QMM DEA Assignment

Varshini

2022-10-30

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

```
library("Benchmarking")
```

```
datadfvalues <- 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(datadfvalues) <- c("DMU", "Staff_Hours_Per_Day","Supplies_Per_Day","Reimbursed_Patient_Days","I

tabledf <- as.table(datadfvalues)
tabledf
```

```
##   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")

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

```r
DEAcrs
```

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

```r
peers(DEAcrs)
```

```
##      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(DEAcrs)
```

```
##             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 shown,Facility 1, 2, 3 and 4 are efficient.

2. And also, Facility 1, 2 and 4 are the peer members for Facility 5 and 6 which are also inefficient 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

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

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

```r
peers(DEAdrs)
```

```
##      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(DEAdrs)
```

```
##             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.As depicted, Facility 1, 2, 3 and 4 are efficient.

2. And,Facility 1, 2 and 4 are the peer members for Facility 5 and 6 which are inefficient 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 IRS

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

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

```
peers(DEAirs)
```

```
##      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(DEAirs)
```

```
##             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. As observed, Facility 1, 2, 3, 4 and 5 are efficient.

2.We can also see that Facility 1, 2 and 5 are the peer members for Facility 6 which is the only inefficient facility.

3. Facility 6 is 89.63 % efficient making 10.37 % as inefficient.

Calculating VRS

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

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

```
peers(DEAvrs)
```

```
##      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(DEAvrs)
```

```
##             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 get to see that Facility 1, 2, 3, 4 and 5 are efficient.

2. Facility 1, 2 and 5 are the peer members for Facility 6 which is an inefficient facility.

3. Facility 6 is 89.63 % efficient thereby making 10.37 % as inefficient.

Calculating FDH

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

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

```
peers(DEAfdh)
```

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

```
lambda(DEAfdh)
```

```
##      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
```

FDH Observations -

Each and every DMU is efficient,due to the principal which FDH technique follows which detects any level of efficiency.

Calculating FRH

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

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

```
peers(DEAfrh)
```

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

```
## [6,]     6
```

```
lambda(DEAfrh)
```

```
##      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


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

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", "Facility 1, 2, 3, 4 & 5", "Al

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 findings show that DMUs 1, 2, 3, and 4 are effective. Only 97.75% of DMU(5) and 86.7% of DMU(6) are effectively used. On the basis of our initial analysis, we discovered this. In addition, DMU(4peer)'s units are 1, 2, and 4, with respective weights of 0.2, 0.08, and 0.54. The peer units for DMU(6) are 1, 2, and 4, and their weights are 0.34, 0.4, and 0.13, respectively.

In essence, CRS enables us to determine whether any potential DMUs, in this example DMUs 1, 2, 3, and 4, may be scaled up or down.

DRS - Decreasing Returns to Scale

The findings show that DMUs 1, 2, 3, and 4 are effective. Only 97.75% of DMU(5) and 86.7% of DMU(6) are efficient. On the basis of our initial analysis, we discovered this. In addition, DMU(4peer)'s units are 1, 2, and 4, with respective weights of 0.2, 0.08, and 0.54. The peer units for DMU(6) are 1, 2, and 4, with respective weights of 0.34, 0.4, and 0.13.

By looking at the inefficient DMUs, in this case DMUs 5 and 6, we may determine whether there are any alternative DMUs where we can scale the processes. As the base original scale, the CRS values can also be used to obtain this information.

IRS - Increasing Returns to Scale

The outcomes show that DMUs 1, 2, 3, 4, and 5 are productive. Only 89.63% of the DMU(6) is effective. On the basis of our initial analysis, we discovered this. Additionally, the peer units for DMU(6) are 1, 2, and 5, with corresponding relative weights of 0.4, 0.34, and 0.26.

By examining the efficiency scores, it enables any firm to determine whether it can arbitrarily expand its operations.

VRS - Variable Returns to Scale

The outcomes show that DMUs 1, 2, 3, 4, and 5 are productive. Only 89.63% of the DMU(6) is effective. On the basis of our initial analysis, we discovered this. Additionally, the peer units for DMU(6) are 1, 2, and 5, with corresponding relative weights of 0.4, 0.34, and 0.26.

Understanding the scale of operations with adjustments in the input and output factor—either by increasing or decreasing or by using both—is made possible by varying or variable returns to scale.

