DECISION MAKING

The goal of decision analysis is to help individuals make good decisions that maximize return and minimize risk. But good decisions do not always result in good outcomes. Take a look at this wonderful example to understand how good decisions can lead to bad outcomes!

Andre-Francois Raffray thought he had a great deal in 1965 when he agreed to pay a 90-year-old woman named Jeanne Calment $500 a month until she died to acquire her grand apartment in Arles, northwest of Marseilles in the south of France—a town Vincent Van Gogh once roamed. Buying apartments "for life" is common in France. The elderly owner gets to enjoy a monthly income from the buyer who gambles on getting a real estate bargain—betting the owner doesn't live too long. Upon the owner's death, the buyer inherits the apartment regardless of how much was paid. But in December of 1995, Raffray died at age 77, having paid more than $180,000 for an apartment he never got to live in.

On the same day, Calment, then the world's oldest living person at 120, dined on foie gras, duck thighs, cheese, and chocolate cake at her nursing home near the sought-after apartment. And she does not need to worry about losing her $500 monthly income. Although the amount Raffray already paid is twice the apartment's current market value, his widow is obligated to keep sending the monthly check to Calment. If Calment also outlives her, then the Raffray children will have to pay. "In life, one sometimes makes bad deals," said Calment of the outcome of Raffray's decision. (Source: The Savannah Morning News, *12/29/95.)*

On the good side, over the past decades, scores of operations research and management science projects saved companies millions of dollars.

- At the turn of the century, Motorolafaced a crisis due to economic conditions in its marketplaces; the company needed to reduce costs dramatically and quickly. A natural target was its purchases of goods and services, as these expenses account for more than half of Motorola's costs. Motorola decided to create an Internet-based system to conduct multi-step negotiations and auctions for supplier negotiation. The system can handle complex bids and constraints, such as bundled bids, volume-based discounts, and capacity limits. In addition, it can optimize multi-product, multi-vendor awards subject to these constraints and nonlinear price schedules.

  - Benefits:In 2003, Motorola used this system to source 56 percent of its total spending, with 600 users and a total savings exceeding $600 million.

- Waste Managementis the leading company in North America in the waste collection industry. The company has a fleet of over 26,000 vehicles for collecting waste from nearly 20 million residential customers, plus another two million commercial customers. To improve trash collection and make its operations more efficient,Waste Management implemented a vehicle-routing application to optimize its collection routes.

  - Benefits: The successful deployment of this system brought benefits including the elimination of nearly 1,000 routes within one year of implementation and an estimated annual savings of $44 million.

MATHEMATICAL OPTIMIZATION

To help you understand the purpose of optimization and the types of problems for which it can be used, let's consider several examples of decision-making situations in which MP techniques have been applied.

Determining Product Mix: Most manufacturing companies can make a variety of products however, each product usually requires different amounts of raw materials and labor similarly, the amount of profit generated by the products varies. The manager of such a company must decide how many of each product to produce to maximize profits or to satisfy demand at minimum cost.

Manufacturing: Printed circuit boards, like those used in most computers, often have hundreds or thousands of holes drilled in them to accommodate the different electrical components that must be plugged into them. To manufacture these boards, a computer controlled drilling machine must be programmed to drill in a given location, then move the drill bit to the next location and drill again. This process is repeated hundreds or thousands of times to complete all the holes on a circuit board. Manufacturers of these boards would benefit from determining the drilling order that minimizes the total distance the drill bit must be moved.

Routing and Logistics: Many retail companies have warehouses around the country that are responsible for keeping stores supplied with merchandise to sell. The amount of merchandise available at the warehouses and the amount needed at each store tends to fluctuate, as does the cost of shipping or delivering merchandise from the warehouses to the retail locations. Large amounts of money can be saved by determining the least costly method of transferring merchandise from the warehouses to the stores.

Financial Planning: The federal government requires individuals to begin withdrawing money from individual retirement accounts (IRAs) and other tax-sheltered retirement programs no later than age 70.5. There are various rules that must be followed to avoid paying penalty taxes on these withdrawals. Most individuals want to withdraw their money in a manner that minimizes the amount of taxes they must pay while still obeying the tax laws.

So, all optimization problems are alike. They want to maximize (normally things related to profit) or minimize (normally things related to loss) with or without constraints. In this program, we get exposed to some commonly used optimization techniques like linear programming, quadratic programming and dynamic programming in the context of convex functions.

Mathematically, an optimization problem can be expressed as

$$\text{Optimize } z = f(x_1, x_2, \ldots, x_n)$$

$$\text{Subject to } g_1(x_1, x_2, \ldots, x_n) \le or \ge or = b_1$$

$$g_2(x_1, x_2, \ldots, x_n) \le or \ge or = b_2$$

$$g_3(x_1, x_2, \ldots, x_n) \le or \ge or = b_3$$

Z is called either the objective function or fitness function. The functions g1, g2 etc. are called constraints.

If both the objective function and constraints are linear, then the problems are referred to as linear optimization.

$$f(x_1, x_2, \ldots, x_n) = c_1x_1+c_2x_2+\ldots c_nx_n$$

$$g_1(x_1, x_2, \ldots, x_n) = a_1x_1+a_2x_2+\ldots a_nx_n$$

The problem is usually expressed in matrix form and then it becomes:

$$\text{Maximize } C^TX$$

$$\text{Subject to } Ax \le b, x \ge 0$$

Where A is a m X n matrix

If there is an additional constraint that x_1, x_2 etc. can only take integer values then the problems are called integer optimization problems. If the objective function is quadratic and the constraints are linear, they are called quadratic programming problems.

Non-linear optimization is where objective function is non-linear. Quadratic is a special case of non-linear optimization problems. In quadratic optimization, objective functions take the following form

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\, x_i x_j + \sum_{i=1}^{n} d_i x_i$$

So, you have square terms in the objective function.

SETTING UP OPTIMIZATION PROBLEMS

We can build a simple framework to set up optimization problems.

- Identify and name the decision variables consistently
- Mathematically define the objective/fitness function in terms of the variables
- Identify all stipulated requirements, restrictions and limitations
- Express any hidden constraints (generally non-negative or integer only like constraints)
- Identify the class of optimization it belongs to
- Pick the solution method

In this section, let us practice setting up optimization problems. We will begin our study of LP by considering a simple example. You should not interpret this to mean that LP cannot solve more complex or realistic problems. LP has been used to solve extremely complicated problems, saving companies millions of dollars.

Case 1

Blue Ridge Hot Tubs manufactures and sells two models of hot tubs: the Aqua-Spa and the Hydro-Lux. Howie Jones, the owner and manager of the company, needs to decide how many of each type of hot tub to produce during his next production cycle. Howie buys prefabricated fiberglass hot tub shells from a local supplier and adds the pump and tubing to the shells to create his hot tubs. (This supplier has the capacity to deliver as many hot tub shells as Howie needs.) Howie installs the same type of pump into both hot tubs. He will have only 200 pumps available during his next production cycle. From a manufacturing standpoint, the main difference between the two models of hot tubs is the amount of tubing and labor required. Each Aqua-Spa requires 9 hours of labor and 12 feet of tubing. Each Hydro-Lux requires 6 hours of labor and 16 feet of tubing. Howie expects to have 1,566 production labor hours and 2,880 feet of tubing available during the next production cycle. Howie earns a profit of $350 on each Aqua-Spa he sells and $300 on each Hydro-Lux he sells. He is confident that he can sell all the hot tubs he produces. The question is, how many Aqua-Spas and Hydro-Luxes should Howie produce if he wants to maximize his profits during the next production cycle?

Identify the decision variables

In our example, the fundamental decision Howie faces is this: How many Aqua-Spas and Hydro-Luxes should be produced? In this problem, we will let X_1 represent the number of Aqua-Spas to produce and X_2 represent the number of Hydro-Luxes to produce.

State the objective function as a linear combination of the decision variables

In our example, Howie earns a profit of $350 on each Aqua-Spa ($X_1$) he sells and $300 on each Hydro-Lux ($X_2$) he sells. Thus, Howie's objective of maximizing the profit he earns is stated mathematically as:

$$\text{MAX: } 350X_1 + 300X_2$$

State the constraints as linear combinations of the decision variables.

In our example, Howie faces three major constraints. Because only 200 pumps are available and each hot tub requires one pump, Howie cannot produce more than a total of 200 hot tubs. This restriction is stated mathematically as:

$$1X_1 + 1X_2 \leq 200$$

Another restriction Howie faces is that he has only 1,566 labor hours available during the next production cycle. Because each Aqua-Spa he builds (each unit of X_1) requires 9 labor hours and each Hydro-Lux (each unit of X_2) requires 6 labor hours, the constraint on the number of labor hours is stated as:

$$9X_1 - 6X_2 \leq 1{,}566$$

The final constraint specifies that only 2,880 feet of tubing is available for the next production cycle. Each Aqua-Spa produced (each unit of X_1) requires 12 feet of tubing, and each Hydro-Lux produced (each unit of X_2) requires 16 feet of tubing. The following constraint is necessary to ensure that Howie's production plan does not use more tubing than is available:

$$12X_1 - 16X_2 \leq 2{,}880$$

Hidden constraints

In our example, there are simple lower bounds of zero on the variables X_1 and X_2 because it is impossible to produce a negative number of hot tubs. Therefore, thefollowing two constraints also apply to this problem:

$$X_1 \geq 0; \; X_2 \geq 0$$

$$\begin{array}{lrrcr}
\text{MAX:} & 350X_1 + & 300X_2 & & \\
\text{Subject to:} & 1X_1 + & 1X_2 & \leq & 200 \\
 & 9X_1 + & 6X_2 & \leq & 1{,}566 \\
 & 12X_1 + & 16X_2 & \leq & 2{,}880 \\
 & 1X_1 & & \geq & 0 \\
 & & 1X_2 & \geq & 0
\end{array}$$

A graphical solution

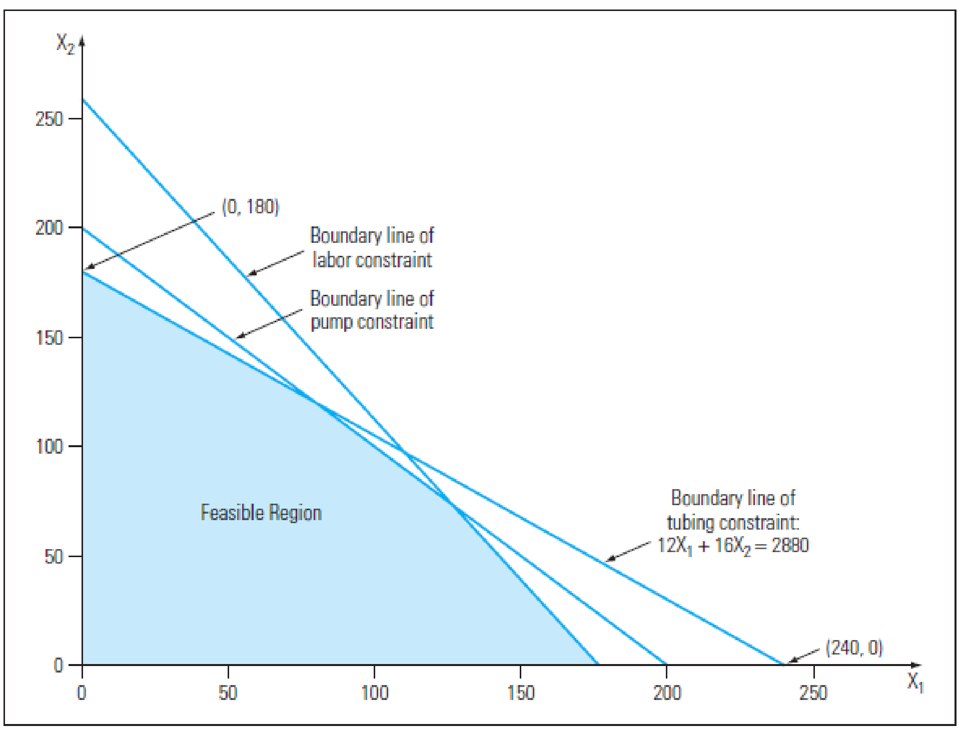

Figure 1

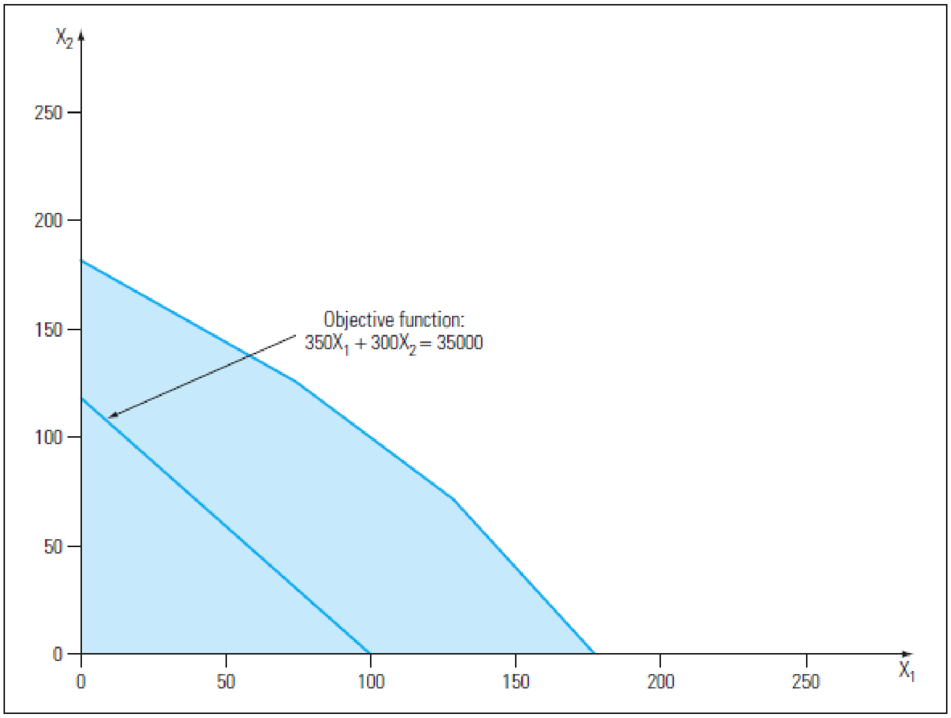

Figure 2

The lines representing the two objective function values are sometimes referred to as level curvesbecause they represent different levels or values of the objective. Note that the two level curves are parallelto one another. If we repeat this process of drawing lines associated with larger and larger values of our objective function, we will continue to observe a series of parallel lines shifting away from the origin—that is, away from the point (0, 0). The very last level curve we can draw that still intersects the feasible region will determine the maximum profit we can achieve. This point of intersection, shown in Figure 2.6, represents the optimal feasible solution to the problem.

