QMM Assignment DEA

Laya Sree Gangula

2022-10-30

```
knitr::opts_chunk$set(message = FALSE)
knitr::opts_chunk$set(warning = FALSE)

library("Benchmarking")

data_dfvalues <- matrix(c("Facility 1","Facility 2","Facility 3","Facility 4"
,"Facility 5", "Facility 6",
                150,400,320,520,350,320,
                0.2,0.7,1.2,2.0,1.2,0.7,
                14000,14000,42000,28000,19000,14000,
                3500,21000,10500,42000,25000,15000), ncol=5, byrow=F)

colnames(data_dfvalues) <- c("DMU", "Staff_Hours_Per_Day","Supplies_Per_Day",
"Reimbursed_Patient_Days","Privately_Paid_Patient_Days")

table_df <- as.table(data_dfvalues)
table_df

##   DMU        Staff_Hours_Per_Day Supplies_Per_Day Reimbursed_Patient_Days
## A Facility 1 150                 0.2              14000
## B Facility 2 400                 0.7              14000
## C Facility 3 320                 1.2              42000
## D Facility 4 520                 2                28000
## E Facility 5 350                 1.2              19000
## F Facility 6 320                 0.7              14000
##   Privately_Paid_Patient_Days
## A 3500
## B 21000
## C 10500
## D 42000
## E 25000
## F 15000
```

Calculating CRS

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

y <- matrix(c(14000,14000,42000,28000,19000,14000,
                3500,21000,10500,42000,25000,15000),ncol=2)

colnames(y) <- c("Reimbursed_Patient_Days","Privately_Paid_Patient_Days")
```

```
colnames(x) <- c("Staff_Hours_Per_Day","Supplies_Per_Day")

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

## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675

peers(DEA_crs)

##      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(DEA_crs)

##             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
```

CRS findings -

1.As demonstrated, Facilities 1, 2, 3, and 4 are efficient.

2.Additionally, facilities 1, 2, and 4 are peers of facilities 5 and 6, which are likewise inadequate facilities..

3. Facility 5 is 97.75 % efficient making 2.25 % as inefficient. Facility 6 is 86.75 % efficient making 13.25 % as inefficient.

Calculating DRS

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

## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675

peers(DEA_drs)

##      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(DEA_drs)
```

```
##              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
```

DRS Observations - 1.Facilities 1, 2, 3, and 4 are effective as shown..

2.Additionally, the ineffective facilities Facility 5 and Facility 6 are peers with Facilities 1, 2, and 4..

3.Facility 5 is 2.75% less efficient than Facility 4, or 97.75%. Since Facility 6 is 86.75% efficient, its inefficiency is 13.25%..

Calculating IRS

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

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

```
peers(DEA_irs)
```

```
##      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(DEA_irs)
```

```
##              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 Observations -

1.Facilities 1, 2, 3, 4, and 5 are effective, as seen.

2.We can also see that Facility 6 is the only inefficient facility that has as peer members Facilities 1, 2, and 5.

3. Facility 6 has an efficiency of 89.63%, which leaves 10.37% inefficient.

Calculating VRS

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

## [1] 1.0000 1.0000 1.0000 1.0000 1.0000 0.8963

peers(DEA_vrs)

##      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(DEA_vrs)

##             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
```

VRS Observations -

1.We see how effective Facilities 1, 2, 3, 4, and 5 are.

2. Facility 6 is an ineffective facility, and its peer facilities are Facilities 1, 2, and 5.

3.Because Facility 6 is 89.63% efficient, 10.37% of it is inefficient..

Calculating FDH

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

## [1] 1 1 1 1 1 1

peers(DEA_fdh)

##      peer1
## [1,]     1
## [2,]     2
## [3,]     3
```

```
## [4,]     4
## [5,]     5
## [6,]     6
```

```
lambda(DEA_fdh)
```

```
##      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
```

Observations of FDH -

Due to the principle that the FDH technique follows, which can detect any level of effectiveness, every DMU is effective..

Calculating FRH

```
#here FRH is calculated by specifying RTS = "add"
DEA_frh <- dea(x, y, RTS = "add")
DEA_frh
```

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

```
peers(DEA_frh)
```

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

```
lambda(DEA_frh)
```

```
##      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
```

Summary of Inefficient DMUs

```
inefficient.data.df.summarise <- matrix(c("CRS","DRS","IRS","VRS","FDH","FRH"
,
2,2,1,1,0,0,
"Facility 5 & 6", "Facility 5 & 6","Facility 6", "Facility 6", "-","-",
```

