Loading the library

library(lpSolveAPI)

#Primal Problem Creating an lp with zero constraints and 9 variables

rm(Wcorp) # free up resources and memory

## Warning in rm(Wcorp): object 'Wcorp' not found

Wcorp<-make.lp(0,9)

Creating an objective function and setting it for maximization

set.objfn(Wcorp,rep(c(420,360,300),3)) # Objective function with 9 decision variables  
lp.control(Wcorp,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"

Adding the constraints to the problem

add.constraint(Wcorp,c(1,1,1),"<=",750,indices = c(1,2,3))   
add.constraint(Wcorp,c(1,1,1),"<=",900,indices = c(4,5,6))  
add.constraint(Wcorp,c(1,1,1),"<=",450,indices = c(7,8,9))  
add.constraint(Wcorp,c(20,15,12),"<=",13000,indices = c(1,2,3))  
add.constraint(Wcorp,c(20,15,12),"<=",12000,indices = c(4,5,6))  
add.constraint(Wcorp,c(20,15,12),"<=",5000,indices = c(7,8,9))  
add.constraint(Wcorp,c(1,1,1),"<=",900,indices = c(1,4,7))  
add.constraint(Wcorp,c(1,1,1),"<=",1200,indices = c(2,5,8))  
add.constraint(Wcorp,c(1,1,1),"<=",750,indices = c(3,6,9))  
add.constraint(Wcorp,c(rep(c(900,-750),each=3)),"=",0,indices = c(1:6))  
add.constraint(Wcorp,c(rep(c(450,-750),each=3)),"=",0,indices = c(1,2,3,7,8,9))

solve(Wcorp)

## [1] 0

get.objective(Wcorp)

## [1] 696000

get.constraints(Wcorp)

## [1] 6.944444e+02 8.333333e+02 4.166667e+02 1.300000e+04 1.200000e+04  
## [6] 5.000000e+03 5.166667e+02 8.444444e+02 5.833333e+02 -2.037268e-10  
## [11] 0.000000e+00

get.variables(Wcorp)

## [1] 516.6667 177.7778 0.0000 0.0000 666.6667 166.6667 0.0000 0.0000  
## [9] 416.6667

Reduced Cost

get.sensitivity.obj(Wcorp)

## $objfrom  
## [1] 3.60e+02 3.45e+02 -1.00e+30 -1.00e+30 3.45e+02 2.52e+02 -1.00e+30  
## [8] -1.00e+30 2.04e+02  
##   
## $objtill  
## [1] 4.60e+02 4.20e+02 3.24e+02 4.60e+02 4.20e+02 3.24e+02 7.80e+02 4.80e+02  
## [9] 1.00e+30

Shadow Prices

get.sensitivity.rhs(Wcorp)

## $duals  
## [1] 0.00 0.00 0.00 12.00 20.00 60.00 0.00 0.00 0.00  
## [10] -0.08 0.56 0.00 0.00 -24.00 -40.00 0.00 0.00 -360.00  
## [19] -120.00 0.00  
##   
## $dualsfrom  
## [1] -1.000000e+30 -1.000000e+30 -1.000000e+30 1.122222e+04 1.150000e+04  
## [6] 4.800000e+03 -1.000000e+30 -1.000000e+30 -1.000000e+30 -2.500000e+04  
## [11] -1.250000e+04 -1.000000e+30 -1.000000e+30 -2.222222e+02 -1.000000e+02  
## [16] -1.000000e+30 -1.000000e+30 -2.000000e+01 -4.444444e+01 -1.000000e+30  
##   
## $dualstill  
## [1] 1.000000e+30 1.000000e+30 1.000000e+30 1.388889e+04 1.250000e+04  
## [6] 5.181818e+03 1.000000e+30 1.000000e+30 1.000000e+30 2.500000e+04  
## [11] 1.250000e+04 1.000000e+30 1.000000e+30 1.111111e+02 1.000000e+02  
## [16] 1.000000e+30 1.000000e+30 2.500000e+01 6.666667e+01 1.000000e+30

Sens\_1<-data.frame(get.sensitivity.rhs(Wcorp)$duals[1:11],get.sensitivity.rhs(Wcorp)$dualsfrom[1:11],get.sensitivity.rhs(Wcorp)$dualstill[1:11])  
names(Sens\_1)<-c("Price","low","High")  
Sens\_1

## Price low High  
## 1 0.00 -1.000000e+30 1.000000e+30  
## 2 0.00 -1.000000e+30 1.000000e+30  
## 3 0.00 -1.000000e+30 1.000000e+30  
## 4 12.00 1.122222e+04 1.388889e+04  
## 5 20.00 1.150000e+04 1.250000e+04  
## 6 60.00 4.800000e+03 5.181818e+03  
## 7 0.00 -1.000000e+30 1.000000e+30  
## 8 0.00 -1.000000e+30 1.000000e+30  
## 9 0.00 -1.000000e+30 1.000000e+30  
## 10 -0.08 -2.500000e+04 2.500000e+04  
## 11 0.56 -1.250000e+04 1.250000e+04

