Module 4 Assignment 2

Lynsey Scheneman

2022-09-20

```
library(lpSolveAPI)
```

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.

Define the decision variables

The decision variables are the # of units of each new product (small, medium and large) that should be produced on each plant (1, 2, and 3) to maximize the company's profit.

x1 = small product, plant 1 x2 = medium product, plant 1 x3 = large product, plant 1 x4 = small product, plant 2 x5 = medium product, plant 2 x6 = large product, plant 2 x7 = small product, plant 3 x8 = medium product, plant 3 x9 = large product, plant 3

Formulate a linear programming model for this problem.

Let x1 = small product, plant 1 x2 = medium product, plant 1 x3 = large product, plant 1 x4 = small product, plant 2 x5 = medium product, plant 2 x6 = large product, plant 2 x7 = small product, plant 3 x8 = medium product, plant 3 x9 = large product, plant 3

Max Z = 300(x1 + x4 + x7) + 360(x2 + x5 + x8) + 420(x3 + x6 + x9)

Subject to

Excess Capacity x1 + x2 + x3 <= 750 x4 + x5 + x6 <= 900 x7 + x8 + x9 <= 450

In-process Storage Space (SQ FT) 12(x1) + 15(x2) + 20(x3) <= 13000 12(x4) + 15(x5) + 20(x6) <= 12000 12(x7) + 15(x8) + 20(x9) <= 5000

Sales Forecast x1 + x4 + x7 <= 750 x2 + x5 + x8 <= 1200 x3 + x6 + x9 <= 900

Percentage 900(x1 + x2 + x3) - 750(x4 + x5 + x6) = 0 450(x4 + x5 + x6) - 900(x7 + x8 + x9) = 0 450(x1 + x2 + x3) - 750(x7 + x8 + x9) = 0

Other x1,x2, x3, x4, x5, x6, x7, x8, x9 >= 0

#Solve, identify variables and objective funtion

```
dvar <- make.lp(0,9)
set.objfn(dvar,c(420,420,420,360,360,360,300,300,300))
lp.control(dvar,sense='max')
```

```
## $anti.degen
## [1] "fixedvars" "stalling"
##
## $basis.crash
## [1] "none"
##
## $bb.depthlimit
## [1] -50
##
## $bb.floorfirst
## [1] "automatic"
##
## $bb.rule
## [1] "pseudononint" "greedy"       "dynamic"      "rcostfixing"
##
## $break.at.first
## [1] FALSE
##
## $break.at.value
## [1] 1e+30
##
## $epsilon
##       epsb       epsd      epsel     epsint epsperturb   epspivot
##      1e-10      1e-09      1e-12      1e-07      1e-05      2e-07
##
## $improve
## [1] "dualfeas" "thetagap"
##
## $infinite
## [1] 1e+30
##
## $maxpivot
## [1] 250
##
## $mip.gap
## absolute relative
##    1e-11    1e-11
##
## $negrange
## [1] -1e+06
```

```
##
## $obj.in.basis
## [1] TRUE
##
## $pivoting
## [1] "devex"    "adaptive"
##
## $presolve
## [1] "none"
##
## $scalelimit
## [1] 5
##
## $scaling
## [1] "geometric"   "equilibrate" "integers"
##
## $sense
## [1] "maximize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

#Add constraints Excess Capacity

```
add.constraint(dvar, c(1,0,0,1,0,0,1,0,0), "<=", 750)
add.constraint(dvar, c(0,1,0,0,1,0,0,1,0), "<=", 900)
add.constraint(dvar, c(0,0,1,0,0,1,0,0,1), "<=", 450)
```

Storage Space (SQ FT)

```
add.constraint(dvar, c(20,0,0,15,0,0,12,0,0), "<=", 13000)
add.constraint(dvar, c(0,20,0,0,15,0,0,12,0), "<=", 12000)
add.constraint(dvar, c(0,0,20,0,0,15,0,0,12), "<=", 5000)
```

Sales Forecast

```
add.constraint(dvar, c(1,1,1,0,0,0,0,0,0), "<=", 900)
add.constraint(dvar, c(0,0,0,1,1,1,0,0,0), "<=", 1200)
add.constraint(dvar, c(0,0,0,0,0,0,1,1,1), "<=", 750)
```

Percentage

```
add.constraint(dvar, c(0.001333,-0.00111,0,0.001333,-0.00111,0,0.001333,-0.00111,0), "=", 0)
add.constraint(dvar, c(0,-0.00111,0.00222,0,-0.00111,0.00222,0,-0.00111,0.00222), "=", 0)
```

Set up linear Program and ensure program ran correctly

```
Rows <- c("1", "2", "3", "4", "5,", "6", "7", "8", "9", "10", "11")
Columns <- c("1", "2", "3", "4", "5,", "6", "7", "8", "9")
dimnames(dvar) <- list(Rows,Columns)
dvar
```

```
## Model name:
##   a linear program with 9 decision variables and 11 constraints
```

```
solve(dvar)
```

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

Get answers

```
get.objective(dvar)
```

```
## [1] 695906.2
```

```
get.variables(dvar)
```

```
## [1] 518.2296   0.0000   0.0000 175.6939 666.6667   0.0000   0.0000 166.6667
## [9] 416.6667
```

```
get.constraints(dvar)
```

```
##  [1]   693.9235   833.3333   416.6667 13000.0000 12000.0000  5000.0000
##  [7]   518.2296   842.3606   583.3333     0.0000     0.0000
```

#Description of Results Maximum Profit = $695,906.2 To achieve this maximum profit, the company needs to produce: - Large: Plant 1 at 518 units, Plant 2 at 0 units, Plant 3 at 0 units - Medium: Plant 1 at 176 units, Plant 2 at 667 units, Plant 3 at 0 units - Small: Plant 1 at 0 units, Plant 2 at 167 units, Plant 3 at 417 units

**ALL QUANTITIES HAVE BEEN ROUNDED UP BECAUSE THEY CAN ONLY PRODUCE WHOLE UNITS