = 0$, then we use x as a feasible solution for the original linear program. Otherwise, the original linear program is not feasible. We still need an initial feasible solution for the auxiliary linear program: $x = 0$ and $y = b$ is one (since $b \geq 0$).

6 Running time of simplex

If we think of the basic variables x_B as the state of the simplex algorithm, then simplex can be in at most $\left(\begin{array}{c} m+n \\ m \end{array}\right)$ states. This is because we are not changing the equations, but simply deciding which variables (the basic ones) appear on the left hand side. Therefore, simplex either terminates after at most $\binom{m+n}{m}$ iterations, or it cycles. This, however, does not mean that the objective function will increase indefinitely, as it is possible for an iteration not to increase the objective. Here's an example.

$$\begin{array}{rl} \text{maximize} & x_1 + x_2 + x_3 \\ \text{subject to} & x_4 = 8 - x_1 - x_2 \\ & x_5 = x_2 - x_3 \\ & x \geq 0 \end{array}$$

Suppose we choose x_1 and increase it to 8 to make $x_4 = 0$, we get:

$$\begin{array}{rl} \text{maximize} & 8 + x_3 - x_4 \\ \text{subject to} & x_1 = 8 - x_2 - x_4 \\ & x_5 = x_2 - x_3 \\ & x \geq 0 \end{array}$$

The only choice now is x_3, but we can only increase it up to 0; otherwise, x_5 becomes negative. Doing so will not increase the objective function, but will change the state.

$$\begin{array}{rl} \text{maximize} & 8 + x_2 - x_4 - x_5 \\ \text{subject to} & x_1 = 8 - x_2 - x_4 \\ & x_3 = x_2 - x_5 \\ & x \geq 0 \end{array}$$

We can continue by choosing x_2 now.

Cycling is possible but rare. It is avoidable by making more careful choices about which non-basic variable becomes basic. We are not going to explore this any further.

7 Duality and optimality

We now prove that when simplex terminates we have the optimal solution. Given the linear program (in slack form here):

$$\begin{array}{rl} \text{maximize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

construct its dual:

$$\begin{array}{rl} \text{minimize} & y^T b \\ \text{subject to} & y^T A \geq c^T \\ & y \geq 0 \end{array}$$

Note that $y^T A \geq c^T$ and, since $x \geq 0$, $y^T Ax \geq c^T x$. But $y^T Ax = y^T(Ax) = y^T b$. Therefore, $c^T x \leq y^T b$. This means that whenever $c^T x = y^T b$, both x and

y are optimal solutions for their respective programs. We will prove that this is in deed the case. When simplex terminates, the solution is given by $x_N = 0$, $x_B = B^{-1}b$, and $c^T x = c_B^T B^{-1} b$. Moreover, $c_j - c_B^T B^{-1} A_j \leq 0$ for all $x_j \in x_N$. Let $y^T = c_B^T B^{-1}$. It is obvious that $y^T b = c^T x$. We will show that y is feasible for the dual linear program.

$$y^T B = c_B^T B^{-1} B = c_B^T$$

$$y^T A_j = c_B^T B^{-1} A_j \geq c_j \quad \text{for all } x_j \in x_N$$

Therefore, $y^T A \geq c^T$.