```
"97.75% & 86.7%","97.75% & 86.7%","89.63%","89.63%","-","-",
"Facility 1, 2 & 4","Facility 1, 2 & 4","Facility 1, 2 & 5","Facility 1, 2 &
5","-","-",
"0.2, 0.08, 0.54 and 0.34, 0.4, 0.13", "0.2, 0.08, 0.54 and 0.34, 0.4, 0.13",
"0.4, 0.34 and 0.26", "0.4, 0.34 and 0.26", "-","-"),ncol=6,byrow=F)


colnames(inefficient.data.df.summarise) <- c("RTS","Count_Inefficient_DMUs","
Name_DMUs","%_Inefficiency","Peers","Lambda")

as.table(inefficient.data.df.summarise)

##   RTS Count_Inefficient_DMUs Name_DMUs      %_Inefficiency Peers
## A CRS 2                      Facility 5 & 6 97.75% & 86.7% Facility 1, 2 &
4
## B DRS 2                      Facility 5 & 6 97.75% & 86.7% Facility 1, 2 &
4
## C IRS 1                      Facility 6     89.63%         Facility 1, 2 &
5
## D VRS 1                      Facility 6     89.63%         Facility 1, 2 &
5
## E FDH 0                      -              -              -
## F FRH 0                      -              -              -
##   Lambda
## A 0.2, 0.08, 0.54 and 0.34, 0.4, 0.13
## B 0.2, 0.08, 0.54 and 0.34, 0.4, 0.13
## C 0.4, 0.34 and 0.26
## D 0.4, 0.34 and 0.26
## E -
## F -
```

Summary of Efficient DMUs

```
efficient.data.df.summarise <- matrix(c("CRS","DRS","IRS","VRS","FDH","FRH",
"Facility 1, 2, 3 & 4","Facility 1, 2, 3 & 4","Facility 1, 2, 3, 4 & 5", "Fac
ility 1, 2, 3, 4 & 5", "All DMUs", "All DMUs"), ncol = 2, byrow=F)

colnames(efficient.data.df.summarise) <- c("RTS", "Efficient_DMUs")

as.table(efficient.data.df.summarise)

