jnoxon_5.R

Jason

2021-11-08

```r
# Assignment 5

library(lpSolveAPI)
library(Benchmarking)
```

```
## Warning: package 'Benchmarking' was built under R version 4.0.5
```

```
## Loading required package: ucminf
```

```
## Loading required package: quadprog
```

```r
#Question 1

inputs <- matrix(c(150,400,320,520,350, 320, 0.2, 0.7, 1.2, 2.0, 1.2, 0.7), ncol = 2)

outputs <- matrix(c(14000,14000,42000,28000,19000, 14000, 3500, 21000, 10500, 42000, 25000, 15000), nco

colnames(outputs) <- c("reimbursed patient days","privately paid patient days")
colnames(inputs) <- c("staff hours per day", "supplies per day")

inputs
```

```
##      staff hours per day supplies per day
## [1,]                 150              0.2
## [2,]                 400              0.7
## [3,]                 320              1.2
## [4,]                 520              2.0
## [5,]                 350              1.2
## [6,]                 320              0.7
```

```r
outputs
```

```
##      reimbursed patient days privately paid patient days
## [1,]                   14000                        3500
## [2,]                   14000                       21000
## [3,]                   42000                       10500
## [4,]                   28000                       42000
## [5,]                   19000                       25000
## [6,]                   14000                       15000
```

```
fdh <- dea(inputs,outputs,RTS = "fdh")
fdh
```

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

```
peers(fdh) #identify peers
```

```
##      peer1
## [1,]     1
## [2,]     2
## [3,]     3
## [4,]     4
## [5,]     5
## [6,]     6
```

```
lambda(fdh) #identify lambda
```

```
##      L1 L2 L3 L4 L5 L6
## [1,]  1  0  0  0  0  0
## [2,]  0  1  0  0  0  0
## [3,]  0  0  1  0  0  0
## [4,]  0  0  0  1  0  0
## [5,]  0  0  0  0  1  0
## [6,]  0  0  0  0  0  1
```

```
crs <- dea(inputs,outputs,RTS = "crs")
crs
```

```
## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675
```

```
peers(crs)  #identify peers
```

```
##      peer1 peer2 peer3
## [1,]     1    NA    NA
## [2,]     2    NA    NA
## [3,]     3    NA    NA
## [4,]     4    NA    NA
## [5,]     1     2     4
## [6,]     1     2     4
```

```
lambda(crs) #identify lambda
```

```
##             L1         L2 L3        L4
## [1,] 1.0000000 0.00000000  0 0.0000000
## [2,] 0.0000000 1.00000000  0 0.0000000
## [3,] 0.0000000 0.00000000  1 0.0000000
## [4,] 0.0000000 0.00000000  0 1.0000000
## [5,] 0.2000000 0.08048142  0 0.5383307
## [6,] 0.3428571 0.39499264  0 0.1310751
```

```
vrs <- dea(inputs,outputs,RTS = "vrs")
vrs
```

```
## [1] 1.0000 1.0000 1.0000 1.0000 1.0000 0.8963
```

```
peers(vrs)    #identify peers
```

```
##      peer1 peer2 peer3
## [1,]     1    NA    NA
## [2,]     2    NA    NA
## [3,]     3    NA    NA
## [4,]     4    NA    NA
## [5,]     5    NA    NA
## [6,]     1     2     5
```

```
lambda(vrs) #identify lambda
```

```
##              L1        L2 L3 L4        L5
## [1,] 1.0000000 0.0000000  0  0 0.0000000
## [2,] 0.0000000 1.0000000  0  0 0.0000000
## [3,] 0.0000000 0.0000000  1  0 0.0000000
## [4,] 0.0000000 0.0000000  0  1 0.0000000
## [5,] 0.0000000 0.0000000  0  0 1.0000000
## [6,] 0.4014399 0.3422606  0  0 0.2562995
```

```
irs <- dea(inputs,outputs,RTS = "irs")
irs
```

```
## [1] 1.0000 1.0000 1.0000 1.0000 1.0000 0.8963
```

```
peers(irs)    #identify peers
```

```
##      peer1 peer2 peer3
## [1,]     1    NA    NA
## [2,]     2    NA    NA
## [3,]     3    NA    NA
## [4,]     4    NA    NA
## [5,]     5    NA    NA
## [6,]     1     2     5
```

```
lambda(irs) #identify lambda
```

```
##              L1        L2 L3 L4        L5
## [1,] 1.0000000 0.0000000  0  0 0.0000000
## [2,] 0.0000000 1.0000000  0  0 0.0000000
## [3,] 0.0000000 0.0000000  1  0 0.0000000
## [4,] 0.0000000 0.0000000  0  1 0.0000000
## [5,] 0.0000000 0.0000000  0  0 1.0000000
## [6,] 0.4014399 0.3422606  0  0 0.2562995
```

```r
drs <- dea(inputs,outputs,RTS = "drs")
drs
```

```
## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675
```

```r
peers(drs)    #identify peers
```

```
##      peer1 peer2 peer3
## [1,]     1    NA    NA
## [2,]     2    NA    NA
## [3,]     3    NA    NA
## [4,]     4    NA    NA
## [5,]     1     2     4
## [6,]     1     2     4
```

```r
lambda(drs) #identify lambda
```

```
##               L1         L2 L3        L4
## [1,] 1.0000000 0.00000000  0 0.0000000
## [2,] 0.0000000 1.00000000  0 0.0000000
## [3,] 0.0000000 0.00000000  1 0.0000000
## [4,] 0.0000000 0.00000000  0 1.0000000
## [5,] 0.2000000 0.08048142  0 0.5383307
## [6,] 0.3428571 0.39499264  0 0.1310751
```

```r
frh <- dea(inputs,outputs,RTS = "add")
frh
```

