Qmm Assignment_4

VENU DODDA

2022-10-30

##Loading Packages

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

```
## Loading required package: lpSolveAPI

## Loading required package: ucminf

## Loading required package: quadprog

##
## Loading Benchmarking version 0.30h, (Revision 244, 2022/05/05 16:31:31) ...

## Build 2022/05/05 16:31:40
```

Formulation of Data Envelopment Analysis model

```
data<-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) <- c("DMU", "Staff_Hours_Per_Day","Supplies_Per_Day","Reimbursed_Patient_Days","Privately
data <- as.table(data)
data
```

```
##   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
```

Formulation of all DEA assumptions of FDH, CRS, VRS, IRS, DRS,and FRH for above file.

#Set the Peers and Lambdas command under each of the above assumptions # Free Disposability Hull (FDH)

```
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")
print(x)
```

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

```
print(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
```

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

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

```
peers(FDH)
```

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

```
lambda(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
```

Constant Returns to Scale (CRS)

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

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

```
peers(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(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
```

Variable Returns to Scale (VRS)

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

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

```
peers(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(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
```

#Increasing Returns to Scale (IRS)

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

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

```
peers(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(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
```

Decreasing Returns to Scale (DRS)

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

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

```
peers(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(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
```

Free Replicability Hull (FRH)

```
FRH <- dea(x, y, RTS = "fdh+") #Additivity (scaling up and down, but only with integers) and local cons
FRH
```

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

```
peers(FRH)
```

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

```
lambda(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 Results for above tables in tabular format

```
as.table(data)
```

```
##   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
```

```
data1 <-table (CRS = c(1.0000, 1.0000, 1.0000, 1.0000, 0.9775, 0.8675),
FDH= c(1,1,1,1,1,1), VRS= c(1.0000, 1.0000, 1.0000, 1.0000, 1.0000, 0.8963),IRS =c( 1.0000, 1.0000, 1.0
data1
```

```
## , , VRS = 0.8963, IRS = 0.8963, DRS = 0.8675, FRH = 1
##
##         FDH
## CRS       1
##   0.8675 1
##   0.9775 0
##   1      0
##
## , , VRS = 1, IRS = 0.8963, DRS = 0.8675, FRH = 1
##
##         FDH
## CRS       1
##   0.8675 0
##   0.9775 0
##   1      0
##
## , , VRS = 0.8963, IRS = 1, DRS = 0.8675, FRH = 1
##
##         FDH
## CRS       1
##   0.8675 0
##   0.9775 0
##   1      0
##
## , , VRS = 1, IRS = 1, DRS = 0.8675, FRH = 1
##
##         FDH
## CRS       1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 0.8963, IRS = 0.8963, DRS = 0.9775, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 1, IRS = 0.8963, DRS = 0.9775, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 0.8963, IRS = 1, DRS = 0.9775, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 1, IRS = 1, DRS = 0.9775, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 1
##   1      0
## 
## , , VRS = 0.8963, IRS = 0.8963, DRS = 1, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 1, IRS = 0.8963, DRS = 1, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
## 
## , , VRS = 0.8963, IRS = 1, DRS = 1, FRH = 1
## 
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      0
##
## , , VRS = 1, IRS = 1, DRS = 1, FRH = 1
##
##         FDH
## CRS      1
##   0.8675 0
##   0.9775 0
##   1      4
```

Compare and contrast the above results

```
Results <- cbind(data,data1)
```

```
## Warning in cbind(data, data1): number of rows of result is not a multiple of
## vector length (arg 2)
```

