QMM-Assignment 04

Chandima Attanayake

2024-10-27

**import required packages**

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

## Loading required package: lpSolveAPI

## Loading required package: ucminf

## Loading required package: quadprog

##   
## Loading Benchmarking version 0.32h (Revision 263, 2024/03/13 15:04:04) ...

## Build 2024/03/13 15:05:00

# Load Benchmarking package  
library(Benchmarking)

if(!require(knitr)){  
 library(knitr)  
}

## Loading required package: knitr

library(kableExtra)

# Input data for DEA

df <- data.frame(  
 DMU = c("Facility 1", "Facility 2", "Facility 3", "Facility 4", "Facility 5", "Facility 6" ),   
 Staff\_hours\_per\_day = c(100,300,320,500,350,340),   
 Supplies\_per\_day = c(0.3,0.6,1.2,2,1.4,0.7),  
 Reimbursed\_patient\_days = c(15000,15000,40000,28000,20000,14000),  
 Privetly\_paid\_patient\_days = c(3500,20000,11000,42000,25000,15000)  
)

kable(df,format = "pandoc",caption = "Hope Valley Health Care Association")

Hope Valley Health Care Association

| DMU | Staff\_hours\_per\_day | Supplies\_per\_day | Reimbursed\_patient\_days | Privetly\_paid\_patient\_days |
| --- | --- | --- | --- | --- |
| Facility 1 | 100 | 0.3 | 15000 | 3500 |
| Facility 2 | 300 | 0.6 | 15000 | 20000 |
| Facility 3 | 320 | 1.2 | 40000 | 11000 |
| Facility 4 | 500 | 2.0 | 28000 | 42000 |
| Facility 5 | 350 | 1.4 | 20000 | 25000 |
| Facility 6 | 340 | 0.7 | 14000 | 15000 |

**create the matrix with given data**

x- contains input data (Supplies per day,Staff hours per day) y- contains output data (Reimbursed patient days, Privetly paid patient days)

x <- matrix(c(100,300,320,500,350,340,   
 0.3,0.6,1.2,2,1.4,0.7), ncol = 2)  
y <- matrix(c(15000,15000,40000,28000,20000,14000,   
 3500,20000,11000,42000,25000,15000), ncol = 2)  
colnames(x) <- c("Supplies\_per\_day","Staff\_hours\_per\_day")  
colnames(y) <- c("Reimbursed\_patient\_days", "Privetly\_paid\_patient\_days")

**Questions** 1. Formulate and perform DEA analysis under all DEA assumptions of FDH, CRS, VRS, IRS, DRS, and FRH. 2. Determine the Peers and Lambdas under each of the above assumptions 3. Summarize your results in a tabular format 4. Compare and contrast the above result

# DEA Analysis using FDH

We will now perform a DEA analysis using the Free Disposability Hull (FDH) approach. FDH evaluates a company’s efficiency in generating outputs from inputs, assuming the company can freely discard any unwanted inputs and outputs without incurring costs.

**input and output values**

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

**calculate efficiency and name the column**

eff\_fdh <- as.data.frame(fdh$eff)  
colnames(eff\_fdh)<- c("efficiency\_fdh")  
eff\_fdh

## efficiency\_fdh  
## 1 1.0000000  
## 2 1.0000000  
## 3 1.0000000  
## 4 1.0000000  
## 5 1.0000000  
## 6 0.8823529

Under the FDH analysis, Facilities 1 to 5 are fully efficient, with a score of 1.0000000, indicating optimal input-output utilization. Facility 6, with an efficiency score of 0.8823529, has some inefficiencies, suggesting potential improvements in resource management. Specifically, Facility 6 could reduce staffing hours or supply costs without compromising quality, possibly by introducing efficiency initiatives or negotiating better supplier rates.

**Find the peers**

peer\_fdh<-peers(fdh)  
colnames(peer\_fdh)<- c("peer\_fdh")  
peer\_fdh

## peer\_fdh  
## [1,] 1  
## [2,] 2  
## [3,] 3  
## [4,] 4  
## [5,] 5  
## [6,] 2

According to the peer comparison results, Facility 1 is compared to itself with a peer index of 1, as are Facilities 2, 3, 4, and 5, each with a peer index of their respective facility numbers. Facility 6 is compared to Facility 2, giving it a peer index of 2. The peer index of 1 for Facilities 1 through 5 indicates they are deemed efficient, as they are their own peers. In contrast, Facility 6 is compared to Facility 2, suggesting it may not be as efficient, as it does not qualify as its own peer under the FDH assumption.