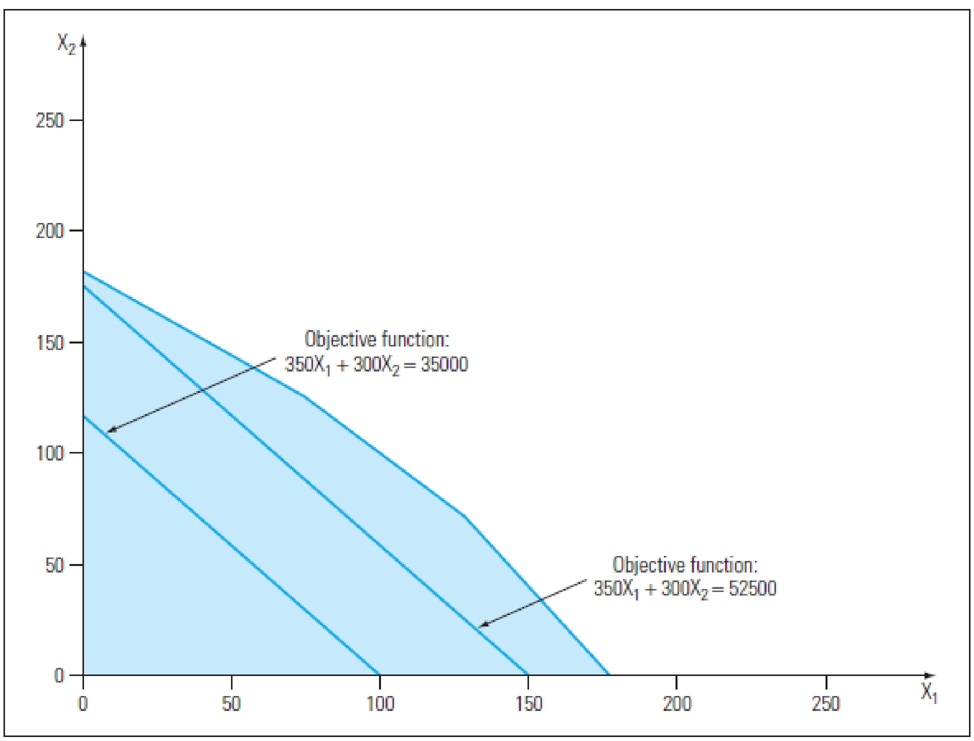

Figure 3

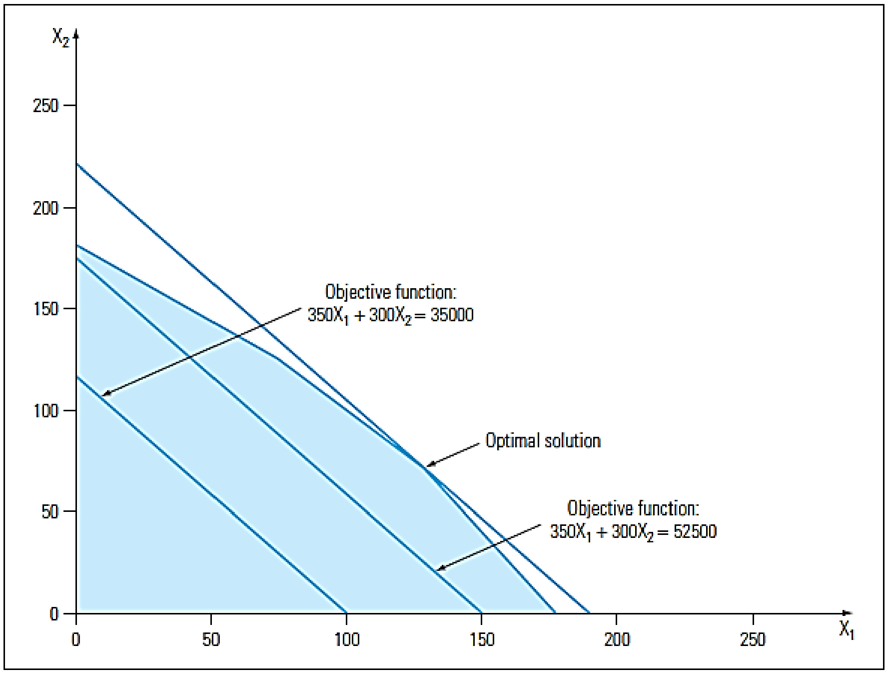

Figure 4

If an LP problem has a finite optimal solution, this solution always will occur at some corner point of the feasible region. So, another way of solving an LP problem is to identify all the corner points, or extreme points, of the feasible region and calculate the value of the objective function at each of these points. The corner point with the largest objective function value is the optimal solution to the problem.

Special Conditions in LP Models

Several special conditions can arise in LP modeling: alternate optimal solutions, redundant constraints, unbounded solutions, and infeasibility. The first two conditions do not preventyou from solving an LP model and are not really problems—they are just anomalies thatsometimes occur. On the other hand, the last two conditions represent real problemsthat prevent us from solving an LP model.

We show these issues graphically below

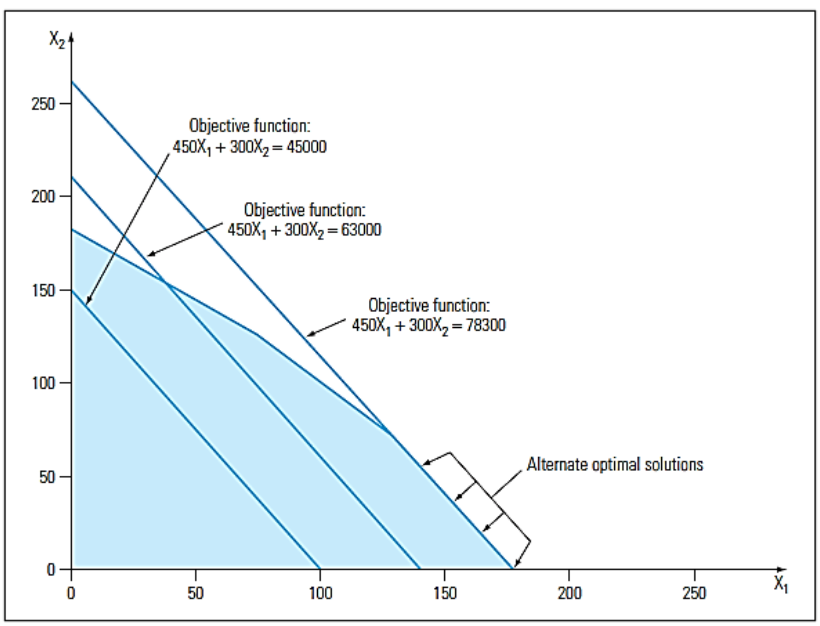

Figure 5: Alternate optimal solutions

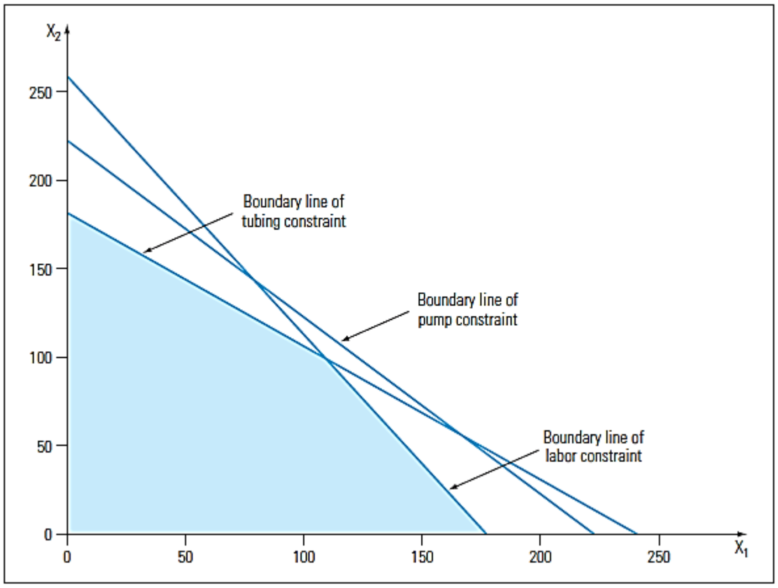

Figure 6: Redundant constraints

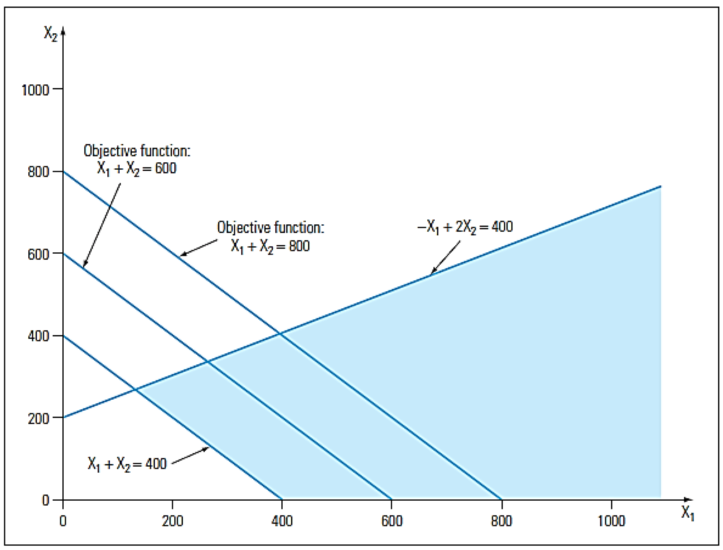

Figure 7: Unbounded Solutions

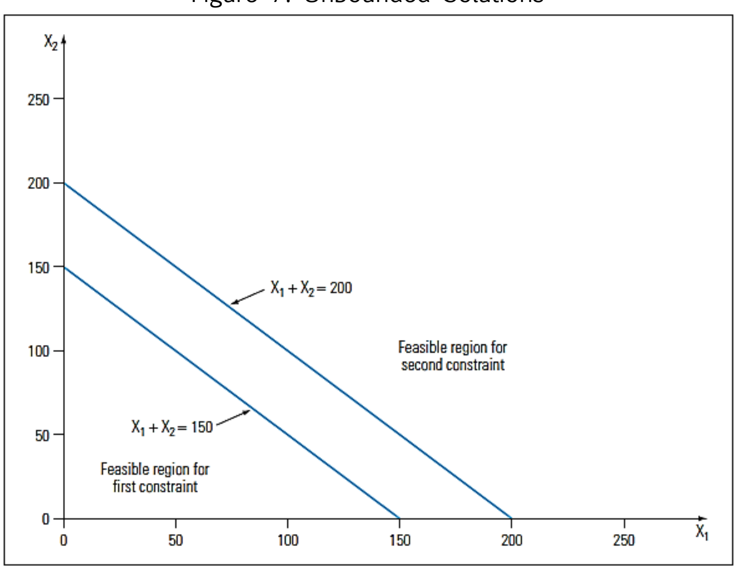

Figure 8: Non-Feasible solution

SOLVING LP USING R

install.packages("lpsolve")

library(lpsolve)

Description

Interface to lp_solve linear/integer programming system

Usage

lp (direction = "min", objective.in, const.mat, const.dir, const.rhs, transpose.constraints = TRUE, int.vec, presolve=0, compute.sens=0, binary.vec, all.int=FALSE, all.bin=FALSE, scale = 196, dense.const, num.bin.solns=1, use.rw=FALSE)

Arguments

- direction Character string giving direction of optimization: "min" (default) or "max."
- objective.in Numeric vector of coefficients of objective function
- const.mat Matrix of numeric constraint coefficients, one row per constraint, one columnper variable (unless transpose.constraints = FALSE; see below).
- const.dir Vector of character strings giving the direction of the constraint: each valueshould be one of "<," "<=," "=," "==," ">," or ">=". (In each pair the two valuesare identical.)
- const.rhs Vector of numeric values for the right-hand sides of the constraints.
- transpose.constraintsBy default each constraint occupies a row of const.mat, and that matrix needs tobe transposed before being passed to the optimizing code. For very large constraintmatrices it may be wiser to construct the constraints in a matrix columnby-column. In that case set transpose.constraints to FALSE.
- int.vec Numeric vector giving the indices of variables that are required to be integer.The length of this vector will therefore be the number of integer variables.
- presolve Numeric: presolve? Default 0 (no); any non-zero value means "yes." Currentlyignored.
- compute.sens Numeric: compute sensitivity? Default 0 (no); any non-zero value means "yes."
- binary.vec Numeric vector like int.vec giving the indices of variables that are required tobe binary.
- all.int Logical: should all variables be integer? Default: FALSE.
- all.bin Logical: should all variables be binary? Default: FALSE.
- scale Integer: value for lpSolve scaling. Details can be found in the lpSolve documentation.Set to 0 for no scaling. Default: 196
- dense.const Three column dense constraint array. This is ignored if const.mat is supplied.Otherwise the columns are constraint number, column number, and value; thereshould be one row for each non-zero entry in the constraint matrix.
- num.bin.solns Integer: if all.bin=TRUE, the user can request up to num.bin.solns optimal solutionsto be returned.
- use.rw Logical: if TRUE and num.bin.solns> 1, write the lp out to a file and read itback in for each solution after the first. This is just to defeat a bug somewhere.Although the default is FALSE, we recommend you set this to TRUE if youneed num.bin.solns> 1, until the bug is found.

Solution

$$
\begin{array}{llrcrcr}
\text{MAX:} & 350X_1 + 300X_2 & & & & & \\
\text{Subject to:} & & 1X_1 + & 1X_2 & \leq & 200 \\
 & & 9X_1 + & 6X_2 & \leq & 1{,}566 \\
 & & 12X_1 + & 16X_2 & \leq & 2{,}880 \\
 & & 1X_1 & & \geq & 0 \\
 & & & 1X_2 & \geq & 0
\end{array}
$$

As you can see, we defined the coefficients of objective function in “obj”. Similarly, the coefficients, directions and rhs of constraints are given in the next three steps.

```
> obj=c(350,300)
> con=rbind(c(1,1), c(9,6), c(12, 16), c(1,0), c(0, 1))
> dir=c("<=", "<=", "<=", ">=", ">=")
> rhs=c(200, 1566, 2880, 0, 0)
> res=lp("max", obj, con, dir, rhs)
```

“res” computes the minimum of the function using simplex method. Solutions are given as shown there.

```
> res$solution
[1] 122  78
> res$sens.coef.from
[1] 300.0000 233.3333
> res$sens.coef.to
[1] 450 350
> res$duals
[1] 200.00000  16.66667   0.00000   0.00000   0.00000   0.00000   0.00000
> res$duals.from
[1]  1.74e+02  1.44e+03 -1.00e+30 -1.00e+30 -1.00e+30 -1.00e+30 -1.00e+30
```

A FEW MORE EXAMPLES

Problem 1: A manufacturing unit has two machines separately to produce mountain bikes and race bikes. The machine for mountain bikes produces 2 bikes per day and the machine for race bikes produces 3 bikes per day. There is a common polishing machine that can polish 4 bikes a day. Profit on mountain bike is Rs. 1000 and race bike is Rs. 1500. How many bikes should the firm produce to maximize profits

Identify and name the variables: Number of mountain bikes (M) and number of race bikes (R) per day

Objective function: Maximizing profits.

Profit from each mountain bike: Rs. 1000.

Profit from M bikes: 1000M

Profit from race bike: Rs. 1500

Profit from R bikes: 1500R

Total profit: 1000M + 1500 R

Write all the specified constraints as mathematical expressions (equality or inequality) in terms of variables.

Max number of mountain bikes per day = 2; $M \leq 2$

Max number of race bikes per day = 3; $R \leq 3$

Total number of bikes that can be polished on any day = 4; $M+R \leq 4$

Write any hidden constraints (use common sense)

Obviously, the bikes cannot be negative and also can only take integer values.

So, this is a integer linear programming problem.

Problem 2: A furniture maker has 6 units of wood and 28 hours of free time. Two models were sold well in the past. Model 1 requires 2 units of wood and 7 hours and model 2 requires 1 unit of wood and 8 hours of time. Prices are Rs. 200 and 150 each. How many of each should he make to maximize the revenues?

Variables: Number of units of model 1: n1 and number of units of model 2: n2

Objective function

Revenue earned from making n1 pieces of model 1: 200n1 and

Revenues earned from making n2 pieces of model 2: 150n2

Total revenues = 200n1 + 150n2

Explicit constraints

The total wood consumed from n1 and n2 pieces: 2n1+n2

Available wood: 6

Constraint 1: 2n1+n2≤6

The total time needed to make n1 and n2 pieces: 7n1+8n2

Available time: 28

Constraint 2: 7n1+8n2 ≤ 28

Implicit constraints

n1 and n2≥ 0

This is a linear integer programming problem.

Linear and Integer Programming in R:

R-command:

lp (direction = "min", objective.in, const.mat, const.dir, const.rhs,int.vec)

This is exactly as in the previous problem. We also add the constraint of integers using int.vec

#converting data into matrix format

>obj = c(200, 150)

>con = matrix (c(2, 1, 7, 8), nrow=2, byrow=TRUE) (we could have used rbind also)

>dir = c("<=", "<=")

>rhs = c(6,28)

#Linear Programming

>lp ("max", obj, con, dir, rhs)

Success: the objective function is 677.7778

>lp ("max", f.obj, f.con, f.dir, f.rhs)$solution

[1] 2.222222 1.555556

Here, as you can see, we did not constrain it to be an integer.

>lp ("max", f.obj, f.con, f.dir, f.rhs, int.vec=1:2) #Both solutions must be integers.

