Quantitative Management Assignment#8

Rakhee Moolchandani

11/01/2020

This notebook contains the code for the Assignment 8.

Using Benchmarking Libraries for DEA

We will now run DEA analysis using the benchmarking library. First, install the library, if you don't have it already.

```
#install.packages("Benchmarking")
#install.packages("readxl")
library(Benchmarking)
library(readxl)
```

Now, we read our input data. We will read the data from an excel file. Remember our problem had 6 DMUs with two inputs and two outputs.

Inputs: Staffing Labor, Cost of Supplies

Outputs: No of patient-days reimbursed by third party, No of patient-days reimbursed privately

```
#Read the data from excel file
data <- read_excel("DEA.xlsx")
#See the data
data
```

```
## # A tibble: 6 x 5
##   DMU    `Staff Hours per~ `Supplies per Da~ `Reimbursed Pati~ `Privately Paid ~
##   <chr>              <dbl>             <dbl>             <dbl>             <dbl>
## 1 Facil~               150               0.2             14000              3500
## 2 Facil~               400               0.7             14000             21000
## 3 Facil~               320               1.2             42000             10500
## 4 Facil~               520               2               28000             42000
## 5 Facil~               350               1.2             19000             25000
## 6 Facil~               320               0.7             14000             15000
```

```
#Facility1 to Facility 6 are the DMUs
namesDMU <- data[1]
namesDMU
```

```
## # A tibble: 6 x 1
##   DMU
##   <chr>
```

```
## 1 Facility 1
## 2 Facility 2
## 3 Facility 3
## 4 Facility 4
## 5 Facility 5
## 6 Facility 6
```

```
#Lets see the Inputs
inputs <- data[c(2,3)]
inputs
```

```
## # A tibble: 6 x 2
##   `Staff Hours per Day` `Supplies per Day`
##                   <dbl>              <dbl>
## 1                   150                0.2
## 2                   400                0.7
## 3                   320                1.2
## 4                   520                2
## 5                   350                1.2
## 6                   320                0.7
```

```
#Now, see the outputs
outputs <- data[c(4,5)]
outputs
```

```
## # A tibble: 6 x 2
##   `Reimbursed Patient-Days` `Privately Paid Patient-Days`
##                       <dbl>                         <dbl>
## 1                     14000                          3500
## 2                     14000                         21000
## 3                     42000                         10500
## 4                     28000                         42000
## 5                     19000                         25000
## 6                     14000                         15000
```

```
#Create the input matrix
x <- matrix(c(data$`Staff Hours per Day`,data$`Supplies per Day`),ncol = 2)
#Lets see the input matrix
x
```

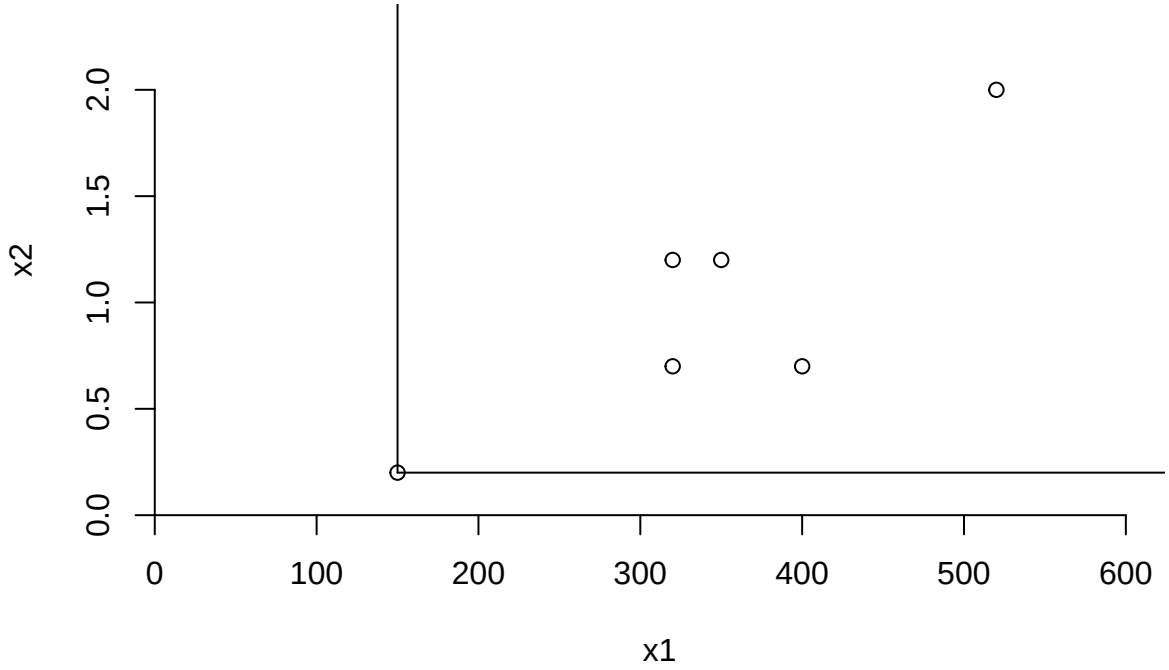
```
##      [,1] [,2]
## [1,]  150  0.2
## [2,]  400  0.7
## [3,]  320  1.2
## [4,]  520  2.0
## [5,]  350  1.2
## [6,]  320  0.7
```

```
#Create the output matrix
y <- matrix(c(data$`Reimbursed Patient-Days`,data$`Privately Paid Patient-Days`),ncol = 2)
#Lets see the output matrix
y
```

