QMM\_cwood19\_4

Chris Wood

2025-09-16

library(lpSolveAPI)  
  
# Create LP model with 0 constraints and 9 variables  
lpmodel <- make.lp(0, 9)  
  
# Name variables  
colnames(lpmodel) <- c("x11", "x12", "x13", "x21", "x22", "x23", "x31", "x32", "x33")  
  
# Objective (max)  
set.objfn(lpmodel, c(420, 360, 300, 420, 360, 300, 420, 360, 300))  
lp.control(lpmodel, 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"

# Capacity constraints  
add.constraint(lpmodel, c(1, 1, 1, 0, 0, 0, 0, 0, 0), "<=", 750)  
add.constraint(lpmodel, c(0, 0, 0, 1, 1, 1, 0, 0, 0), "<=", 900)  
add.constraint(lpmodel, c(0, 0, 0, 0, 0, 0, 1, 1, 1), "<=", 450)  
  
# Storage constraints  
add.constraint(lpmodel, c(20, 15, 12, 0, 0, 0, 0, 0, 0), "<=", 13000)  
add.constraint(lpmodel, c(0, 0, 0, 20, 15, 12, 0, 0, 0), "<=", 12000)  
add.constraint(lpmodel, c(0, 0, 0, 0, 0, 0, 20, 15, 12), "<=", 5000)  
  
# Demand constraints  
add.constraint(lpmodel, c(1, 0, 0, 1, 0, 0, 1, 0, 0), "<=", 900)  
add.constraint(lpmodel, c(0, 1, 0, 0, 1, 0, 0, 1, 0), "<=", 1200)  
add.constraint(lpmodel, c(0, 0, 1, 0, 0, 1, 0, 0, 1), "<=", 750)  
  
# Equal use constraints  
add.constraint(lpmodel, c(900, 900, 900, -750, -750, -750, 0, 0, 0), "=", 0)  
add.constraint(lpmodel, c(0, 0, 0, 450, 450, 450, -900, -900, -900), "=", 0)  
  
# Non-negativity bounds (default)  
set.bounds(lpmodel, lower = rep(0, 9))  
  
# Solve  
solve(lpmodel)

## [1] 0

# Results  
# Max objective value  
objective\_value <- get.objective(lpmodel)  
cat(sprintf("Maximum Profit: $%.2f\n", objective\_value))

## Maximum Profit: $696000.00

# Get decision variable values & names  
solution <- get.variables(lpmodel)  
var\_names <- colnames(lpmodel)  
  
  
# Print results  
results <- data.frame(Variable = var\_names, Value = solution)  
print(results)

## Variable Value  
## 1 x11 516.6667  
## 2 x12 177.7778  
## 3 x13 0.0000  
## 4 x21 0.0000  
## 5 x22 666.6667  
## 6 x23 166.6667  
## 7 x31 0.0000  
## 8 x32 0.0000  
## 9 x33 416.6667