Success: the objective function is 600

>lp ("max", f.obj, f.con, f.dir, f.rhs, int.vec=1:2)$solution

3 0

UNDERSTANDING SENSITIVITY AND DUALITY

Let us look at the sensitivity and duality in another problem

Let us consider a simplified model of an automobile manufacturer that produces cars and trucks. Manufacturing is organized into four departments: sheet metal stamping, engine assembly, automobile assembly, and truck assembly. The capacity of each department is limited. The following table provides the percentages of each department's monthly capacity that would be consumed by constructing a thousand cars or a thousand trucks:

Department	Automobile	Truck
metal stamping	4%	2.86%
engine assembly	3%	6%
automobile assembly	4.44%	0%
truck assembly	0%	6.67%

Table 1

The marketing department estimates a profit of \$3000 per car produced and \$2500 per truck produced. If the company decides only to produce cars, it could produce 22,500 of them, generating a total profit of \$67.5 million. On the other hand, if it only produces trucks, it can produce 15,000 of them, with a total profit of \$37.5 million. So should the company only produce cars? No. It turns out that profit can be increased if the company produces a combination of cars and trucks.

$$
\begin{array}{lll}
\text{maximize} & 3x_1 + 2.5x_2 & \text{(profit in thousands of dollars)} \\
\text{subject to} & 4.44x_1 \leq 100 & \text{(car assembly capacity)} \\
& 6.67x_2 \leq 100 & \text{(truck assembly capacity)} \\
& 4x_1 + 2.86x_2 \leq 100 & \text{(metal stamping capacity)} \\
& 3x_1 + 6x_2 \leq 100 & \text{(engine assembly capacity)} \\
& x \geq 0 & \text{(nonnegative production).}
\end{array}
$$

Written in matrix notation, the linear program becomes

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

where

$$
c = \begin{bmatrix} 3 \\ 2.5 \end{bmatrix}, \qquad A = \begin{bmatrix} 4.44 & 0 \\ 0 & 6.67 \\ 4 & 2.86 \\ 3 & 6 \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} 100 \\ 100 \\ 100 \\ 100 \end{bmatrix}
$$

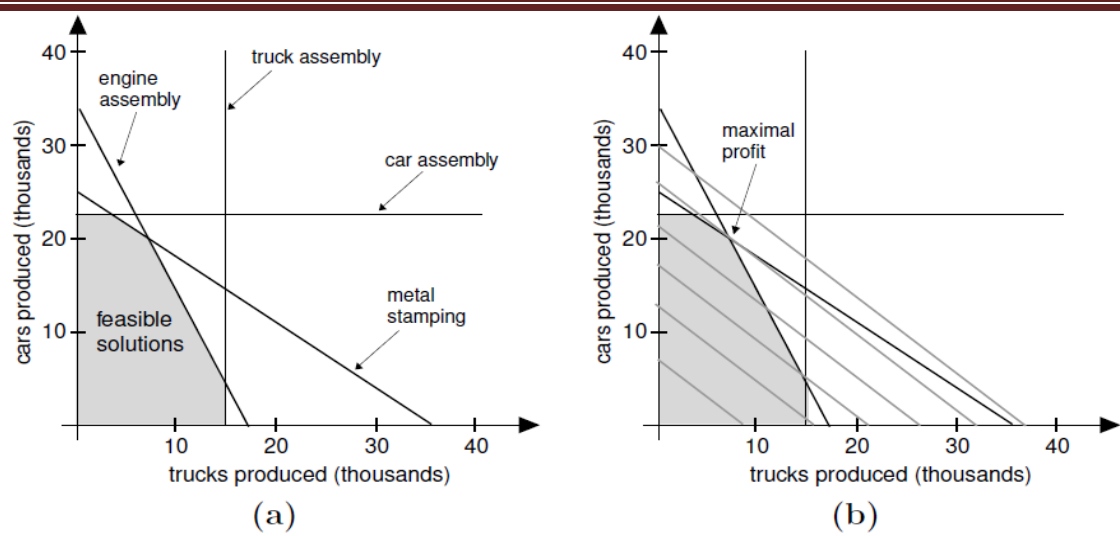

Figure 9

The *sensitivity* y_i of profit to quantity of the ith resource is the rate at which $z(\Delta)$ increases as Δ_i increases, starting from $\Delta_i = 0$. It is clear that small changes in non binding capacities do not influence profit. Hence, $y_1 = y_2 = 0$. From the preceding discussion, we have $z(\Delta) - z(0) = c^T\overline{A}^{-1}\Delta$, and therefore

$$\begin{bmatrix} y_3 & y_4 \end{bmatrix} = c^T\overline{A}^{-1} = \begin{bmatrix} 3 & 2.5 \end{bmatrix} \begin{bmatrix} 0.389 & -0.185 \\ -0.195 & 0.259 \end{bmatrix} = \begin{bmatrix} 0.681 & 0.092 \end{bmatrix}.$$

In other words, the sensitivity is about \$0.681 million per percentage ofmetal stamping capacity and \$0.092 million per percentage of engine assemblycapacity. If a 1% increase in metal stamping capacity requires the sameinvestment as a 1% increase in engine assembly, we should invest in metalstamping.

The sensitivities of profit to resource quantities are commonly called shadow prices.

A shadow price represents the maximal price at which we should be willing to buy additional units of a resource. It also represents the minimal price at which we should be willing to sell units of the resource. A shadow price might therefore be thought of as the value per unit of a resource. Remarkably, if we compute the value of our entire stock of resources based on shadow prices, we get our optimal profit!

Shadow prices solve another linear program, called the dual. In order to distinguish it from the dual, the original linear program of interest – in this case, the one involving decisions on quantities of cars and trucks to build in order to maximize profit – is called the primal.

Again, here is the primal

$$\begin{array}{lll} \text{maximize} & 3x_1 + 2.5x_2 & \text{(profit in thousands of dollars)} \\ \text{subject to} & 4.44x_1 \leq 100 & \text{(car assembly capacity)} \\ & 6.67x_2 \leq 100 & \text{(truck assembly capacity)} \\ & 4x_1 + 2.86x_2 \leq 100 & \text{(metal stamping capacity)} \\ & 3x_1 + 6x_2 \leq 100 & \text{(engine assembly capacity)} \\ & x \geq 0 & \text{(nonnegative production).} \end{array}$$

The corresponding dual is

$$\begin{array}{lll} \text{minimize} & 100y_1 + 100y_2 + 100y_3 + 100y_4 & \text{(cost of capacity)} \\ \text{subject to} & 4.44y_1 + 4y_3 + 3y_4 \geq 3 & \text{(car production)} \\ & 6.67y_2 + 2.86y_3 + 6y_4 \geq 2.5 & \text{(truck production)} \\ & y \geq 0 & \text{(nonnegative prices).} \end{array}$$

Linear programming problems are optimization problems in which the objective function and the constraints are all linear. In the primal problem, the objective function is a linear combination of *n* variables. There are *m* constraints, each of which places an upper bound on a linear combination of the *n* variables. The goal is to maximize the value of the objective function subject to the constraints. A *solution* is a vector (a list) of *n* values that achieves the maximum value for the objective function.

In the dual problem, the objective function is a linear combination of the *m* values that are the limits in the *m* constraints from the primal problem. There are *n* dual constraints, each of which places a lower bound on a linear combination of *m* dual variables.

INTEGER PROGRAMMING

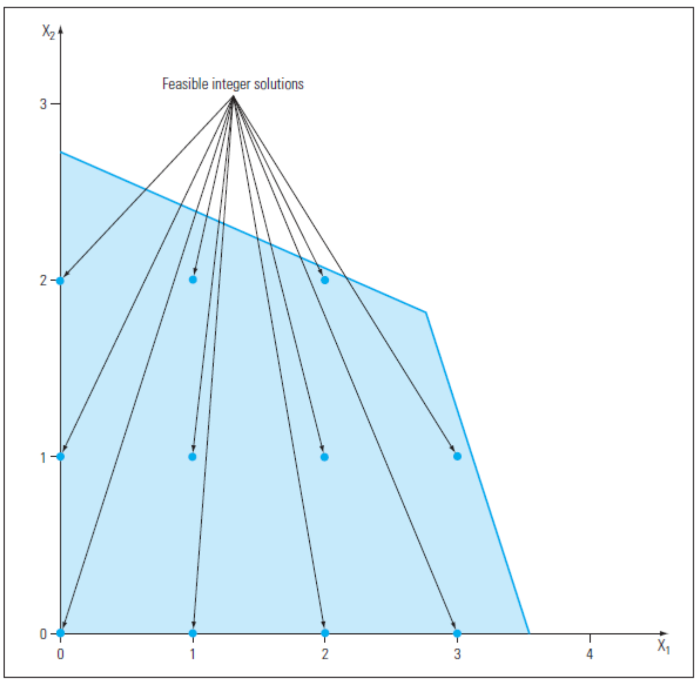

Figure 10

The search in integer programming is for only those solutions where all or some of the decision variables take integer values.

For maximizationproblems, the objective function value at the optimal solution to the LP relaxation represents an upper boundon the optimal objective function value of the original ILP problem. For minimizationproblems, the objective function value at the optimal solution to the LP relaxation represents a lower boundon the optimal objective function value of the original ILP problem.

Let us solve an integer problem using R. Suppose, for example, that Blue Ridge Hot Tubs has only 1,520 hours of labor and 2,650 feet of tubing available during its next production cycle. The company might be interested in solving the following ILP problem:

MAX:	$350X_1 + 300X_2$	} profit
Subject to:	$1X_1 + 1X_2 \leq 200$	} pump constraint
	$9X_1 + 6X_2 \leq 1{,}520$	} labor constraint
	$12X_1 + 16X_2 \leq 2{,}650$	} tubing constraint
	$X_1, X_2 \geq 0$	} nonnegativity conditions
	X_1, X_2 must be integers	} integrality conditions

```
> obj=c(350,300)
> con=rbind(c(1,1), c(9,6), c(12, 16), c(1,0), c(0, 1))
> dir=c("<=", "<=", "<=", ">=", ">=")
> rhs=c(200, 1520, 2650, 0, 0)
> res=lp("max", obj, con, dir, rhs, compute.sens=1)
> resi=lp("max", obj, con, dir, rhs, int.vec=1:2, compute.sens=1)
>
```

res runs the regular linear program and resi runs the ILP.

```
> res$solution
[1] 116.94444  77.91667
> resi$solution
[1] 118  76
>
```

APPLICATIONS OF LINEAR PROGRAMMING

ASSIGNMENT PROBLEM

Assignment problems involve scheduling workers to jobs on a one to one basis. Number of workers and jobs must be made equal through dummy variables (fictitious workers and jobs). The time that worker does a job is different from each other. Objective is to minimize it.

This is a simple variation of a LP problem. Here, demand and supply are just 1, costs are times. As a corollary, no worker can be assigned to multiple jobs.

A 400-meter medley relay involves four different swimmers, who successively swim 100 meters of the backstroke, breaststroke, butterfly and freestyle. A coach has six very fast swimmers whose expected times (in seconds) in the individual events are given in following table

	Event 1 (backstroke)	Event 2 (breaststroke)	Event 3 (butterfly)	Event 4 (freestyle)
Swimmer 1	65	73	63	57
Swimmer 2	67	70	65	58
Swimmer 3	68	72	69	55
Swimmer 4	67	75	70	59
Swimmer 5	71	69	75	57
Swimmer 6	69	71	66	59

Table 2

Integer Programming for the Assignment Problem:

R-command:

lp.assign (cost.mat, direction = "min", presolve = 0, compute.sens = 0)

Arguments

cost.mat- Matrix of costs: the ij-th element is the cost of assigning source i to destination j.

direction - Character vector, length 1, containing either "min" (the default) or "max"

presolve - Numeric: presolve? Default 0 (no); any non-zero value means "yes." Currently ignored.

compute.sens - Numeric: compute sensitivity? Default 0 (no); any non-zero value means "yes." In that case presolving is attempted.

We need to first make it a square matrix. As it is time and we are minimizing it, we add zeroes to make the matrix square.

>assign.costs = matrix(c(65,67,68,67,71,69,73,70,72,75,69,71,63,65,69,70,75,66,57,58,55,59,57,59,0,0,0,0,0,0,0,0,0,0,0,0),6,6)

>assign.costs

```
> assign.costs = matrix(c(65,67,68,67,71,69,73,70,72,75,69,71,63,65,69,70,75,66$
> assign.costs
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]   65   73   63   57    0    0
[2,]   67   70   65   58    0    0
[3,]   68   72   69   55    0    0
[4,]   67   75   70   59    0    0
[5,]   71   69   75   57    0    0
[6,]   69   71   66   59    0    0
> |
```

```
> lp.assign (assign.costs, direction = "min", compute.sens = 0)
Success: the objective function is 254
> lp.assign (assign.costs)$solution
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    1    0    0    0    0    0
[2,]    0    0    1    0    0    0
[3,]    0    0    0    1    0    0
[4,]    0    0    0    0    0    1
[5,]    0    1    0    0    0    0
[6,]    0    0    0    0    1    0
```

EMPLOYEE SCHEDULING PROBLEM

Anyone responsible for creating work schedules for several employees can appreciate the difficulties in this task. It can be very difficult to develop a feasible schedule, much less an optimal schedule. Trying to ensure that a sufficient number of workers is available when needed is a complicated task when you must consider multiple shifts, rest breaks, and lunch or dinner breaks. However, some sophisticated LP models have been devised to solve these problems. We will consider a simple example of an employee scheduling problem to give you an idea of how LP models are applied in this area.

Air-Express is an express shipping service that guarantees overnight delivery of packages anywhere in the continental United States. The company has various operations centers, called hubs, at airports in major cities across the country. Packages are received at hubs from other locations and then shipped to intermediate hubs or to their final destinations.

The manager of the Air-Express hub in Baltimore, Maryland, is concerned about labor costs at the hub and is interested in determining the most effective way to schedule workers. The hub operates seven days a week, and the number of packages it handles each day varies from one day to the next. Using historical data on the average number of packages received each day, the manager estimates the number of workers needed to handle the packages as:

Day of Week	Workers Required
Sunday	18
Monday	27
Tuesday	22
Wednesday	26
Thursday	25
Friday	21
Saturday	19

Table 3

The package handlers working for Air-Express are unionized and are guaranteed a five-day work week with two consecutive days off. The base wage for the handlers is $655 per week. Because most workers prefer to have Saturday or Sunday off, the union has negotiated bonuses of $25 per day for its members who work on these days. The possible shifts and salaries for package handlers are:

Shift	Days Off	Wage
1	Sunday and Monday	$680
2	Monday and Tuesday	$705
3	Tuesday and Wednesday	$705
4	Wednesday and Thursday	$705
5	Thursday and Friday	$705
6	Friday and Saturday	$680
7	Saturday and Sunday	$655

Table 4

The manager wants to keep the total wage expense for the hub as low as possible. With this in mind, how many package handlers should be assigned to each shift if the manager wants to have a sufficient number of workers available each day?

In this problem, the manager must decide how many workers to assign to each shift. Because there are seven possible shifts, we need the following seven decision variables:

X_1 _ the number of workers assigned to shift 1

X_2 _ the number of workers assigned to shift 2

X_3 _ the number of workers assigned to shift 3

X_4 _ the number of workers assigned to shift 4

X_5 _ the number of workers assigned to shift 5

X_6 _ the number of workers assigned to shift 6

X_7 _ the number of workers assigned to shift 7

The LP model for the Air-Express scheduling problem is summarized as:

MIN: $680X_1 + 705X_2 + 705X_3 + 705X_4 + 705X_5 + 680X_6 + 655X_7$ } total wage expense

Subject to:

$0X_1 + 1X_2 + 1X_3 + 1X_4 + 1X_5 + 1X_6 + 0X_7 \geq 18$ } workers required on Sunday
$0X_1 + 0X_2 + 1X_3 + 1X_4 + 1X_5 + 1X_6 + 1X_7 \geq 27$ } workers required on Monday
$1X_1 + 0X_2 + 0X_3 + 1X_4 + 1X_5 + 1X_6 + 1X_7 \geq 22$ } workers required on Tuesday
$1X_1 + 1X_2 + 0X_3 + 0X_4 + 1X_5 + 1X_6 + 1X_7 \geq 26$ } workers required on Wednesday
$1X_1 + 1X_2 + 1X_3 + 0X_4 + 0X_5 + 1X_6 + 1X_7 \geq 25$ } workers required on Thursday
$1X_1 + 1X_2 + 1X_3 + 1X_4 + 0X_5 + 0X_6 + 1X_7 \geq 21$ } workers required on Friday
$1X_1 + 1X_2 + 1X_3 + 1X_4 + 1X_5 + 0X_6 + 0X_7 \geq 19$ } workers required on Saturday
$X_1, X_2, X_3, X_4, X_5, X_6, X_7 \geq 0$
All X_i must be integers

PRODUCT MIX PROBLEM

National Petroleum produces two types of unleaded gasoline: regular and premium. It sells these at Rs. 600 and 800 per barrel. These are blended from their internal domestic oil and foreign oil and must meet the following constraints

	Maximum vapor pressure	Minimum octane rating	Maximum demand (barrels/wk)	Minimum deliverables (barrels/wk)
Regular	23	88	100,000	50,000
Premium	23	93	20,000	5000

Table 5

The characteristics of the refined oils in inventory are as follows

	vapor pressure	Octane rating	Inventory(barrels)	Cost(barrels)
Domestic	27	87	40,000	400
Foreign	15	98	60,000	500

Table 6

How much of regular and premium should we produce to maximize the profits

Variables

Let us say we use d1 barrels of domestic for regular oil and d2 barrels for premium oil. Similarly, we use f1 barrels of foreign for regular and f2 barrels for premium.

Objective function

Total domestic oil consumed = d1+d2

Total foreign oil consumed: f1+f2

Total regular blended: d1+f1

Total premium blended: d2+f2

The cost of domestic oil: (d1+d2)400; The cost of foreign oil: 500(f1+f2)

The price of regular oil: 600(d1+f1); The cost of premium oil: 800(d2+f2)

The profit: total price - total cost = 600(d1+f1) + 800(d2+f2) – [400(d1+d2)+500(f1+f2)]

=200d1+100f1+400d2+300f2

So, we have the objective function in terms of the variables.

