Module 8 DEA

Hannah Strickling

2022-10-10

Reference: https://www.rdocumentation.org/packages/Benchmarking/versions/0.30/topics/dea

Questions 1 & 2

Run the DEA model under the different assumtions and determine the peers and lambda for each model

First we will read our input data. We will read the data as input and output as vectors. For this problem we have 6 DMUs with Staff Hours per Day and Supplies per Day as input variables and Reimbursed Patient Days and Privately Paid Patient Days as output variables.

```
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")

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
```

```
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
```

Now we will run the DEA analysis using the benchmarking library. We will use the option of CRS. CRS = Constant returns to scale, convexity and free disposability

```
e1 <- dea(x,y,RTS = "crs")
e1
```

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

Next we will determine the Peers and Lamba

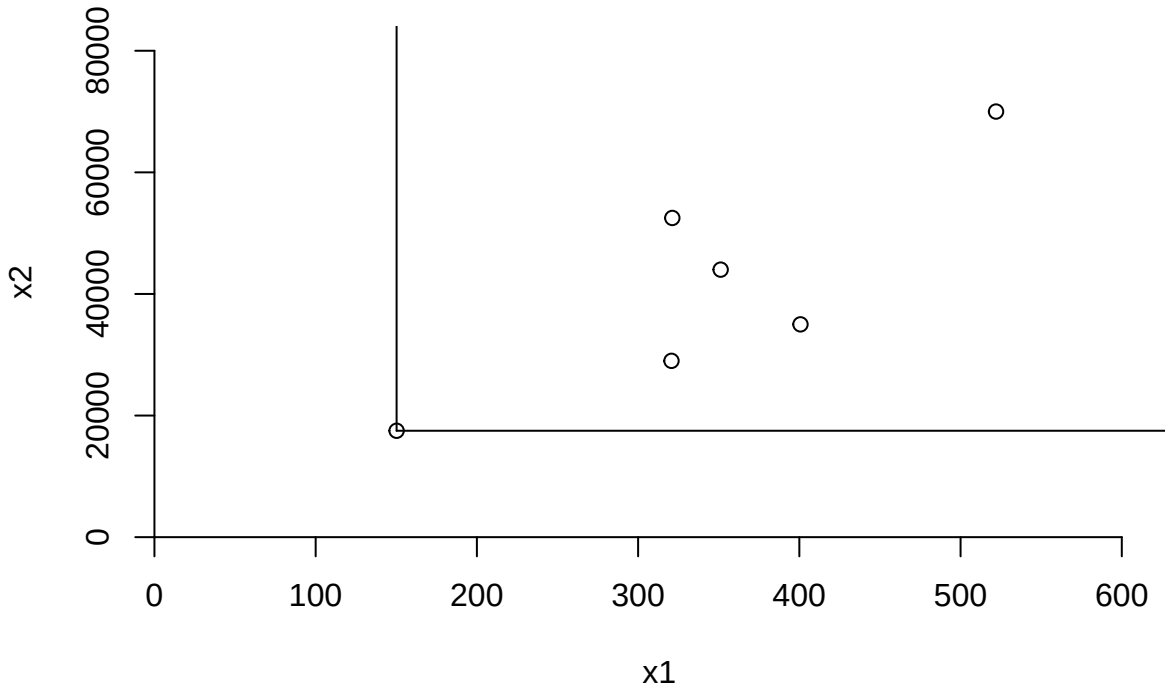
```
peers(e1)
```

```
##      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(e1)
```

```
##             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
```

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



Now we will run under the assumption of FDH. FDH = Free disposability hull, no convexity assumption

```
e2 <- dea(x,y,RTS = "fdh")
e2
```