**Find the lambda**

lambda\_fdh <- lambda(fdh)  
lambda\_fdh

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

The results indicate that L1, L2, L3, L4, and L5 represent the peer facilities for all six facilities, including Facility 6.

Facilities 1 through 5 are 100% efficient, as their efficiency scores are based on comparisons with themselves. However, Facility 6 is less efficient and is compared to Facility 2 for its efficiency score. This suggests that Facility 6 has opportunities to enhance its efficiency by optimizing the use of its inputs and outputs or by reducing inputs and increasing outputs.

**Tabular data for FDH**

tabular\_fdh <- cbind(peer\_fdh,lambda\_fdh,eff\_fdh)  
tabular\_fdh

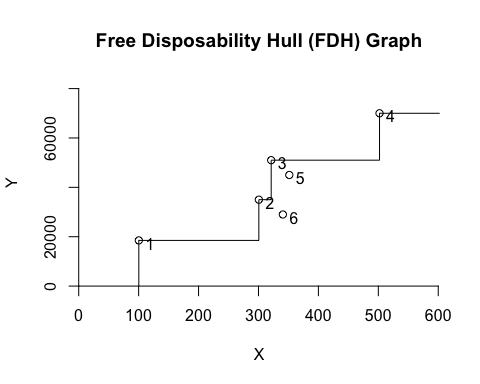
## peer\_fdh L1 L2 L3 L4 L5 efficiency\_fdh  
## 1 1 1 0 0 0 0 1.0000000  
## 2 2 0 1 0 0 0 1.0000000  
## 3 3 0 0 1 0 0 1.0000000  
## 4 4 0 0 0 1 0 1.0000000  
## 5 5 0 0 0 0 1 1.0000000  
## 6 2 0 1 0 0 0 0.8823529

**Interpretation of lambda**

In DEA, lambda values represent the extent to which a Decision Making Unit (DMU) borrows or learns from its efficient peers. Higher lambda values indicate that a DMU gains more insights from its peers to achieve efficiency. According to the data, only Facility 6 is inefficient, with an inefficiency score calculated as 1−0.88=0.121. This means that DMU6 can reduce its 12% inefficiency by learning from its peer, DMU2.

**Plot the result**

dea.plot(x,y,RTS = "fdh", ORIENTATION = "in-out", txt = TRUE, main= "Free Disposability Hull (FDH) Graph")



# DEA Analysis using CRS

We will now perform a DEA analysis using the Constant Returns to Scale (CRS) assumption. CRS is a scaling assumption that allows us to assess whether there are feasible combinations for scaling inputs or outputs without affecting overall efficiency.

**input and output values**

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

**calculate efficiency and name the column**

eff\_crs <- as.data.frame(crs$eff)  
colnames(eff\_crs)<- c("efficiency\_crs")  
eff\_crs

## efficiency\_crs  
## 1 1.0000000  
## 2 1.0000000  
## 3 0.8793468  
## 4 1.0000000  
## 5 0.8941998  
## 6 0.7047619

Based on the results, Facilities 1, 2, and 4 are operating at optimal efficiency under the CRS assumption. However, Facilities 3, 5, and 6 have opportunities for improvement in their resource utilization and output generation. These efficiency scores can be used to rank and compare the facilities, helping to identify those with the most significant efficiency gaps.

**Find the peers**

peer\_crs<-peers(crs)  
peer\_crs

## peer1 peer2  
## [1,] 1 NA  
## [2,] 2 NA  
## [3,] 1 4  
## [4,] 4 NA  
## [5,] 1 4  
## [6,] 1 2

The matrix above shows the potential peers for each of the six entities (DMUs). DMU1, DMU2, and DMU4 are their own peers, which classifies them as efficient.

However, DMU3, 5 and 6 have multiple peers and therefore, they are considered ineffecient.

**Find the lambda**

lambda\_crs <- lambda(crs)  
lambda\_crs

## L1 L2 L4  
## [1,] 1.0000000 0.0000000 0.00000000  
## [2,] 0.0000000 1.0000000 0.00000000  
## [3,] 2.5789474 0.0000000 0.04699248  
## [4,] 0.0000000 0.0000000 1.00000000  
## [5,] 0.2631579 0.0000000 0.57330827  
## [6,] 0.2222222 0.7111111 0.00000000

Rows 1 to 6 correspond to Facilities 1 to 6.

