Assignment_Module 8_DEA

Venkata Naga Siddartha Gutha

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

```
library("Benchmarking")

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

table <- as.table(table_values)
table_values
```

```
##      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"              "28000"
## [5,] "Facility 5" "350"               "1.2"            "19000"
## [6,] "Facility 6" "320"               "0.7"            "14000"
##      Privately_Paid_Patient_Days
## [1,] "3500"
## [2,] "21000"
## [3,] "10500"
## [4,] "42000"
## [5,] "25000"
## [6,] "15000"
```

calculating Constant that Returns to Scale (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(x) <- c("Staff_Hours_Per_Day","Supplies_Per_Day")
colnames(y) <- c("Reimbursed_Patient_Days","Privately_Paid_Patient_Days")


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

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

```
peers(D_E_A_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(D_E_A_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 Observations:-

a. From the above output it can be seen that Facility 1, Facility 2, Facility 3 and Facility 4 are efficient.

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

c. Facility 5 is 97.75 % efficient leaving 2.25 % as inefficient

d. Facility 6 is 86.75 % efficient leaving 13.25 % as inefficient.

Calculating the Decreasing that returns to Scale (DRS)

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

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

```
peers(D_E_A_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(D_E_A_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. Facility 1, Facility 2, Facility 3 and Facility 4 are efficient.
2. Facility 1, Facility 2 and Facility 4 are the peer members for Facility 5 and Facility 6 which are inefficient facilities.
3. Facility 5 is 97.75 % efficient leaving 2.25 % as inefficient
4. Facility 6 is 86.75 % efficient leaving 13.25 % as inefficient.

Calculating Increasing Returns to Scale (IRS)

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

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

```
peers(D_E_A_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(D_E_A_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. Facility 1, Facility 2, Facility 3, Facility 4 and Facility 5 are efficient.
2. Facility 1, Facility 2 and Facility 5 are the peer members for Facility 6 which is inefficient facility.
3. Facility 6 is 89.63 % efficient leaving 10.37 % as inefficient.

Calculating Variable Returns to Scale (VRS)

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

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

```
peers(D_E_A_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(D_E_A_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. Facility 1, Facility 2, Facility 3, Facility 4 and Facility 5 are efficient.
2. Facility 1, Facility 2 and Facility 5 are the peer members for Facility 6 which is the only inefficient facility.
3. Facility 6 is 89.63 % efficient leaving 10.37 % as inefficient.

Calculating the Free Disposability Hull (FDH)

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

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

```
peers(D_E_A_fdh)
```

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

```
lambda(D_E_A_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
```

FDH Observations

All the DMUs are efficient this is due to the principal which FDH technique follows.

Calculating Free Replicability Hull (FRH)

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

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

```
peers(D_E_A_frh)
```

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

```
lambda(D_E_A_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
```

FRH Observations

All the DMUs are efficient. It follows the no convexity assumption it ensures that the o/p is free from disposal and replication.

Summary of Results (Inefficient DMUs)

```
summary_Inefficient_DMUs <- 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(summary_Inefficient_DMUs) <- c("RTS","Count_Inefficient_DMUs",
                                        "Name_DMUs","%_Inefficiency",
                                        "Peers","Lambda")

as.table(summary_Inefficient_DMUs)
```

```
##   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, \n0.54 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 Results (Efficient DMUs)

```
summary_Inefficient_DMUs<- 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", "All DMUs", "All DMUs"), ncol = 2, byrow=F)

colnames(summary_Inefficient_DMUs) <- c("RTS", "Efficient_DMUs")