Constraints

Amount of domestic oil consumed: d1+d2

Inventory available: 40,000 barrels

So, d1+d2 ≤40,000

Similarly, f1+f2 ≤ 60,000

Amount of regular produced: d1+f1

The maximum demand: 100,000 and minimum deliverables = 50000 barrels

So, 50,000 ≤ d1+f1 ≤ 100,000

Similarly, 5000≤ d2+f2≤ 20,000

Vapor pressure is based on the weight fractions

Vapor pressure of d1+f1 of regular = (weight fraction of domestic)* vapor pressure of domestic + (weight fraction of foreign)*vapor pressure of foreign

Vapor pressure of regular =

$$\frac{d1}{d1+f1}(Vapor\, pressure\, of\, domestic) + \frac{f1}{d1+f1}\,(vapor\, pressure\, of\, foreign)$$

$$\frac{d1}{d1+f1}(27) + \frac{f1}{d1+f1}\,(15) \le 23,$$

27d1+15f1 = 23d1+23f1 -> 4d1-8f1 ≤ 0

$$\text{similarly}\frac{d2}{d2+f2}(27) + \frac{f2}{d2+f2}(15) = 23$$

27d2+15f2 ≤ 23d2+23f2 ->4d2-8f2≤0

Extending the same logic to octane rating

$$\frac{d1}{d1+f1}(87) + \frac{f1}{d1+f1}(98) \ge 88$$

-d1+10f1≥ 0

$$\text{and}\frac{d2}{d2+f2}(87) + \frac{f2}{d2+f2}(98) = 93$$

-6d2+5f2≥ 0

Hidden constraints: d1, d2, f1, f2 ≥ 0.

This is a linear optimization problem.

Solution In R

obj = c(200,100,400,300) #The objective function

con = rbind(c(1,0,1,0), c(0,1,0,1),c(1,1,0,0),c(1,1,0,0),c(0,0,1,1),c(0,0,1,1),c(1,-2,0,0),c(0,0,1,-2),c(-1,10,0,0),c(0,0,-6,5), c(1,0,0,0), c(0,1,0,0), c(0,0,1,0), c(0,0,0,1))

#Coefficients for a total of 14 constraints

dir=c("<=", "<=", ">=", "<=", ">=", "<=", "<=", "<=", ">=", ">=", ">=", ">=", ">=", ">=")

#All directions

rhs = c(40000,60000,50000,100000, 5000,20000,0,0,0,0,0,0,0,0)

```
res = lp("max", obj, con, dir, rhs)

res$solution

>40000  40000      0  20000
```

STOCK SELECTION PROBLEM

Brian Givens is a financial analyst for Retirement Planning Services, Inc. who specializes in designing retirement income portfolios for retirees using corporate bonds. He has just completed a consultation with a client who expects to have $750,000 in liquid assets to invest when she retires next month. Brian and his client agreed to consider upcoming bond issues from the following six companies:

Company	Return	Years to Maturity	Rating
Acme Chemical	8.65%	11	1-Excellent
DynaStar	9.50%	10	3-Good
Eagle Vision	10.00%	6	4-Fair
MicroModeling	8.75%	10	1-Excellent
OptiPro	9.25%	7	3-Good
Sabre Systems	9.00%	13	2-Very Good

Table 7

The column labeled "Return" in this table represents the expected annual yield on each bond, the column labeled "Years to Maturity" indicates the length of time over which the bonds will be payable, and the column labeled "Rating" indicates an independent underwriter's assessment of the quality or risk associated with each issue.

Brian believes that all of the companies are relatively safe investments. However, to protect his client's income, Brian and his client agreed that no more than 25% of her money should be invested in any one investment and at least half of her money should be invested in long-term bonds that mature in ten or more years. Also, even though DynaStar, Eagle Vision, and OptiPro offer the highest returns, it was agreed that no more than 35% of the money should be invested in these bonds because they also represent the highest risks (i.e., they were rated lower than "very good").

Brian needs to determine how to allocate his client's investments to maximize her income while meeting their agreed-upon investment restrictions.

Decision variables

- X_1 = amount of money to invest in Acme Chemical
- X_2 = amount of money to invest in DynaStar
- X_3 = amount of money to invest in Eagle Vision
- X_4 = amount of money to invest in MicroModeling
- X_5 = amount of money to invest in OptiPro
- X_6 = amount of money to invest in Sabre Systems

The objective in this problem is to maximize the investment income for Brian's client. Because each dollar invested in Acme Chemical (X_1) earns 8.65% annually, each dollar invested in DynaStar (X_2) earns 9.50%, and so on, the objective function for the problem is expressed as:

MAX: $.0865X_1 + .095X_2 + .10X_3 + .0875X_4 + .0925X_5 + .09X_6$ -------------total annual return

Subject to:

$X_1 \leq 187{,}500$	} 25% restriction per investment
$X_2 \leq 187{,}500$	} 25% restriction per investment
$X_3 \leq 187{,}500$	} 25% restriction per investment
$X_4 \leq 187{,}500$	} 25% restriction per investment
$X_5 \leq 187{,}500$	} 25% restriction per investment
$X_6 \leq 187{,}500$	} 25% restriction per investment
$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 = 750{,}000$	} total amount invested
$X_1 + X_2 + X_4 + X_6 \geq 375{,}000$	} long-term investment
$X_2 + X_3 + X_5 \leq 262{,}500$	} higher-risk investment
$X_1, X_2, X_3, X_4, X_5, X_6 \geq 0$	} nonnegativity conditions

TRANSPORTATION PROBLEM

Tropicsun currently has 275,000 bags of citrus at Mt. Dora, 400,000 bags at Eustis, and 300,000 bags at Clermont. Tropicsun has citrus processing plants in Ocala, Orlando, and Leesburg with processing capacities to handle 200,000, 600,000, and 225,000 bags, respectively.

Tropicsun contracts with a local trucking company to transport its fruit from the groves to the processing plants. The trucking company charges a flat rate for every mile that each bushel of fruit must be transported. Each mile a bushel of fruit travels is known as a bushel-mile. The following table summarizes the distances (in miles) between the groves and processing plants:

	Distances (in miles) Between Groves and Plants		
Grove	**Ocala**	**Orlando**	**Leesburg**
Mt. Dora	21	50	40
Eustis	35	30	22
Clermont	55	20	25

Table 8

Tropicsun wants to determine how many bushels to ship from each grove toeach processing plant to minimize the total number of bushel-miles the fruit must beshipped.

MIN:

$$\left.\begin{array}{l} 21X_{14} + 50X_{15} + 40X_{16} + \\ 35X_{24} + 30X_{25} + 22X_{26} + \\ 55X_{34} + 20X_{35} + 25X_{36} \end{array}\right\} \text{total distance fruit is shipped (in bushel-miles)}$$

Subject to:

$X_{14} + X_{24} + X_{34} \leq 200{,}000$	} capacity restriction for Ocala
$X_{15} + X_{25} + X_{35} \leq 600{,}000$	} capacity restriction for Orlando
$X_{16} + X_{26} + X_{36} \leq 225{,}000$	} capacity restriction for Leesburg
$X_{14} + X_{15} + X_{16} = 275{,}000$	} supply available at Mt. Dora
$X_{24} + X_{25} + X_{26} = 400{,}000$	} supply available at Eustis
$X_{34} + X_{35} + X_{36} = 300{,}000$	} supply available at Clermont
$X_{ij} \geq 0$, for all i and j	} nonnegativity conditions

This class of problems are so important that there is a specific library called lp.transport. It requires slightly different formulation.

You start with the cost matrix as above but add dummy source or receiver to ensure that demand = supply

	Ocala	Orlando	Leesburg	Supply available
Mt. Dora	21	50	40	275000
Eustis	35	30	22	400000
Clermont	55	20	25	300000
Dummy	0	0	0	50000
Capacities	200000	600000	225000	

Table 9

PRODUCTION PROBLEM

The Upton Corporation manufactures heavy-duty air compressors for the home andlight industrial markets. Upton is presently trying to plan its production and inventorylevels for the next six months. Because of seasonal fluctuations in utility andraw material costs, the per unit cost of producing air compressors varies from monthto month—as does the demand for air compressors. Production capacity also variesfrom month to month due to differences in the number of working days, vacations,and scheduled maintenance and training. The following table summarizes themonthly production costs, demands, and production capacity Upton's managementexpects to face over the next six months.

	Month					
	1	2	3	4	5	6
Unit Production Cost	$ 240	$ 250	$ 265	$ 285	$ 280	$ 260
Units Demanded	1,000	4,500	6,000	5,500	3,500	4,000
Maximum Production	4,000	3,500	4,000	4,500	4,000	3,500

Table 10

Given the size of Upton's warehouse, a maximum of 6,000 units can be held in inventoryat the end of any month. The owner of the company likes to keep at least 1,500units in inventory as safety stock to meet unexpected demand contingencies. To maintaina stable workforce, the company wants to produce no less than one half of its maximumproduction capacity each month. Upton's controller estimates that the cost ofcarrying a unit in any given month is approximately equal to 1.5% of the unit productioncost in the same month. Upton estimates the number of units carried in inventoryeach month by averaging the beginning and ending inventory for each month.There are 2,750 units currently in inventory. Upton wants to identify theproduction and inventory plan for the next six months that will meet the expecteddemand each month while minimizing production and inventory costs.

The objective in this problem is to minimize the total production and inventory costs.

The total production cost is computed easily as:

Production Cost = 240P1 + 250P2 + 265P3 + 285P4 + 280P5 + 260P6

The inventory cost is tricky to compute. The cost of holding a unit in inventory each month is 1.5% of the production cost in the same month. So, the unit inventory cost is $3.60 in month 1 (i.e., 1.5% x $240 = $3.60), $3.75 in month 2 (i.e., 1.5% x $250 = $3.75), and so on. The number of units held each month is to be computed as the average of the beginning and ending inventory for the month. Of course, the beginning inventory in any given month is equal to the ending inventory from the previous month. So let B*i* represent the beginning inventory for month *i*.

Inventory Cost = 3.6(B1 + B2)/2 + 3.75(B2 + B3)/2 + 3.98(B3 + B4)/2+ 4.28(B4 + B5)/2 + 4.20(B5 + B6)/2 + 3.9(B6 + B7)/2

There are three sets of constraints that apply to this problem. First, the number of units produced each month cannot exceed the maximum production levels stated in the problem. However, we also must make sure that the number of units produced each month is no less than one half of the maximum production capacity for the month. These conditions can be expressed concisely as follows:

$$
\begin{array}{ll}
2{,}000 \le P_1 \le 4{,}000 & \text{\} production level for month 1} \\
1{,}750 \le P_2 \le 3{,}500 & \text{\} production level for month 2} \\
2{,}000 \le P_3 \le 4{,}000 & \text{\} production level for month 3} \\
2{,}250 \le P_4 \le 4{,}500 & \text{\} production level for month 4} \\
2{,}000 \le P_5 \le 4{,}000 & \text{\} production level for month 5} \\
1{,}750 \le P_6 \le 3{,}500 & \text{\} production level for month 6}
\end{array}
$$

We must ensure that the ending inventory each month falls between the minimum and maximum allowable inventory levels of 1,500 and 6,000, respectively. In general, the ending inventory for anymonth is computed as:

Ending Inventory = Beginning Inventory + Units Produced – Units Sold

Thus, the following restrictions indicate that the ending inventory in each of the next six months (after meeting the demand for the month) must fall between 1,500 and 6,000.

$$
\begin{array}{ll}
1{,}500 \le B_1 + P_1 - 1{,}000 \le 6{,}000 & \text{\} ending inventory for month 1} \\
1{,}500 \le B_2 + P_2 - 4{,}500 \le 6{,}000 & \text{\} ending inventory for month 2} \\
1{,}500 \le B_3 + P_3 - 6{,}000 \le 6{,}000 & \text{\} ending inventory for month 3} \\
1{,}500 \le B_4 + P_4 - 5{,}500 \le 6{,}000 & \text{\} ending inventory for month 4} \\
1{,}500 \le B_5 + P_5 - 3{,}500 \le 6{,}000 & \text{\} ending inventory for month 5} \\
1{,}500 \le B_6 + P_6 - 4{,}000 \le 6{,}000 & \text{\} ending inventory for month 6}
\end{array}
$$

Finally, to ensure that the beginning balance in one month equals the ending balance from the previous month, we have the following additional restrictions:

$$
\begin{aligned}
B_2 &= B_1 + P_1 - 1{,}000 \\
B_3 &= B_2 + P_2 - 4{,}500 \\
B_4 &= B_3 + P_3 - 6{,}000 \\
B_5 &= B_4 + P_4 - 5{,}500 \\
B_6 &= B_5 + P_5 - 3{,}500 \\
B_7 &= B_6 + P_6 - 4{,}000
\end{aligned}
$$

$$
\text{MIN:} \quad \left.\begin{array}{l}
240P_1 + 250P_2 + 265P_3 + 285P_4 + 280P_5 + 260P_6 \\
+ 3.6(B_1 + B_2)/2 + 3.75(B_2 + B_3)/2 + 3.98(B_3 + B_4)/2 \\
+ 4.28(B_4 + B_5)/2 + 4.20(B_5 + B_6)/2 + 3.9(B_6 + B_7)/2
\end{array}\right\} \text{ total cost}
$$

$$
\begin{array}{lll}
\text{Subject to:} & 2{,}000 \le P_1 \le 4{,}000 & \text{\} production level for month 1} \\
& 1{,}750 \le P_2 \le 3{,}500 & \text{\} production level for month 2} \\
& 2{,}000 \le P_3 \le 4{,}000 & \text{\} production level for month 3} \\
& 2{,}250 \le P_4 \le 4{,}500 & \text{\} production level for month 4} \\
& 2{,}000 \le P_5 \le 4{,}000 & \text{\} production level for month 5} \\
& 1{,}750 \le P_6 \le 3{,}500 & \text{\} production level for month 6} \\
& 1{,}500 \le B_1 + P_1 - 1{,}000 \le 6{,}000 & \text{\} ending inventory for month 1} \\
& 1{,}500 \le B_2 + P_2 - 4{,}500 \le 6{,}000 & \text{\} ending inventory for month 2} \\
& 1{,}500 \le B_3 + P_3 - 6{,}000 \le 6{,}000 & \text{\} ending inventory for month 3} \\
& 1{,}500 \le B_4 + P_4 - 5{,}500 \le 6{,}000 & \text{\} ending inventory for month 4} \\
& 1{,}500 \le B_5 + P_5 - 3{,}500 \le 6{,}000 & \text{\} ending inventory for month 5} \\
& 1{,}500 \le B_6 + P_6 - 4{,}000 \le 6{,}000 & \text{\} ending inventory for month 6}
\end{array}
$$

where:

$$B_2 = B_1 + P_1 - 1{,}000$$
$$B_3 = B_2 + P_2 - 4{,}500$$
$$B_4 = B_3 + P_3 - 6{,}000$$
$$B_5 = B_4 + P_4 - 5{,}500$$
$$B_6 = B_5 + P_5 - 3{,}500$$
$$B_7 = B_6 + P_6 - 4{,}000$$

PRODUCTION PROBLEM 2

An industrial firm must plan for each of the four seasons over the next year. The company's production capacities and the expected demands (all in units) are as follows:

	Spring	Summer	Fall	Winter
Demand	250	100	400	500
Regular Capacity	200	300	350	---
Overtime Capacity	100	50	100	150

Table 11

Regular production costs for the firm are \$7.00 per unit. The unit cost of overtime varies seasonally being \$8.00 in spring and fall, \$9.00 in summer and \$10.00 in winter.

The company has 200 units of inventory on January 1, but as it plans to discontinue the product at the end of the year, it wants no inventory after the winter season. Units produced on regular shifts are not available for shipment during the season of production; generally, they are sold during the following season. Those that are not are added to inventory and carried forward at a cost of \$0.70 per unit per season. In contrast, units produced on overtime shifts must be shipped in the same season as produced. Determine a production schedule that meets all demands at minimum total cost.

The cost matrix is formed as given below

Costs						Supply
From/To	Spring	Summer	Fall	Winter	Dummy	
RegSpr	10000	7	7.7	8.4	0	200
RegSum	10000	10000	7	7.7	0	300
RegFall	10000	10000	10000	7	0	350
Initial	0	0.7	1.4	2.1	10000	200
OTSpr	8	10000	10000	10000	0	100
OTSum	10000	9	10000	10000	0	50
OTFall	10000	10000	8	10000	0	100
OTWinter	10000	10000	10000	10	0	150
Demand	250	100	400	500	200	

Table 12

CAPITAL BUDGETING PROBLEM

In his position as vice president of research and development (R&D) for CRT Technologies,Mark Schwartz is responsible for evaluating and choosing which R&Dprojects to support. The company received 18 R&D proposals from its scientists andengineers, and identified six projects as being consistent with the company's mission.However, the company does not have the funds available to undertake all sixprojects. Mark must determine which of the projects to select. The funding requirementsfor each project are summarized in the following table along with the NPVthe company expects each project to generate.