Column “L1”, “L2” and “L4” represents the lambda value for Facility 1, which is 1.0000000. This indicates that Facility 1 can achieve efficiency independently, without relying on other facilities.

However, Facility 3 has a lambda value of 2.5789474 for “L1” and 0.04699248 for “L4.” This means Facility 3 can achieve efficiency by relying primarily on Facility 1 (2.5789474 times) and, to a lesser extent, on Facility 4 (0.04699248 times).

Facility 5 has a lambda value of 0.2631579 for “L1” and 0.57330827 for “L4.” This indicates that Facility 5 can achieve efficiency through a combination of itself and Facility 1 (0.2631579 times) and with Facility 4 (0.57330827 times).

Facility 6 has a lambda value of 0.2222222 for “L1” and 0.7111111 for “L2.” therefore, Facility 6 can attain efficiency by relying on Facility 1 (0.2222222 times) and more significantly on Facility 2 (0.7111111 times).

**Tabular data for CRS**

tabular\_crs <- cbind(peer\_crs,lambda\_crs,eff\_crs)  
tabular\_crs

## peer1 peer2 L1 L2 L4 efficiency\_crs  
## 1 1 NA 1.0000000 0.0000000 0.00000000 1.0000000  
## 2 2 NA 0.0000000 1.0000000 0.00000000 1.0000000  
## 3 1 4 2.5789474 0.0000000 0.04699248 0.8793468  
## 4 4 NA 0.0000000 0.0000000 1.00000000 1.0000000  
## 5 1 4 0.2631579 0.0000000 0.57330827 0.8941998  
## 6 1 2 0.2222222 0.7111111 0.00000000 0.7047619

According to the CRS tabular data, DMU3 learns from its peers, DMU1(L1) and DMU4(L4). The lambda values for L1L1L1 and L4L4L4 indicate the extent to which DMU3 benefits from these peers. Since the lambda value for DMU1 is 2.5789474, DMU3 can potentially maximize its efficiency gain from this peer, with the remaining inefficiency being addressed through DMU4.

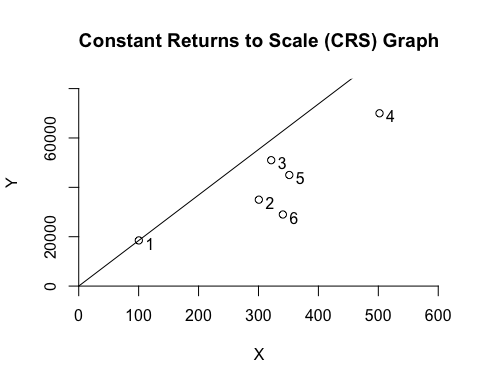
Similarly, DMU5 is inefficient, with an inefficiency score of 1−(0.2631579+0.57330827)=0.163533831, indicating it can reduce this 16.35% inefficiency by learning from peers DMU1(L1) and DMU4. The ratio of DMU1(L1) to DMU4 is 4:5, so DMU5 can learn 0.1058002×4/9=0.047022311 (4.7%) from DMU1 and 0.1058002×5/9= 0.058777889 (5.88%) from DMU4.

For DMU6, the inefficiency is calculated as 1−(0.2222222+0.7111111)=0.06666671 or 6.67%. DMU6 can address this inefficiency by learning from peers DMU1(L1) and DMU2(L2) (L2). The ratio of DMU1(L1) to DMU2(L2) is 14:15. Hence, DMU6 can learn 0.0666667×14/29=0.032183924 (3.22%) from DMU1 and 0.0666667×15/29=0.034482776 (3.45%) from DMU2.

The other facilities, DMU1, DMU2, and DMU4, are operating at full efficiency with a score of 1.0000.

**Plot the result**

dea.plot(x,y,RTS = "crs", ORIENTATION = "in-out", txt = TRUE, main= "Constant Returns to Scale (CRS) Graph")

 # DEA Analysis using VRS

We will now perform a DEA analysis using the Variable Returns to Scale (VRS) assumption. **input and output values**

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

**calculate efficiency and name the column**

eff\_vrs <- as.data.frame(vrs$eff)  
colnames(eff\_vrs)<- c("efficiency\_vrs")  
eff\_vrs

## efficiency\_vrs  
## 1 1.0000000  
## 2 1.0000000  
## 3 1.0000000  
## 4 1.0000000  
## 5 0.9239332  
## 6 0.7272727

Facilities 1, 2, 3, and 4 all have an efficiency score of 1.000, indicating they are efficient under the VRS assumption. These facilities are maximizing their inputs relative to their specific circumstances.