as.table(summary_Inefficient_DMUs)
```

```
##   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
```

Interpretation of the DEA Analysis

Constant Returns to Scale (CRS) is regarded as the original scale and is utilized by the majority of businesses.The Free Disposability and Free Replicability Hull (FDH & FRH), which makes no assumptions about convexity, is regarded as a non-parametric way to evaluate the effectiveness of DMUs.The dispersion scales known as Decreasing, Increasing and Varying Returns to Scale (DRS, IRS, and VRS) aid us in determining what to increase and what to decrease based on the deployment of information.

CRS - Constant Returns to Scale

DMUs 1, 2, 3 and 4 are efficient. DMU(5) is only 97.75% efficient, and DMU(6) is 86.7% efficient. the peer units for DMU(4) are 1, 2 and 4 with relative weights of 0.2, 0.08 and 0.54.The peer units OF DMU(6) are 1, 2 and 4 with weights of 0.34, 0.4 and 0.13 respectively.CRS helps us to know if any possible DMUs can be scaled up or down. Here DMUs 1, 2, 3 and 4 can be scaled up.

DRS - Decreasing Returns to Scale

DMUs 1, 2, 3 and 4 are efficient. DMU(5) is only 97.75% efficient and DMU(6) is 86.7% efficient. the peer units for DMU(4) are 1, 2 and 4 with relative weights of 0.2, 0.08 and 0.54. Similarly for DMU(6), the peer units are 1, 2 and 4 with weights of 0.34, 0.4 and 0.13 respectively. In this case DMU 5 & 6 can be scaled down.

IRS - Increasing Returns to Scale DMUs 1, 2, 3, 4 and 5 are efficient. DMU(6) is only 89.63% efficient. the peer units for DMU(6) are 1, 2 and 5 with relative weights of 0.4, 0.34 and 0.26 respectively. IRS helps in increasing the scale of operation by looking at the efficiency scores

VRS - Variable Returns to Scale