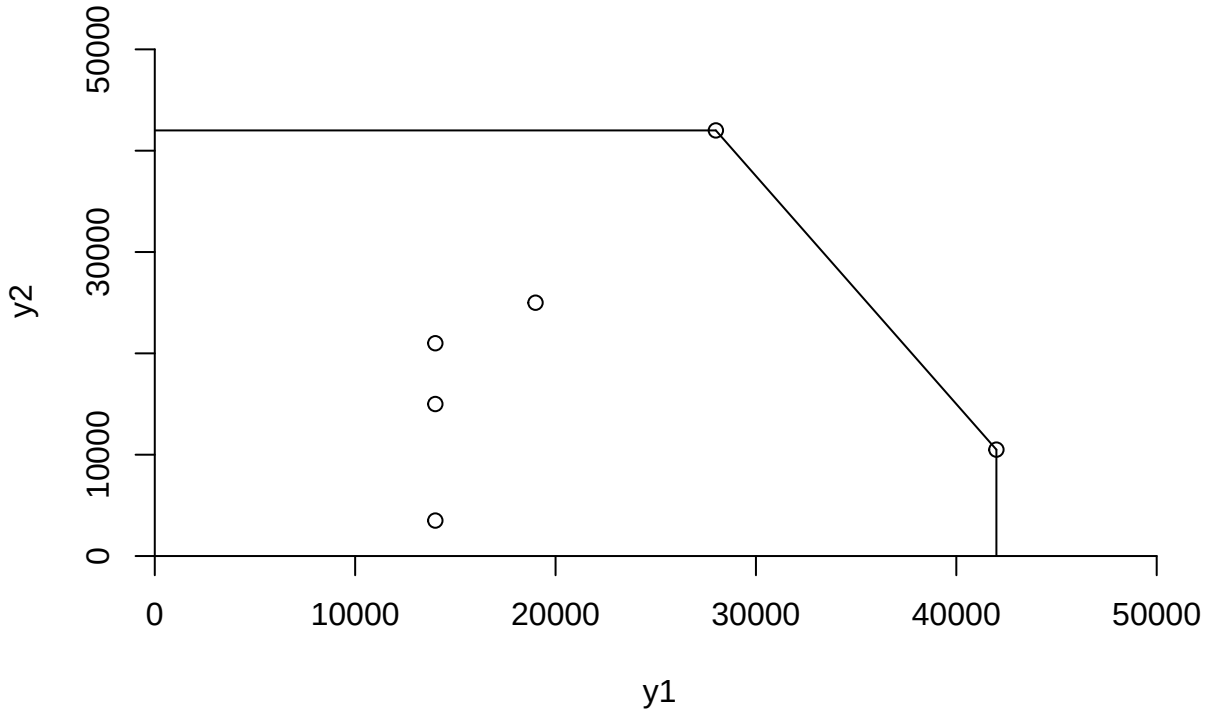
```
##       [,1]  [,2]
## [1,] 14000  3500
## [2,] 14000 21000
## [3,] 42000 10500
```

```
## [4,] 28000 42000
## [5,] 19000 25000
## [6,] 14000 15000
```

```
#plot the graph for Inputs
dea.plot.isoquant(x[,1],x[,2])
```




```
#plot the graph for Outputs
dea.plot.transform(y[,1],y[,2])
```



We now run the DEA analysis for different assumptions:

We use the option of FDH, Free disposability hull, no convexity assumption

```
#DEA input or output efficiency measures, peers, lambdas and slacks
e1 <- dea(x,y,RTS = "FDH")
#Show the Efficiency
e1
```

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

```
#Show the list of objects calculated
str(e1)
```

```
## List of 7
##  $ eff        : num [1:6] 1 1 1 1 1 1
##  $ objval     : num [1:6] 1 1 1 1 1 1
##  $ peers      : int [1:6] 1 2 3 4 5 6
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0 0 0 1 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ RTS        : chr "fdh"
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  - attr(*, "class")= chr "Farrell"
```

```
#Show the peers
peers(e1)
```

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

```
#Show the lambda
lambda(e1)
```

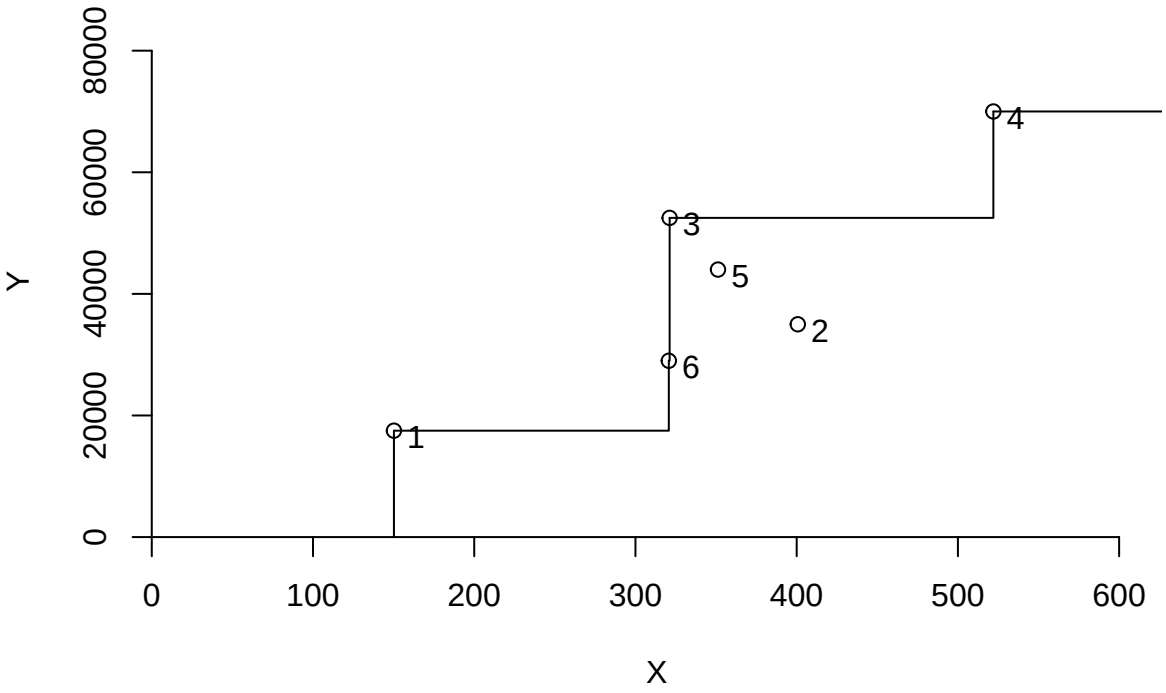
```
##      L1 L2 L3 L4 L5 L6
## [1,]  1  0  0  0  0  0
## [2,]  0  1  0  0  0  0
## [3,]  0  0  1  0  0  0
## [4,]  0  0  0  1  0  0
## [5,]  0  0  0  0  1  0
## [6,]  0  0  0  0  0  1
```

```
#Add the Efficiency, Peers & Lambda values in the table
report1 <- cbind(data, e1$eff, e1$lambda, e1$peers)
#Name the columns of the table
colnames(report1)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Sow the table
report1
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                 150              0.2                    14000
```

```
## 2 Facility 2                 400             0.7                  14000
## 3 Facility 3                 320             1.2                  42000
## 4 Facility 4                 520             2.0                  28000
## 5 Facility 5                 350             1.2                  19000
## 6 Facility 6                 320             0.7                  14000
##   Privately Paid Patient-Days Efficiency Lambda1 Lambda2 Lambda3 Lambda4
## 1                       3500          1       1       0       0       0
## 2                      21000          1       0       1       0       0
## 3                      10500          1       0       0       1       0
## 4                      42000          1       0       0       0       1
## 5                      25000          1       0       0       0       0
## 6                      15000          1       0       0       0       0
##   Lambda5 Lambda6 Peers
## 1       0       0     1
## 2       0       0     2
## 3       0       0     3
## 4       0       0     4
## 5       1       0     5
## 6       0       1     6
```

```
#plot the graph for FDH Assumption
dea.plot(x,y,RTS="FDH",txt = rownames(report1))
```