Project	Expected NPV (in $1,000s)	Capital (in $1,000s) Required in Year 1	Year 2	Year 3	Year 4	Year 5
1	$141	$ 75	$25	$20	$15	$10
2	$187	$ 90	$35	$ 0	$ 0	$30
3	$121	$ 60	$15	$15	$15	$15
4	$ 83	$ 30	$20	$10	$ 5	$ 5
5	$265	$100	$25	$20	$20	$20
6	$127	$ 50	$20	$10	$30	$40

Table 13

The company currently has $250,000 available to invest in new projects. It has budgeted$75,000 for continued support for these projects in year 2 and $50,000 per yearfor years 3, 4, and 5. Surplus funds in any year are re-appropriated for other useswithin the company and may not be carried over to future years.

$$
\begin{array}{ll}
\text{MAX:} & 141X_1 + 187X_2 + 121X_3 + 83X_4 + 265X_5 + 127X_6 \\
\text{Subject to:} & 75X_1 + 90X_2 + 60X_3 + 30X_4 + 100X_5 + 50X_6 \le 250 \\
& 25X_1 + 35X_2 + 15X_3 + 20X_4 + 25X_5 + 20X_6 \le 75 \\
& 20X_1 + 0X_2 + 15X_3 + 10X_4 + 20X_5 + 10X_6 \le 50 \\
& 15X_1 + 0X_2 + 15X_3 + 5X_4 + 20X_5 + 30X_6 \le 50 \\
& 10X_1 + 30X_2 + 15X_3 + 5X_4 + 20X_5 + 40X_6 \le 50 \\
& \text{All } X_i \text{ must be binary}
\end{array}
$$

MINIMUM SPANNING

Jon Fleming is responsible for setting up a local area network (LAN) in the designengineering department of Windstar Aerospace Company. A LAN consists of anumber of individual computers connected to a centralized computer or file server.Each computer in the LAN can access information from the file server and communicatewith the other computers in the LAN.Installing a LAN involves connecting all the computers together with communicationscables. Not every computer has to be connected directly to the file server,but there must be some link between each computer in the network. Figure below summarizes all the possible connections that Jon could make. Each node in this figurerepresents one of the computers to be included in the LAN. Each line connectingthe nodes represents a possible connection between pairs of computers. The dollaramount on each line represents the cost of making the connection.

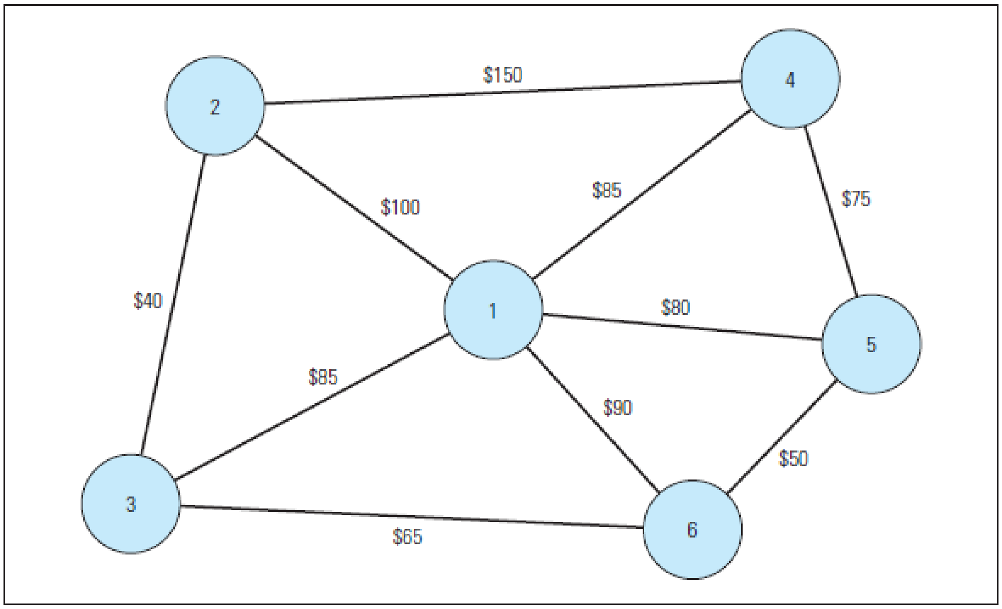

Figure 11

The arcs have no specific directional orientation, indicating that informationcan move in either direction across the arcs. Also note that the communication links represented by the arcs do not exist yet. Jon's challenge is to determine which links to establish. Because the network involves n = 6 nodes, a spanning tree for this problem consists of $n - 1 = 5$ arcs that results in a path existing between any pair of nodes. The objective is to find the minimal (least costly) spanning tree for this problem.

You can apply a simple algorithm to solve minimal spanning tree problems. The stepsto this algorithm are:1. Select any node. Call this the current sub-network.2. Add to the current sub-network the cheapest arc that connects any node within thecurrent sub-network to any node not in the current sub-network. (Ties for the cheapestarc can be broken arbitrarily.)

Call this the current sub-network.3. If all the nodes are in the sub-network, stop; this is the optimal solution. Otherwise,return to step 2.

- **Step 1.** If we select node 1 inFigure 11, then node 1 is the current sub-network.
- **Step 2.** The cheapest arc connecting the current sub-network to a node not in the currentsub-network is the $80 arc connecting nodes 1 and 5. This arc and node 5are added to the current sub-network.
- **Step 3.** Four nodes (nodes 2, 3, 4, and 6) remain unconnected—therefore, return tostep 2.
- **Step 2.** The cheapest arc connecting the current sub-network to a node not in the current sub-network is the $50 arc connecting nodes 5 and 6. This arc and node 6are added to the current sub-network.
- **Step 3.** Three nodes (nodes 2, 3, and 4) remain unconnected—therefore, return tostep 2.
- **Step 2.** The cheapest arc connecting the current sub-network to a node not in the current sub-network is the $65 arc connecting nodes 6 and 3. This arc and node 3are added to the current sub-network.
- **Step 3.** Two nodes (nodes 2 and 4) remain unconnected—therefore, return to step 2.
- **Step 2.** The cheapest arc connecting the current sub-network to a node not in the current sub-network is the $40 arc connecting nodes 3 and 2. This arc and node 2are added to the current sub-network.
- **Step 3.** One node (node 4) remains unconnected—therefore, return to step 2.
- **Step 2.** The cheapest arc connecting the current sub-network to a node not in the current sub-network is the $75 arc connecting nodes 5 and 4. This arc and node 4are added to the current sub-network.
- **Step 3.** All the nodes are now connected. Stop; the current sub-network is optimal.

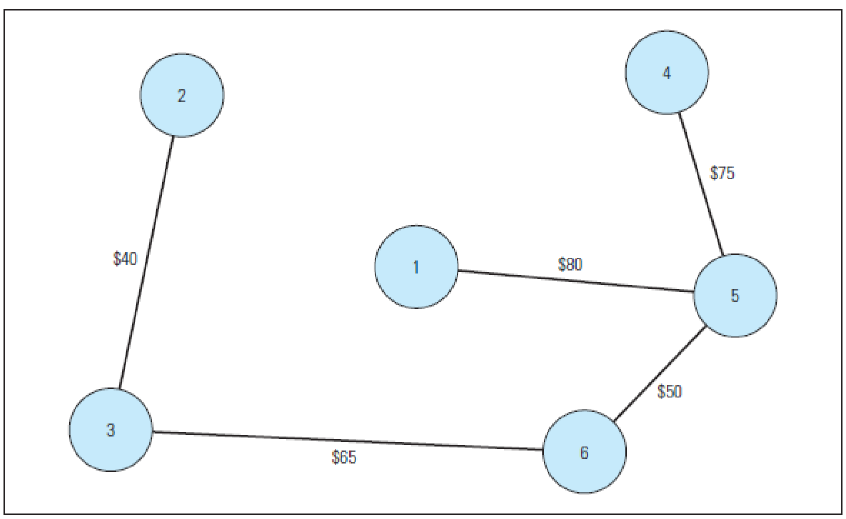

Figure 12

DATA ENVELOPMENT ANALYSIS

Managers often are interested in determining how efficiently various units within acompany operate. Similarly, investment analysts might be interested in comparing theefficiency of several competing companies within an industry. Data EnvelopmentAnalysis (DEA) is an LP-based methodology for performing this type of analysis. DEAdetermines how efficiently an operating unit (or company) converts inputs to outputswhen compared with other units. We will consider how DEA may be applied via the following example.

Mike Lister is a district manager for the Steak & Burger fast-food restaurant chain.The region Mike manages contains 12 company-owned units. Mike is in the processof evaluating the performance of these units during the past year to make recommendationson how much of an annual bonus to pay each unit's manager. He wantsto base this decision, in part, on how efficiently each unit has been operated. Mikehas collected the data shown in the following table on each of the 12 units. The outputshe has chosen includes each unit's net profit (in $100,000s), average customersatisfaction rating, and average monthly cleanliness score. The inputs include total labor hours (in 100,000s) and total operating costs (in $1,000,000s). He wants toapply DEA to this data to determine an efficiency score of each unit.

	Outputs			Inputs	
Unit	Profit	Satisfaction	Cleanliness	Labor Hours	Operating Costs
1	5.98	7.7	92	4.74	6.75
2	7.18	9.7	99	6.38	7.42
3	4.97	9.3	98	5.04	6.35
4	5.32	7.7	87	3.61	6.34
5	3.39	7.8	94	3.45	4.43
6	4.95	7.9	88	5.25	6.31
7	2.89	8.6	90	2.36	3.23
8	6.40	9.1	100	7.09	8.69
9	6.01	7.3	89	6.49	7.28
10	6.94	8.8	89	7.36	9.07
11	5.86	8.2	93	5.46	6.69
12	8.35	9.6	97	6.58	8.75

Table 14

$$\text{Efficiency of unit } i = \frac{\text{Weighted sum of unit } i\text{'s outputs}}{\text{Weighted sum of unit } i\text{'s inputs}} = \frac{\sum_{j=1}^{n_O} O_{ij} w_j}{\sum_{j=1}^{n_I} I_{ij} v_j}$$

Here, O_{ij} represents the value of unit *i* on *output j*, I_{ij} represents the value of unit *i* on *input j*, w_j is a nonnegative weight assigned to output *j*, v_j is a nonnegative weight assigned to input *j*, n_O is the number of output variables, and n_I is the number of input variables. The problem in DEA is to determine values for the weights w_j and v_j. Thus, w_j and v_j represent the decision variables in a DEA problem.

A separate LP problem is solved for each unit in a DEA problem. However, for each unit the objective is the same: to maximize the weighted sum of that unit's outputs. For an arbitrary unit *i*, the objective is stated as:

$$\text{MAX: } \sum_{j=1}^{n_O} O_{ij} w_j$$

Thus, as each LP problem is solved, the unit under investigation is given the opportunity to select the best possible weights for itself (or the weights that maximize the weighted sum of its output), subject to the following constraints.

When applying DEA, it is assumed that for output variables "more is better" (e.g.,profit) and for input variables "less is better" (e.g., costs). Any output or inputvariables that do not naturally conform to these rules should be transformed beforeapplying DEA. For example, the percentage of defective products produced isnot a good choice for an output because fewer defects is actually a good thing.However, the percentage of non-defective products produced would be an acceptablechoice for an output because "more is better" in that case.

MAX:	$5.98w_1 + 7.7w_2 + 92w_3$	} weighted output for unit 1
Subject to:	$5.98w_1 + 7.7w_2 + 92w_3 - 4.74v_1 - 6.75v_2 \leq 0$	} efficiency constraint for unit 1
	$7.18w_1 + 9.7w_2 + 99w_3 - 6.38v_1 - 7.42v_2 \leq 0$	} efficiency constraint for unit 2
	and so on to . . .	
	$8.35w_1 + 9.6w_2 + 97w_3 - 6.58v_1 - 8.75v_2 \leq 0$	} efficiency constraint for unit 12
	$4.74v_1 + 6.75v_2 = 1$	} input constraint for unit 1
	$w_1, w_2, w_3, v_1, v_2 \geq 0$	} nonnegativity conditions

GOAL PROGRAMMING

Goalprogramming—involves solving problems containing not one specific objective function,but rather a collection of goals that we would like to achieve. As we will see, a goal canbe viewed as a constraint with a flexible, or soft, Right-Hand Side (RHS) value.Hard constraints are appropriate in many situations; however, these constraints mightbe too restrictive in other situations. For example, when you buy a new car, you probablyhave in mind a maximum purchase price that you do not want to exceed. We might callthis your goal. However, you probably will find a way to spend more than this amountif it is impossible to acquire the car you really want for your goal amount. So, the goalyou have in mind is *not* a hard constraint that cannot be violated. We might view it moreaccurately as a **soft constraint**—representing a target that you would like to achieve.

Numerous managerial decision-making problems can be modeled more accuratelyusing goals rather than hard constraints. Often, such problems do not have one explicitobjective function to be maximized or minimized over a constraint set but, instead, canbe stated as a collection of goals that also might include hard constraints. These types ofproblems are known as **goal programming** (GP)

Problem

Davis McKeown is the owner of a resort hotel and convention center in Myrtle Beach,South Carolina. Although his business is profitable, it is also highly seasonal; thesummer months are the most profitable time of year. To increase profits during therest of the year, Davis wants to expand his convention business but, to do so, he needsto expand his conference facilities. Davis hired a marketing research firm to determinethe number and sizes of conference rooms that would be required by the conventionshe wants to attract. The results of this study indicated that Davis's facilities should include at least 5 small (400 square foot) conference rooms, 10 medium(750 square foot) conference rooms, and 15 large (1,050 square foot) conferencerooms. Additionally, the marketing research firm indicated that if the expansion consistedof a total of 25,000 square feet, Davis would have the largest convention centeramong his competitors—which would be desirable for advertising purposes. Whilediscussing his expansion plans with an architect, Davis learned that he can expect topay $18,000 for each small conference room in the expansion, $33,000 for each mediumconference room, and $45,150 for each large conference room. Davis wants to limit hisexpenditures on the convention center expansion to approximately $1,000,000.

DEFINING THE DECISION VARIABLES

In this problem, the fundamental decision facing the hotel owner is how many small,medium, and large conference rooms to include in the conference center expansion.These quantities are represented by X1, X2, and X3, respectively.

DEFINING THE GOALS

This problem is somewhat different from the problems presented earlier.Rather than one specific objective, this problem involves several goals, which are stated(in no particular order) as:

- Goal 1: The expansion should include approximately 5 small conference rooms.
- Goal 2: The expansion should include approximately 10 medium conference rooms.
- Goal 3: The expansion should include approximately 15 large conference rooms.
- Goal 4: The expansion should consist of approximately 25,000 square feet.
- Goal 5: The expansion should cost approximately $1,000,000.

Notice that the word "approximately" appears in each goal. This word reinforcesthat these are soft goals rather than hard constraints. For example, if the first fourgoals could be achieved at a cost of $1,001,000, it is very likely that the hotel ownerwould not mind paying an extra $1,000 to achieve such a solution. However, we mustdetermine if we can find a solution that exactly meets all of the goals in this problemand, if not, what trade-offs can be made among the goals to determine an acceptablesolution. We can formulate an LP model for this GP problem to help us make thisdetermination.

DEFINING THE GOAL CONSTRAINTS The first step in formulating an LP model for a GP problem is to create a goal constraintfor each goal in the problem. A**goal constraint** allows us to determine how close a givensolution comes to achieving the goal. To understand how these constraints should beformulated, let's begin with the three goal constraints associated with the number ofsmall, medium, and large conference rooms in the expansion.If we wanted to make sure that *exactly* 5 small, 10 medium, and 15 large conferencerooms were included in the planned expansion, we would include the following hardconstraints in our GP modelX1 = 5X2 = 10X3 = 15However, the goals stated that the expansion should include *approximately* 5 smallconference rooms, *approximately* 10 medium conference rooms, and *approximately* 15large conference rooms. If it is impossible to achieve all the goals, the hotel owner mightconsider a solution involving only 14 large conference rooms. The hard constraintswould not allow for such a solution; they are too restrictive.

However, we can modifythem easily to allow for departures from the stated goals, as:

$$\begin{aligned}
&X_1 + d_1^- - d_1^+ = 5 && \text{\} small rooms} \\
&X_2 + d_2^- - d_2^+ = 10 && \text{\} medium rooms} \\
&X_3 + d_3^- - d_3^+ = 15 && \text{\} large rooms} \\
&\text{where } d_i^-, d_i^+ \geq 0 \text{ for all } i
\end{aligned}$$

The RHS value of each goal constraint (the values 5, 10, and 15 in the previousconstraints) is the **target value** for the goal because it represents the level of achievementthat the decision maker wants to obtain for the goal. The variables *d i*– and *di*+ are called**deviational variables** because they represent the amount by which each goal deviatesfrom its target value. The *di* _ represents the amount by which each goal's target valueis *underachieved*, and the *di*+ represents the amount by which each goal's target value is*overachieved*.

