BA64018_Assignment2

Ruthvick Bulagakula

2023-09-24

Problem

The Weigelt Corporation has three branch plants with excess production capacity. Fortunately, the corporation has a new product ready to begin production, and all three plants have this capability, so some of the excess capacity can be used in this way. This product can be made in three sizes–large, medium, and small–that yield a net unit profit of $420, $360, and $300, respectively. Plants 1, 2, and 3 have the excess capacity to produce 750, 900, and 450 units per day of this product, respectively, regardless of the size or combination of sizes involved. The amount of available in-process storage space also imposes a limitation on the production rates of the new product. Plants 1, 2, and 3 have 13,000, 12,000, and 5,000 square feet, respectively, of in-process storage space available for a day's production of this product. Each unit of the large, medium, and small sizes produced per day requires 20, 15, and 12 square feet, respectively. Sales forecasts indicate that if available, 900, 1,200, and 750 units of the large, medium, and small sizes, respectively, would be sold per day. At each plant, some employees will need to be laid off unless most of the plant's excess production capacity can be used to produce the new product. To avoid layoffs if possible, management has decided that the plants should use the same percentage of their excess capacity to produce the new product. Management wishes to know how much of each of the sizes should be produced by each of the plants to maximize profit.

Solve the problem using lpsolve, or any other equivalent library in R.

LP Model

This represents data in the table format.

```r
data_plant = matrix(c(750,13000,
                900,12000,
                450,5000), ncol=2, byrow = TRUE)

colnames(data_plant) = c('Capacity', 'Square Feet')
rownames(data_plant) = c('P1', 'P2', 'P3')

final_plant = as.table(data_plant)

print(final_plant)
```

```
##    Capacity Square Feet
## P1      750       13000
## P2      900       12000
## P3      450        5000
```

```r
data_size = matrix(c(20,900,'$420',
                     15,1200,'360',
                     12,750,'$900'), ncol=3, byrow = TRUE)

colnames(data_size) = c('Square Feet', 'Sales', 'Profit')
rownames(data_size) = c('Large', 'Medium', 'Small')

final_size = as.table(data_size)

print(final_size)
```

```
##        Square Feet Sales Profit
## Large  20          900   $420
## Medium 15          1200  360
## Small  12          750   $900
```

Decision Variable

Let us assume, $p1_l$, $p1_m$, $p1_s$ for plant 1, $p2_l$, $p2_m$, $p2_s$ for plant 2, and $p3_l$, $p3_m$, $p3_s$ for plant 3.

Objective Function

Objective for Weigelt Corporation is to increase profits. So, all large units in all plants are making 420 Dollars profit, all medium units in all plants are making 360 Dollars profit, and all small units in all plants are making 300 Dollars profit

Objective function for Weigelt Corporation:

$Z = 420(p1_l + p2_l + p3_l) + 360(p1_m + p2_m + p3_m) + 300(p1_s + p2_s + p3_s)$

Constraints

Capacity Constraint:

Total units produced by plant 1 per week is 750, Total units produced by plant 1 per week is 900, and Total units produced by plant 1 per week is 450.

Plant 1: $p1_l + p1_m + p1_s <= 750$

Plant 2: $p2_l + p2_m + p2_s <= 900$

Plant 3: $p3_l + p3_m + p3_s <= 450$

Storage Constraint:

Total square feet required for large per week is 20, Total square feet required for medium per week is 15, and Total square feet required for small per week is 12

Total square feet required for plant 1 is 13000, Total square feet required for plant 2 is 12000, and Total square feet required for plant 3 is 5000

Plant 1: $20p1_l + 15p1_m + 12p1_s <= 13000$

Plant 2: $20p2_l + 15p2_m + 12p2_s <= 12000$

Plant 3: $20p3_l + 15p3_m + 12p3_s <= 5000$

Salesforecast Capacity

Sales forecasts indicate that if available, 900, 1,200, and 750 units of the large, medium, and small sizes, respectively, would be sold per day.

