hcronin_#6

Hannah Cronin

2022-10-17

```
library(lpSolve)
library(lpSolveAPI)
```

1. Formulate problem

- (Combined the cost of shipping/production for each decision variable) Objective function- min(Z) = 622 PA1 + 614PA2 + 630 PA3 + 641 PB1 + 645 PB2 + 649 PB3

Constraints- PA1 + PA2 + PA3 <= 100 #plant A capacity PB1 + PB2 + PB3 <= 120 #plant B capacity PA1 + PB1 >= 80 #WH1 demand PA2 + PB2 >= 60 #WH2 demand PA3 + PB3 >= 70 #WH3 demand

```
# 0 constraints, 8 variables
shipping = make.lp(0,6)
```

```
set.objfn(shipping, c(622, 614, 630, 641, 645, 649)) #objective function
lp.control(shipping, sense = 'min' ) #setting problem to min to minimize costs
```

```
## $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"
```

```
add.constraint(shipping, c(1,1,1,0,0,0), "<=", 100) #PlantA Capacity
add.constraint(shipping, c(0,0,0,1,1,1), "<=", 120) #PlantB Capacity
add.constraint(shipping, c(1,0,0,1,0,0), ">=", 80) #WH1 Demand
add.constraint(shipping, c(0,1,0,0,1,0), ">=", 60) #WH2 Demand
add.constraint(shipping, c(0,0,1,0,0,1), ">=", 70) #WH3 Demand
```

```
Rownames = c('P1C', 'PAC', 'WH1D', 'WH2', 'WH3D')
Colnames = c('PA1', 'PA2', 'PA3', 'PB1', 'PB2', 'PB3')
dimnames(shipping) = list(Rownames, Colnames)
```

```
shipping #formula model
```

```
## Model name:
##            PA1   PA2   PA3   PB1   PB2   PB3
## Minimize   622   614   630   641   645   649
## P1C          1     1     1     0     0     0  <=  100
## PAC          0     0     0     1     1     1  <=  120
## WH1D         1     0     0     1     0     0  >=   80
## WH2          0     1     0     0     1     0  >=   60
## WH3D         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
```

```
solve(shipping)
```

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

```
get.objective(shipping)
```

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

```
get.variables(shipping)
```

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

```
get.constraints(shipping)
```

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

Answer is: min(Z) = 0 PA1 + 60 PA2 + 40 PA3 + 80 PB1 + 0 PB2 + 30 PB3 # There is an unused capacity of 10 units left in Plant B - however producing more of any unit would exceed # demand for any available warehouse.

```
 write.lp(shipping, filename = 'shipping.lp', type = 'lp')
```