```
Results[,-c(1:4)]
```

```
##   Privately_Paid_Patient_Days data1
## A "3500"                      "1"
## B "21000"                     "0"
## C "10500"                     "0"
## D "42000"                     "0"
## E "25000"                     "0"
## F "15000"                     "0"
```

Interpret DEA results

For FDH the results indicate that every DMU is efficient. This is primarily because the scale is able to assess even the lowest degree of efficiency because there is no geometrical constraint.

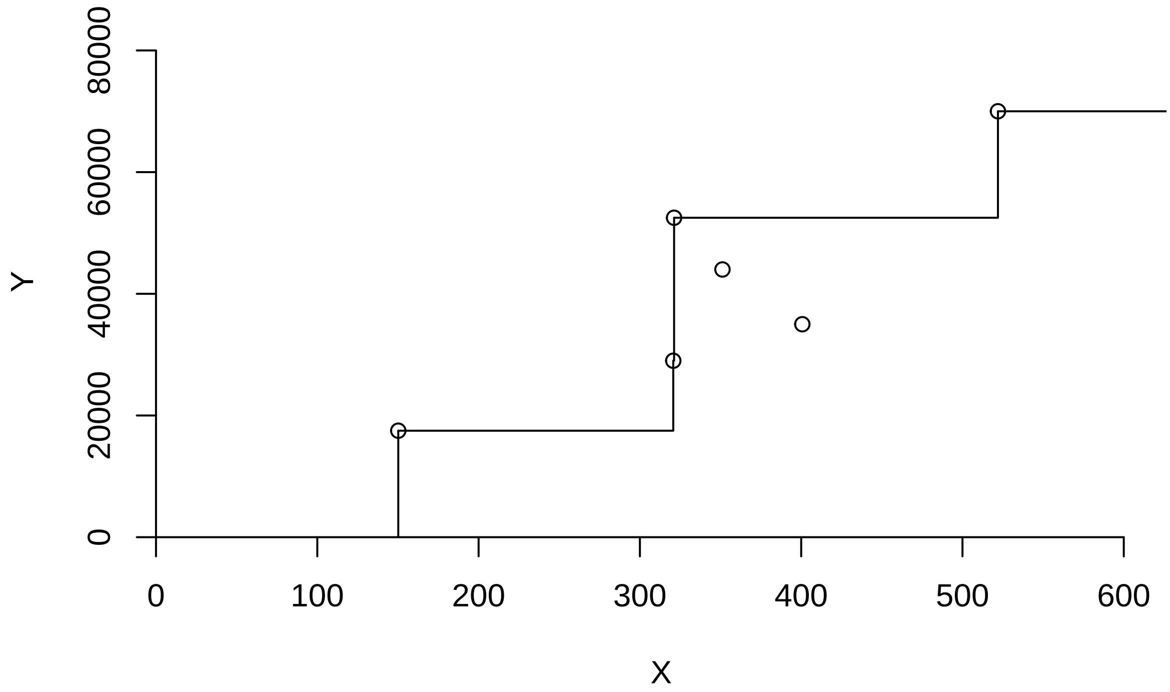