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

```
peers(e2)
```

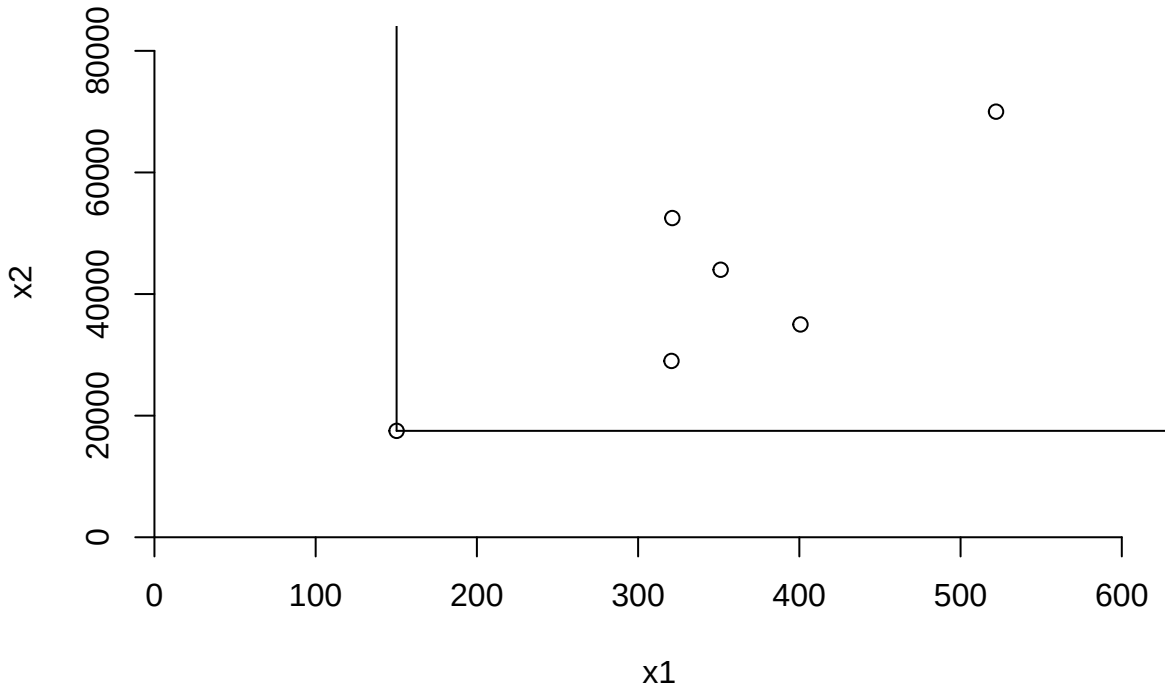
```
##      peer1
## [1,]     1
## [2,]     2
## [3,]     3
## [4,]     4
## [5,]     5
## [6,]     6
```

```
lambda(e2)
```

```
##      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
```

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



Now we will use the vrs assumption. VRS = Variable returns to scale, convexity and free disposability

```
e3 <- dea(x,y,RTS = "vrs")
e3
```

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

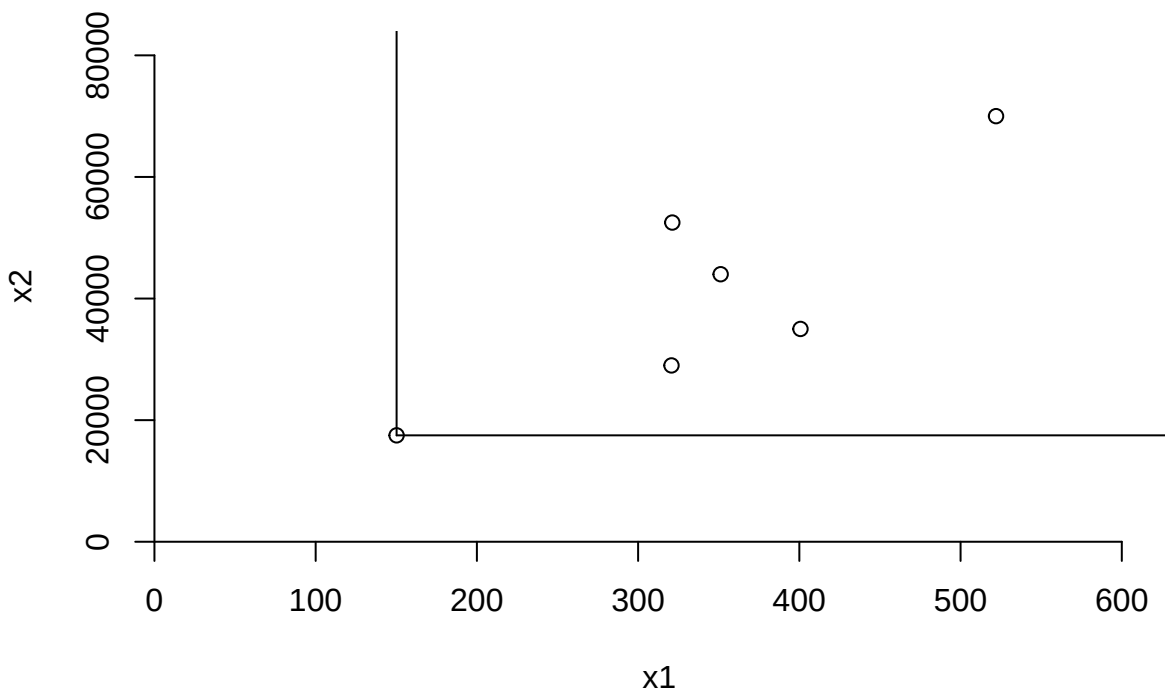
```
peers(e3)
```

```
##      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(e3)
```

```
##             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
```