The results indicate that DMUs 1, 2, 3 , 4, 5 and 6 all are efficient.

We use the option of CRS, Constant Return to Scale, convexity and free disposability

```
#DEA input or output efficiency measures, peers, lambdas and slacks
e2 <- dea(x,y,RTS = "CRS")
#Show the Efficiency
e2
```

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

```
#Show the list of objects calculated
str(e2)
```

```
## List of 12
##  $ eff        : num [1:6] 1 1 1 1 0.977 ...
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0.2 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ objval     : num [1:6] 1 1 1 1 0.977 ...
##  $ RTS        : chr "crs"
##  $ primal     : NULL
##  $ dual       : NULL
##  $ ux         : NULL
##  $ vy         : NULL
##  $ gamma      :function (x)
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  $ param      : NULL
##  - attr(*, "class")= chr "Farrell"
```

```
#Show the peers
peers(e2)
```

```
##      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
```

```
#Show the lambda
lambda(e2)
```

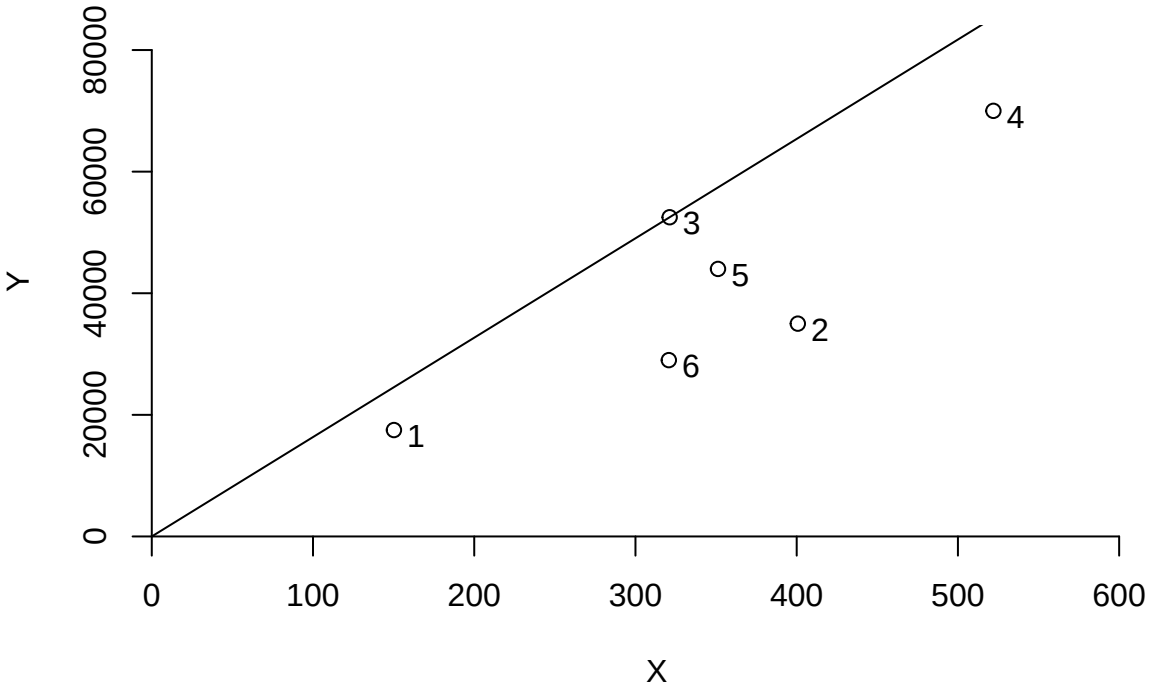
```
##             L1         L2 L3        L4
## [1,] 1.0000000 0.00000000  0 0.0000000
## [2,] 0.0000000 1.00000000  0 0.0000000
## [3,] 0.0000000 0.00000000  1 0.0000000
## [4,] 0.0000000 0.00000000  0 1.0000000
## [5,] 0.2000000 0.08048142  0 0.5383307
## [6,] 0.3428571 0.39499264  0 0.1310751
```

```
#Add the Efficiency & Lambda values in the table
report2 <- cbind(data, e2$eff, e2$lambda)
#Name the columns of the table
colnames(report2)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Sow the table
report2
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                 150              0.2                   14000
## 2 Facility 2                 400              0.7                   14000
## 3 Facility 3                 320              1.2                   42000
## 4 Facility 4                 520              2.0                   28000
```

```
## 5 Facility 5                 350             1.2                   19000
## 6 Facility 6                 320             0.7                   14000
##   Privately Paid Patient-Days Efficiency   Lambda1    Lambda2 Lambda3   Lambda4
## 1                        3500  1.0000000 1.0000000 0.00000000       0 0.0000000
## 2                       21000  1.0000000 0.0000000 1.00000000       0 0.0000000
## 3                       10500  1.0000000 0.0000000 0.00000000       1 0.0000000
## 4                       42000  1.0000000 0.0000000 0.00000000       0 1.0000000
## 5                       25000  0.9774987 0.2000000 0.08048142       0 0.5383307
## 6                       15000  0.8674521 0.3428571 0.39499264       0 0.1310751
##   Lambda5 Lambda6
## 1       0       0
## 2       0       0
## 3       0       0
## 4       0       0
## 5       0       0
## 6       0       0
```

```r
#plot the graph for CRS Assumption
dea.plot(x,y,RTS="CRS",txt = rownames(report2))
```



