Linear Programming

Introduction to Linear Programming:-

Linear programming was developed during World War II, when a system with which to maximize the efficiency of resources was of utmost importance. New war-related projects demanded attention and spread resources thin. “Programming” was a military term that referred to activities such as planning schedules efficiently or deploying men optimally. George Dantzig, a member of the U.S. Air Force, developed the Simplex method of optimization in 1947 in order to provide an efficient algorithm for solving programming problems that had linear structures. Since then, experts from a variety of fields, especially mathematics and economics, have developed the theory behind “linear programming” and explored its applications [1]. This paper will cover the main concepts in linear programming, including examples when appropriate. First, in Section 1 we will explore simple properties, basic definitions and theories of linear programs. In order to illustrate some applications of linear programming, we will explain simplified “real-world” examples in Section 2. Section 3 presents more definitions, concluding with the statement of the General Representation Theorem (GRT). In Section 4, we explore an outline of the proof of the GRT and in Section 5 we work through a few examples related to the GRT. After learning the theory behind linear programs, we will focus methods of solving them. Section 6 introduces concepts necessary for introducing the Simplex algorithm, which we explain in Section 7. In Section 8, we explore the Simplex further and learn how to deal with no initial basis in the Simplex tableau. Next, Section 9 discusses cycling in Simplex tableaux and ways to counter this phenomenon. We present an overview of sensitivity analysis in Section 10. Finally, we put all of these concepts together in an extensive case study .

What is a linear program?

We can reduce the structure that characterizes linear programming problems (perhaps after several manipulations) into the following form:

Minimize

$c_1x_1 + c_2x_2 + \cdots + c_nx_n = z$

Subject to

$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1$

$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2$

$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m$

$x_1, x_2, \ldots, x_n \geq 0.$

In linear programming z, the expression being optimized, is called the objective function. The variables $x_1, x_2 \ldots x_n$ are called decision variables, and their values are subject to m + 1 constraints (every line ending with a b_i , plus the nonnegativity constraint). A set of $x_1, x_2 \ldots x_n$ satisfying all the constraints is called a feasible point and the set of all such points is called the feasible region. The solution of the linear program must be a point $(x_1, x_2, \ldots, x_n)$ in the feasible region, or else not all the constraints would be satisfied. The following example from Chapter 3 of Winston [3] illustrates that geometrically interpreting the feasible region is a useful tool for solving linear programming problems with two decision variables. The linear program is:

Minimize $4x_1 + x_2 = z$

Subject to $3x_1 + x_2 \geq 10$

$x_1 + x_2 \geq 5$

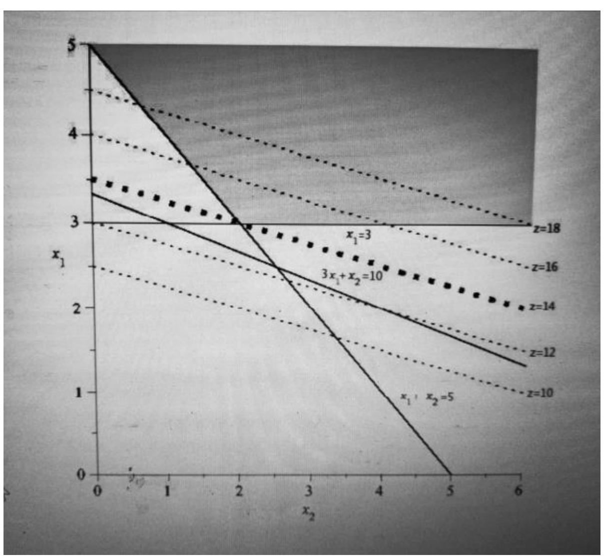
$x_1 \quad \geq 3$

$x_1, \; x_2 \geq 0.$

We plotted the system of inequalities as the shaded region in Figure 1. Since all of the constraints are "greater than or equal to" constraints, the shaded region above all three lines is the feasible region. The solution to this linear program must lie within the shaded region. Recall that the solution is a point (x_1, x_2) such that the value of z is the smallest it can be, while still lying in the feasible region. Since $z = 4x_1 + x_2$, plotting the line $x_1 = (z - x_2)/4$ for various values of z results in is cost lines, which have the same slope. Along these lines, the value of z is constant. In Figure 1, the dotted lines represent is cost lines for different values of z. Since isocost lines are parallel to each other, the thick dotted is cost line for which $z = 14$ is clearly the line that intersects the feasible region at the smallest possible value for z. Therefore, $z = 14$ is the smallest possible value of z give

the constraints. This value occurs at the intersection of the lines $x_1 = 3$ and $x_1 + x_2 = 5$, where $x_1 = 3$ and $x_2 = 2$.