$p1_l + p2_l + p3_l <= 900$

$p1_m + p2_m + p3_m <= 1200$

$p1_s + p2_s + p3_s <= 750$

Same Capaticy Constraint:

To ensure same percentage of excess capacity does not exceed at each plant. So, we can use constraint like this

$p1_l + p1_m + p1_s/750 = p2_2 + p2_m + p2_s/900 = p3_3 + p3_m + p3_s/450$

$900(p1_l + p1_m + p1_s) - 750(p2_2 + p2_m + p2_s) = 0$

$450(p2_2 + p2_m + p2_s) - 900(p3_3 + p3_m + p3_s) = 0$

$450(p3_3 + p3_m + p3_s) - 750(p1_l + p1_m + p1_s) = 0$

Non Negative Constraint:

$p1_l, p1_m, p1_s, p2_l, p2_m, p2_s, p3_l, p3_m, p3_s >= 0$

Mathematical Formula

Decision Variable:

$p1_l$, $p1_m$, $p1_s$, $p2_l$, $p2_m$, $p2_s$, $p3_l$, $p3_m$, $p3_s$

Objective Function:

$Z = 420(p1_l + p2_l + p3_l) + 360(p1_m + p2_m + p3_m) + 300(p1_s + p2_s + p3_s)$

Constraints:

Capacity Constraint:

Plant 1: $p1_l + p1_m + p1_s <= 750$

Plant 2: $p2_l + p2_m + p2_s <= 900$

Plant 3: $p3_l + p3_m + p3_s <= 450$

Storage Capacity:

Plant 1: $20p1_l + 15p1_m + 12p1_s <= 13000$

Plant 2: $20p2_l + 15p2_m + 12p2_s <= 12000$

Plant 3: $20p3_l + 15p3_m + 12p3_s <= 5000$

Salesforecast Capacity:

$p1_l + p2_l + p3_l <= 900$

$p1_m + p2_m + p3_m <= 1200$

$p1_s + p2_s + p3_s <= 750$

Same Capacity Constraint:

$p1_l + p1_m + p1_s/750 = p2_2 + p2_m + p2_s/900 = p3_3 + p3_m + p3_s/450$

$900(p1_l + p1_m + p1_s) - 750(p2_2 + p2_m + p2_s) = 0$

$450(p2_2 + p2_m + p2_s) - 900(p3_3 + p3_m + p3_s) = 0$

$450(p3_3 + p3_m + p3_s) - 750(p1_l + p1_m + p1_s) = 0$

Non Negative Constraint:

$p1_l, p1_m, p1_s, p2_l, p2_m, p2_s, p3_l, p3_m, p3_s >= 0$

So, this above mathematical formulas will help Weigelt Corporation to maximize profits.

Solution in R

Calling Installed Library*

```r
library(lpSolve)
```

Defining Objective Function

```r
MaxZ = c(420,360,300,420,360,300,420,360,300)
```

Defining Constraints

```r
constraints = matrix(c(1,1,1,0,0,0,0,0,0,
                    0,0,0,1,1,1,0,0,0,
                    0,0,0,0,0,0,1,1,1,
                    20,15,12,0,0,0,0,0,0,
                    0,0,0,20,15,12,0,0,0,
                    0,0,0,0,0,0,20,15,12,
                    1,0,0,1,0,0,1,0,0,
                    0,1,0,0,1,0,0,1,0,
                    0,0,1,0,0,1,0,0,1,
                    900,900,900,-750,-750,-750,0,0,0,
                    0,0,0,450,450,450,-900,-900,-900,
                    450,450,450,0,0,0,-750,-750,-750), ncol = 9, byrow = TRUE)
```

Defining Signs used for constraints

```r
signs = c("<=","<=","<=","<=","<=","<=", "<=","<=","<=","=","=","=")
```

Defining Sign values used for constraints

```r
sign_values = c(750,900,450,13000,12000,5000,900,1200,750,0,0,0)
```

Objective Function

```r
lp_result = lp("max", MaxZ, constraints, signs, sign_values)

print(lp_result)
```

```
## Success: the objective function is 696000
```

LP Model Solution

```r
lp_result$solution
```

```
## [1] 516.6667 177.7778   0.0000   0.0000 666.6667 166.6667   0.0000   0.0000
## [9] 416.6667
```