Assignment 1

Rohith Chandra

2022-10-31

# Question 1 -

# Step 1 - Creating matrix for the problem  
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)  
  
# Assign column names to the matrix  
colnames(x) <- c("Staff\_Hours\_per\_day","Supplies\_per\_day")  
colnames(y) <- c("Reimbursed\_patient\_days", "Privately\_paid\_patient-days")

# Formulating and performing DEA analysis

1. DEA Analysis using FDH

library(Benchmarking)

## Warning: package 'Benchmarking' was built under R version 4.2.1

## Loading required package: lpSolveAPI

## Warning: package 'lpSolveAPI' was built under R version 4.2.1

## Loading required package: ucminf

## Warning: package 'ucminf' was built under R version 4.2.1

## Loading required package: quadprog

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

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

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.7 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1

## Warning: package 'ggplot2' was built under R version 4.2.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

# Analyse DEA using FDH with the input x and output y variables  
fdh\_ana = dea(x,y,RTS = "fdh")  
  
#Finding the efficiency value to dataset  
fdh\_eff = as.data.frame(fdh\_ana$eff)  
print(fdh\_eff)

## fdh\_ana$eff  
## 1 1  
## 2 1  
## 3 1  
## 4 1  
## 5 1  
## 6 1

1. DEA Analysis using CRS

# Analyse DEA using CRS with input x and output y variables  
crs\_ana = dea(x,y,RTS = "crs")  
  
#Finding the efficiency value to dataset  
crs\_eff = as.data.frame(crs\_ana$eff)  
print(crs\_ana)

## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675

1. DEA Analysis using VRS

# Analyse DEA using VRS with input x and output y variables  
vrs\_ana = dea(x,y,RTS = "vrs")  
  
#Finding the efficiency value to dataset  
vrs\_eff = as.data.frame(vrs\_ana$eff)  
print(vrs\_ana)

## [1] 1.0000 1.0000 1.0000 1.0000 1.0000 0.8963

1. DEA Analysis using IRS

# Analyse DEA using IRS with input x and output y values  
irs\_ana <- dea(x,y,RTS = "irs")  
  
#Finding efficiency value to dataset  
irs\_eff = as.data.frame(irs\_ana$eff)  
print(irs\_ana)

## [1] 1.0000 1.0000 1.0000 1.0000 1.0000 0.8963

1. DEA Analysis using DRS

# Analyse DEA using DRS with input x and output y values  
drs\_ana = dea(x,y,RTS = "drs")  
  
#Finding efficiency value to dataset  
drs\_eff = as.data.frame(drs\_ana$eff)  
print(drs\_ana)

## [1] 1.0000 1.0000 1.0000 1.0000 0.9775 0.8675

1. DEA Analysis using FRH

# Analyse DEA using FRH with input x and output y values  
frh\_ana = dea(x,y,RTS = "add")  
  
#Finding efficiency value to dataset  
frh\_eff = as.data.frame(frh\_ana$eff)  
print(frh\_ana)