To illustrate how deviational variables work, suppose that we have a solution whereX1 = 3, X2 = 13, and X3 = 15. To satisfy the first goal constraint listed previously, its deviationalvariables would assume the values *d*1 _ = 2 and *d*1 + = 0 to reflect that the goal ofhaving 5 small conference rooms is *underachieved* by 2. Similarly, to satisfy the secondgoal constraint, its deviational variables would assume the values *d* 2– = 0 and *d* 2+ = 3to reflect that the goal of having 10 medium conference rooms is *overachieved* by 3.Finally, to satisfy the third goal constraint, its deviational variables would assume thevalues *d*3– = 0 and *d*3+ = 0, to reflect that the goal of having 15 medium conference roomsis *exactly* achieved.

We can formulate the goal constraints for the remaining goals in the problem in a similarmanner. Because each small, medium, and large conference room requires 400, 750,and 1,050 square feet, respectively, and the hotel owner wants the total square footage ofthe expansion to be 25,000, the constraint representing this goal is:

$400X1 + 750X2 + 1{,}050X3 + d4^{-} - d4^{+} = 25{,}000$} square footage

Because each small, medium, and large conference room results in building costs of$18,000, $33,000, and $45,150, respectively, and the hotel owner wants to keep the costof the expansion at approximately $1,000,000, the constraint representing this goal is:

$18{,}000X1 + 33{,}000X2 + 45{,}150X3 + d5^{-} - d5^{+} = 1{,}000{,}000$} building cost

The deviational variables in each of these goal constraints represent the amounts bywhich the actual values obtained for the goals deviate from their respective targetvalues.

DEFINING THE HARD CONSTRAINTS

As noted earlier, not all of the constraints in a GP problem have to be goal constraints. AGP problem also can include one or more hard constraints typically found in LP problems.In our example, if $1,000,000 was the absolute maximum amount that the hotelowner was willing to spend on the expansion, this could be included in the model as ahard constraint. (As we'll see, it is also possible to change a soft constraint into a hardconstraint during the analysis of a GP problem.)

Although it is fairly easy to formulate the constraints for a GP problem, identifying an appropriate objective function can be quite tricky and usually requires some mental effort. Before formulating the objective function for our sample problem, let's consider some of the issues and options involved in this process.

The objective in a GP problem is to determine a solution that achieves all the goals as closely as possible. The *ideal* solution to any GP problem is one in which each goal is achieved exactly at the level specified by its target value. (In such an ideal solution, all the deviational variables in all the goal constraints would equal 0.) Often, it is not possible to achieve the ideal solution because some goals might conflict with others. In such a case, we want to find a solution that deviates as little as possible from the ideal solution. One possible objective for our example GP problem is:

Minimize the sum of the deviations: MIN: $\sum_i (d_i^- + d_i^+)$

With this objective, we attempt to find a solution to the problem where all the deviationalvariables are 0—or where all the goals are met exactly. But if such a solution is not possible,will this objective always produce a desirable solution? The answer is "probably not."

The previous objective has several shortcomings. First, the deviational variables measure entirely different things. In our example problem, $d1$to d3 measure rooms of one size or another, whereas $d4$ are measures of square footage, and $d5$ are financial measures of building costs.

One obvious criticism of the previous objective is that it is unclear how to interpret any numerical value that the objective assumes (7 rooms + 1,500 dollars = 1,507 units of what?).

One solution to this problem is to modify the objective function so that it measures the sum of *percentage deviations* from the various goals. This is accomplished as follows, where t_i represents the target value for goal i:

Minimize the sum of the percentage deviations: MIN: $\sum_i \frac{1}{t_i}(d_i^- + d_i^+)$

In our example problem, suppose that we arrive at a solution where the first goal is underachieved by 1 room ($d1$ _ _ 1) and the fifth goal is overachieved by $20,000 ($d5+ = 20{,}000$) and all other goals are achieved exactly (all other d_i^- and d_i^+ equal 0). Using the sum of percentage deviations objective, the optimal objective function value is

$$\frac{1}{t_1}d_1^- + \frac{1}{t_5}d_5^+ = \frac{1}{5}\times 1 + \frac{1}{1{,}000{,}000}\times 20{,}000 = 20\% + 2\% = 22\%$$

Note that the percentage deviation objective can be used only if all the target values for all the goals are non-zero; otherwise a division by zero error will occur.

Another potential criticism of the previous objective functions concerns how they evaluate deviations. In the previous example, where the objective function value is 22%, the objective function implicitly assumes that having 4 small conference rooms (rather than 5) is 10 times worse than being $20,000 over the desired building cost budget. That is, the budget overrun of $20,000 would have to increase 10 times to $200,000 before the percentage deviation on this goal equaled the 20% deviation caused by being one room below the goal of having 5 small conference rooms. Is having one fewer conference room really as undesirable as having to pay $200,000 more than budgeted? Only the decision maker in this problem can answer this question. It would be nice to provide the decision maker a way to evaluate and change the implicit trade-offs among the goals if he or she wanted to do so.

Both of the previous objective functions view a deviation from any goal in any direction as being equally undesirable. For example, according to both of the previous objective functions, a solution resulting in a building cost of $900,000 is treated just like a solution resulting in 1.1 million!

Another practice is to assign weights

Minimize the weighted sum of the deviations: MIN: $\sum_i (w_i^- d_i^- + w_i^+ d_i^+)$

or

Minimize the weighted sum of the percentage deviations: MIN: $\sum_i \frac{1}{t_i}(w_i^- d_i^- + w_i^+ d_i^+)$

Unfortunately, no standard procedure is available for assigning values to the w_i^- and w_i^+ in a way that guarantees that you will find the most desirable solution to a GP problem. Rather, you need to follow an iterative procedure in which you try a particular set of weights, solve the problem, analyze the solution, and then refine the weights and solve the problem again. You might need to repeat this process many times to find a solution that is the most desirable to the decision maker.

In our example problem, assume that the decision maker considers it undesirable to underachieve any of the first three goals related to the number of small, medium, and large conference rooms, but is indifferent about overachieving these goals. Also assume that the decision maker considers it undesirable to underachieve the goal of adding 25,000 square feet, but equally undesirable to overachieve this goal. Finally, assume that the decision maker finds it undesirable to spend more than $1,000,000, but is indifferent about spending less than this amount. In this case, if we want to minimize the weighted percentage deviation for our example problem, we use the following objective:

$$\text{MIN:} \quad \frac{w_1^-}{5}d_1^- + \frac{w_2^-}{10}d_2^- + \frac{w_3^-}{15}d_3^- + \frac{w_4^-}{25{,}000}d_4^- + \frac{w_4^+}{25{,}000}d_4^+ + \frac{w_5^+}{1{,}000{,}000}d_5^+$$

Notice that this objective omits (or assigns weights of 0 to) the deviational variables about which the decision maker is indifferent. Thus, this objective would not penalize a solution where, for example, 7 small conference rooms were selected (and therefore $d_1^+ = 2$) because we assume that the decision maker would not view this as an undesirable deviation from the goal of having 5 small conference rooms.

To summarize, the LP model for our example GP problem is:

MIN: $\frac{w_1^-}{5} d_1^- + \frac{w_2^-}{10} d_2^- + \frac{w_3^-}{15} d_3^- + \frac{w_4^-}{25,000} d_4^- + \frac{w_4^+}{25,000} d_4^+ + \frac{w_5^+}{1,000,000} d_5^+$

Subject to:

$$
\begin{array}{rlr}
X_1 + d_1^- - d_1^+ = & 5 & \text{\} small rooms} \\
X_2 + d_2^- - d_2^+ = & 10 & \text{\} medium rooms} \\
X_3 + d_3^- - d_3^+ = & 15 & \text{\} large rooms} \\
400X_1 + 750X_3 + 1,050X_3 + d_4^- - d_4^+ = & 25,000 & \text{\} square footage} \\
18,000X_1 + 33,000X_2 + 45,150X_3 + d_5^- - d_5^+ = & 1,000,000 & \text{\} building cost} \\
d_i^-, d_i^+ \geq 0 \text{ for all } i & & \text{\} non-negativity conditions} \\
X_i \geq 0 \text{ for all } i & & \text{\} non-negativity conditions} \\
X_i \text{ must be integers} & &
\end{array}
$$

We will start with a set of weights (say all equal to 1). If the solution is not acceptable, we change the weights and try out and continue until we find a satisfactory solution.

Summary of Goal Programming

- Identify the decision variables in the problem.
- Identify any hard constraints in the problem and formulate them in the usual way.
- State the goals of the problem along with their target values.
- Create constraints using the decision variables that would achieve the goals exactly.
- Transform the above constraints into goal constraints by including deviational variables.
- Determine which deviational variables represent undesirable deviations from the goals.
- Formulate an objective that penalizes the undesirable deviations.
- Identify appropriate weights for the objective.
- Solve the problem.
- Inspect the solution to the problem. If the solution is unacceptable, return to step 8 and revise the weights as needed.

Some additional comments should be made before we leave the topic of GP. First, it isimportant to note that different GP solutions cannot be compared simply on the basis of their optimal objective function values. The user changes the weights in the objective functions from iteration to iteration; therefore, comparing their values is not appropriate because they measure different things. The objective function in a GP problem serves more of a mechanical purpose, allowing us to explore possible solutions.

Thus, we should compare the solutions that are produced—not the objective function values. Second, in some GP problems, one or more goals might be viewed as being infinitely more important than the other goals. In this case, we could assign arbitrarily large weights to deviations from these goals to ensure that undesirable deviations from them never occur.

This is sometimes referred to as *preemptive* GPbecause certain goals preempt others in order of importance. If the target values for these goals can be achieved, the use of preemptive weights effectively makes these goals hard constraints that should never be violated.

Third, we can place hard constraints on the amount by which we can deviate from a goal. For example, suppose that the owner of the hotel in our example problem wants to eliminate from consideration any solution that exceeds the target building cost by more than \$50,000. We could build this requirement into our model easily with the hard constraint:

$d5_+ \leq 50,000$

Fourth, the concept of deviational variables is not limited to GP. These types of variables can be used in other problems that are quite different from GP problems. So, understanding deviational variables can prove useful in other types of mathematical programming situations.

Finally, another type of objective function, called the MINIMAX objective, is sometimes helpful in GP when you want to minimize the maximum deviation from any goal. To implement the MINIMAX objective, we must create one additional constraint for each deviational variable as follows, where Q is the MINIMAX variable:

$$d1_{-} \leq Q$$

$$d1_{+} \leq Q$$

$$d2_{-} \leq Q$$

and so on . . .

The objective is to minimize the value of Q, stated as:

MIN: Q

Because the variable Q must be greater than or equal to the values of all the deviational variables, and because we are trying to minimize it, Q will always be set equal to the maximum value of the deviational variables. At the same time, this objective function tries to find a solution where the maximum deviational variable (and the value of Q) is as small as possible. Therefore, this technique allows us to minimize the maximum deviation from all the goals. As we will see shortly, this type of objective is especially valuable if a GP problem involves hard constraints.

MULTIPLE OBJECTIVE OPTIMIZATION

We now consider how to solve LP problems involving multiple objective functions. These problems are called multiple objective linear programming(MOLP) problems.

For example, if a production process creates a toxic pollutant that is dangerous to the environment, a company might want to minimize this toxic by-product. But this objective is likely to be in direct conflict with the company's other objective of maximizing profits. Increasing profit will likely always result in the creation of additional toxic waste. Figure below shows a hypothetical example of the potential trade-offs between profit and the production of toxic waste. Each point on the curve in this graph corresponds to a possible level of profit and the minimum amount of toxic waste that must be produced to achieve this level of profit. Clearly, reaching higher levels of profit (which is desirable) is associated with incurring greater levels of toxic waste production (which is undesirable). So the decision maker must decide what level of trade-off between profit and toxic waste is most desirable.

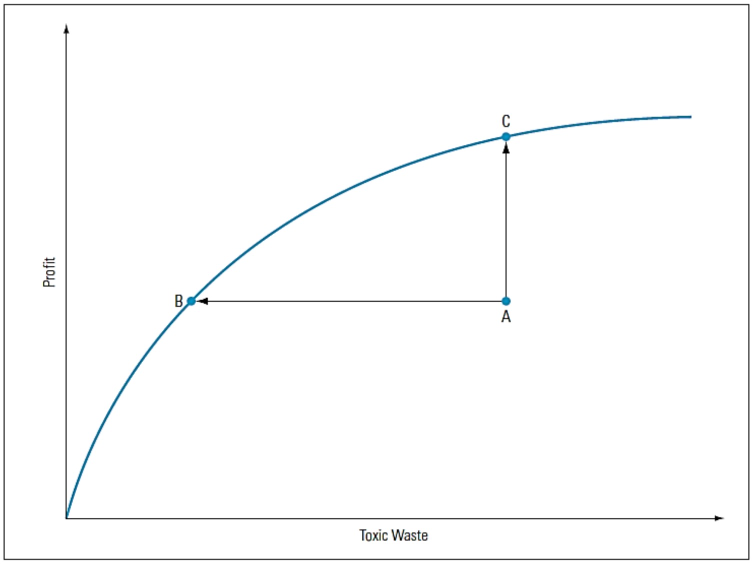

Figure 13

Another important MOLP issue to note in above Figure is the concept of dominated and non-dominated solutions. Accepting a solution that offers the combination of profit and toxic waste indicated by point A is clearly undesirable. There is another alternative (i.e., point B on the graph) that offers less toxic waste production for the same level of profit.

Also, there is another alternative (i.e., point C on the graph) that offers more profit for the same level of toxic waste. So points B and C either would be preferable to (or dominate) point A. Indeed, all the points along the curve connecting point B to point C dominate point A. In MOLP, a decision alternative is **dominated** if there is another alternative that produces a better value for at least one objective without worsening the value of the other objectives. Clearly, rational decision makers should want to consider only decision alternatives that are non-dominated. The technique for MOLP presented in this chapter guarantees that the solutions presented to the decision maker are non-dominated.

Fortunately, MOLP problems can be viewed as special types of GP problems where, as part of solving the problem, we also must determine target values for each goal or objective. Analyzing these problems effectively also requires that we use the MINIMAX objective described earlier.

Lee Blackstone is the owner of the Blackstone Mining Company, which operates two different coal mines in Wythe and Giles counties in Southwest Virginia. Due to increased commercial and residential development in the primary areas served by these mines, Lee is anticipating an increase in demand for coal in the coming year. Specifically, her projections indicate a 48-ton increase in the demand for high-grade coal, a 28-ton increase in the demand for medium-grade coal, and a 100-ton increase in the demand for low-grade coal. To handle this increase in demand, Lee must schedule extra shifts of workers at the mines. It costs $40,000 per month to run an extra shift of workers at the Wythe County mine and $32,000 per month at the Giles mine. Only one additional shift can be scheduled each month at each mine. The amount of coal that can be produced in a month's time at each mine by a shift of workers is summarized in the following table.

Type of Coal	Wythe Mine	Giles Mine
High grade	12 tons	4 tons
Medium grade	4 tons	4 tons
Low grade	10 tons	20 tons

Table 15

Unfortunately, the methods used to extract coal from these mines produce toxic water that enters the local groundwater aquifers. At the Wythe mine, approximately 800 gallons of toxic water per month will be generated by running an extra shift, whereas the mine in Giles County will generate about 1,250 gallons of toxic water. Although these amounts are within EPA guidelines, Lee is concerned about the environment and doesn't want to create any more pollution than is absolutely necessary.

Additionally, although the company follows all OSHA safety guidelines, company records indicate that approximately 0.20 life-threatening accidents occur per shift each month at the Wythe mine, whereas 0.45 accidents occur per shift each month at the Giles mine. Lee knows that mining is a hazardous occupation, but she cares about the health and welfare of her workers and wants to keep the number of life-threatening accidents to a minimum.

In this problem, Lee has to determine the number of months to schedule an extra shift at each of the company's mines. Thus, we can define the decision variables as:

X_1 = number of months to schedule an extra shift at the Wythe county mine

X_2 = number of months to schedule an extra shift at the Giles county mine

DEFINING THE OBJECTIVES

This problem is different from the other types of LP problems we have considered in that three different objective functions are possible. Lee might be interested in minimizing costs, minimizing the production of toxic waste water, or minimizing the expected number of life-threatening accidents. These three different objectives would be formulated as follows:

$$\begin{array}{lll}
\text{Minimize:} & \$40X_1 + \$32X_2 & \}\text{ Production costs (in \$1,000s)} \\
\text{Minimize:} & 800X_1 + 1250X_2 & \}\text{ Toxic water produced (in gallons)} \\
\text{Minimize:} & 0.20X_1 + 0.45X_2 & \}\text{ Life-threatening accidents}
\end{array}$$

In an LP model, Lee would be forced to decide which of these three objectives is most important or most appropriate, and use that single objective in the model. However, in an MOLP model, Lee can consider how both of these objectives (and any others she might want to formulate) can be incorporated into the analysis and solution of the problem.