Facility 5 has an efficiency score of approximately 0.924, suggesting it is less efficient than the fully efficient facilities and has potential for improvement in resource utilization.

Facility 6, with an efficiency score of approximately 0.727, is the least efficient among the group, highlighting a need for significant enhancements in resource allocation and management.

**Find the peers**

peer\_vrs<-peers(vrs)  
peer\_vrs

## peer1 peer2  
## [1,] 1 NA  
## [2,] 2 NA  
## [3,] 3 NA  
## [4,] 4 NA  
## [5,] 1 4  
## [6,] 1 2

facility 1,2,3 and 4 are their own peers and no any secondary peer. However, for facility 5 has two peers, facility 1 and 4.For facility 6 also has two peers as facility 1 and facility 2.

**Find the lambda**

lambda\_vrs <- lambda(vrs)  
lambda\_vrs

## L1 L2 L3 L4  
## [1,] 1.0000000 0.0000000 0 0.0000000  
## [2,] 0.0000000 1.0000000 0 0.0000000  
## [3,] 0.0000000 0.0000000 1 0.0000000  
## [4,] 0.0000000 0.0000000 0 1.0000000  
## [5,] 0.4415584 0.0000000 0 0.5584416  
## [6,] 0.3030303 0.6969697 0 0.0000000

According to the table, Facilities 1, 2, 3, and 4 are efficient, each having an efficiency score of 1.000. However, Facility 5 has an efficiency score of 0.4415584, indicating that it relies on Facility 1 to achieve efficiency with a weight of 0.4415584, and also depends on Facility 4, contributing to its overall efficiency.

Facility 6 has an efficiency score of 0.3030303, indicating that it relies on Facility 1 to achieve efficiency with a weight of 0.3030303, while it also depends on Facility 2, which has a weight of 0.6969697 in its efficiency calculation.

**Tabular data for VRS**

tabular\_vrs <- cbind(peer\_vrs,lambda\_vrs,eff\_vrs)  
tabular\_vrs

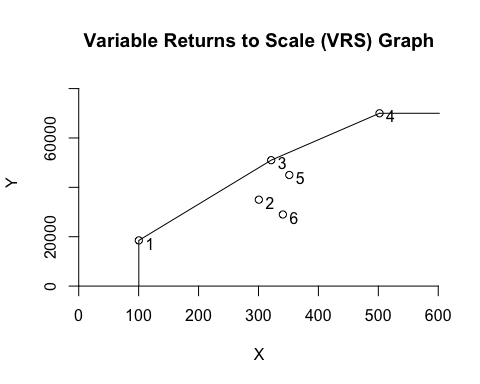
## peer1 peer2 L1 L2 L3 L4 efficiency\_vrs  
## 1 1 NA 1.0000000 0.0000000 0 0.0000000 1.0000000  
## 2 2 NA 0.0000000 1.0000000 0 0.0000000 1.0000000  
## 3 3 NA 0.0000000 0.0000000 1 0.0000000 1.0000000  
## 4 4 NA 0.0000000 0.0000000 0 1.0000000 1.0000000  
## 5 1 4 0.4415584 0.0000000 0 0.5584416 0.9239332  
## 6 1 2 0.3030303 0.6969697 0 0.0000000 0.7272727

According to the tabular data, DMU1, DMU2, DMU3, and DMU4 are identified as efficient. However, DMU5 is inefficient, with an inefficiency score of 1−(0.44+0.55)=0.01. This indicates that DMU5 can reduce its 1% inefficiency by learning from its peers DMU1(L1) and DMU4(L4). The ratio of DMU1(L1) to DMU4(L4)DMU\_4 (L4) is 4:5. Therefore, DMU5 can learn 0.01×4/9=0.004444444 (approximately 0.4%) from DMU1 and 0.01×5/9 = 0.005555556 (approximately 0.56%) from DMU4.

