mbruner3_mod4

Mark Bruner

9/21/2020

```r
library(lpSolveAPI)
```

defining decision variables and objective function

```r
factory <- make.lp(9, 9)
lp.control(factory, 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"
```

```r
set.objfn(factory, c(420, 420, 420, 360, 360, 360, 300, 300, 300))
```

set constraints

```r
add.constraint(factory, c(20, 0, 0, 15, 0, 0, 12, 0, 0), "<=", 13000)
add.constraint(factory, c(0, 20, 0, 0, 15, 0, 0, 12, 0), "<=", 12000)
add.constraint(factory, c(0, 0, 20, 0, 0, 15, 0, 0, 12), "<=", 5000)
add.constraint(factory, c(1, 1, 1), indices = c(1, 4, 7), "<=", 750)
add.constraint(factory, c(1, 1, 1), indices = c(2, 5, 8), "<=", 900)
add.constraint(factory, c(1, 1, 1), indices = c(3, 6, 9), "<=", 450)
add.constraint(factory, c(1, 1, 1), indices = c(1, 4, 7), "<=", 900)
add.constraint(factory, c(1, 1, 1), indices = c(2, 5, 8), "<=", 1200)
add.constraint(factory, c(1, 1, 1), indices = c(3, 6, 9), "<=", 750)
```

```
add.constraint(factory, c(1/750, 1/750, 1/750, -1/900, -1/900, -1/900), indices = c(1, 2, 4, 5, 7, 8),
add.constraint(factory, c(1/750, 1/750, 1/750, -1/900, -1/900, -1/900), indices = c(1, 3, 4, 6, 7, 9),
```

Solve LP model

```
solve(factory)
```

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

```
get.objective(factory)
```

```
## [1] 694448.2
```

```
get.variables(factory)
```

```
## [1] 481.81818 203.48837   0.00000  48.48485   0.00000   0.00000 219.69697
## [8] 660.85271 416.66667
```

Interpreting the output from optimization routines

L1 = 481.81818 L2 = 203.48837 L3 = 0.00000 M1 = 48.48485 M2 = 0.00000 M3 = 0.00000 S1 = 219.69697 S2 = 660.85271 S3 = 416.66667

In order to satisfy the constraints on the LP Model, the Weigelt Corporation should produce the following quantities of L, M, and S at each factory:

Factory 1 (space = 13000 sq ft and capacity is 750 units, rounded due to the inability to produce partial products.) L1 = 481 M1 = 48 S1 = 220

Factory 2 (space = 12000 sq ft and capacity is 900 units, rounded due to the inability to produce partial products.) L2 = 203 M2 = 0 S2 = 661

Factory 3 (space = 5000 sq ft and capacity is 450 units, rounded due to the inability to produce partial products.) L3 = 0 M3 = 0 S3 = 417

Interpreting optimal objective function value:

If the above production occurs at each factory, optimal amount of money they can expect in profit per day. ~$694,448.20