The results indicate that DMUs 1, 2, 3 and 4 are efficient. DMU(5) is only 97.7% efficient, and DMU(6) is 86.7% efficient.

We use the option of VRS, Variable returns to scale, convexity and free disposability

```r
#DEA input or output efficiency measures, peers, lambdas and slacks
e3 <- dea(x,y,RTS = "VRS")
#Show the Efficiency
e3
```

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

```r
#Show the list of objects calculated
str(e3)
```

```
## List of 12
##  $ eff        : num [1:6] 1 1 1 1 1 ...
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ objval     : num [1:6] 1 1 1 1 1 ...
##  $ RTS        : chr "vrs"
##  $ primal     : NULL
##  $ dual       : NULL
##  $ ux         : NULL
##  $ vy         : NULL
##  $ gamma      :function (x)
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  $ param      : NULL
##  - attr(*, "class")= chr "Farrell"
```

```r
#Show the peers
peers(e3)
```

```
##      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
```

```r
#Show the lambda
lambda(e3)
```

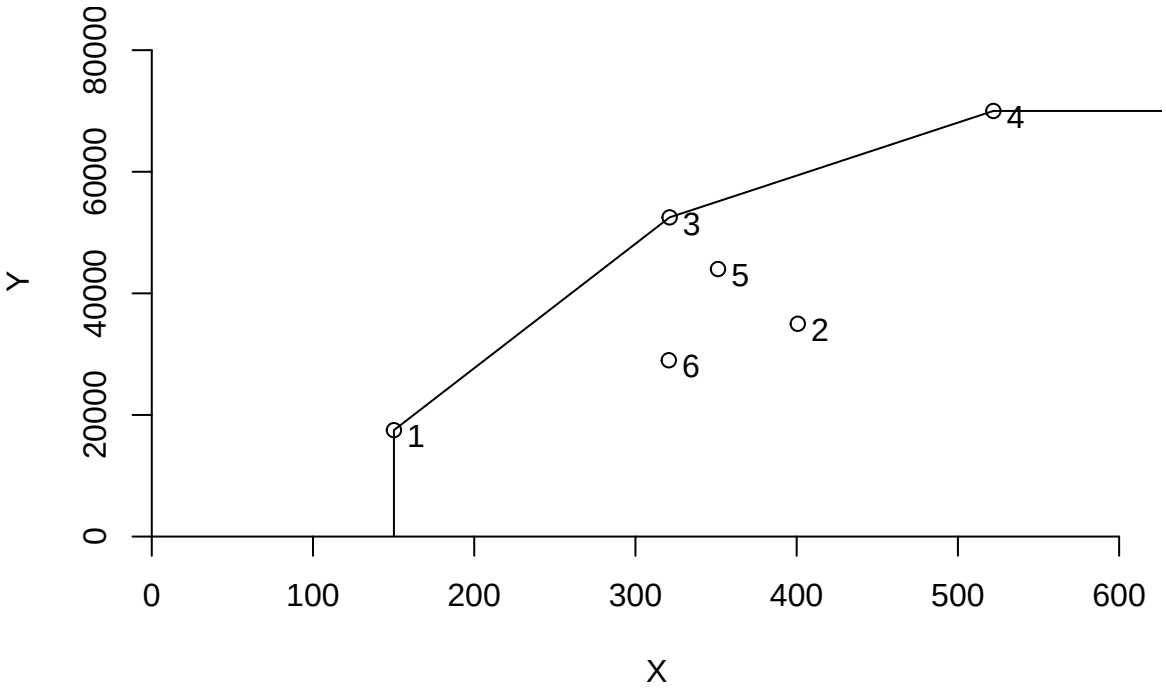
```
##             L1        L2 L3 L4        L5
## [1,] 1.0000000 0.0000000  0  0 0.0000000
## [2,] 0.0000000 1.0000000  0  0 0.0000000
## [3,] 0.0000000 0.0000000  1  0 0.0000000
## [4,] 0.0000000 0.0000000  0  1 0.0000000
## [5,] 0.0000000 0.0000000  0  0 1.0000000
## [6,] 0.4014399 0.3422606  0  0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