Figure 1: The shaded region above all three solid lines is the feasible region (one of the constraints does not contribute to defining the feasible region). The dotted lines are isocost lines. The thick isocost line that passes through the intersection of the two defining constraints represents the minimum possible value of $z = 14$ while still passing through the feasible region.



Example:- 1. Consider the problem of locating a new machine to an existing layout consisting of four machines. These machines are located at the following (x, y) coordinates: (3, 0), (0, −3), (−2, 1), and (1, 4). Let the coordinates of the new machine be (x1, x2). Formulate the problem of finding an optimal location as a linear program if the sum of the distances from the new machine to the existing four machines is minimized. Use street distance; for example, the distance from (x1, x2) to the first machine at 1 (3, 0) is $|x1 - 3| + |x2|$. This means that the distance is not defined by the length of a line between two points, rather it is the sum of the lengths of the horizontal and vertical components of such a line

Solution:--

Since absolute value signs cannot be included in a linear program, recall that:

$|x| = \max \{x, -x\}$.

With this in mind, the following linear program models the problem:

Minimize $z = (P1 + P2) + (P3 + P4) + (P5 + P6) + (P7 + P8)$

Subject to
$P1 \geq -(x1 - 3)$

$P1 \geq x1 - 3$

$P2 \geq -(x2)$

$P2 \geq x2$

$P3 \geq -(x1 - 1)$

$P3 \geq x1 - 1$

$P4 \geq -(x2 - 4)$

$P4 \geq x2 - 4$

$P5 \geq -(x1 + 2)$

$P5 \geq x1 + 2$

$P6 \geq -(x2 - 1)$

$P6 \geq x2 - 1$

$P7 \geq -(x1)$

$P7 \geq x1$

$P8 \geq -(x2 + 3)$

$P8 \geq x2 + 3$

all variables ≥ 0.

Each $P2i-1$ represents the horizontal distance between the new machine and the i th old machine for i = 1, 2, 3, 4. Also for i = 1, 2, 3, 4, $P2i$ represents the vertical distance between the new machine and the i th old machine. The objective function reflects the desire to minimize total distance between the new machine and all the others. The constraints relate the P variables to the distances in terms of x1 and x2. Two constraints for each P variable allow each Pi (i = 1, 2, . . . , 8) to equal the maximum 12 of xj − cj and $-(x_j - c_j)$ (for j = 1, 2 and where c is the j the component of the position of one of the old machines). Since this program is a minimization problem and the smallest any of the variables can be is max $\{(x_j - c_j), -(x_j - c_j)\}$, each P_i will naturally equal its least possible value. This value will be the absolute value of $x_j - c_j$. In the next problem we will also interpret a "real-world" situation as a linear program. Perhaps the most notable aspect of this problem is the concept of inventory and recursion in constraints.

ASSIGNMENT:- 1

A company is opening a new franchise and wants to try minimizing their quarterly cost using linear programming. Each of their workers gets paid $500 per quarter and works 3 contiguous quarters per year. Additionally, each worker can only make 50 pairs of shoes per quarter. The demand (in pairs of shoes) is 600 for quarter 1, 300 for quarter 2, 800 for quarter 3, and 100 for quarter 4. Pairs of shoes may be put in inventory, but this costs $50 per quarter per pair of shoes, and inventory must be empty at the end of quarter 4.?

(2) Suppose that there are m sources that generate waste and n disposal sites. The amount of waste generated at source i is a_i and the capacity of site j is b_j . It is desired to select appropriate transfer facilities from among K candidate facilities. Potential transfer facility k has fixed cost f_k, capacity 14 q_k and unit processing cost α_k per ton of waste. Let c_{ik} and $\bar{c}_{kj}$ be the unit shipping costs from source i to transfer station k and from transfer station k to disposal site j respectively. The problem is to choose the transfer facilities and the shipping pattern that minimize the total capital and operating costs of the transfer stations plus the transportation costs ?

Queuing Theory

What Is Queuing Theory?