#For CRS the findings show that DMUs 1, 2, 3, and 4 are effective. Only 97.75% of DMU(5) and 86.7% of DMU(6) are effectively used. On the basis of our initial analysis, we discovered this. In addition, DMU(4peer)’s units are 1, 2, and 4, with weights of 0.2, 0.08, and 0.54. The peer units for DMU(6) are 1, 2, and 4, with respective weights of 0.34, 0.4, and 0.13. #For VRS the results showed 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. #For IRS the results showed that DMUs 1, 2, 3, 4, and 5 are effective. Only 89.63% of DMU(6) is effective. Based on our initial study, we discovered this. Additionally, the peer units for DMU(6) are 1, 2, and 5, with respective relative weights of 0.4, 0.34, and 0.26. #For DRS the results showed 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. #For FRH all DMUs are efficient, according to the FRH data. This is mainly because there isn't a convexity assumption used, and most of the time, this technique enables the scale to capture even the tiniest amount of efficiency that is not subject to replication or disposal.

plotting for all DEA acceptance of FDH, CRS, VRS, IRS, DRS,and FRH

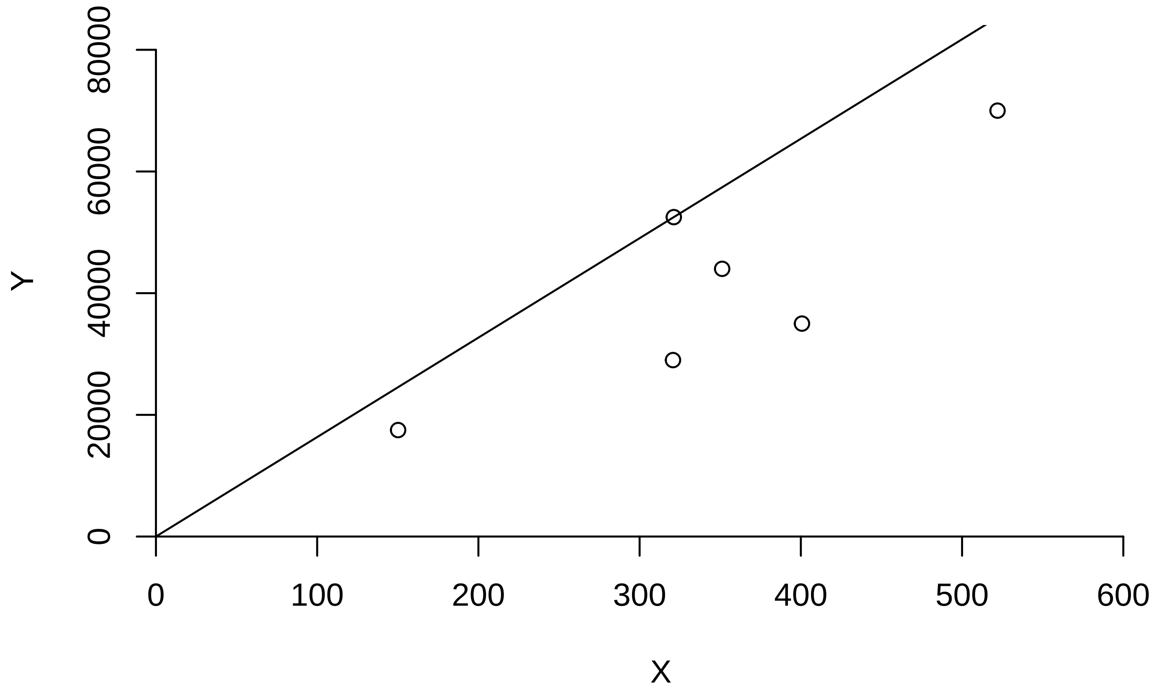
Plot for Free Disposability Hull(FDH)

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



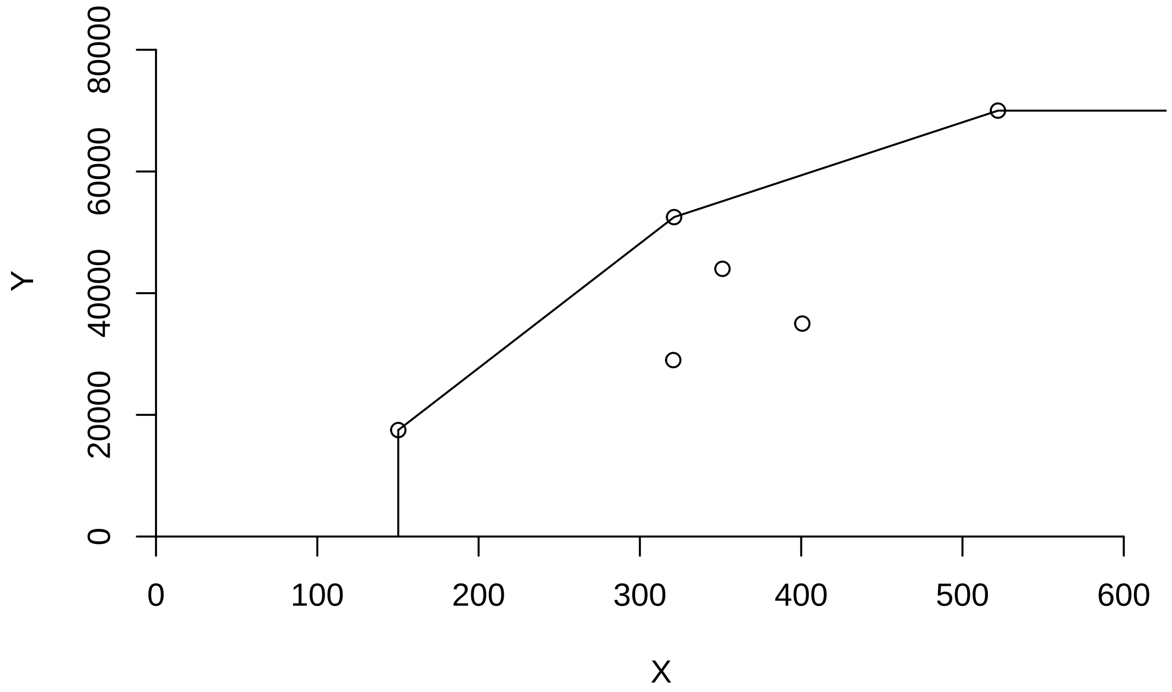