FRH - Free Replicability Hull

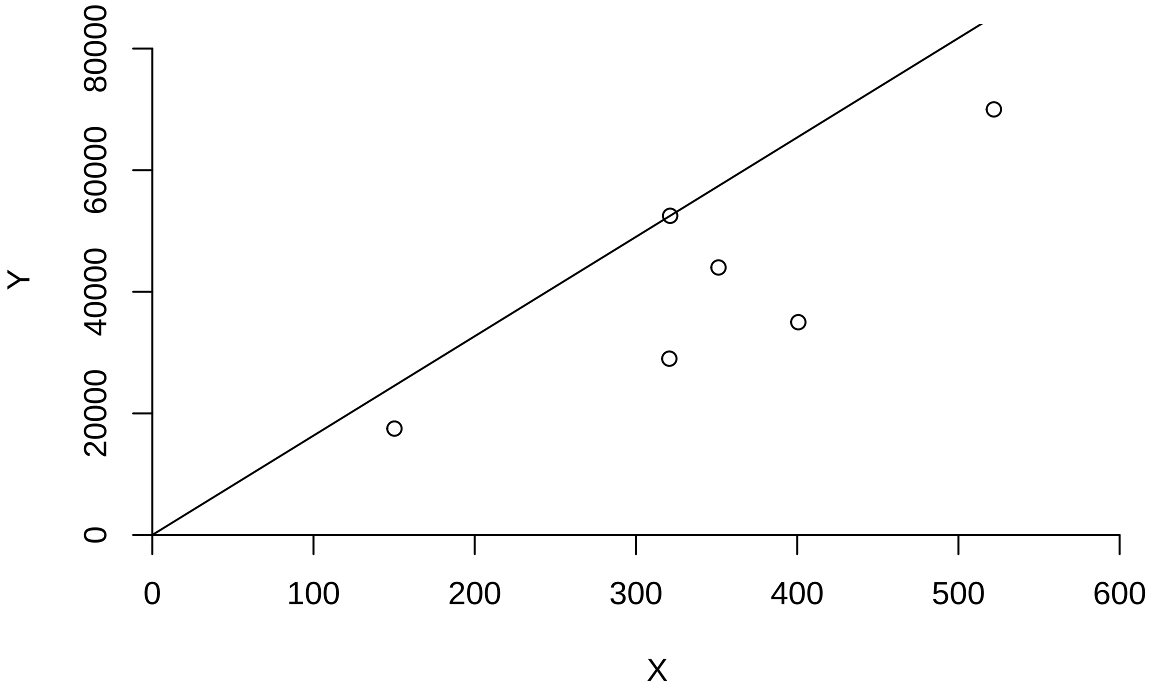
All DMUs are effective, according to the FRH data. This is primarily because the convexity assumption is not made, and because this technique enables the scale to record even the tiniest efficiency level that is not subject to replication or disposal.

Conclusion

It is important to keep in mind that DEA is a very helpful tool for any company in deciding which of the Decision Making Units should be maximized so that there would be an increase, decrease, or any other kind of variations to the output by feeding input into it. Additionally, a company can choose which of the RTS it wants to employ, i.e. Returns to Scale based on their specifications; each of these scales is significant in its own right.