DMUs 1, 2, 3, 4 and 5 are efficient. DMU(6) is only 89.63% efficient. the peer units for DMU(6) are 1, 2 and 5 with relative weights of 0.4, 0.34 and 0.26 respectively. VRS provides idea on increasing or decreasing the scale of operations with variations towards the input and output factor.

FDH - Free Disposability Hull

The results indicate that all the DMUs are efficient. This is because there is no convexity assumption

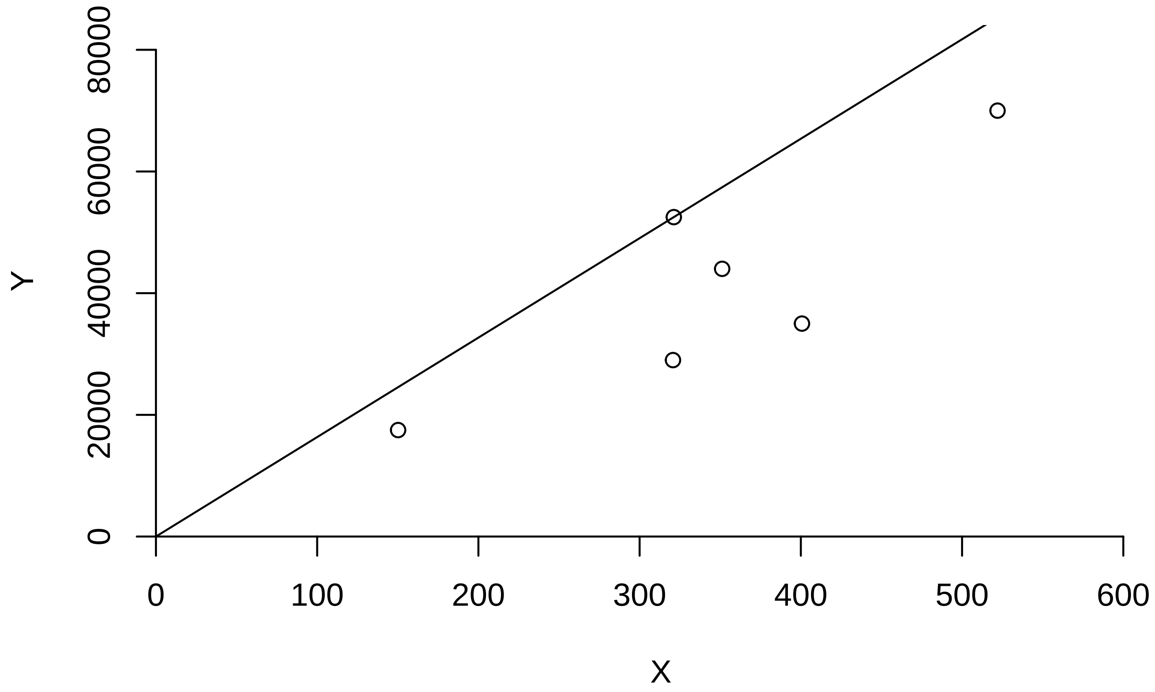
FRH - Free Replicability Hull

The results indicate that all the DMUs are efficient. This is because there is no convexity assumption

Plotting the Graphs

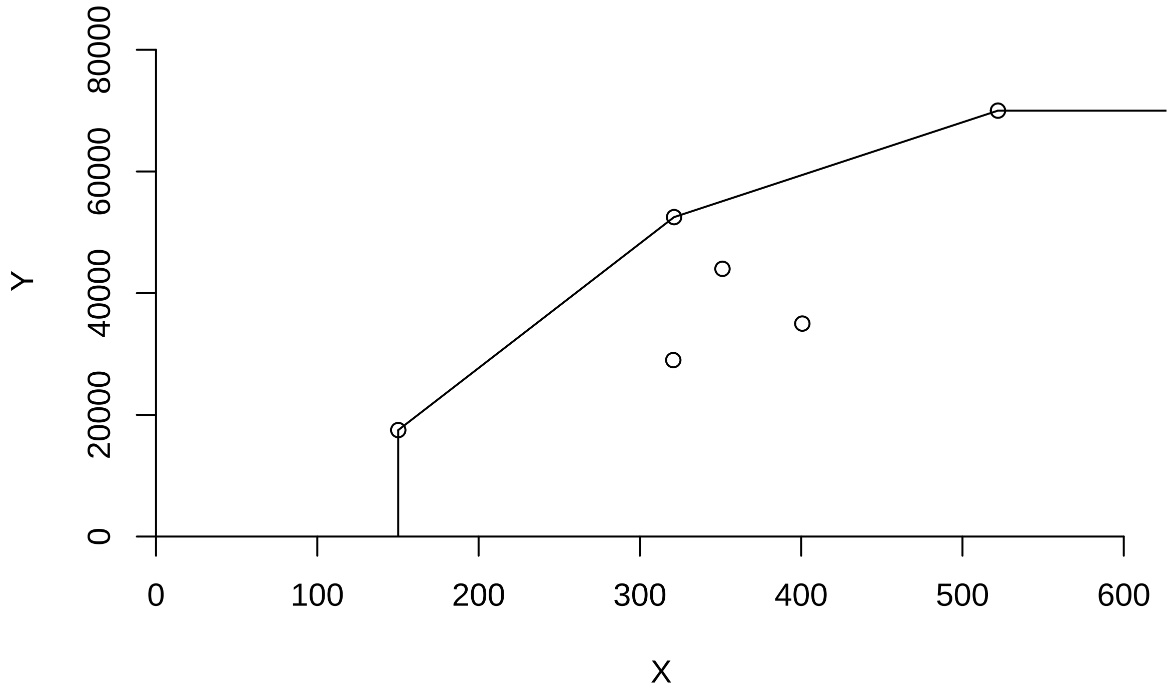
CRS Plot

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



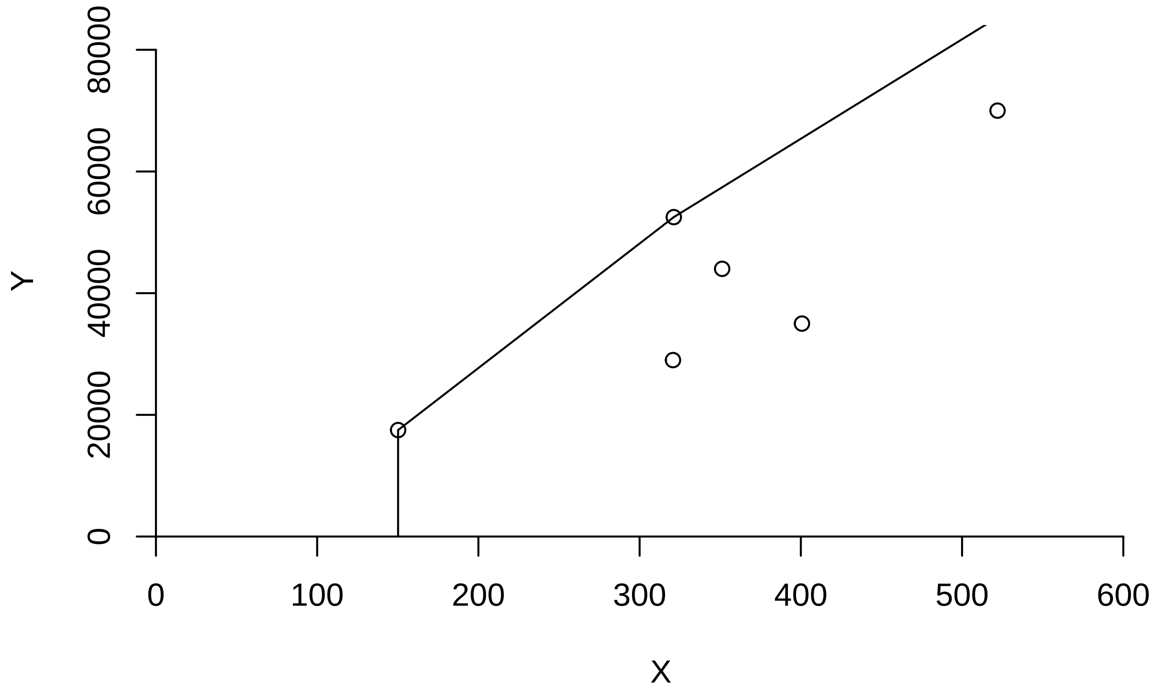
DRS Plot

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



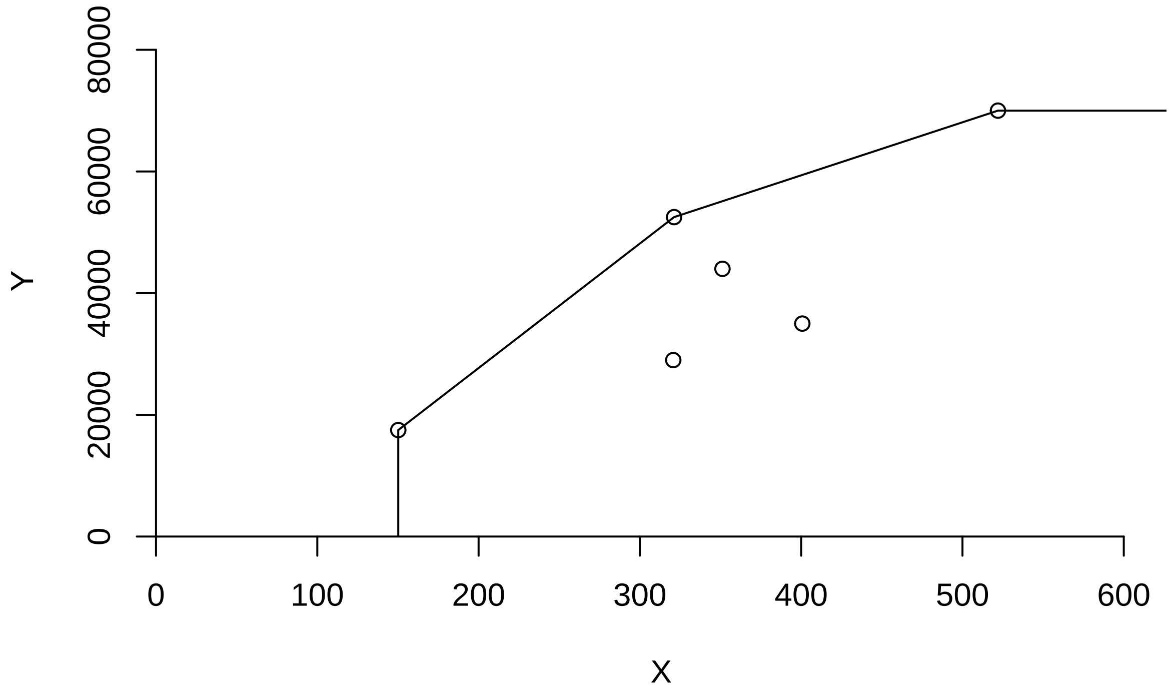
IRS Plot

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



VRS Plot

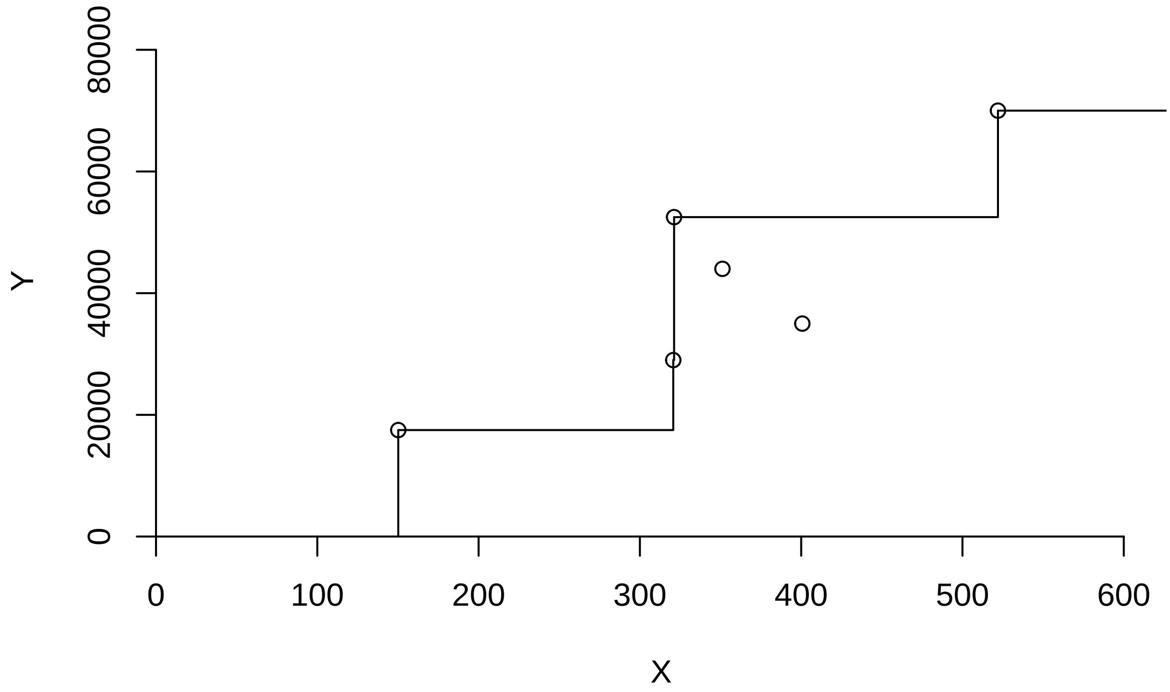
```
dea.plot(x,y,RTS="vrs")
```