```r
#Add the Efficiency & Lambda values in the table
report3 <- cbind(data, e3$eff, e3$lambda)
#Name the columns of the table
colnames(report3)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Show the table
report3
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                 150              0.2                    14000
## 2 Facility 2                 400              0.7                    14000
## 3 Facility 3                 320              1.2                    42000
## 4 Facility 4                 520              2.0                    28000
## 5 Facility 5                 350              1.2                    19000
```

```
## 6 Facility 6                 320             0.7                    14000
##   Privately Paid Patient-Days Efficiency   Lambda1   Lambda2 Lambda3 Lambda4
## 1                        3500  1.0000000 1.0000000 0.0000000       0       0
## 2                       21000  1.0000000 0.0000000 1.0000000       0       0
## 3                       10500  1.0000000 0.0000000 0.0000000       1       0
## 4                       42000  1.0000000 0.0000000 0.0000000       0       1
## 5                       25000  1.0000000 0.0000000 0.0000000       0       0
## 6                       15000  0.8963283 0.4014399 0.3422606       0       0
##     Lambda5 Lambda6
## 1 0.0000000       0
## 2 0.0000000       0
## 3 0.0000000       0
## 4 0.0000000       0
## 5 1.0000000       0
## 6 0.2562995       0
```

```
#plot the graph for VRS Assumption
dea.plot(x,y,RTS="VRS",txt = rownames(report3))
```



The results indicate that DMUs 1, 2, 3, 4 and 5 are efficient. DMU(6) is only 89.6% efficient.

We use the option of IRS, Increasing returns to scale, convexity and free disposability

```
#DEA input or output efficiency measures, peers, lambdas and slacks
e4 <- dea(x,y,RTS = "IRS")
#Show the Efficiency
e4
```

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

```r
#Show the list of objects calculated
str(e4)
```

```
## List of 12
##  $ eff        : num [1:6] 1 1 1 1 1 ...
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ objval     : num [1:6] 1 1 1 1 1 ...
##  $ RTS        : chr "irs"
##  $ primal     : NULL
##  $ dual       : NULL
##  $ ux         : NULL
##  $ vy         : NULL
##  $ gamma      :function (x)
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  $ param      : NULL
##  - attr(*, "class")= chr "Farrell"
```

```r
#Show the peers
peers(e4)
```

```
##      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
```

```r
#Show the lambda
lambda(e4)
```

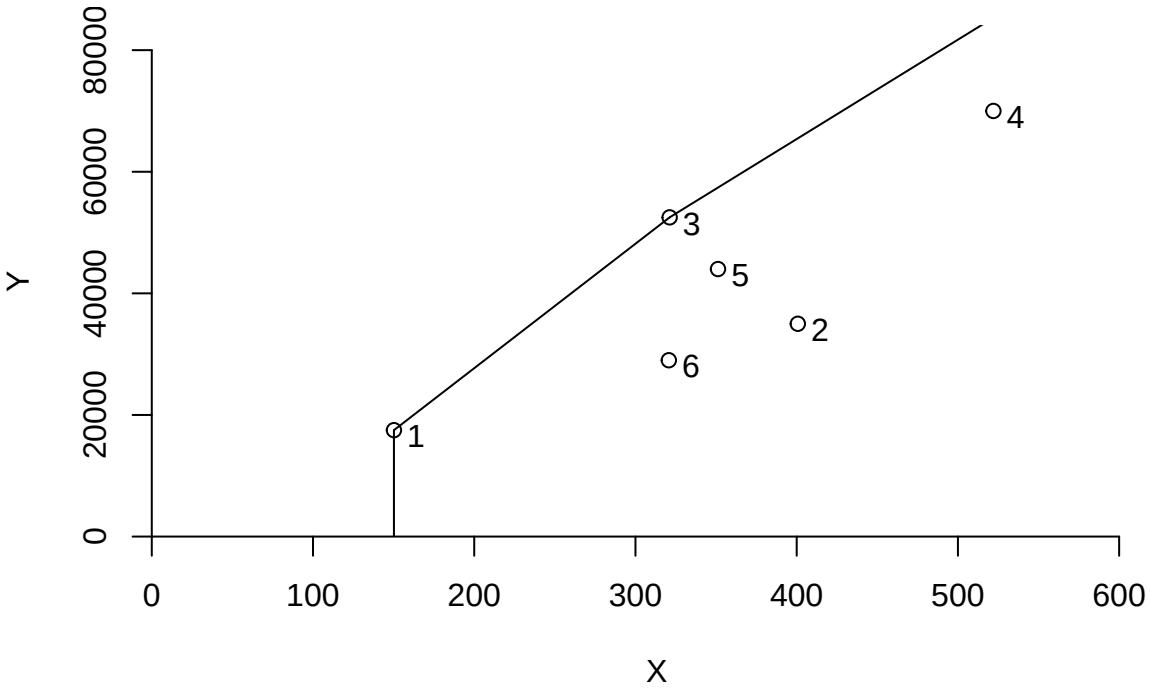
```
##             L1        L2 L3 L4        L5
## [1,] 1.0000000 0.0000000  0  0 0.0000000
## [2,] 0.0000000 1.0000000  0  0 0.0000000
## [3,] 0.0000000 0.0000000  1  0 0.0000000
## [4,] 0.0000000 0.0000000  0  1 0.0000000
## [5,] 0.0000000 0.0000000  0  0 1.0000000
## [6,] 0.4014399 0.3422606  0  0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

```r
#Add the Efficiency & Lambda values in the table
report4 <- cbind(data, e4$eff, e4$lambda)
#Name the columns of the table
colnames(report4)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Show the table
report4
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                 150              0.2                    14000
## 2 Facility 2                 400              0.7                    14000
## 3 Facility 3                 320              1.2                    42000
## 4 Facility 4                 520              2.0                    28000
## 5 Facility 5                 350              1.2                    19000
```

```
## 6 Facility 6                 320             0.7                    14000
##   Privately Paid Patient-Days Efficiency   Lambda1   Lambda2 Lambda3 Lambda4
## 1                        3500  1.0000000 1.0000000 0.0000000       0       0
## 2                       21000  1.0000000 0.0000000 1.0000000       0       0
## 3                       10500  1.0000000 0.0000000 0.0000000       1       0
## 4                       42000  1.0000000 0.0000000 0.0000000       0       1
## 5                       25000  1.0000000 0.0000000 0.0000000       0       0
## 6                       15000  0.8963283 0.4014399 0.3422606       0       0
##     Lambda5 Lambda6
## 1 0.0000000       0
## 2 0.0000000       0
## 3 0.0000000       0
## 4 0.0000000       0
## 5 1.0000000       0
## 6 0.2562995       0
```

```
#plot the graph for IRS Assumption
dea.plot(x,y,RTS="IRS",txt = rownames(report4))
```



The results indicate that DMUs 1, 2, 3, 4 and 5 are efficient. DMU(6) is only 89.6% efficient.

