Assignment3QT_4Only

```
install.packages("lpSolveAPI")
```

```
library(lpSolveAPI)
```

Let us set up the problem. Note that we have two decision variables, and three constraints. In the first formulation, we will directly create the objective function and constraints

```
install.packages("lpSolveAPI")
```

```
## Warning: package 'lpSolveAPI' is in use and will not be installed
```

```
library(lpSolveAPI)
```

```
library(lpSolveAPI)
# make an lp object with 0 constraints and 9 decision variables
lprec <- make.lp(0, 9)

# Now create the objective function. The default is a minimization problem.
set.objfn(lprec, c(750, 900, 450, 13000, 12000, 5000, 900, 1200,750 ))
```

```
library(lpSolveAPI)
# As the default is a minimization problem, we change the direction to set maximization
lp.control(lprec,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"
```

```
# Add the price per three Squared Footage constraints
# Plant production by day
```

```
add.constraint(lprec,c(1,20,1,900,450,0,0,0,0), ">=",420)
add.constraint(lprec,c(0,0,1,15,1,900,450,0,0), ">=",360)
add.constraint(lprec,c(0,0,0,0,1,12,1,900,450), ">=",300)

#Price per Land storage for products by day
add.constraint(lprec,c(1,20,1,-750,0,0,0,0,0), ">=",420)
add.constraint(lprec,c(0,1,15,1,-750,0,0,0,0), ">=",360)
add.constraint(lprec,c(0,0,1,12,1,-750,0,0,0), ">=",300)

#Price per Sales for products by day
add.constraint(lprec,c(1,20,1,-750,0,0,0,0,0), ">=",420)
add.constraint(lprec,c(0,1,15,1,-750,0,0,0,0), ">=",360)
add.constraint(lprec,c(0,0,1,12,1,-750,0,0,0), ">=",300)


# Price of the plants with the same % capacity
add.constraint(lprec,c(-6,-6,-6,5,5,5,0,0,0), "=",0)
add.constraint(lprec,c(-3,-3,-3,0,0,0,5,5,5), "=",0)

RN<-c("Ccon1","Ccon2","Ccon3","SCon1", "SCon2","SCon3","saCon1","saCon2","saCon3","%C1","%C2")
CN<-c("P1L","P1M","P1S","P2L","P2M","P2S","P3L","P3M","P3S")
dimnames(lprec)<-list(RN, CN)
lprec
```

```
## Model name:
##   a linear program with 9 decision variables and 11 constraints
```

We now solve the above LP problem

```r
solve(lprec)
```

```
## [1] 2
```

```
get.objective(lprec)
```

```
## [1] -1e+30
```

```
get.variables(lprec)
```

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

```
get.dual.solution(lprec)
```

```
##  [1] 1.000000e+00 0.000000e+00 3.333333e-01 0.000000e+00 3.806075e+02
##  [6] 0.000000e+00 6.781209e+03 0.000000e+00 0.000000e+00 0.000000e+00
## [11] 1.043692e+03 1.500000e+02 7.081542e+03 0.000000e+00 0.000000e+00
## [16] 2.118576e+05 0.000000e+00 5.085388e+06 0.000000e+00 4.500000e+02
## [21] 0.000000e+00
```

```
get.sensitivity.rhs(lprec)
```

```
## $duals
##  [1] 0.000000e+00 3.333333e-01 0.000000e+00 3.806075e+02 0.000000e+00
##  [6] 6.781209e+03 0.000000e+00 0.000000e+00 0.000000e+00 1.043692e+03
## [11] 1.500000e+02 7.081542e+03 0.000000e+00 0.000000e+00 2.118576e+05
## [16] 0.000000e+00 5.085388e+06 0.000000e+00 4.500000e+02 0.000000e+00
##
## $dualsfrom
##  [1] -1.000000e+30 -6.000000e+01 -1.000000e+30  0.000000e+00 -1.000000e+30
##  [6]  0.000000e+00 -1.000000e+30 -1.000000e+30 -1.000000e+30 -1.836000e+03
## [11] -4.268800e+02 -3.191162e+02 -1.000000e+30 -1.000000e+30 -3.756773e-01
## [16] -1.000000e+30 -3.659121e-01 -1.000000e+30 -8.537600e+01 -1.000000e+30
##
## $dualstill
##  [1] 1.000000e+30 3.850476e+04 1.000000e+30 4.580000e+03 1.000000e+30
##  [6] 6.000000e+01 1.000000e+30 1.000000e+30 1.000000e+30 1.374000e+03
## [11] 1.000000e+30 2.410526e+02 1.000000e+30 1.000000e+30 4.907143e+00
## [16] 1.000000e+30 3.636364e-02 1.000000e+30 8.566106e+01 1.000000e+30
```