Similarly, DMU6 is also inefficient, with an inefficiency score of 1−(0.3+0.69) = 0.01. This inefficiency can be reduced by learning from its peers DMU1(L1) and DMU2(L2). The ratio between these peers is 14:15. Hence, DMU6 can learn 0.01×14/29=0.004827586 (approximately 0.48%) from DMU1(L1) and 0.01×15/29=0.005172414 (approximately 0.52%) from DMU2(L2).

**Plot the result**

dea.plot(x,y,RTS = "vrs", ORIENTATION = "in-out", txt = TRUE, main= "Variable Returns to Scale (VRS) Graph")

 # DEA Analysis using IRS

**input and output values**

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

**calculate efficiency and name the column**

eff\_irs <- as.data.frame(irs$eff)  
colnames(eff\_irs)<- c("efficiency\_irs")  
eff\_irs

## efficiency\_irs  
## 1 1.0000000  
## 2 1.0000000  
## 3 0.8793468  
## 4 1.0000000  
## 5 0.9239332  
## 6 0.7272727

Based on the results under the Increasing Returns to Scale (IRS) assumption, Facilities 1, 2, and 4 are identified as fully efficient, each with an efficiency score of 1. However, Facilities 5 and 6 are not fully efficient, as their efficiency scores fall below 1.

**Find the peers**

peer\_irs<-peers(irs)  
peer\_irs

## peer1 peer2  
## [1,] 1 NA  
## [2,] 2 NA  
## [3,] 1 4  
## [4,] 4 NA  
## [5,] 1 4  
## [6,] 1 2

According to the table results, DMU1, DMU2, and DMU4 do not have any secondary peers and have become their own peers, allowing them to be identified as fully efficient. However, DMU3 and DMU5 each have two peers as DMU1 and DMU4. Additionally, DMU6 also has two peers of DMU1 and DMU2.

**Find the lambda**

lambda\_irs <- lambda(irs)  
lambda\_irs

## L1 L2 L4  
## [1,] 1.0000000 0.0000000 0.00000000  
## [2,] 0.0000000 1.0000000 0.00000000  
## [3,] 2.5789474 0.0000000 0.04699248  
## [4,] 0.0000000 0.0000000 1.00000000  
## [5,] 0.4415584 0.0000000 0.55844156  
## [6,] 0.3030303 0.6969697 0.00000000

Facility 1: L1 = 1.000 (Fully efficient under IRS) Facility 2: L2 = 1.000 (Fully efficient under IRS) Facility 3: L1 = 2.579, L4 = 0.047 (Inefficient under IRS, but it can improve by reallocating resources) Facility 4: L4 = 1.000 (Fully efficient under IRS) Facility 5: L1 = 0.442, L4 = 0.558 (Inefficient under IRS, and it needs to improve resource allocation) Facility 6: L1 = 0.303, L2 = 0.697 (Inefficient under IRS, and it needs to improve resource allocation)

**Tabular data for IRS**

tabular\_irs <- cbind(peer\_irs,lambda\_irs,eff\_irs)  
tabular\_irs

## peer1 peer2 L1 L2 L4 efficiency\_irs  
## 1 1 NA 1.0000000 0.0000000 0.00000000 1.0000000  
## 2 2 NA 0.0000000 1.0000000 0.00000000 1.0000000  
## 3 1 4 2.5789474 0.0000000 0.04699248 0.8793468  
## 4 4 NA 0.0000000 0.0000000 1.00000000 1.0000000  
## 5 1 4 0.4415584 0.0000000 0.55844156 0.9239332  
## 6 1 2 0.3030303 0.6969697 0.00000000 0.7272727

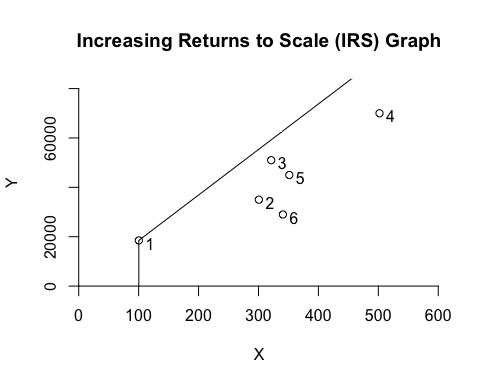
According to the IRS table data, DMU1, DMU2, and DMU4 are identified as efficient. DMU3 is deemed inefficient and can reduce its inefficiency primarily by learning from its peer DMU1, as its lambda value is greater than 1. The remaining inefficiency can be addressed by learning from its other peer, DMU4.