## [1] 1 1 1 1 1 1

# Question 2 - Determining the Peers and Lambdas under each of the DEA assumption

1. Peers and Lambdas for FDH

# Determining peers  
fdh\_peers <- peers(fdh\_ana)  
  
# Determining the weights using lambda function for the peer values  
fdh\_lamda <- lambda(fdh\_ana)

1. Peers and Lambdas for CRS

# Determining peers  
crs\_peers <- peers(crs\_ana)  
  
# Weights using lambda function for the peer values  
crs\_lamda <- lambda(crs\_ana)

1. Peers and Lambdas for VRS

# Determining peers  
vrs\_peers <- peers(vrs\_ana)  
  
# Weights using lambda function for the peer values  
vrs\_lamda <- lambda(vrs\_ana)

1. Peers and Lambdas for IRS

# Determining peers  
irs\_peers <- peers(irs\_ana)  
  
# Weights using lambda function for the peer values  
irs\_lamda <- lambda(irs\_ana)

1. Peers and Lambdas for DRS

# Determining peers  
drs\_peers <- peers(drs\_ana)  
  
# Weights using lambda function for the peer values  
drs\_lamda <- lambda(drs\_ana)

1. Peers and Lambdas for FDH

# Identify the peers  
frh\_peers <- peers(frh\_ana)  
  
# Weights given to the peers using lambda function  
frh\_lamda <- lambda(frh\_ana)

# Question 3 - Summarizing results in a tabular format

fdh\_result <- data.frame(fdh\_eff,fdh\_peers, fdh\_lamda)  
crs\_result <- data.frame(crs\_eff,crs\_peers, crs\_lamda)  
vrs\_result <- data.frame(vrs\_eff,vrs\_peers, vrs\_lamda)  
irs\_result <- data.frame(irs\_eff,irs\_peers, irs\_lamda)  
drs\_result <- data.frame(drs\_eff,drs\_peers, drs\_lamda)  
frh\_result <- cbind(frh\_eff,frh\_peers, frh\_lamda)

fdh\_result

## fdh\_ana.eff peer1 L1 L2 L3 L4 L5 L6  
## 1 1 1 1 0 0 0 0 0  
## 2 1 2 0 1 0 0 0 0  
## 3 1 3 0 0 1 0 0 0  
## 4 1 4 0 0 0 1 0 0  
## 5 1 5 0 0 0 0 1 0  
## 6 1 6 0 0 0 0 0 1

crs\_result

## crs\_ana.eff peer1 peer2 peer3 L1 L2 L3 L4  
## 1 1.0000000 1 NA NA 1.0000000 0.00000000 0 0.0000000  
## 2 1.0000000 2 NA NA 0.0000000 1.00000000 0 0.0000000  
## 3 1.0000000 3 NA NA 0.0000000 0.00000000 1 0.0000000  
## 4 1.0000000 4 NA NA 0.0000000 0.00000000 0 1.0000000  
## 5 0.9774987 1 2 4 0.2000000 0.08048142 0 0.5383307  
## 6 0.8674521 1 2 4 0.3428571 0.39499264 0 0.1310751

vrs\_result

## vrs\_ana.eff peer1 peer2 peer3 L1 L2 L3 L4 L5  
## 1 1.0000000 1 NA NA 1.0000000 0.0000000 0 0 0.0000000  
## 2 1.0000000 2 NA NA 0.0000000 1.0000000 0 0 0.0000000  
## 3 1.0000000 3 NA NA 0.0000000 0.0000000 1 0 0.0000000  
## 4 1.0000000 4 NA NA 0.0000000 0.0000000 0 1 0.0000000  
## 5 1.0000000 5 NA NA 0.0000000 0.0000000 0 0 1.0000000  
## 6 0.8963283 1 2 5 0.4014399 0.3422606 0 0 0.2562995

irs\_result

## irs\_ana.eff peer1 peer2 peer3 L1 L2 L3 L4 L5  
## 1 1.0000000 1 NA NA 1.0000000 0.0000000 0 0 0.0000000  
## 2 1.0000000 2 NA NA 0.0000000 1.0000000 0 0 0.0000000  
## 3 1.0000000 3 NA NA 0.0000000 0.0000000 1 0 0.0000000  
## 4 1.0000000 4 NA NA 0.0000000 0.0000000 0 1 0.0000000  
## 5 1.0000000 5 NA NA 0.0000000 0.0000000 0 0 1.0000000  
## 6 0.8963283 1 2 5 0.4014399 0.3422606 0 0 0.2562995