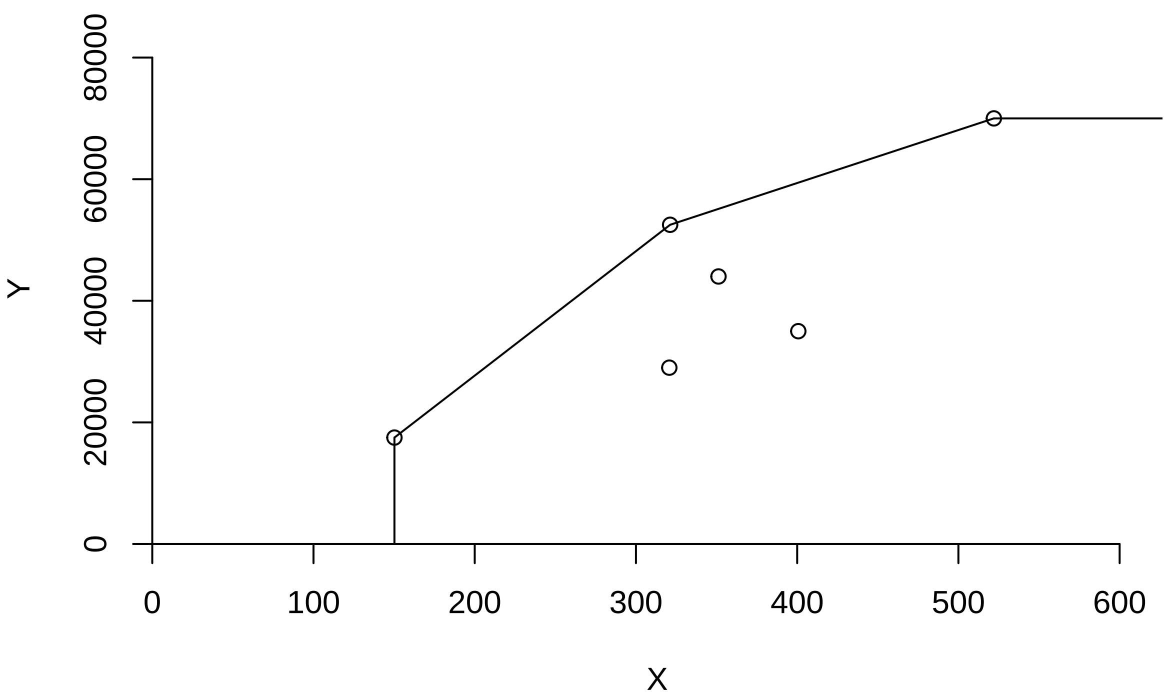
CRS Plot

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



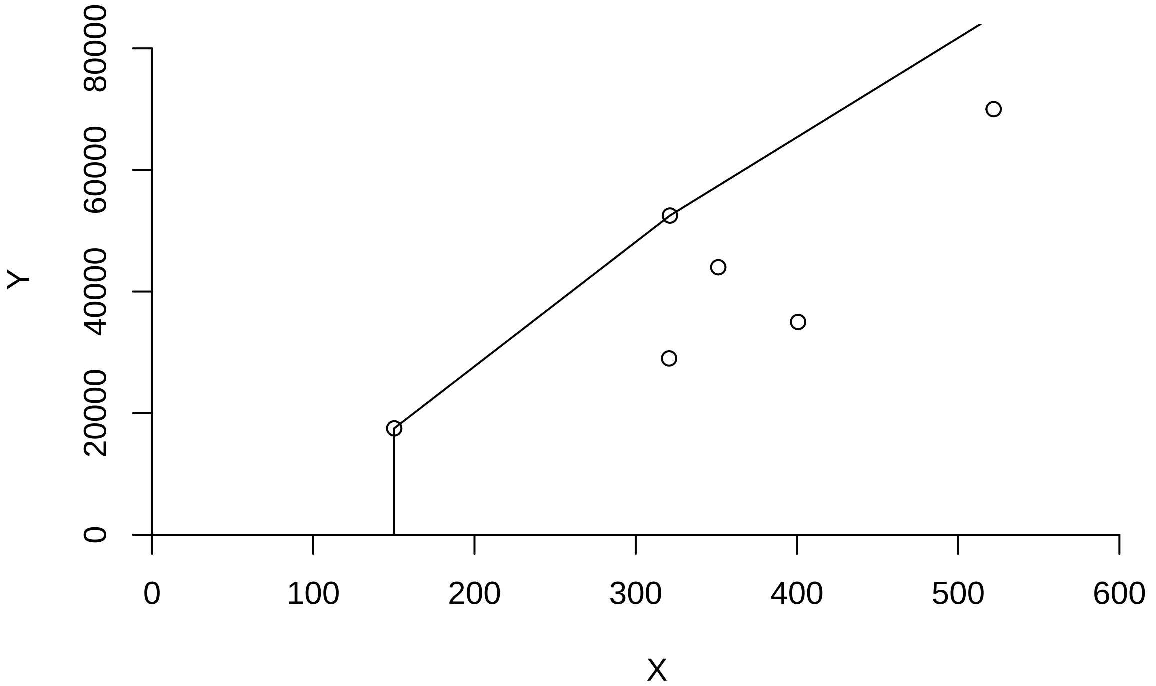
DRS Plot

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



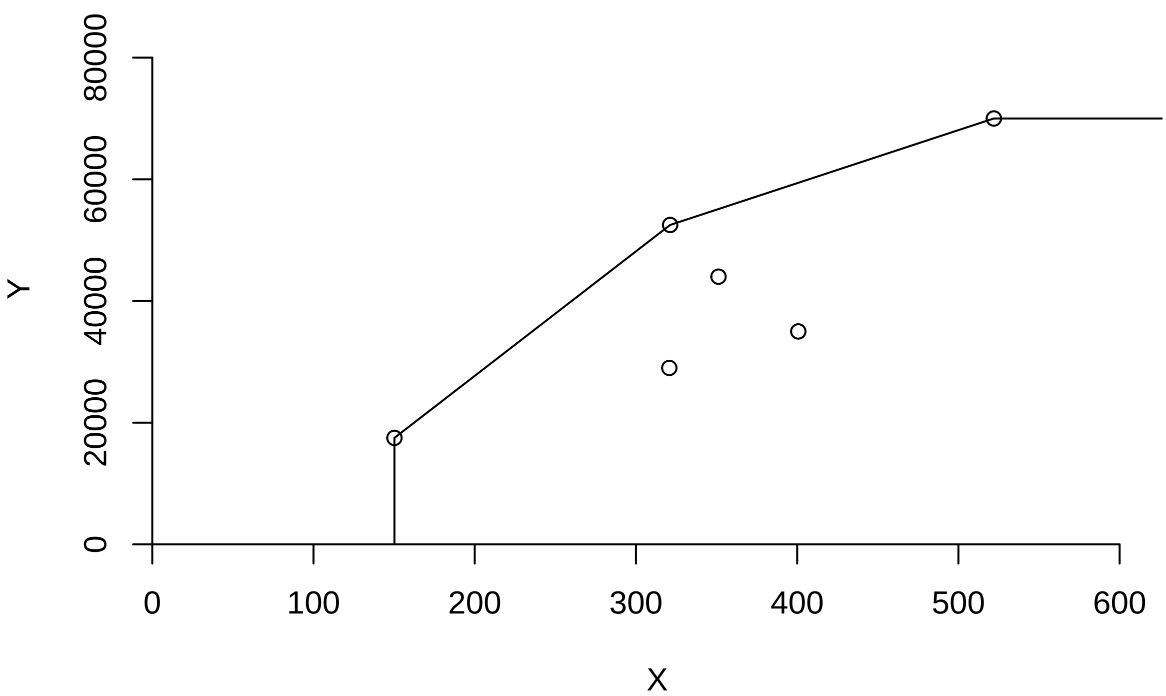
IRS Plot