Queuing theory is the mathematical study of the congestion and delays of waiting in line. Queuing theory (or "queueing theory") examines every component of waiting in line to be served, including the arrival process, service process, number of servers, number of system places, and the number of customers—which might be people, data packets, cars, etc.

As a branch of operations research, queuing theory can help users make informed business decisions on how to build efficient and cost-effective workflow systems. Real-life applications of queuing theory cover a wide range of applications, such as how to provide faster customer service, improve traffic flow, efficiently ship orders from a warehouse, and design of telecommunications systems, from data networks to call centers.

Benefits of Queuing Theory

By applying queuing theory, a business can develop more efficient queuing systems, processes, pricing mechanisms, staffing solutions, and arrival management strategies to reduce customer wait times and increase the number of customers that can be served.

Queuing theory as an operations management technique is commonly used to determine and streamline staffing needs, scheduling, and inventory, which helps improve overall customer service. It is often used by Six Sigma practitioners to improve processes.

Basic Queueing Formulas

Little's rule provides the following results:

$L = \lambda W; Lq = \lambda Wq;$

the first of the above applies to the system and the second to the queue, which is a part of the system. Another useful relationship in the queue is:

$W = Wq + 1\ \mu\ ;$

the above is intuitive (we prove it later): it says the mean wait in the system is the sum of the mean wait in the queue and the service time ($1/\mu$).

For the M/M/1

queue, we can prove that (Ross, 2014)

$Lq = \rho\ 2\ 1 - \rho\ .$

For the M/G/1 queue,

we can prove that $Lq = \lambda\ 2\sigma\ 2\ s + \rho\ 2\ 2(1 - \rho)$

The above is called the Pollazcek-Khintichine formula (named after its inventors and discovered in the 1930s; see Ross (2014)

For the G/G/1 queue, we do not have an exact result. The following approximation (derived in Marchal (1976)) is popular in industry:

$Lq \approx \rho\ 2\ (1 + C\ 2\ s\ )(C\ 2\ a + \rho\ 2C\ 2\ s\ )\ 2(1 - \rho)(1 + \rho\ 2C2\ s\ )\ .$

(2) In the above, if the mean rate of arrival is λ and $\sigma\ 2\ a$ denotes the variance of the inter-arrival time, then:

$C\ 2\ a = \sigma\ 2\ a\ (1/\lambda)$

2 . Similarly, if μ denotes the service rate and $\sigma\ 2\ s$ denotes the variance of the service time, then:

$C\ 2\ s = \sigma\ 2\ s\ (1/\mu)$

2 . 4 Another approximation from Kraemer and Langenbach-Belz (1976) is also quite powerful:

Lq ≈ ρ 2 (C 2 a + C 2 s) 2(1 − ρ) g (3)

where g = exp −2(1 − ρ)(1 − C 2 a) 2 3ρ(C2 a + C2 s) !

when C 2 a ≤ 1; (4) g = exp (1 − ρ)(1 − C 2 a) C2 a + 4C2 s ! when C 2 a > 1. (5)

ASSIGNMENT PROBLEMS:--

1. X is a discrete stochastic variable, pk = P(X = k) = a k k! e −a , k = 0, 1, 2, ... and a is a positive constant. a) Prove that P∞ k=0 pk = 1. b) Determine the z-transform (generating function) P(z) = P∞ k=0 z kpk. c) Calculate E[X], Var[X] and E[X(X − 1)...(X − r + 1)], r = 1, 2, ... with and without using z-transforms?

2. X is a random variable chosen from X1 with probability a and from X2 with probability b. Calculate E[X] and σX for α = 0.2 and b = 0.8. X1 is an exponentially distributed r.v. with parameter λ1 = 0.1 and X2 is an exponentially distributed r.v. with parameter λ2 = 0.02. Let the r.v. Y be chosen from D1 with probability α and from D2 with probability b, where D1 and D2 are deterministic r.v.s. Calculate the values D1 and D2 so that E[X] = E[Y] and σX = σY .?

3. X is a discrete stochastic variable, pk = P(X = k) = a k k! e −a , k = 0, 1, 2, ... and a is a positive constant. a) Prove that P∞ k=0 pk = 1. b) Determine the z-transform (generating function) P(z) = P∞ k=0 z kpk. c) Calculate E[X], Var[X] and E[X(X − 1)...(X − r + 1)], r = 1, 2, ... with and without using z-transforms.?