Assignment 6

[msasnur@kent.edu](mailto:msasnur@kent.edu)

04/11/2019

library(lpSolveAPI)  
library(Benchmarking)

## Loading required package: ucminf

library(ucminf)

Question 1

DMU 1

Max, z= 14000 u1 + 3500 u2

Subject to constraints,

14000 u1 + 3500 u2 – 150 v1 – 0.2 v2 <= 0

14000 u1 + 21000 u2 – 400 v1 – 0.7 v2 <= 0

42000 u1 + 10500 u2 – 320 v1 – 1.2 v2 <= 0

28000 u1 + 42000 u2 – 520 v1 – 2.0 v2 <= 0

19000 u1 + 25000 u2 – 350 v1 – 1.2 v2 <= 0

14000 u1 + 15000 u2 – 320 v1 – 0.7 v2 <= 0

150 v1 + 0.2 v2 = 1

v1, v2, u1, u2, >=0

lpdea1<-make.lp(0,4)  
lp.control(lpdea1,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"

set.objfn(lpdea1,c(0,0,14000,3500))  
# Constraints for Facility 1  
add.constraint(lpdea1,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea1,c(150,0.2),"=",1,indices = c(1,2))  
solve(lpdea1)

## [1] 0

get.objective(lpdea1)

## [1] 1

get.variables(lpdea1)

## [1] 5.172414e-03 1.120690e+00 7.142857e-05 0.000000e+00

#Here the objective value is 1, which indicates that we can achieve maximum efficiency from DMU(1).  
#This happens when we use the weights 0 and 0.00714 for the outputs, 0.5172414 and 1.12069 for the input.

DMU 2

Max, z= 14000 u1 + 21000 u2

Subject to constraints,

14000 u1 + 3500 u2 – 150 v1 – 0.2 v2 <= 0

14000 u1 + 21000 u2 – 400 v1 – 0.7 v2 <= 0

42000 u1 + 10500 u2 – 320 v1 – 1.2 v2 <= 0

28000 u1 + 42000 u2 – 520 v1 – 2.0 v2 <= 0

19000 u1 + 25000 u2 – 350 v1 – 1.2 v2 <= 0

14000 u1 + 15000 u2 – 320 v1 – 0.7 v2 <= 0

400 v1 + 0.7 v2 = 1

v1, v2, u1, u2, >=0

lpdea2<-make.lp(0,4)  
lp.control(lpdea2,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"

set.objfn(lpdea2,c(0,0,14000,21000))  
#Constraints for Facility 2  
add.constraint(lpdea2,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea2,c(400,0.7),"=",1,indices = c(1,2))  
solve(lpdea2)

## [1] 0

get.objective(lpdea2)

## [1] 1

get.variables(lpdea2)

## [1] 1.376147e-03 6.422018e-01 0.000000e+00 4.761905e-05

#Here the objective value is 1, which indicates that we can achieve maximum efficiency for DMU(2).   
#This happens when we use the weights 0 and 0.00476 for the outputs, 0.1376147 and 0.6422 for the input.

DMU 3

Max, z= 42000 u1 + 10500 u2

Subject to constraints,

14000 u1 + 3500 u2 – 150 v1 – 0.2 v2 <= 0

14000 u1 + 21000 u2 – 400 v1 – 0.7 v2 <= 0

42000 u1 + 10500 u2 – 320 v1 – 1.2 v2 <= 0

28000 u1 + 42000 u2 – 520 v1 – 2.0 v2 <= 0

19000 u1 + 25000 u2 – 350 v1 – 1.2 v2 <= 0

14000 u1 + 15000 u2 – 320 v1 – 0.7 v2 <= 0

320 v1 + 1.2 v2 = 1

v1, v2, u1, u2, >=0

lpdea3<-make.lp(0,4)  
lp.control(lpdea3,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"

set.objfn(lpdea3,c(0,0,42000,10500))  
# Constraints for Facility 3  
add.constraint(lpdea3,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea3,c(320,1.2),"=",1,indices = c(1,2))  
solve(lpdea3)

## [1] 0

get.objective(lpdea3)

## [1] 1

get.variables(lpdea3)

## [1] 1.724138e-03 3.735632e-01 2.380952e-05 0.000000e+00

#Here the objective value is 1, which indicates that we can achieve maximum efficiency for DMU(3).   
#This happens when we use the weights 0 and 0.00238 for the outputs, 0.001724 and 0.3735 for the input.