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



Now we will use the irs assumption. IRS = Increasing returns to scale, (up-scaling, but not down-scaling), convexity and free disposability

```
e4 <- dea(x,y,RTS = "irs")
e4
```

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

```
peers(e4)
```

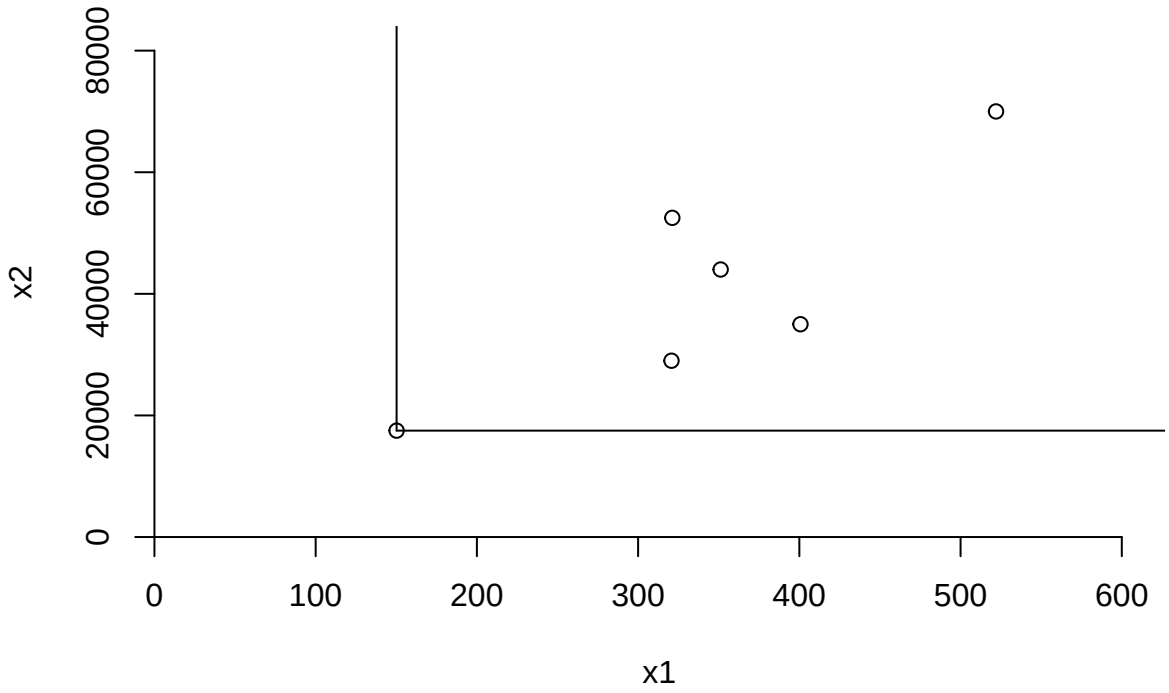
```
##      peer1 peer2 peer3
## [1,]     1    NA    NA
## [2,]     2    NA    NA
## [3,]     3    NA    NA
## [4,]     4    NA    NA
```

```
## [5,]     5    NA    NA
## [6,]     1     2     5
```

```
lambda(e4)
```

```
##             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
```