We use the option of DRS, Decreasing returns to scale, convexity, down-scaling and free disposability

```
#DEA input or output efficiency measures, peers, lambdas and slacks
e5 <- dea(x,y,RTS = "DRS")
#Show the Efficiency
e5
```

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

```r
#Show the list of objects calculated
str(e5)
```

```
## List of 12
##  $ eff        : num [1:6] 1 1 1 1 0.977 ...
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0.2 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ objval     : num [1:6] 1 1 1 1 0.977 ...
##  $ RTS        : chr "drs"
##  $ primal     : NULL
##  $ dual       : NULL
##  $ ux         : NULL
##  $ vy         : NULL
##  $ gamma      :function (x)
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  $ param      : NULL
##  - attr(*, "class")= chr "Farrell"
```

```r
#Show the peers
peers(e5)
```

```
##      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
#Show the lambda
lambda(e5)
```

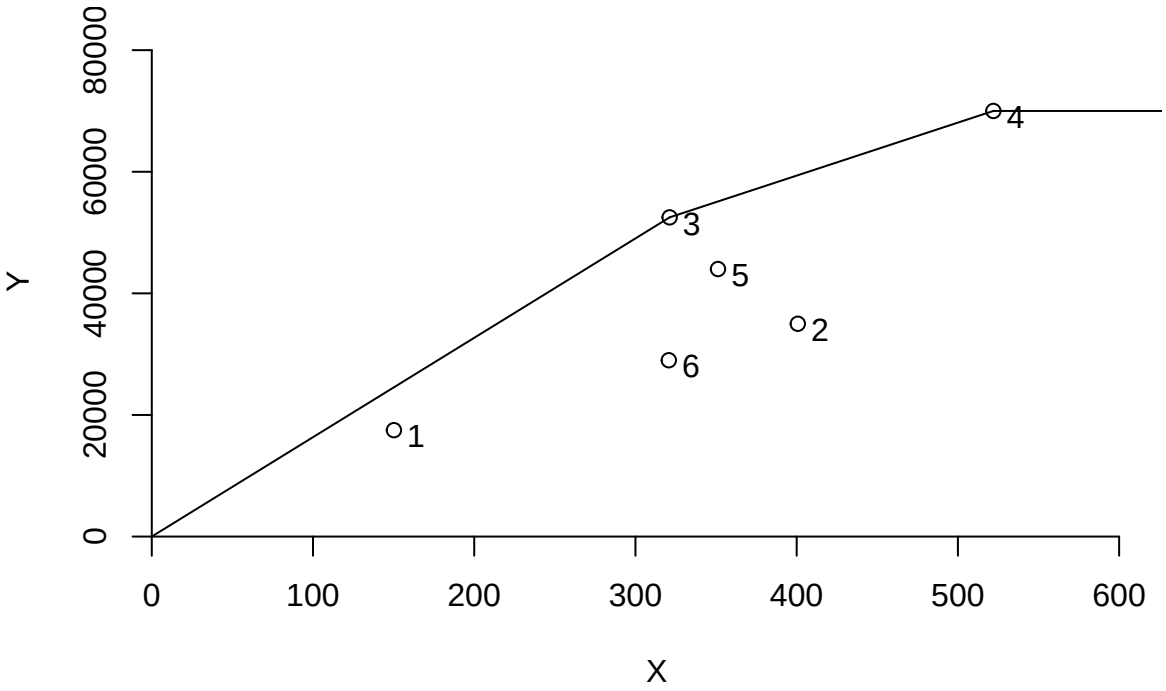
```
##             L1         L2 L3        L4
## [1,] 1.0000000 0.00000000  0 0.0000000
## [2,] 0.0000000 1.00000000  0 0.0000000
## [3,] 0.0000000 0.00000000  1 0.0000000
## [4,] 0.0000000 0.00000000  0 1.0000000
## [5,] 0.2000000 0.08048142  0 0.5383307
## [6,] 0.3428571 0.39499264  0 0.1310751
```

```r
#Add the Efficiency, Peers & Lambda values in the table
report5 <- cbind(data, e5$eff, e5$lambda)
#Name the columns of the table
colnames(report5)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Show the table
report5
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                 150              0.2                    14000
## 2 Facility 2                 400              0.7                    14000
## 3 Facility 3                 320              1.2                    42000
## 4 Facility 4                 520              2.0                    28000
## 5 Facility 5                 350              1.2                    19000
```

```
## 6 Facility 6                 320              0.7                    14000
##   Privately Paid Patient-Days Efficiency   Lambda1    Lambda2 Lambda3   Lambda4
## 1                        3500  1.0000000 1.0000000 0.00000000       0 0.0000000
## 2                       21000  1.0000000 0.0000000 1.00000000       0 0.0000000
## 3                       10500  1.0000000 0.0000000 0.00000000       1 0.0000000
## 4                       42000  1.0000000 0.0000000 0.00000000       0 1.0000000
## 5                       25000  0.9774987 0.2000000 0.08048142       0 0.5383307
## 6                       15000  0.8674521 0.3428571 0.39499264       0 0.1310751
##   Lambda5 Lambda6
## 1       0       0
## 2       0       0
## 3       0       0
## 4       0       0
## 5       0       0
## 6       0       0
```

```
#plot the graph for IRS Assumption
dea.plot(x,y,RTS="DRS",txt = rownames(report5))
```



The results indicate that DMUs 1, 2, 3 and 4 are efficient. DMU(5) is only 97.7% efficient, and DMU(6) is 86.7% efficient.