DMU 4: Max, z= 28000 u1 + 42000 u2

Subject to constraints,

14000 u1 + 3500 u2 – 150 v1 – 0.2 v2 <= 0

14000 u1 + 21000 u2 – 400 v1 – 0.7 v2 <= 0

42000 u1 + 10500 u2 – 320 v1 – 1.2 v2 <= 0

28000 u1 + 42000 u2 – 520 v1 – 2.0 v2 <= 0

19000 u1 + 25000 u2 – 350 v1 – 1.2 v2 <= 0

14000 u1 + 15000 u2 – 320 v1 – 0.7 v2 <= 0

520 v1 + 2.0 v2 = 1 v1, v2, u1, u2, >=0

lpdea4<-make.lp(0,4)  
lp.control(lpdea4,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"

set.objfn(lpdea4,c(0,0,28000,42000))  
  
#Constraints for Facility 4  
add.constraint(lpdea4,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea4,c(520,2.0),"=",1,indices = c(1,2))  
solve(lpdea4)

## [1] 0

get.objective(lpdea4)

## [1] 1

get.variables(lpdea4)

## [1] 6.880734e-04 3.211009e-01 0.000000e+00 2.380952e-05

#The solution indicates that the objective value is 1, which indicates that we can achieve maximum efficiency for DMU(4).  
#This happens when we use the weights 0 and 0.00238 for the outputs, 0.0688 and 0.3211 for the input.

DMU 5

Max, z= 19000 u1 + 25000 u2

Subject to constraints,

14000 u1 + 3500 u2 – 150 v1 – 0.2 v2 <= 0

14000 u1 + 21000 u2 – 400 v1 – 0.7 v2 <= 0

42000 u1 + 10500 u2 – 320 v1 – 1.2 v2 <= 0

28000 u1 + 42000 u2 – 520 v1 – 2.0 v2 <= 0

19000 u1 + 25000 u2 – 350 v1 – 1.2 v2 <= 0

14000 u1 + 15000 u2 – 320 v1 – 0.7 v2 <= 0

350 v1 + 1.2 v2 = 1

v1, v2, u1, u2, >=0

lpdea5<-make.lp(0,4)  
lp.control(lpdea5,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"

set.objfn(lpdea5,c(0,0,19000,25000))  
  
# Constraints for Facility 5  
add.constraint(lpdea5,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea5,c(350,1.2),"=",1,indices = c(1,2))  
solve(lpdea5)

## [1] 0

get.objective(lpdea5)

## [1] 0.9774987

get.variables(lpdea5)

## [1] 0.0010989011 0.5128205128 0.0000115123 0.0000303506

#Here the objective value is 0.9775, which indicates that we are not able to achieve maximum efficiency for DMU(5).   
#This happens when we use the weights 0.00001151 and 0.000030350 for the outputs, 0.001098 and 0.5128 for the input.   
  
##DMU 6  
lpdea6<-make.lp(0,4)  
lp.control(lpdea6,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"

set.objfn(lpdea6,c(0,0,14000,15000))  
# Constraints for Facility 6  
add.constraint(lpdea6,c(-400,-0.7,14000,21000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(-150,-0.2,14000,3500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(-320,-1.2,42000,10500),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(-520,-2.0,28000,42000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(-350,-1.2,19000,25000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(-320,-0.7,14000,15000),"<=",0,indices = c(1,2,3,4))  
add.constraint(lpdea6,c(320,0.7),"=",1,indices = c(1,2))  
solve(lpdea6)

## [1] 0

get.objective(lpdea6)

## [1] 0.8674521

get.variables(lpdea6)

## [1] 1.546392e-03 7.216495e-01 1.620029e-05 4.270987e-05

#The solution indicates that the objective value is 0.08674521, which indicates that we are not able to achieve maximum efficiency for DMU(6). This happens when we use the weights 0.00162 and 0.00427 for the outputs, 0.1546 and 0.7216 for the input.   
  