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

```r
peers(frh)    #identify peers
```

```
##      peer1
## [1,]     1
## [2,]     2
## [3,]     3
## [4,]     4
## [5,]     5
## [6,]     6
```

```r
lambda(frh) #identify lambda
```

```
##      L1 L2 L3 L4 L5 L6
## [1,]  1  0  0  0  0  0
## [2,]  0  1  0  0  0  0
## [3,]  0  0  1  0  0  0
## [4,]  0  0  0  1  0  0
## [5,]  0  0  0  0  1  0
## [6,]  0  0  0  0  0  1
```

```r
#the efficiency ratings for each analysis method vary for respective DMU's
#the number for peer columns vary by analysis
#a commonality between the analyses: facility 6 has effeciency <1 in several

#Question 2

# Z = P - 6C = 3D

# 20x1 + 15x2 + 25x3 (unbound)
# 6x1 + 4x2 + 5x3 = 50
# 8x1 + 7x2 + 5x3 >= 75

# P = 20x1 + 15x2 + 25x3
# y1 = 6x1 + 4x2 + 5x3 - 50
# y2 = 8x1 + 7x2 + 5x3 - 75

#y1(+) - y1(-) = 6x1 + 4x2 + 5x3 - 50 or 6x1 + 4x2 + 5x3 - (y1(+) - y1(-)) = 50
#y2(+) - y2(-) = 8x1 + 7x2 + 5x3 - 75 or 8x1 + 7x2 + 5x3 - (y2(+) - y2(-)) = 75

#objective function:
####### Z = 20x1 + 15x2 + 25x3 - 6y1(+) + 6y1(-) - 3y2(+) + 3y2(-)

goal <- make.lp(0, 7)

goal
```

```
## Model name:
##             C1    C2    C3    C4    C5    C6    C7
## Minimize     0     0     0     0     0     0     0
## Kind       Std   Std   Std   Std   Std   Std   Std
## Type      Real  Real  Real  Real  Real  Real  Real
## Upper      Inf   Inf   Inf   Inf   Inf   Inf   Inf
## Lower        0     0     0     0     0     0     0
```

```r
set.objfn(goal, c(20, 15, 25, 6, 6, 3, 3))

lp.control(goal, sense = "max")
```

```
## $anti.degen
## [1] "none"
##
## $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.constraint(goal, c(6, 4, 5, 1, 1, 0, 0 ), "=", 50)
add.constraint(goal, c(8, 7, 5, 0, 0, 1, 1 ), "=", 75)



solve(goal)
```

[1] 0

```
get.objective(goal)
```

[1] 525

```
get.variables(goal)
```

[1] 0 0 0 50 0 75 0