We use the option of FRH, Additivity (scaling up and down, but only with integers), and free disposability

```
#DEA input or output efficiency measures, peers, lambdas and slacks
e6 <- dea(x,y,RTS = "ADD")
#Show the Efficiency
e6
```

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

```r
#Show the list of objects calculated
str(e6)
```

```
## List of 12
##  $ eff        : num [1:6] 1 1 1 1 1 1
##  $ lambda     : num [1:6, 1:6] 1 0 0 0 0 0 0 1 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "L1" "L2" "L3" "L4" ...
##  $ objval     : num [1:6] 1 1 1 1 1 1
##  $ RTS        : chr "add"
##  $ primal     : NULL
##  $ dual       : NULL
##  $ ux         : NULL
##  $ vy         : NULL
##  $ gamma      :function (x)
##  $ ORIENTATION: chr "in"
##  $ TRANSPOSE  : logi FALSE
##  $ param      : NULL
##  - attr(*, "class")= chr "Farrell"
```

```r
#Show the peers
peers(e6)
```

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

```r
#Show the lambda
lambda(e6)
```

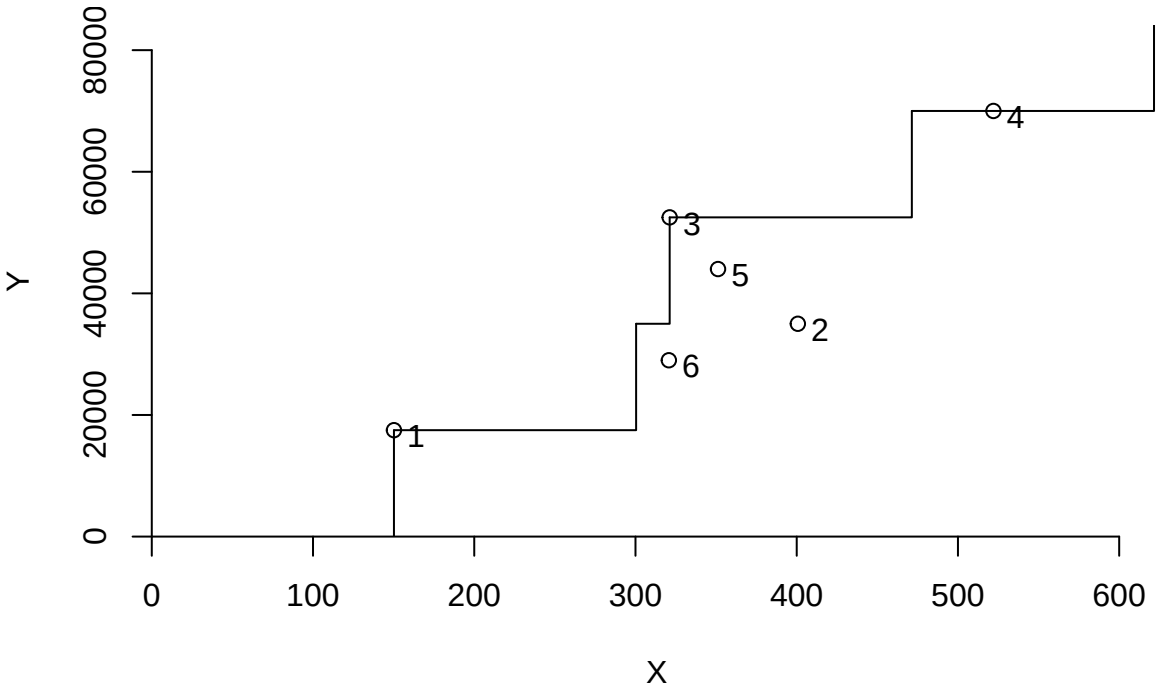
```
##      L1 L2 L3 L4 L5 L6
## [1,]  1  0  0  0  0  0
## [2,]  0  1  0  0  0  0
## [3,]  0  0  1  0  0  0
## [4,]  0  0  0  1  0  0
## [5,]  0  0  0  0  1  0
## [6,]  0  0  0  0  0  1
```

```r
#Add the Efficiency, Peers & Lambda values in the table
report6 <- cbind(data, e6$eff, e6$lambda)
#Name the columns of the table
colnames(report6)<- c(names(namesDMU),names(inputs), names(outputs),'Efficiency','Lambda1','Lambda2','La
#Sow the table
report6
```

```
##          DMU Staff Hours per Day Supplies per Day Reimbursed Patient-Days
## 1 Facility 1                150              0.2                    14000
## 2 Facility 2                400              0.7                    14000
## 3 Facility 3                320              1.2                    42000
## 4 Facility 4                520              2.0                    28000
## 5 Facility 5                350              1.2                    19000
```

```
## 6 Facility 6                  320             0.7                    14000
##   Privately Paid Patient-Days Efficiency Lambda1 Lambda2 Lambda3 Lambda4
## 1                        3500          1       1       0       0       0
## 2                       21000          1       0       1       0       0
## 3                       10500          1       0       0       1       0
## 4                       42000          1       0       0       0       1
## 5                       25000          1       0       0       0       0
## 6                       15000          1       0       0       0       0
##   Lambda5 Lambda6
## 1       0       0
## 2       0       0
## 3       0       0
## 4       0       0
## 5       1       0
## 6       0       1
```

```
#plot the graph for FDH Assumption
dea.plot(x,y,RTS="ADD",txt = rownames(report5))
```



The results indicate that DMUs 1, 2, 3 , 4, 5 and 6 all are efficient.