# DEA Analysis  
a<-matrix(c(150,400,320,520,350,320,0.2,0.7,1.2,2.0,1.2,0.7),ncol=2)  
b<-matrix(c(14000,14000,42000,28000,19000,14000,3500,21000,10500,42000,25000,15000),ncol=2)  
colnames(a)<-c("Staff Hours/Day","Supplies/Day")  
colnames(b)<-c("Reimbursed Patients/Day","Privately Paid Patients/Day")  
rownames(a)<-paste0(rep("Facility",6),seq(1,6,1))  
rownames(b)<-paste0(rep("Facility",6),seq(1,6,1))  
r1 <-dea(a,b,RTS = "crs")  
r2 <-dea(a,b,RTS = "fdh")  
r3 <-dea(a,b,RTS = "vrs")  
r4 <-dea(a,b,RTS = "irs")  
r5 <-dea(a,b,RTS = "drs")  
r6 <-dea(a,b,RTS = "add")  
r7 <-dea(a,b,RTS = "irs2")  
r8 <-dea(a,b,RTS = "fdh+")  
r9 <-dea(a,b,RTS = "vrs+")  
  
###Question 2:  
#Peers and Lambdas values for each assumption.  
peers(r1)

## peer1 peer2 peer3  
## Facility1 1 NA NA  
## Facility2 2 NA NA  
## Facility3 3 NA NA  
## Facility4 4 NA NA  
## Facility5 1 2 4  
## Facility6 1 2 4

lambda(r1)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4  
## Facility1 1.0000000 0.00000000 0 0.0000000  
## Facility2 0.0000000 1.00000000 0 0.0000000  
## Facility3 0.0000000 0.00000000 1 0.0000000  
## Facility4 0.0000000 0.00000000 0 1.0000000  
## Facility5 0.2000000 0.08048142 0 0.5383307  
## Facility6 0.3428571 0.39499264 0 0.1310751

# The peer units for DMU(5) are [1,2,4], with relative weights [0.2000000,0.08048142,0.5383307]. Similary for DMU(6), the peer units are [1,2,4], with weights [0.3428571,0.39499264,0.1310751], respectively.  
  
peers(r2)

## peer1  
## Facility1 1  
## Facility2 2  
## Facility3 3  
## Facility4 4  
## Facility5 5  
## Facility6 6

lambda(r2)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1 0 0 0 0  
## Facility2 0 1 0 0 0  
## Facility3 0 0 1 0 0  
## Facility4 0 0 0 1 0  
## Facility5 0 0 0 0 1  
## Facility6 0 0 0 0 0  
## L\_Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 1

peers(r3)

## peer1 peer2 peer3  
## Facility1 1 NA NA  
## Facility2 2 NA NA  
## Facility3 3 NA NA  
## Facility4 4 NA NA  
## Facility5 5 NA NA  
## Facility6 1 2 5

lambda(r3)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1.0000000 0.0000000 0 0 0.0000000  
## Facility2 0.0000000 1.0000000 0 0 0.0000000  
## Facility3 0.0000000 0.0000000 1 0 0.0000000  
## Facility4 0.0000000 0.0000000 0 1 0.0000000  
## Facility5 0.0000000 0.0000000 0 0 1.0000000  
## Facility6 0.4014399 0.3422606 0 0 0.2562995

peers(r4)

## peer1 peer2 peer3  
## Facility1 1 NA NA  
## Facility2 2 NA NA  
## Facility3 3 NA NA  
## Facility4 4 NA NA  
## Facility5 5 NA NA  
## Facility6 1 2 5

lambda(r4)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1.0000000 0.0000000 0 0 0.0000000  
## Facility2 0.0000000 1.0000000 0 0 0.0000000  
## Facility3 0.0000000 0.0000000 1 0 0.0000000  
## Facility4 0.0000000 0.0000000 0 1 0.0000000  
## Facility5 0.0000000 0.0000000 0 0 1.0000000  
## Facility6 0.4014399 0.3422606 0 0 0.2562995

peers(r5)

## peer1 peer2 peer3  
## Facility1 1 NA NA  
## Facility2 2 NA NA  
## Facility3 3 NA NA  
## Facility4 4 NA NA  
## Facility5 1 2 4  
## Facility6 1 2 4

lambda(r5)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4  
## Facility1 1.0000000 0.00000000 0 0.0000000  
## Facility2 0.0000000 1.00000000 0 0.0000000  
## Facility3 0.0000000 0.00000000 1 0.0000000  
## Facility4 0.0000000 0.00000000 0 1.0000000  
## Facility5 0.2000000 0.08048142 0 0.5383307  
## Facility6 0.3428571 0.39499264 0 0.1310751

peers(r6)