The constraints for this problem are formulated in the same way as for any LP problem. The following three constraints ensure that required amounts of high-grade, medium grade, and low-grade coal are produced.

$$\begin{array}{ll}
12X_1 + 4X_2 \geq 48 & \}\text{ High-grade coal required} \\
4X_1 + 4X_2 \geq 28 & \}\text{ Medium-grade coal required} \\
10X_1 + 20X_2 \geq 100 & \}\text{ Low-grade coal required}
\end{array}$$

$$\begin{array}{lll}
\text{Minimize:} & \$40X_1 + \$32X_2 & \}\text{ Production costs (in \$1,000s)} \\
\text{Minimize:} & 800X_1 + 1250X_2 & \}\text{ Toxic water produced (in gallons)} \\
\text{Minimize:} & 0.20X_1 + 0.45X_2 & \}\text{ Life-threatening accidents} \\
\text{Subject to:} & 12X_1 + 4X_2 \geq 48 & \}\text{ High-grade coal required} \\
 & 4X_1 + 4X_2 \geq 28 & \}\text{ Medium-grade coal required} \\
 & 10X_1 + 20X_2 \geq 100 & \}\text{ Low-grade coal required} \\
 & X_1, X_2 \geq 0 & \}\text{ non-negativity conditions}
\end{array}$$

An LP problem can have only one objective function, so how can we include three objectives in our model? If these objectives had target values, we could treat them the same way as the goals in our example earlier in this chapter. That is, the objectives in this problem can be stated as the following goals if we have appropriate values for t_1, t_2, and t_3:

Goal 1: The total production cost should be approximately t_1.

Goal 2: The gallons of toxic water produced should be approximately t_2.

Goal 3: The number of life-threatening accidents should be approximately t_3.

Unfortunately, the problem did not provide explicit values for t_1, t_2, and t_3. However, if we solve our model to find the solution that minimizes the first objective (total production cost), the optimal value of this objective function would be a reasonable value to use as t_1 in the first goal. Similarly, if we solve the problem two more times minimizing the second and third objectives, respectively, the optimal objective function values for these solutions would provide reasonable values to use as t_2 and t_3 in the second and third goals. We could then view our MOLP problem in the format of a GP problem.

Now that we have target values for the three objectives in our problem, we can formulate a weighted GP objective to allow the decision maker to explore possible solutions. Earlier in this chapter, we discussed several GP objectives and illustrated the use of an objective that minimized the weighted percentage deviation from the goals' target values. Let's consider how to formulate this same type of objective for the current problem.

We can restate the objectives of this problem as the goals:

Goal 1: The total production cost should be approximately $244.

Goal 2: The gallons of toxic water produced should be approximately 6,950.

Goal 3: The number of life-threatening accidents should be approximately 2.0.

We now know that the actual total production cost never can be smaller than its target (optimum) value of $244, so the percentage deviation from this goal may be computed as:

$$\frac{\text{actual value} - \text{target value}}{\text{target value}} = \frac{(40X_1 + 32X_2) - 244}{244}$$

Using the same logic

$$\frac{\text{actual value} - \text{target value}}{\text{target value}} = \frac{(800X_1 + 1250X_2) - 6950}{6950}$$

$$\frac{\text{actual value} - \text{target value}}{\text{target value}} = \frac{(0.20X_1 + 0.45X_2) - 2}{2}$$

As it turns out, the MINIMAX objective, described earlier, can be used to explore the points on the edge of the feasible region—in addition to corner points. To illustrate this, let's attempt to minimize the maximum weighted percentage deviation from the target values for the goals in our example problem using the objective:

We implement this objective by establishing a MINIMAX variable Q that we minimize with the objective:

$$\text{MIN: } Q$$

subject to the additional constraints:

$$w_1\left(\frac{(40X_1 + 32X_2) - 244}{244}\right) \leq Q$$

$$w_2\left(\frac{(800X_1 + 1250X_2) - 6950}{6950}\right) \leq Q$$

$$w_3\left(\frac{(0.20X_1 + 0.45X_2) - 2}{2}\right) \leq Q$$

The first constraint indicates that the weighted percentage deviation from the target production cost must be less than or equal to Q. The second constraint indicates that the weighted percentage deviation from the target level of toxic water production also must be less than or equal to Q. The third constraint indicates that the weighted percentage deviation from the target expected number of life-threatening accidents also must be less than or equal to Q. Thus, as we minimize Q, we also are minimizing the weighted percentage deviations from the target values for each of our goals. In this way, the maximum weighted deviation from any of the goals is minimized—or we have MINImized the MAXimum deviation (hence the term MINIMAX).

MIN: Q
Subject to:

$$\begin{array}{lll}
12X_1 + 4X_2 \geq 48 & & \text{} \} \text{ High-grade coal required} \\
4X_1 + 4X_2 \geq 28 & & \text{} \} \text{ Medium-grade coal required} \\
10X_1 + 20X_2 \geq 100 & & \text{} \} \text{ Low-grade coal required} \\
w_1 (40X_1 + 32X_2 - 244)/\ 244 \leq Q & & \text{} \} \text{ goal 1 MINIMAX constraint} \\
w_2(800X_1 + 1250X_2 - 6950)/\ 6950 \leq Q & & \text{} \} \text{ goal 2 MINIMAX constraint} \\
w_3(0.20X_1 + 0.45X_2 - 2)/\ 2 \leq Q & & \text{} \} \text{ goal 3 MINIMAX constraint} \\
X_1, X_2 \geq 0 & & \text{} \} \text{ non-negativity conditions} \\
w_1, w_2, w_3 \text{ are positive constants} & &
\end{array}$$

Summary of Multiple Objective Optimization

1. Identify the decision variables in the problem.

2. Identify the objectives in the problem and formulate them in the usual way.

3. Identify the constraints in the problem and formulate them in the usual way.

4. Solve the problem once for each of the objectives identified in step 2 to determine the optimal value of each objective.

5. Restate the objectives as goals using the optimal objective values identified in step 4 as the target values.

6. For each goal, create a deviation function that measures the amount by which any given solution fails to meet the goal (either as an absolute or a percentage).

7. For each of the deviation functions identified in step 6, assign a weight to the deviation function and create a constraint that requires the value of the weighted deviation function to be less than the MINIMAX variable Q.

8. Solve the resulting problem with the objective of minimizing Q.

9. Inspect the solution to the problem. If the solution is unacceptable, adjust the weights in step 7 and return to step 8.

Although the MOLP example in this chapter was somewhat simple, the same basic process applies in virtually any MOLP problem, regardless of the number of objectives or the complexity of the problem.

One advantage of using the MINIMAX objective to analyze MOLP problems is that the solutions generated are always ***Pareto optimal***. That is, given any solution generated using this approach, we can be certain that no other feasible solution allows an increase in any objective without decreasing at least one other objective. (There are one or two exceptions to this statement, but they go beyond the scope of this text.)

Although the MINIMAX objective is helpful in the analysis of MOLPs, its usefulness is not limited to these problems. Like deviational variables, the MINIMAX technique can prove useful in other types of mathematical programming situations.

In the example MOLP problem presented here, all of the goals were derived from minimization objectives. Because of this, we knew that the actual value for any goal never could be less than its derived target value, and we used the following formula to calculate the percentage deviation for each goal constraint:

$$\frac{\text{actual value} - \text{target value}}{\text{target value}}$$

For goals derived from maximization objectives, we know that the actual value of the goal never can be greater than its derived target value, and the percentage deviation for such goals should be calculated as:

If the target value of a goal is zero, it is not possible to use weighted percentage deviations in the solution to the MOLP (because division by zero is not permissible). In this case, you can simply use weighted deviations.

$$\frac{\text{target value} - \text{actual value}}{\text{target value}}$$

MULTIVARIABLE CALCULUS

In calculus up to this time, the functions you have dealt with have mostly been functions of one variable, like $f(x) = x2 - x + 1$.

Pick an x, plug it in, and calculate. But you have worked with functions of more than one variable, even if they were not called that at the time. Consider the basic formulas from geometry.

Some, like those for circles, are functions of one variable. We could write area of a circle as a function of radius: $A(r) = \pi r^2$.

The rectangle formulas require two variables, one for each dimension. If we let x = length and y = width, then the perimeter and area formulas are $P(x, y) = 2x + 2y$ and $A(x, y) = xy$.

The objective function from linear programming, which was a function of two variables, something like $z = 4x + 7y$.

A shed with a rectangular base and a flat roof is being built. The concrete for the base costs \$8 per square foot. The wooden studs and panels for the sides will cost \$5 per square foot. The heavier material for the roof costs \$7 per square foot. Write an equation for the cost to build the shed.

$$C(x, y, z) = 8xy + 2(5yz) + 2(5xz) + 7xy$$

A function of two variables, $z = f(x, y)$, can be graphed on a three-dimensional grid. Picture the corner of a room where the length and width of the floor form the x-axis and y-axis. The vertical where the two walls meet is the z-axis. This "standard" orientation, while helpful from a conceptual view, will not always provide a clear picture of the function. A three-dimensional shape translated into two dimensions can obscure the true nature of the shape. It would be nice to be able to rotate the picture through the three dimensions to get a clearer idea.

A very good, and free, graphing calculator for your computer can be downloaded from www.graphcalc.com.This utility will allow you to graph both two- and three-dimensional graphs, and rotate the three-dimensionalgraphs either automatically or manually. (It's the utility I used for the examples below.)

It also has the ability todo some evaluations, as well as to graph parametric and polar equations. (You have your choice of colors todraw the graph in, too.)Example D: Graph and explore the rectangle perimeter function z = 2x + 2y.The graph shown to the left below has been drawn on the axes as oriented above, but it's hard to tell what shapeit really has. The graph to the right has been rotated to illustrate more clearly that the shape is a flat surface—ingeometry terms a plane.

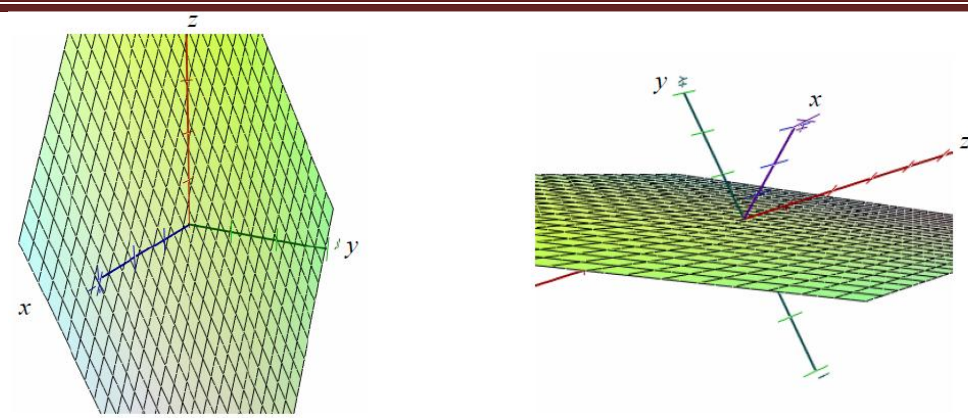

Figure 14

Let's go back briefly to the type of problems we've encountered before this. Even when we needed to use a function of more than one variable, there was enough information given so that we could do some sort of substitution and turn it into an equation with one variable that we could solve.

Example E: Farmer Bob has a rectangular corral and 120 feet of fencing. Write an equation in one variable that represents the area of the corral. *Answer*: $A = x(60 - x)$

A similar process provides a means of taking a three-dimensional object and considering it in two dimensions. The concept is the same one used in drawing topographical maps that show the elevations of terrain. The flat surface is laid out in latitude and longitude (the x and y) while a series of curves show the elevation (z) of the terrain. Tracing along a curve shows the places on the ground that all have the same elevation. Widely spaced curves indicate a gentler slope. Curves close together indicate a steep slope.

If we choose a series of values for z, the result is a series of equations in x and y that can be graphed as usual on the Cartesian grid. These are called *level curves*.

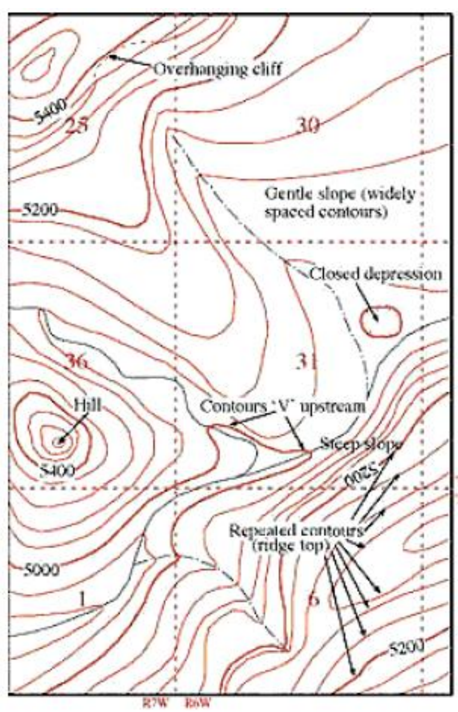

Figure 15

Draw level curves for $z = 2x + 2y$ for $z = 10, 8, 6, 4$, and 2. Given $z = 10$, then $10 = 2x + 2y \Rightarrow 10 - 2x = 2y \Rightarrow 5 - x = y$. The others are found in a similar fashion.

Each level curve on the grid represents a different value for *z*. The flat surface of *z* = 2*x* + 2*y* rises at an angle as *z* increases.

z	$z = 2x + 2y$	$f(x) = y = \ldots$
10	$10 = 2x + 2y$	$y = 5 - x$

Table 16

Graph and explore the rectangle area function z = xy . Sketch the level curves for z = 10, 8, 6, 4, 2. The graph "standard" orientation of the x-y-z axes is on the left, and a rotation of the graph on the right. The "twist" in the 3-D surface is easier to see in the rotated version. The symmetric nature of the twist centered at the origin is even more apparent from the level curves.

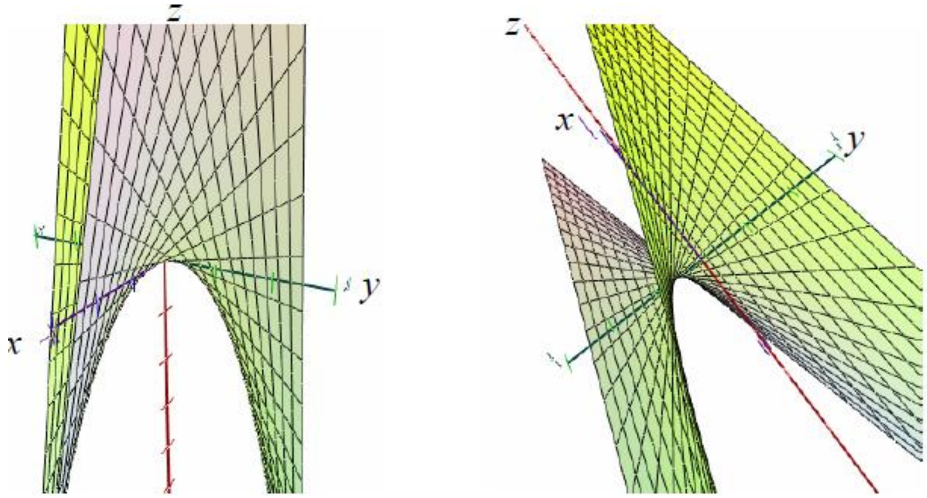

Figure 16

Draw the 3-D graph and level curves for the function z = 9 − x2 − y2.

The surface on the left shows something of the curve, but the top is outside the viewing window. Tilting the graph slightly allows us to see the maximum. The level curves show the circular nature of the shape. At a later time we'll be developing a method of finding the location of that absolute maximum.

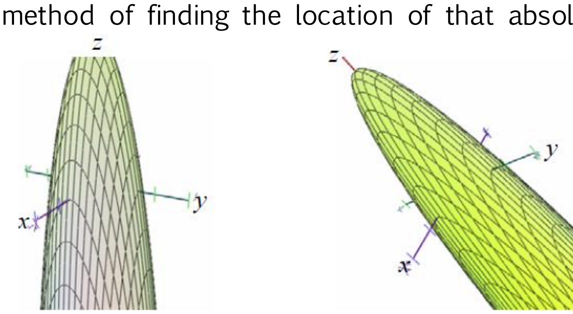

Figure 17

CALCULUS OF MULTIVARIABLE FUNCTIONS

When working with functions of more than one variable, the question in calculus becomes: how can we evaluate the rate of change? The answer is called a *partial derivative*. Given a function *f*(*x*, *y*, *z*), the partial derivative of *f* with respect to *x*, $\frac{\partial f}{\partial x}$, is found by treating all variables other than *x* as constants. The partial derivatives and $\frac{\partial f}{\partial y}$ *and* $\frac{\partial f}{\partial z}$ have analogous definitions.

Example A: Given the function *f*(*x*, *y*) = 2*x* + 2*y* find $\frac{\partial f}{\partial x}$ *and* $\frac{\partial f}{\partial y}$

Answers: 2; 2

Given the function f(x, y) = 9 $-x^2-y^2$find $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$

Answers: $-2x$; $-2y$

A geometric interpretation of partial derivative is pictured below. In each figure *f*(*x*, *y*) is the curved surface. In the figure on the left, with *y* treated as a constant, the tangent line goes the same general direction as the *x*-axis and, $\frac{\partial f}{\partial x}$ is the slope of that tangent at point *P*. With *x* treated as a constant, the tangent line goes the same general direction as the *y*-axis, and $\frac{\partial f}{\partial y}$ is the slope of that tangent at point *P*.