##   RTS Efficient_DMUs
## A CRS Facility 1, 2, 3 & 4
## B DRS Facility 1, 2, 3 & 4
## C IRS Facility 1, 2, 3, 4 & 5
## D VRS Facility 1, 2, 3, 4 & 5
## E FDH All DMUs
## F FRH All DMUs
```

CRS - Constant Returns to Scale

The results demonstrate the efficiency of DMUs 1, 2, 3, and 4. Only 86.7% of DMU(6) and 97.75% of DMU(5) are utilized efficiently. We found this based on our preliminary analysis. Furthermore, the units of DMU(4peer) are 1, 2, and 4, with associated weights of 0.2, 0.08, and 0.54. For DMU(6), the peer units are 1, 2, and 4, and their corresponding values are 0.34, 0.4, and 0.13.

CRS essentially allows us to decide if any prospective DMUs, in this case DMUs 1, 2, 3, and 4, may be moved up or down.

DRS - Decreasing Returns to Scale

The results demonstrate the efficiency of DMUs 1, 2, 3, and 4. Only 86.7% and 97.75% of DMU(6) are effectively used. We found this based on our preliminary analysis. Furthermore, the units of DMU(4peer) are 1, 2, and 4, with weights of 0.2, 0.08, and 0.54. Peer units for DMU(6) are 1, 2, and 4, and their weights are 0.34, 0.4, and 0.13, respectively.

We can find out if there are any other DMUs where we can grow the processes by looking at the inefficient DMUs, in this example DMUs 5 and 6. The CRS values can also be used to get this data as the original base scale.

IRS - Increasing Returns to Scale

The results demonstrate the effectiveness of DMUs 1, 2, 3, 4, and 5. Ineffectiveness for the DMU(6) is only 89.63%. We found this based on our preliminary analysis. Furthermore, DMU(6) has three peer units: 1, 2, and 5, with associated relative weights of 0.4, 0.34, and 0.26.

Any company can decide whether it can arbitrarily grow its operations by looking at the efficiency scores.

VRS - Variable Returns to Scale

The results demonstrate the effectiveness of DMUs 1, 2, 3, 4, and 5. Ineffectiveness for the DMU(6) is only 89.63%. We found this based on our preliminary analysis. Furthermore, DMU(6) has three peer units: 1, 2, and 5, with associated relative weights of 0.4, 0.34, and 0.26.

Variable or variable returns to scale make it feasible to comprehend the scale of operations when input and output factors are changed—either by growing or decreasing, or by employing both.

FRH - Free Replicability Hull

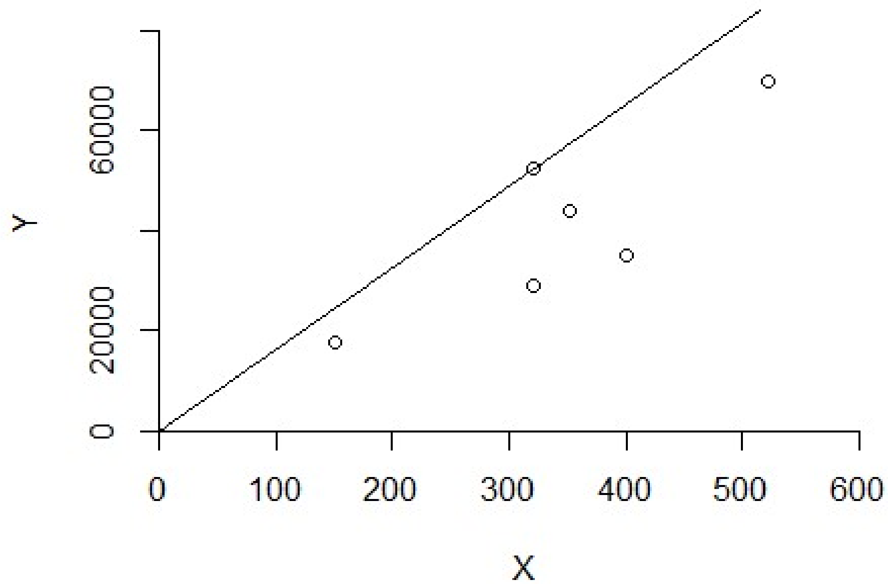
According to FRH data, all DMUs are efficient. This is primarily due to the lack of the convexity assumption and the ability of the scale to record even the smallest efficiency level that is not susceptible to replication or disposal thanks to this technique.

Conclusion

It is critical to remember that DEA is an extremely useful tool for any firm in determining which of the Decision Units must be optimized in order to achieve an increase, decrease, or *any other sort of modifications to the result by feeding input into it. Additionally, a business can decide which RTS, or Returns to Scale, it wishes to use based on its requirements; each of these levels is important in its own right.*

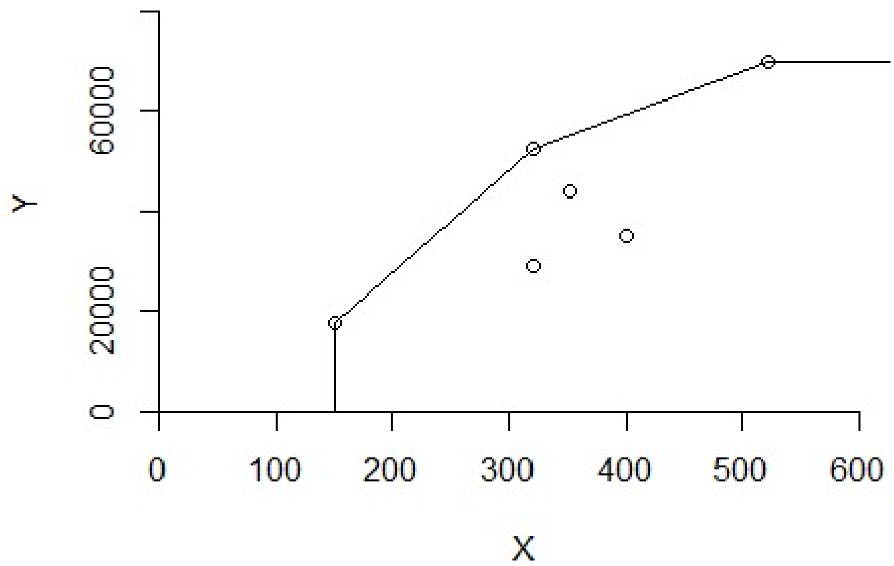
CRS Plot