Sens\_2<-data.frame(get.sensitivity.rhs(Wcorp)$duals[12:20],get.sensitivity.rhs(Wcorp)$dualsfrom[12:20],get.sensitivity.rhs(Wcorp)$dualstill[12:20])  
names(Sens\_2)<-c("Cost","low","High")  
Sens\_2

## Cost low High  
## 1 0 -1.000000e+30 1.000000e+30  
## 2 0 -1.000000e+30 1.000000e+30  
## 3 -24 -2.222222e+02 1.111111e+02  
## 4 -40 -1.000000e+02 1.000000e+02  
## 5 0 -1.000000e+30 1.000000e+30  
## 6 0 -1.000000e+30 1.000000e+30  
## 7 -360 -2.000000e+01 2.500000e+01  
## 8 -120 -4.444444e+01 6.666667e+01  
## 9 0 -1.000000e+30 1.000000e+30

#Dual Problem Creating an lp with zero constraints and 11 variables

Wcorp1<-make.lp(0,11)

Creating objective function and setting it to minimization problem

set.objfn(Wcorp1,c(750,900,450,13000,12000,5000,900,1200,750,0,0))  
lp.control(Wcorp1,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"

Creating Constraints

add.constraint(Wcorp1,c(1,20,1,900,450),">=",420,indices = c(1,4,9,10,11))  
add.constraint(Wcorp1,c(1,15,1,900,450),">=",360,indices = c(1,4,8,10,11))  
add.constraint(Wcorp1,c(1,12,1,900,450),">=",300,indices = c(1,4,7,10,11))  
add.constraint(Wcorp1,c(1,20,1,-750),">=",420,indices = c(2,5,9,10))  
add.constraint(Wcorp1,c(1,15,1,-750),">=",360,indices = c(2,5,8,10))  
add.constraint(Wcorp1,c(1,12,1,-750),">=",300,indices = c(2,5,7,10))  
add.constraint(Wcorp1,c(1,20,1,-750),">=",420,indices = c(3,6,9,11))  
add.constraint(Wcorp1,c(1,15,1,-750),">=",360,indices = c(3,6,8,11))  
add.constraint(Wcorp1,c(1,12,1,-750),">=",300,indices = c(3,6,7,11))  
set.bounds(Wcorp1,lower = c(-Inf,-Inf),columns = 10:11)

solve(Wcorp1)

## [1] 0

get.objective(Wcorp1)

## [1] 696000

get.constraints(Wcorp1)

## [1] 420 360 324 460 360 300 780 480 300

get.variables(Wcorp1)

## [1] 0.00 0.00 0.00 12.00 20.00 60.00 0.00 0.00 0.00 -0.08 0.56

get.sensitivity.obj(Wcorp1)

## $objfrom  
## [1] 694.4444 833.3333 416.6667 11222.2222 11050.0000 4800.0000  
## [7] 583.3333 844.4444 516.6667 -25000.0000 -12500.0000  
##   
## $objtill  
## [1] 1.000000e+30 1.000000e+30 1.000000e+30 1.388889e+04 1.250000e+04  
## [6] 5.345455e+03 1.000000e+30 1.000000e+30 1.000000e+30 1.000000e+30  
## [11] 2.375000e+04

get.sensitivity.rhs(Wcorp1)

## $duals  
## [1] 516.66667 177.77778 0.00000 0.00000 666.66667 166.66667 0.00000  
## [8] 0.00000 416.66667 55.55556 66.66667 33.33333 0.00000 0.00000  
## [15] 0.00000 316.66667 355.55556 233.33333 0.00000 0.00000  
##   
## $dualsfrom  
## [1] 3.600000e+02 3.450000e+02 -1.000000e+30 -1.000000e+30 3.450000e+02  
## [6] 2.880000e+02 -1.000000e+30 -1.000000e+30 2.040000e+02 -1.000000e+30  
## [11] -5.915411e+13 -1.000000e+30 -1.000000e+30 -1.000000e+30 -1.000000e+30  
## [16] -2.400000e+01 -1.500000e+01 -4.000000e+01 -1.000000e+30 -1.000000e+30  
##   
## $dualstill  
## [1] 4.60e+02 4.20e+02 1.00e+30 1.00e+30 3.75e+02 3.24e+02 1.00e+30 1.00e+30  
## [9] 1.00e+30 2.52e+02 6.00e+01 4.80e+02 1.00e+30 1.00e+30 1.00e+30 1.20e+01  
## [17] 1.50e+01 6.00e+01 1.00e+30 1.00e+30

The solution agrees with what we have observed in the primal problem

```