# Plot for Constant Returns to Scale(CRS)

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



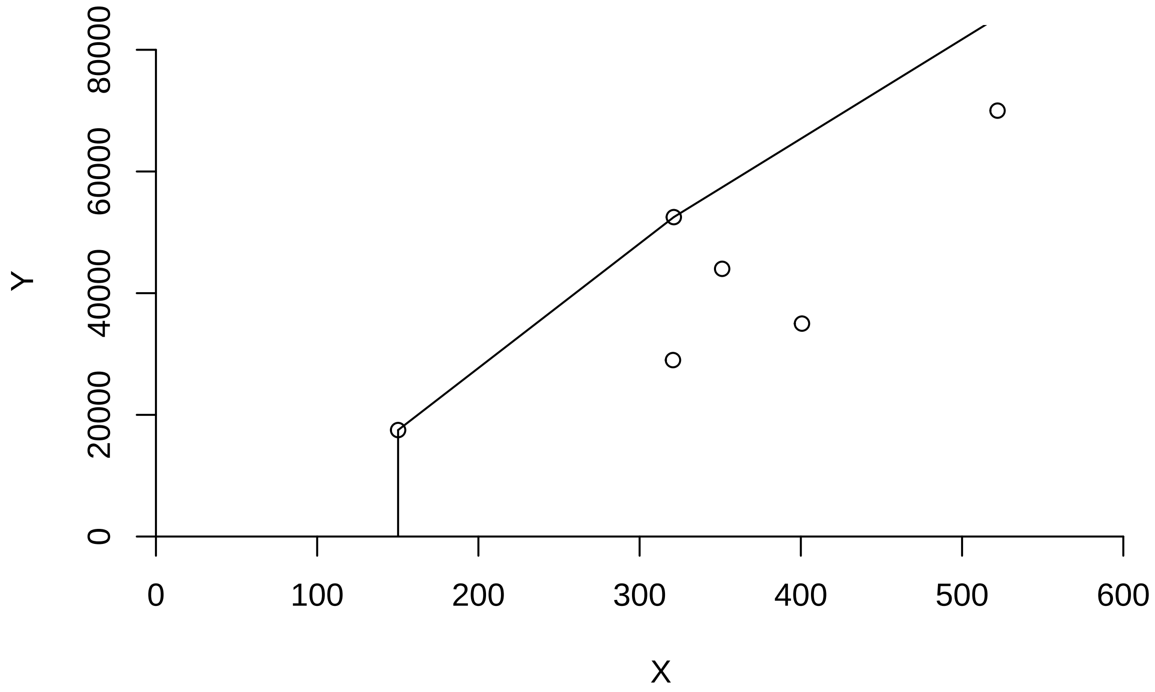
# Plot for Variable Returns to Scale(VRS)

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



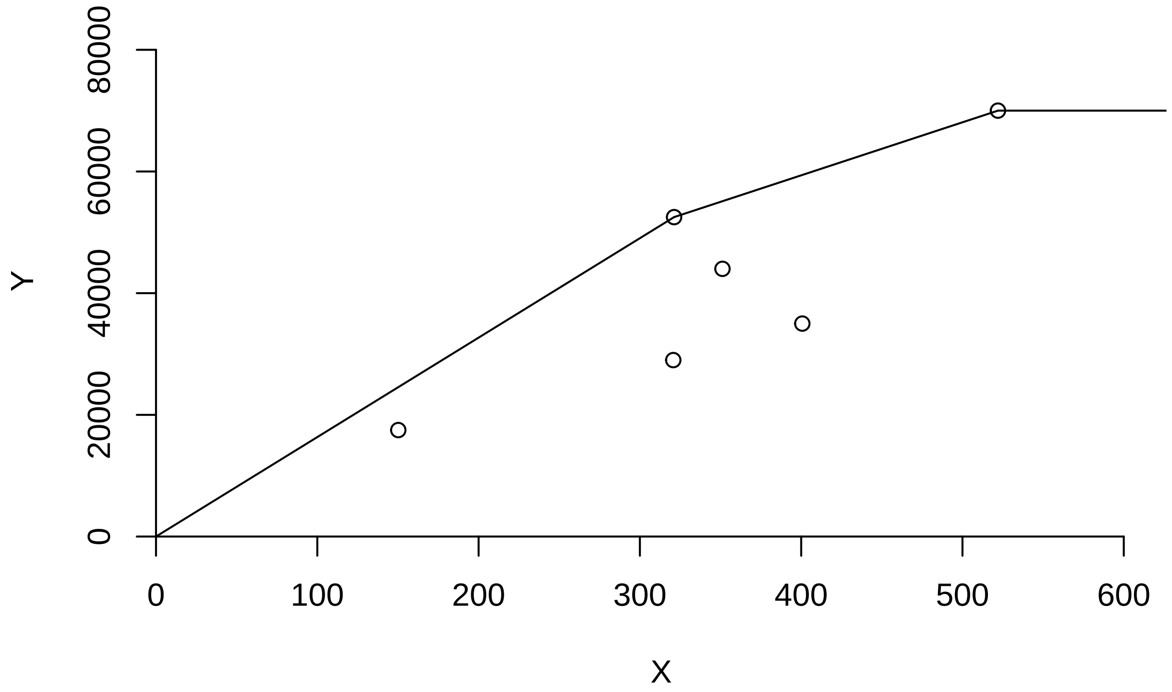
# Plot for Increasing Returns to Scale(IRS)

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



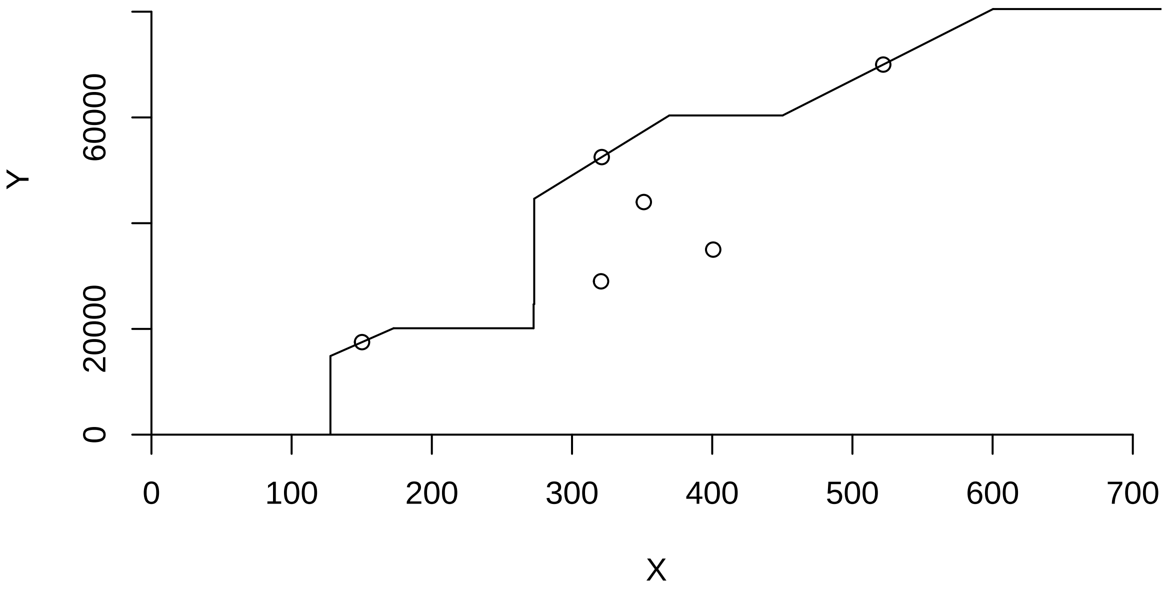
# Plot for Decreasing Returns to Scale(DRS)

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



# Plot for Free Replicability Hull(FRH)

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


Y
0
20000
60000
0
100
200
300
400
500
600
700
X