mbruner3.6

Mark Bruner

9/30/2020

```
library(lpSolveAPI)
```

Defining decision variables and objective function

```
lprec <- make.lp(0, 6)
lp.control(lprec, sense = "min")
```

```
## $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] "minimize"
##
## $simplextype
## [1] "dual"   "primal"
##
## $timeout
## [1] 0
##
## $verbose
## [1] "neutral"
```

```
set.objfn(lprec, c(622, 614, 630, 641, 645, 649))
```

set constraints

Capacity Constraints

```
add.constraint(lprec, c(rep(1, 3)), indices = c(1, 2, 3), "<=", 100)
add.constraint(lprec, c(rep(1, 3)), indices = c(4, 5, 6), "<=", 120)
add.constraint(lprec, c(rep(1, 2)), indices = c(1, 4), "=", 80)
add.constraint(lprec, c(rep(1, 2)), indices = c(2, 5), "=", 60)
add.constraint(lprec, c(rep(1, 2)), indices = c(3, 6), "=", 70)

set.bounds(lprec, lower = c(0, 0, 0, 0, 0, 0))
```

Decision Variable Names

```
RowNames <- c("A", "B", "W1", "W2", "W3")
ColNames <- c("A1","A2","A3", "B1","B2","B3")

dimnames(lprec) <- list(RowNames, ColNames)
lprec
```

```
## Model name:
##             A1    A2    A3    B1    B2    B3
## Minimize   622   614   630   641   645   649
## A            1     1     1     0     0     0  <=  100
## B            0     0     0     1     1     1  <=  120
## W1           1     0     0     1     0     0   =   80
## W2           0     1     0     0     1     0   =   60
## W3           0     0     1     0     0     1   =   70
## Kind       Std   Std   Std   Std   Std   Std
## Type      Real  Real  Real  Real  Real  Real
## Upper      Inf   Inf   Inf   Inf   Inf   Inf
## Lower        0     0     0     0     0     0
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
write.lp(lprec, filename = "lprec.lp")
x <- read.lp(filename = "lprec.lp")

solve(x)
```

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

```
get.objective(x)
```

```
## [1] 132790
```

```
get.constraints(x)     # get constraint RHS values
```

```
## [1] 100 110  80  60  70
```

```r
get.variables(x)
```

```
## [1]  0 60 40 80  0 30
```

Summary of results

The minimum cost of producing and delivering the AED to each of the warehouse is $132,790.

To achieve the minimum cost, Plant A & B should produce 100 and 110 AED's respectively in order to maximize the demand for each warehouse (80, 60, 70 units respectively for Warehouses 1 to 3). Plant A should deliver 60 AED's to Warehouse 2. Plant A should deliver 40 AED's to Warehouse 3. Plant B should deliver 80 AED's to Warehouse 1. Plant B should deliver 30 AED's to Warehouse 3.