```
#tinytex::install_tinytex()
```

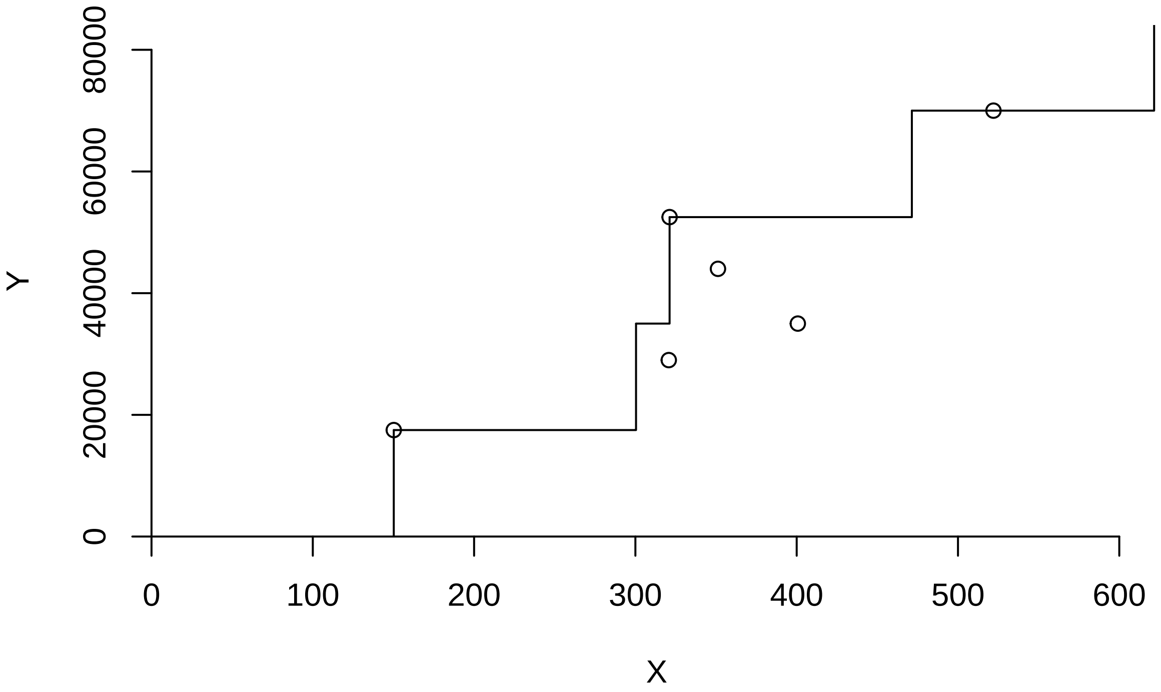
FDH Plot

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



FRH Plot

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


Y
0
20000
40000
60000
80000
X
0
100
200
300
400
500
600