## peer1  
## Facility1 1  
## Facility2 2  
## Facility3 3  
## Facility4 4  
## Facility5 5  
## Facility6 6

lambda(r6)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1 0 0 0 0  
## Facility2 0 1 0 0 0  
## Facility3 0 0 1 0 0  
## Facility4 0 0 0 1 0  
## Facility5 0 0 0 0 1  
## Facility6 0 0 0 0 0  
## L\_Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 1

peers(r7)

## peer1  
## Facility1 1  
## Facility2 2  
## Facility3 3  
## Facility4 4  
## Facility5 5  
## Facility6 6

lambda(r7)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1 0 0 0 0  
## Facility2 0 1 0 0 0  
## Facility3 0 0 1 0 0  
## Facility4 0 0 0 1 0  
## Facility5 0 0 0 0 1  
## Facility6 0 0 0 0 0  
## L\_Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 1

peers(r8)

## peer1  
## Facility1 1  
## Facility2 2  
## Facility3 3  
## Facility4 4  
## Facility5 5  
## Facility6 6

lambda(r8)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1 0 0 0 0  
## Facility2 0 1 0 0 0  
## Facility3 0 0 1 0 0  
## Facility4 0 0 0 1 0  
## Facility5 0 0 0 0 1  
## Facility6 0 0 0 0 0  
## L\_Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 1

peers(r9)

## peer1  
## Facility1 1  
## Facility2 2  
## Facility3 3  
## Facility4 4  
## Facility5 5  
## Facility6 6

lambda(r9)

## L\_Facility1 L\_Facility2 L\_Facility3 L\_Facility4 L\_Facility5  
## Facility1 1 0 0 0 0  
## Facility2 0 1 0 0 0  
## Facility3 0 0 1 0 0  
## Facility4 0 0 0 1 0  
## Facility5 0 0 0 0 1  
## Facility6 0 0 0 0 0  
## L\_Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 1

###Question 3:  
result <-cbind(round(r1$eff,4),round(r1$lambda,4))  
colnames(result)<-c("efficieny",rownames(result))  
result

## efficieny Facility1 Facility2 Facility3 Facility4 Facility5  
## Facility1 1.0000 1.0000 0.0000 0 0.0000 0  
## Facility2 1.0000 0.0000 1.0000 0 0.0000 0  
## Facility3 1.0000 0.0000 0.0000 1 0.0000 0  
## Facility4 1.0000 0.0000 0.0000 0 1.0000 0  
## Facility5 0.9775 0.2000 0.0805 0 0.5383 0  
## Facility6 0.8675 0.3429 0.3950 0 0.1311 0  
## Facility6  
## Facility1 0  
## Facility2 0  
## Facility3 0  
## Facility4 0  
## Facility5 0  
## Facility6 0

# From the above table, except Facility5 and Facility6 nursing homes remaining nursing homes are efficient.  
# The efficiency of these nursing homes can be achieved by using shadow prices (lambda values from the above table, they are the variables related to the constraints limiting the efficiency of each unit to be no greater than 1).  
  
###Question 4  
c1<-cbind(r1$eff,r2$eff,r3$eff,r4$eff,r5$eff,r6$eff,r7$eff,r8$eff,r9$eff)  
colnames(c1)<-c(paste0(rep("r",9),seq(1,9,1)))  
c1

## r1 r2 r3 r4 r5 r6 r7 r8 r9  
## Facility1 1.0000000 1 1.0000000 1.0000000 1.0000000 1 1 1 1  
## Facility2 1.0000000 1 1.0000000 1.0000000 1.0000000 1 1 1 1  
## Facility3 1.0000000 1 1.0000000 1.0000000 1.0000000 1 1 1 1  
## Facility4 1.0000000 1 1.0000000 1.0000000 1.0000000 1 1 1 1  
## Facility5 0.9774987 1 1.0000000 1.0000000 0.9774987 1 1 1 1  
## Facility6 0.8674521 1 0.8963283 0.8963283 0.8674521 1 1 1 1

# For each assumption, efficiency of the units have varied