Assignment - 4- GOAL PROGRAMMING

Jeeva

2023-11-26

Summary:

In summary, this goal programming technique, in my opinion, takes specific objectives and restrictions related to workforce stability and earnings development into account, making it easier to assess production decisions in a systematic manner. The company's objective function is to make a profit of 225 million dollars as its target function.

(a) The corporation must mix X1, X2, and X3 in order to use the combination. To maximize profit, the company's lone product, Product 3, can only be manufactured in 15 units. However, X3 has been adjusted.

(b) Despite the fact that employee levels would be stabilized at a maximum of 50,000 workers (x1p), the firm had 2,500 excess workers in this case. A penalty for the overage would be imposed.

(c) As a result of y2p and y2m, the earnings of the following year are evaluated in respect to the present level; in this case, the current level is understood as "0," indicating that the earnings of the upcoming year do not differ from those of the current year.

Problem Statement : The Research and Development Division of the Emax Corporation has developed three new products. A decision now needs to be made on which mix of these products should be produced. Management wants primary consideration given to three factors: total profit, stability in the workforce, and achieving an increase in the company's earnings next year from the $75 million achieved this year. In particular, using the units given in the following table, they want to Maximize Z = P - 6C - 3D, where P = total (discounted) profit over the life of the new products, C = change (in either direction) in the current level of employment, D = decrease (if any) in next year's earnings from the current year's level. The amount of any increase in earnings does not enter into Z, because management is concerned primarily with just achieving some increase to keep the stockholders happy. (It has mixed feelings about a large increase that then would be difficult to surpass in subsequent years.)

#Loading Required Package

```
library(lpSolveAPI)
```

The file “Assignment_4.lp” contains a representation of the model mentioned above.

```
g <- read.lp("C:/Users/jeeva
thangamani/Documents/GitHub/64018/Assignment_4/Assignment_4.lp")
g
```

```
## Model name:
##             X1    X2    X3   Y1P   Y1M   Y2M   Y2P
## Maximize    20    15    25    -6    -6    -3     0
## R1           6     4     5    -1     1     0     0  =  50
## R2           8     7     5     0     0     1    -1  =  75
## Kind       Std   Std   Std   Std   Std   Std   Std
## Type      Real  Real  Real  Real  Real  Real  Real
## Upper      Inf   Inf   Inf   Inf   Inf   Inf   Inf
## Lower        0     0     0     0     0     0     0
```

Solve

```
solve(g)
```

```
## [1] 0
```

```
get.objective(g)
```

```
## [1] 225
```

```
get.variables(g)
```

```
## [1]  0  0 15 25  0  0  0
```

```
get.constraints(g)
```

```
## [1] 50 75
```

```
#The following table illustrates the effects of each of the new goods (per
unit rate of production) on each of these factors:
# Create the matrix
g_tab <- matrix(c("Total Profit", "Employment Level", "Earnings Next Year",
20, 6, 8,
15, 4, 7,
25, 5, 5,
"Maximize", "=50", ">=75",
"Millions of Dollars", "Hundreds of Employees", "Millions of Dollars"),
ncol = 6, byrow = FALSE)
# Set column names
colnames(g_tab) <- c("Factor", "Product 1", "Product 2", "Product 3", "Goal",
"Units")
as.table(g_tab)
```

```
##   Factor             Product 1 Product 2 Product 3 Goal
## A Total Profit       20        15        25        Maximize
## B Employment Level   6         4         5         =50
```

```
## C Earnings Next Year 8         7         5         >=75
##   Units
## A Millions of Dollars
## B Hundreds of Employees
## C Millions of Dollars
```

```
get.sensitivity.rhs(g)
```

```
## $duals
## [1]   6  -1  -8  -2   0   0 -12  -2  -1
##
## $dualsfrom
## [1] -1.0e+30  5.0e+01 -1.0e+30 -1.0e+30 -1.0e+30 -1.0e+30 -2.5e+01 -
1.0e+30
## [9] -2.5e+01
##
## $dualstill
## [1] 7.500000e+01 1.000000e+30 9.375000e+00 8.333333e+00 1.000000e+30
## [6] 1.000000e+30 1.000000e+30 2.500000e+01 1.000000e+30
```

```
get.sensitivity.objex(g)
```

```
## $objfrom
## [1] -1.000000e+30 -1.000000e+30  2.357143e+01 -6.666667e+00 -1.000000e+30
## [6] -1.000000e+30 -1.000000e+30
##
## $objtill
## [1] 28 17 30 -5  6 -1  1
##
## $objfromvalue
## [1]  9.375000e+00  8.333333e+00 -1.000000e+30 -1.000000e+30 -1.000000e+30
## [6]  2.500000e+01 -1.000000e+30
##
## $objtillvalue
## [1] NA NA NA NA NA NA NA
```