```
dea.plot(x, y, RTS='crs')
```



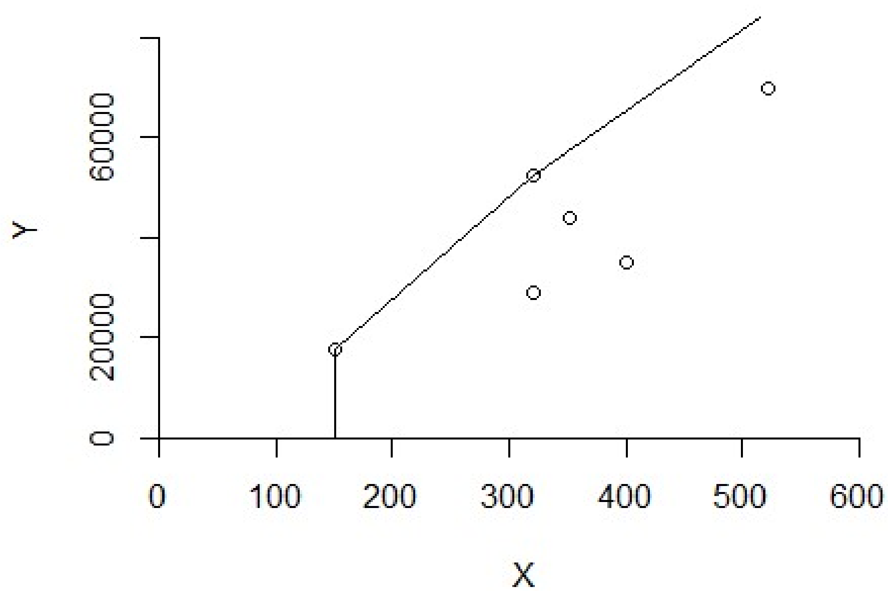
DRS Plot

```
dea.plot(x,y,RTS="vrs")
```



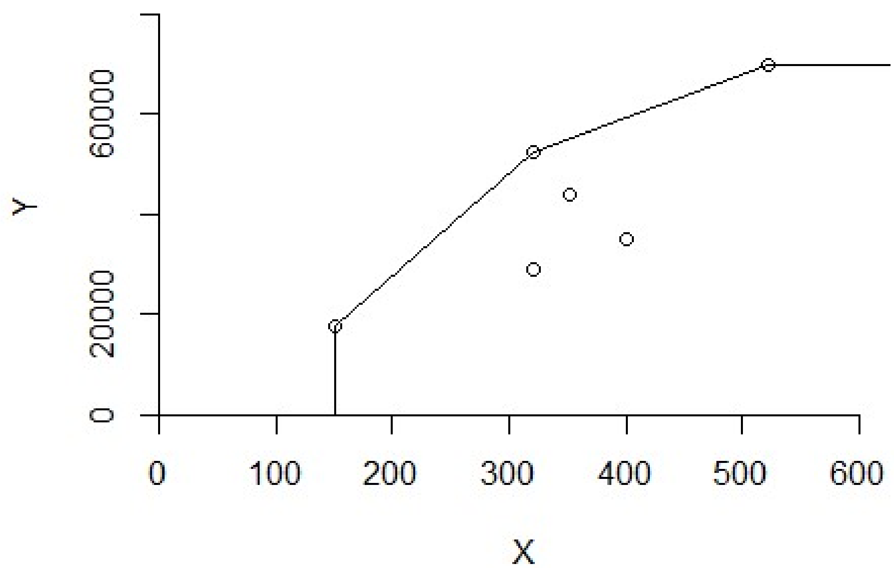
IRS Plot

```
dea.plot(x,y,RTS="irs")
```



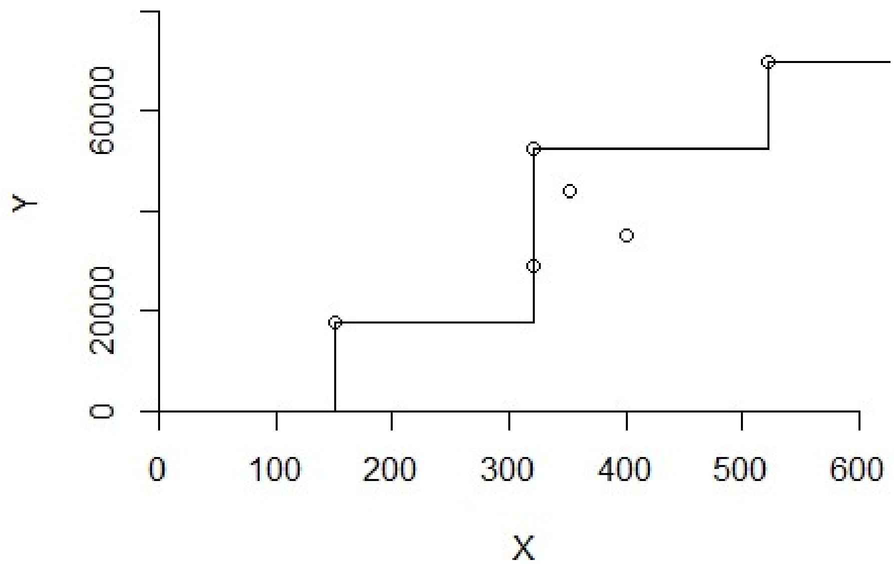
VRS Plot

```
dea.plot(x,y,RTS="vrs")
```



FDH Plot

```
dea.plot(x,y,RTS="fdh")
```



FRH Plot

```
dea.plot(x,y,RTS="add")
```