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



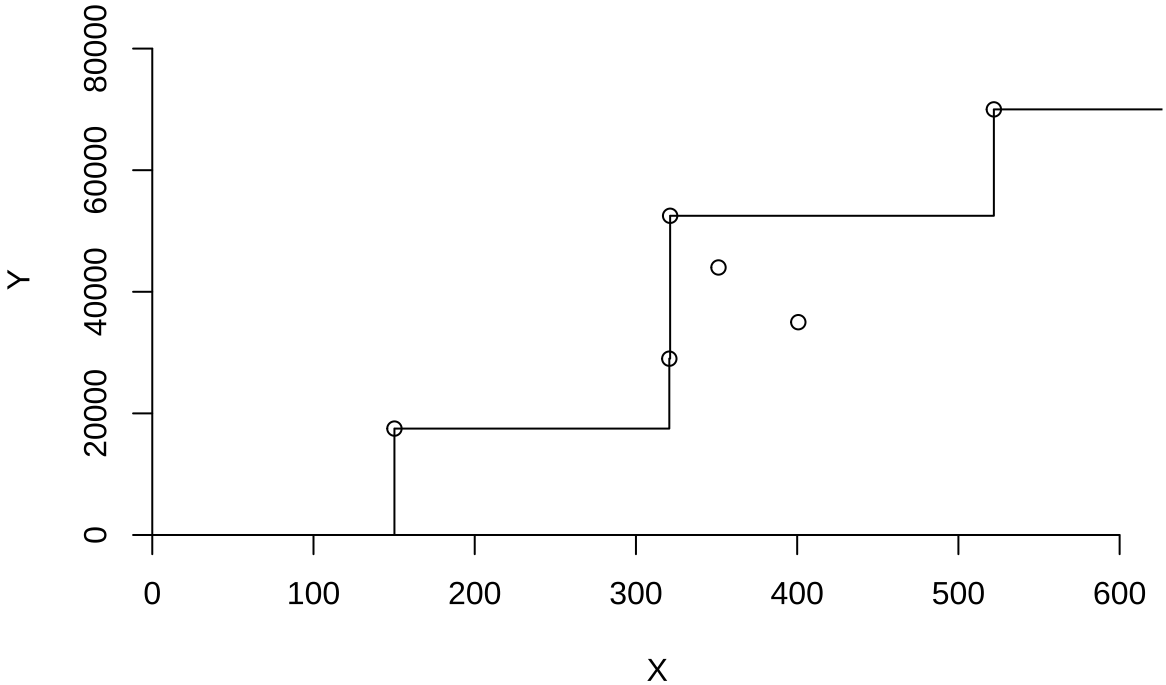
VRS Plot

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



FDH Plot

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



FRH Plot

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