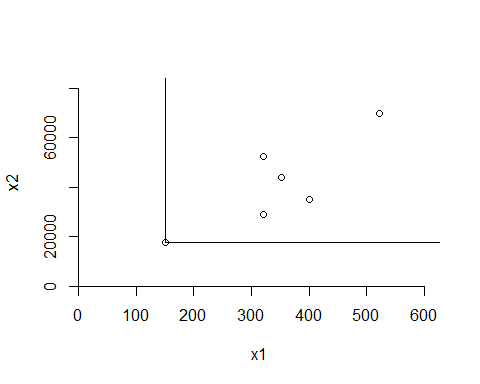
drs\_result

## drs\_ana.eff peer1 peer2 peer3 L1 L2 L3 L4  
## 1 1.0000000 1 NA NA 1.0000000 0.00000000 0 0.0000000  
## 2 1.0000000 2 NA NA 0.0000000 1.00000000 0 0.0000000  
## 3 1.0000000 3 NA NA 0.0000000 0.00000000 1 0.0000000  
## 4 1.0000000 4 NA NA 0.0000000 0.00000000 0 1.0000000  
## 5 0.9774987 1 2 4 0.2000000 0.08048142 0 0.5383307  
## 6 0.8674521 1 2 4 0.3428571 0.39499264 0 0.1310751

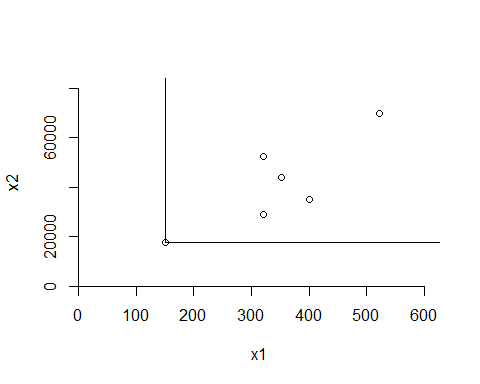
frh\_result

## frh\_ana$eff peer1 L1 L2 L3 L4 L5 L6  
## 1 1 1 1 0 0 0 0 0  
## 2 1 2 0 1 0 0 0 0  
## 3 1 3 0 0 1 0 0 0  
## 4 1 4 0 0 0 1 0 0  
## 5 1 5 0 0 0 0 1 0  
## 6 1 6 0 0 0 0 0 1

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



dea.plot.isoquant(x,y,RTS = "crs")



dea.plot.isoquant(x,y,RTS = "vrs")  
dea.plot.isoquant(x,y,RTS = "irs")  
dea.plot.isoquant(x,y,RTS = "drs")

# Question 4 - Comapre and contrast the above results

In simple terms - In FDH, the peer value was given one unit, both lamda and efficiency values are 1 and from the result table it states that every DMU and facility is having maximum capacity and effectiveness. In CRS, the lamdas and peers it is observed that the efficiency of facilities 1, 2, 3, and 4 are 100% which means they use CRS fully. The efficiency for 5 and 6 is 97.74% and 86.74% respectively which can be improved.In VRS, the lamdas and peers we can tell facilities—numbers 1, 2, 3, 4, and 5 has maximum efficieny/productivity. With facility 6, which has an efficiency of 89.63%. In IRS, Facilities 1, 2, 3, 4, and 5 operate at full productivity, IRS and VRS are both achieved. For facility 6, has 89.63% efficiency requires improvement from units 1, 2, and 5.In DRS, Facilities 1, 2, 3, and 4, Decreasing Returns to Scale (DRS) performs well in terms of efficiency and for facilities 5 and 6 needs improvement and require a portion of facilities 1, 2, and 4 in order to get maximum efficiency of 1.In FRH, all Facilities are efficient which are observed in peer and lambda. For the DMUs - DMU 1,2,3 & 4 have efficiencies of 1 for all DEA analysis.DMU 5 has efficiency of 1 for FDH;VRS;IRS;FRH analysis but efficiency of 0.9775 for both CRS and DRS. DMU 6 has efficiency of 1 for FDH and FRH analysis, for CRS and DRS analysis - 0.8675; VRS and IRS 0.8963.