DMU5 is inefficient, with an inefficiency score of 1−(0.44+0.55)=0.01. Its peers are DMU1(L1) and DMU4(L4), with a peer ratio of 4:5. Therefore, DMU5 can learn 0.01×(4/9)=0.0044 (or 0.44%) from DMU1 and 0.01×(5/9)=0.005555556 (or 0.56%) from DMU4.

Finally, DMU6 is also inefficient, with an inefficiency score of 1−(0.3+0.69) = 0.01. Despite its inefficiency, it can learn from its peers DMU1(L1) and DMU2(L2). The ratio between these peers is 14:15, allowing DMU6 to learn 0.01×(14/29) =0.004827586 (or 0.48%) from DMU1 and 0.01×(15/29)=0.005172414 (or 0.52%) from DMU2

**Plot the result**

dea.plot(x,y,RTS = "irs", ORIENTATION = "in-out", txt = TRUE, main= "Increasing Returns to Scale (IRS) Graph")



# DEA Analysis using DRS

**input and output values**

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

**calculate efficiency and name the column**

drs <- dea(x,y,RTS = "drs")  
eff\_drs <- as.data.frame(drs$eff)  
colnames(eff\_drs)<- c("efficiency\_drs")  
eff\_drs

## efficiency\_drs  
## 1 1.0000000  
## 2 1.0000000  
## 3 1.0000000  
## 4 1.0000000  
## 5 0.8941998  
## 6 0.7047619

According to the table results, DMU1, DMU2, DMU3, and DMU4 are operating at full efficiency. In contrast, DMU5 and DMU6 have efficiency scores below 1, which identifies them as inefficient DMUs under the DRS assumption.

**Find the peers**

peer\_drs<-peers(drs)  
peer\_drs

## peer1 peer2  
## [1,] 1 NA  
## [2,] 2 NA  
## [3,] 3 NA  
## [4,] 4 NA  
## [5,] 1 4  
## [6,] 1 2

According to the above matrix, Facility 1, Facility 2, Facility 3, and Facility 4 are their own peers and do not have any additional peers, making these four DMUs efficient. However, DMU5 has two peers, DMU1 and DMU4, while DMU6 has two peers, DMU1 and DMU2. Therefore, both DMU5 and DMU6 are identified as inefficient.

**Find the lambda**

lambda\_drs <- lambda(drs)  
lambda\_drs

## L1 L2 L3 L4  
## [1,] 1.0000000 0.0000000 0 0.0000000  
## [2,] 0.0000000 1.0000000 0 0.0000000  
## [3,] 0.0000000 0.0000000 1 0.0000000  
## [4,] 0.0000000 0.0000000 0 1.0000000  
## [5,] 0.2631579 0.0000000 0 0.5733083  
## [6,] 0.2222222 0.7111111 0 0.0000000

According to the lambda values, Facility 1, Facility 2, Facility 3, and Facility 4 are fully efficient, as indicated by their lambda values of 1.

For Facility 5, the transformation of the data shows that the first dimension (L1) has a lambda value of approximately 0.2631579, while the fourth dimension (L4) has a lambda value of approximately 0.5733083, indicating inefficiency.

For Facility 6, the transformation reveals that the first dimension (L1) has a lambda value of approximately 0.2222222, and the second dimension (L2) has a lambda value of approximately 0.7111111, also indicating inefficiency. Thus, both Facility 5 and Facility 6 are considered inefficient

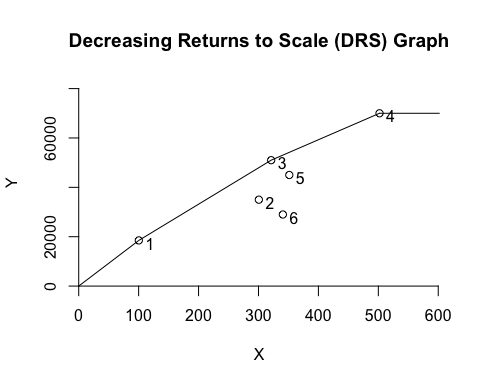
**Tabular data for DRS**

tabular\_drs <- cbind(peer\_drs,lambda\_drs,eff\_drs)  
tabular\_drs

## peer1 peer2 L1 L2 L3 L4 efficiency\_drs  
## 1 1 NA 1.0000000 0.0000000 0 0.0000000 1.0000000  
## 2 2 NA 0.0000000 1.0000000 0 0.0000000 1.0000000  
## 3 3 NA 0.0000000 0.0000000 1 0.0000000 1.0000000  
## 4 4 NA 0.0000000 0.0000000 0 1.0000000 1.0000000  
## 5 1 4 0.2631579 0.0000000 0 0.5733083 0.8941998  
## 6 1 2 0.2222222 0.7111111 0 0.0000000 0.7047619

