Assignment 5

Noorah

11/7/2021

```
###First Loading the libraries
library(lpSolveAPI)
library(Benchmarking)
```

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

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

1. The Hope Valley Health Care Association owns and operates six nursing homes in adjoining states. An evaluation of their efficiency has been undertaken using two inputs and two outputs. The inputs are staffing labor (measured in average hours per day) and the cost of supplies (in thousands of dollars per day). The outputs are the number of patient-days reimbursed by third-party sources and the number of patient-days reimbursed privately. A summary of performance data is shown in the table below.

We are going to solve the problem performing DEA analysis. DEA is a mathematical approach of estimating the best performance standards and then evaluating the relatively efficiency of the entities. These entities (the six nursing homes) are called Decision Making Units (DMUs). We first evaluate each DMU to determine its efficiency along the dimensions that are most favorable to the single DMU, and then for those DMUs that are not fully efficient, we establish a reference set, or technology set, to provide a way for them to improve efficiency. Entities in the technology set are called Peer Units.

One critical assumption in the analysis is that which DMU can belong to this technology set. Our choice of which model to use has significant implications on the technology set and on determining who is efficient and who is not.

We are going to use a specific package for DEA, “Benchmarking”, that tells us not only the weights for each DMU, but also the peer units in the hypothetical comparison set (HCUs) and the relative weights needed to make the DMU efficient.

The first step is to create the inputs and outputs by typing both the values and the relative column names. We have two columns of input (matrix x) and two columns of output (matrix y).

```
x <- matrix(c(150, 400, 320, 520, 350, 320,
              0.2, 0.7, 1.2, 2, 1.2, 0.7),
            ncol = 2)
y <- matrix(c(14000, 14000, 42000, 28000, 19000, 14000,
              3500, 21000, 10500, 42000, 25000, 15000),
            ncol = 2)
colnames(x) <- c("Staff Hours", "Supplies")
colnames(y) <- c("Reimbursed Patient Days","Privately Paid Patient Days")
x
```

```
##      Staff Hours Supplies
## [1,]         150      0.2
## [2,]         400      0.7
## [3,]         320      1.2
## [4,]         520      2.0
## [5,]         350      1.2
## [6,]         320      0.7
```

```
y
```

```
##      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
```

DEA analysis under all DEA assumptions of FDH, CRS, VRS, IRS, DRS, and FRH.

```
a1 <- dea(x,y,RTS = "fdh")
a1
```

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

```
peers(a1)  ##Peer identifying
```

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

```
lambda(a1) ##Peer weights
```

```
##      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
```

```
a2 <- dea(x,y,RTS = "crs")
a2
```

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

```
peers(a2) ##Peer identifying
```

```
##      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(a2) ##Peer weights
```

```
##             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
```

```
a3 <- dea(x,y,RTS = "vrs")
a3
```

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

```
peers(a3) ##Peer identifying
```

```
##      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(a3) ##Peer weights
```

```
##              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
```

```
a4 <- dea(x,y,RTS = "irs")
a4
```

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

```
peers(a4) ##Peer identifying
```

```
##      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(a4) ##Peer weights
```

```
##              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
a5 <- dea(x,y,RTS = "drs")
a5
```

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

```r
peers(a5) ##Peer identifying
```

```
##      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(a5) ##Peer weights
```

```
##             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
a6 <- dea(x,y,RTS = "add")
a6
```

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

```r
peers(a6) ##Peer identifying
```

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

```r
lambda(a6) ##Peer weights
```

```
##      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
```

For conclusion, we will have better estimation whenever we have larger set.

CRS and DRS give same results,and both VRS and IRS gave same results.

2. The Research and Development Division of the Emax Corporation has developed three new products. A decision now needs to be made on which mix of these products should be produced. Management wants primary consideration given to three factors: total profit, stability in the workforce, and achieving an increase in the company's earnings

The variables that will give the optimal production mix are the quantities of the new products, x1, x2, and x3.

create two variables:

y1 = 6x1 + 4x2 + 5x3 - 50 y2 = 8x1 + 7x2 + 5x3 - 75

Objective function:

Z = 20x1 + 15x2 + 25x3 - 6y1p - 6y1m -3 y2m

Creating.lp file that contains the formula for this problem using the lpSolveAPI

```
x <- read.lp("max.lp")
x
```

```
## Model name:
##              x1    x2    x3   y1p   y1n   y2n
## Maximize     20    15    25    -6    -6    -3
## R1            6     4     5    -1     1     0   =  50
## R2            8     7     5     0     0     1  >=  75
## Kind        Std   Std   Std   Std   Std   Std
## Type       Real  Real  Real  Real  Real  Real
## Upper       Inf   Inf   Inf   Inf   Inf   Inf
## Lower         0     0     0     0     0     0
```

```
solve(x)
```

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

```
get.objective(x)
```

```
## [1] 225
```

```
get.variables(x)
```

```
## [1]  0  0 15 25  0  0
```

The maximum result Z = 225

x1 = 0 x2 = 0 x3 = 15 y1p = 25 y1m = 0 y2m = 0

Total profit from the production: P = 25*15 = 375