Compare and Contrast the above Results

Lets compare the efficiency of all the DMUs for all the assumptions

```
#Add the Efficiency of all the DMUs for all the Assumptions in a table
EfficiencyReport <- cbind(data[,1],e1$eff,e2$eff,e3$eff,e4$eff,e5$eff,e6$eff)
#Name the columns of the table
colnames(EfficiencyReport) <- c(names(namesDMU),'FDH Efficiency','CRS Efficiency','VRS Efficiency','IRS
#Show the Efficiency table
EfficiencyReport
```

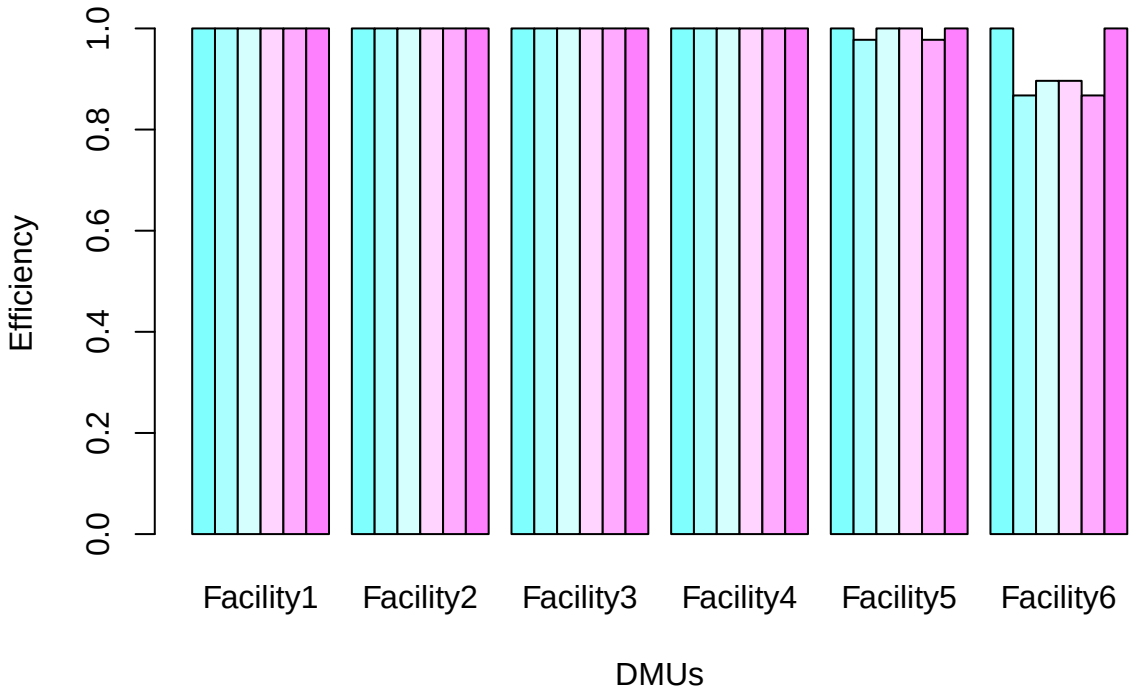
```
##          DMU FDH Efficiency CRS Efficiency VRS Efficiency IRS Efficiency
```

```
## 1 Facility 1              1      1.0000000      1.0000000      1.0000000
## 2 Facility 2              1      1.0000000      1.0000000      1.0000000
## 3 Facility 3              1      1.0000000      1.0000000      1.0000000
## 4 Facility 4              1      1.0000000      1.0000000      1.0000000
## 5 Facility 5              1      0.9774987      1.0000000      1.0000000
## 6 Facility 6              1      0.8674521      0.8963283      0.8963283
##   DRS Efficiency FRH Efficiency
## 1      1.0000000              1
## 2      1.0000000              1
## 3      1.0000000              1
## 4      1.0000000              1
## 5      0.9774987              1
## 6      0.8674521              1
```

```
#Let's compare the Efficiency of all the DMUs for all the assumptions using a plot
#Concatenate the Efficiency
spreadsheet <- cbind(e1$eff,e2$eff,e3$eff,e4$eff,e5$eff,e6$eff)
#Name the rows
rownames(spreadsheet) <- c("Facility1","Facility2","Facility3","Facility4","Facility5","Facility6")
#Name the columns
colnames(spreadsheet) <- c ("FDH","CRS","VRS","IRS","DRS","FRH")
#See the result
spreadsheet
```

```
##           FDH       CRS       VRS       IRS       DRS FRH
## Facility1   1 1.0000000 1.0000000 1.0000000 1.0000000   1
## Facility2   1 1.0000000 1.0000000 1.0000000 1.0000000   1
## Facility3   1 1.0000000 1.0000000 1.0000000 1.0000000   1
## Facility4   1 1.0000000 1.0000000 1.0000000 1.0000000   1
## Facility5   1 0.9774987 1.0000000 1.0000000 0.9774987   1
## Facility6   1 0.8674521 0.8963283 0.8963283 0.8674521   1
```

```
#plot the graph
barplot(t(spreadsheet),col=cm.colors(6),xlab = "DMUs", ylab="Efficiency",beside=TRUE)
```



1) Facility 1 is fully efficient for all the assumptions.

2) Facility 2 is fully efficient for all the assumptions.

3) Facility 3 is fully efficient for all the assumptions.

4) Facility 4 is fully efficient for all the assumptions.

5) Facility 5 is fully efficient for FDH, VRS, IRS and FRH assumptions. For assumptions DRS and CRS, it is 97.7% efficient.

6) Facility 6 is fully efficient for FDH and FRS assumptions. For CRS and DRS assumptions, it is 86.7% efficient. For IRS and VRS assumptions, it is 89.6% efficient.