**Plot the result**

dea.plot(x,y,RTS = "drs", ORIENTATION = "in-out", txt = TRUE, main= "Decreasing Returns to Scale (DRS) Graph")



# DEA Analysis using FRH

**input and output values**

frh <- dea(x,y,RTS = "add")

**calculate efficiency and name the column**

frh <- dea(x,y,RTS = "add")  
eff\_frh<- as.data.frame(frh$eff)  
colnames(eff\_frh)<- c("efficiency\_frh")  
eff\_frh

## efficiency\_frh  
## 1 1.0000000  
## 2 1.0000000  
## 3 1.0000000  
## 4 1.0000000  
## 5 1.0000000  
## 6 0.8823529

According to the above efficiency values,(DMU\_6 is inefficient because its’ efficient value is below 1 and all the other DMUs are identified as efficient under the assumption of FRH.

**Find the peers**

peer\_frh<-peers(frh)  
peer\_frh

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

According to the above peer identification, all the DMUs except (DMU6 become their own peers and only DMU6 consider DMU2 as its’ peer. Therefore DMU6 seems to be inefficient.

**Find the lambdas**

lambda\_frh <- lambda(frh)  
lambda\_frh

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

Facility 1,2,3,4 and 5 are 100% efficient, and its efficiency score is calculated by comparing it to itself.Facility 6 is less efficient than the other facilities, and its efficiency score is calculated by comparing it to Facility 2. Therefore, facility 6 effeciency can be improved.

**Tabular data for FRH**

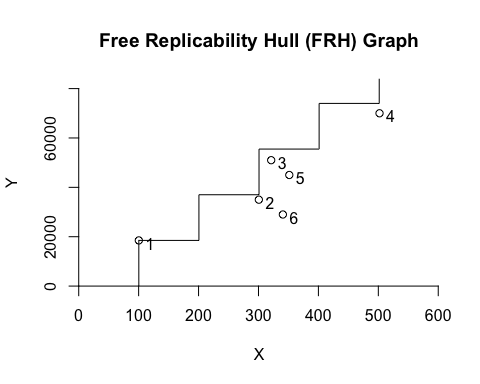
tabular\_frh <- cbind(peer\_frh,lambda\_frh,eff\_frh)  
tabular\_frh

## peer1 L1 L2 L3 L4 L5 efficiency\_frh  
## 1 1 1 0 0 0 0 1.0000000  
## 2 2 0 1 0 0 0 1.0000000  
## 3 3 0 0 1 0 0 1.0000000  
## 4 4 0 0 0 1 0 1.0000000  
## 5 5 0 0 0 0 1 1.0000000  
## 6 2 0 1 0 0 0 0.8823529

According to the tabular data of FRH, (DMU\_6 is inefficient.

**Plot the result**

dea.plot(x,y,RTS = "add", ORIENTATION = "in-out", txt = TRUE, main= "Free Replicability Hull (FRH) Graph")



# Summary

According to the DEA analysis under all DEA assumptions of FDH, CRS, VRS,IRS, DRS, and FRH all the inefficient DMUs can be summarize as follow. Accordingly, the DMS can be identified as inefficient. Under four assumptions DMU5 is identified as inefficient. Also, DMU3 is inefficient according to CRS and IRS assumptions.

df2 <- data.frame(  
 FDH = c("DMU\_6", " ", " "),  
 CRS = c("DMU\_3", "DMU\_5", "DMU\_6"),   
 VRS = c("DMU\_5", "DMU\_6", " "),  
 IRS = c("DMU\_3", "DMU\_5", "DMU\_6"),  
 DRS = c("DMU\_5", "DMU\_6", " "),  
 FRH = c("DMU\_6"," ", " ")  
)  
df2

## FDH CRS VRS IRS DRS FRH  
## 1 DMU\_6 DMU\_3 DMU\_5 DMU\_3 DMU\_5 DMU\_6  
## 2 DMU\_5 DMU\_6 DMU\_5 DMU\_6   
## 3 DMU\_6 DMU\_6