Figure 18

Say a point $P_0=(a_1,a_2,\dots a_n)$ is a maximum point (or a minimum point), of the equation $y= f(x_1,x_2,\dots x_n)$. Then if we hold $(x_2,\dots x_n)$ fixed at $(a_2,\dots a_n)$, *f* becomes a function of x_1 only, with a maximum (or minimum) at $x_1=a_1$.

Therefore at this point

$$\frac{\partial f}{\partial x_1} = 0$$

Likewise, $f_{x2}=0$, $f_{x3}=0$, …$f_{xn}=0$ at point *P*.

A **Critical Point** is any point that satisfies all these *n* equations.

For example, say $f(x,y)=x^5+y^4-5x-32y$. Then,

$$\frac{\partial f}{\partial x} = 5x^4 - 5 = 0$$

$$\frac{\partial f}{\partial y} = 4y^3 - 32 = 0$$

Solving for real critical points, we get (1,2) and (-1,2).

But are these points minimas or maxima, or neither?

We notice that the point $x=1$ is a minimum for the first equation and the point $y=2$ is a minimum for the second equation. The point (1,2) is a minimum for $f(x,y)=x^5+y^4-5x-32y$. The critical point (-1,2) is neither a minimum nor a maximum point for the surface. It is a **saddle point**.

We define the Hessian Matrix for a n-variable function y= f(x1, x,2, … xn), as the n by n matrix whose (i,j)-th entry is the function of the second-order partial derivative

$$\frac{\partial^2 f}{\partial x_i \partial x_j}$$

For two-variable functions, our Hessian matrix will be a 2 by 2 matrix. In such a case, we find the determinant D at each critical point and define that

The point is a maximum point if D>0 and $f_{xx}<0$

a minimum point if D>0 and f_{xx}>0

a saddle point if D<0.

Further, if D=0, then no conclusion can be drawn, and any of the behaviors described above can occur

For the above example,

$$\frac{\partial^2 f}{\partial x^2} = 20x^3$$

$$\frac{\partial^2 f}{\partial x \partial y} = 0$$

$$\frac{\partial^2 f}{\partial y^2} = 12x^2$$

So the determinant of the Hessian is $240x^3y^2$. Clearly for the critical point (1,2), D>0 and f_{xx}>0 indicating (1,2) is a minimum point. On the other hand, for the critical point (-1,2) D<0 indicating (-1,2) is a saddle point. This matches our previous conclusion.

We learned how to find the maxima and minima when the functions are twice differentiable. Let us now look at how to solve the problem when they are not. We will only look at two techniques: One for one variable and one for two variables.

Three point interval search for functions of one variable: The interval under consideration is divided into quarters and the objective function evaluated at the three equally spaced interior points. The interior point yielding the best value of the objective is determined (in case of a tie, one value is arbitrarily chosen) and the subinterval centered at this point and made up of two quarters of the current interval replaces the current interval.

The three point interval search is the most efficient equally spaced search procedure in terms of achieving a prescribed tolerance with a minimum number of functional evaluations.

Practice Problem 1: Use the three point interval search to approximate the location of the global minimum of f(x)=xsin4x on [0,3] to within ε=0.01

Let us take our interval as $[\frac{7\pi}{8}, 3]$. We chose this based on prior knowledge (we could haeve plotted too to get the interval. We can take any interval, a better selection converges fast.

First iteration: Dividing $[\frac{7\pi}{8}, 3]$ into quarters, we take x1=2.8117, x2=2.8744 and x3=2.9372 as the three interior points. Substituting this in f(x) = x sin4x, f(x1) is the lowest value. Now, we will pick our new point as $[\frac{7\pi}{8}, x2]$(for which x1 is the center). We will again divide these in to quarters and iterate until the value does not converge.

Functions with two variables: We use Newton Raphson method. This relies on computing Gradient and Hessian at each point. We start with one point X_k.

$$X_{k+1} = X_k - H^{-1}\nabla f$$

Where H is the Hessian and ∇f is the gradient.

Practice Problem 2:Maximize Z = -$\boldsymbol{(x_1 - \sqrt{5})^2 - (x_2 - \pi)^2 - 10}$ to which the tolerance is 0.05

We take the initial approximation [6.597, 5.891]. Obviously, this was from some previous knowledge. So, $f(X_0)$ = -36.58.

The gradient = $\nabla f = \begin{bmatrix} -2(x_1 - \sqrt{5} \\ -2(x_2 - \pi) \end{bmatrix} H_f = \begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix} \, and \, H_f{}^{-1} = \begin{bmatrix} -0.5 & 0 \\ 0 & -0.5 \end{bmatrix}$

First iteration using the formula gives X_1=[2.236 3.142], $f(X_1)$=26.58. As this is more than the tolerance, we will iterate again.

SOLVING MULTIVARIABLE FUNCTIONS WITH CONSTRAINTS

The method of Lagrange multipliers is a centerpiece of optimization theory. Let us look at a simple way of seeing why Lagrange multipliers actually do what they do -- that is, solve constrained optimization problems through the use of a semi-mysterious Lagrangian function.

OPTIMIZING NON-LINEAR FUNCTIONS

The most important thing to know about gradients is that they always point in the direction of a function's steepest slope at a given point. To help illustrate this, take a look at the drawing below. It illustrates how gradients work for a two-variable function of x1 and x2.

The function f in the drawing forms a hill. Toward the peak I've drawn two regions where we hold the height of f constant at some level a. These are called level curves of f, and they're marked f = a1, and f = a2.

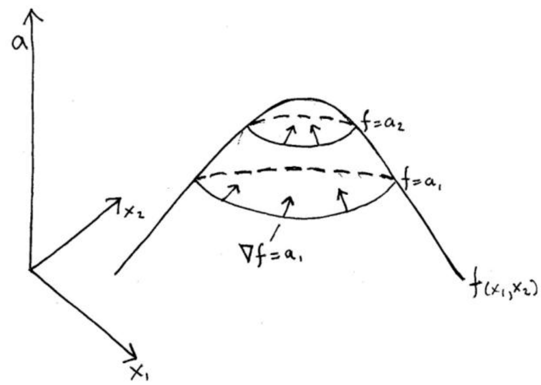

Figure 19

Imagine yourself standing on one of those level curves. Think of a hiking trail on a mountainside. Standing on the trail, in what direction is the mountain steepest? Clearly the steepest direction is straight up the hill, perpendicular to the trail. In the drawing, these paths of steepest ascent are marked with arrows. These are the gradients ∇f at various points along the level curves. Just as the steepest hike is always perpendicular to our trail, the gradients of f are always perpendicular to its level curves.

That's the key idea here: level curves are where $f = a$, and $\nabla f \perp f = a$.

While we're still looking for some optimum value (maximum or minimum), neither the objective nor the constraints are likely to be linear functions, and we'll need somewhat more involved methods of finding the maximum or minimum of the objective, within the given constraints. Given an objective function f and a constraint function g the process looks like this:

Identify the objective function—it's the one that needs to be maximized or minimized.

Write the constraint function in the form g = 0.

Create a function F = f + λg where λ is the Lagrange multiplier.

Find all first partial derivatives, including with respect to λ, and set them equal to 0.

Solve the resulting system of equations for all variables, including λ.

It will usually be best to solve the first equations for λ and set them equal to each other, using a series of substitutions to find the rest of the values.

Find the minimum value of f(x,y) = 2x2+y2+7 subject to the constraint g(x,y)=x+y+18. F(-6, -12)

How the Method Works

To see how Lagrange multipliers work, take a look at the drawing below. I've redrawn the function f from above, along with a constraint g = c. In the drawing, the constraint is a plane that cuts through our hillside. I've also drawn a couple level curves of f.

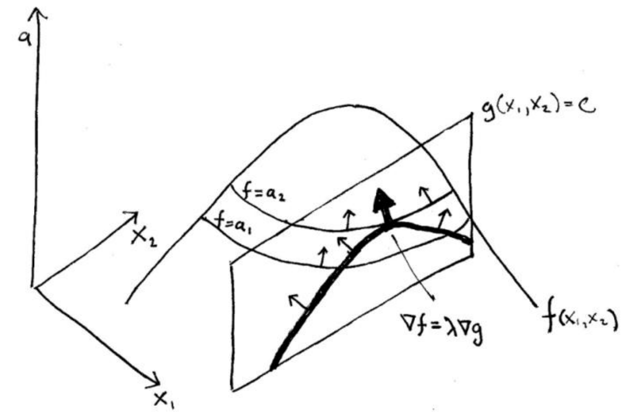

Figure 20

Our goal here is to climb as high on the hill as we can, given that we can't move any higher than where the constraint g = c cuts the hill. In the drawing, the boundary where the constraint cuts the function is marked with a heavy line. Along that line are the highest points we can reach without stepping over our constraint. That's an obvious place to start looking for a constrained maximum.

Imagine hiking from left to right on the constraint line. As we gain elevation, we walk through various level curves of f. I've marked two in the picture. At each level curve, imagine checking its slope -- that is, the slope of a tangent line to it -- and comparing that to the slope on the constraint where we're standing. If our slope is greater than the level curve, we can reach a higher point on the hill if we keep moving right. If our slope is less than the level curve -- say, toward the right where our constraint line is declining -- we need to move backward to the left to reach a higher point.

When we reach a point where the slope of the constraint line just equals the slope of the level curve, we've moved as high as we can. That is, we've reached our constrained maximum. Any movement from that point will take us downhill. In FIGURE 8, this point is marked with a large arrow pointing toward the peak.

At that point, the level curve f = a2 and the constraint have the same slope. That means they're parallel and point in the same direction. But as we saw above, gradients are always perpendicular to level curves. So if these two curves are parallel, their gradients must also be parallel. That means the gradients of f and g both point in the same direction, and differ at most by a scalar. Let's call that scalar λ. That is,

$$\nabla f = \lambda \nabla(g)$$

Solving for zero, we get

$$\nabla f - \lambda \nabla g = 0$$

This is the condition that must hold when we've reached the maximum of f subject to the constraint g = c.

Now, if we're clever we can write a single equation that will capture this idea. This is where the familiar Lagrangian equation comes in:

$$L = f - \lambda(g - c)$$

or more explicitly,

$$L(x_1, x_2, \lambda) = f(x_1, x_2) - \lambda(g(x_1, x_2) - c)$$

To see how this equation works, watch what happens when follow the usual Lagrangian procedure.

First, we find the three partial first derivatives of L,

$\frac{\partial L}{\partial x_1}, \frac{\partial L}{\partial x_2}, \frac{\partial L}{\partial \lambda}$

and set them equal to zero. That is, we need to set the gradient ∇L equal to zero.

To find ∇L, we take the three partial derivatives of L with respect to x1, x2 and lambda. Then we place each as an element in a 3 x 1 vector. That gives us the following:

$$\nabla L = \begin{bmatrix} \frac{\partial f}{\partial x_1} - \lambda \frac{\partial g}{\partial x_1} \\ \frac{\partial f}{\partial x_2} - \lambda \frac{\partial g}{\partial x_2} \\ g(x_1, x_2) - c \end{bmatrix} = 0$$

Recall that we have two "rules" to follow here. First, the gradients of f and g must point in the same direction, or $\nabla f = \lambda \nabla g$. And second, we have to satisfy our constraint, or $g = c$.

The first and second elements of ∇L make sure the first rule is followed. That is, they force $\nabla f - \lambda \nabla g = 0$, assuring that the gradients of f and g both point in the same direction. The third element of ∇L is simply a trick to make sure g = c, which is our constraint. In the Lagrangian function, when we take the partial derivative with respect to λ, it simply returns back to us our original constraint equation.

At this point, we have three equations in three unknowns. So we can solve this for the optimal values of x1 and x2 that maximize f subject to our constraint. And we're done.

So the bottom line is that Lagrange multipliers is really just an algorithm that finds where the gradient of a function points in the same direction as the gradients of its constraints, while also satisfying those constraints.

As with most areas of mathematics, once you see to the bottom of things -- in this case, that optimization is really just hill-climbing, which everyone understands -- things are a lot simpler than most economists make them out to be.

PRACTICE PROBLEM

Haldiram has 20 kgs of seasonal fruit mix and 60 kgs of an expensive Panner. It also has an unlimited supply of some cheap cheese. It makes two different sweets for Diwali: Delux and regular. Delux sweet requires 0.2KG of fruit mix and 0.8 kg of expensive panner. Regular requires 0.2 kg of fruit mix and 0.3 kg of expensive panner and 0.5 kg of cheap cheese. From past data they found that the demand for each sweet on its price as follows. D1 = 190-25P1 and D2=250-50P2. How much should they make and how much should they price each one at to maximize the revenue and be left with no inventory

Variables: Price of the two sweets, P1, P2 and the quantitites X1 and X2

Objective function: We need to maximize the revenue.

The revenue of selling X1 KGs of sweets at price P1 = P1*X1. The revenue of selling X2 KGs of sweet at price P2 = P2*X2

Total = P1X1 + P2X2

However, through the demand function these two are related. We can expect that the price is set such that the demand equals the amount of sweet made.

So, X1 = 190-25P1 and X2 = 250-50P2

So, P1 = $\frac{190-X1}{25}$ and P2 = $\frac{250-X2}{50}$

Now, substituting these values in objective function, we get

$$\frac{190X1 - X1^2}{25} + \frac{250X2 - X2^2}{50}$$

Note that we have quadratic terms in the objective function.

The explicit constraints

The fruit mix needed for 1 KG of delux sweet = 0.2KG. For X1 KGs of delux sweet, we need 0.2X1 KGs of fruit mix. For X2 KGs of regular, we need 0.2X2 KGs of fruit mix.

Total fruit mix needed: 0.2(X1) + 0.2(X2) ≤ 20

Similarly, total expensive cheese: 0.8(X1)+(0.3X2) ≤60

Implicit constraints

Both X1 and X2 have to be ≥ 0

So, the constraints are linear and the objective function is quadratic. So, this is a quadratic optimization problem.

QUADRATIC PROGRAMMING

R-COMMAND

solve.QP(Dmat, dvec, Amat, bvec, meq=0, factorized=FALSE)

Description

This routine implements the dual method of Goldfarb and Idnani (1982, 1983) for solving quadraticprogramming problems of the form min(-d^Tb + 1/2b^TDb) with the constraints A^Tb >= b_0.

Note thatit minimizes and the constraints must be of greater than or zero relation.

Arguments

- Dmat - matrix appearing in the quadratic function to be minimized.
- dvec - vector appearing in the quadratic function to be minimized.
- Amat - matrix defining the constraints under which we want to minimize the quadratic function.
- bvec - vector holding the values of b0 (defaults to zero).
- meq - the first meq constraints are treated as equality constraints, all further as inequality constraints (defaults to 0).
- factorized - logical flag: if TRUE, then we are passing R^{-1} (where D = R^TR) instead of the matrix D in the argument Dmat.

Installing & loading package:

>install.packages("quadprog")

>library(quadprog)

Changing our problem and constraints to minimization and "greater than" relation

$$\frac{-190X1 + X1^2}{25} + \frac{-250X2 + X2^2}{50}$$

Constraints

-0.2(X1) - 0.2(X2) ≥ -20

Similarly, total expensive cheese: -0.8(X1)-(0.3X2) ≥-60

Converting objective function into the required format:

The objective function must be written as

$$\frac{1}{2}b^T Db - d^T b$$

$$\frac{1}{2}(x1 \quad x2)\begin{pmatrix} 2/25 & 0 \\ 0 & 2/50 \end{pmatrix}\begin{pmatrix} x1 \\ x2 \end{pmatrix} - \begin{pmatrix} \frac{190}{25} & \frac{250}{50} \end{pmatrix}\begin{pmatrix} x1 \\ x2 \end{pmatrix}$$

Let us define all the matrices

D = matrix(0,2,2)

D[1,1] = (2/25)

D[2,2] = (2/50)

dvec = c(190/25,250/50)

Amat = matrix(c(-0.2,-0.2,-0.8,-0.3),2,2)

bvec = c(-20,-60)

solve.QP(D,dvec,Amat,bvec)

```
> D = matrix(0,2,2)
> D[1,1] =  (2/25)
> D[2,2] =  (2/50)
> dvec  = c(190/25,250/50)
> Amat  =  matrix(c(-0.2,-0.2,-0.8,-0.3),2,2)
> bvec  = c(-20,-60)
> solve.QP(D,dvec,Amat,bvec)
$solution
[1] 55 45

$value
[1] -481.5

$unconstrained.solution
[1]  95 125

$iterations
[1] 2 0

$Lagrangian
[1] 16  0

$iact
[1] 1
```

If the two constraints were not there, the solution would have been 95 and 125. But, the desired answer is 55 and 45.

Also, the algorithm gives the minimum (that is why we multiplied with a negative). So, the actual maxima we are looking at is 481.5