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



Next is the drs. DRS = Decreasing returns to scale, convexity, down-scaling and free disposability

```
e5 <- dea(x,y,RTS = "drs")
e5
```

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

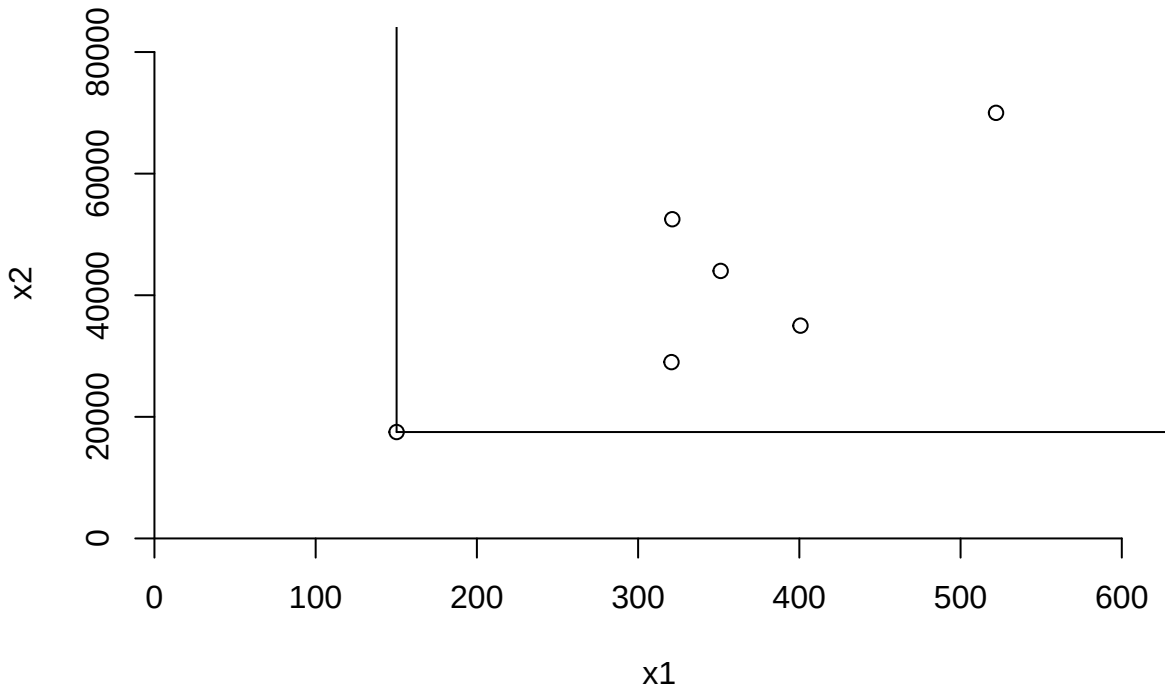
```
peers(e5)
```

```
##      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(e5)
```

```
##             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
```

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



Finally we will use frh. FRH = Free Replication Hull. A combination of free disposability and restricted or local constant return to scale. R would not recognize frh as an rts option so i am utilizing the fdh+ option

```r
e6 <- dea(x,y,RTS = "fdh+")
e6
```

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

```r
peers(e6)
```

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

```r
lambda(e6)
```

```
##      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
```

Question 3 Summarize Results into table

```r
e1
```

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

```r
e2
```

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

```r
e3
```

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

```r
e4
```

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

```r
e5
```

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

```
e6
```

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

```
Q3 <- read.csv("Summary.csv")
Q3
```

```
##                X Facility.1 Facility.2 Facility.3 Facility.4 Facility.5
## 1 Efficiency CRS          1          1          1          1     0.9775
## 2 Efficiency FDH          1          1          1          1     1.0000
## 3 Efficiency VRS          1          1          1          1     1.0000
## 4 Efficiency IRS          1          1          1          1     1.0000
## 5 Efficiency DRS          1          1          1          1     0.9775
## 6 Efficiency FRH          1          1          1          1     1.0000
##   Facility.6
## 1     0.8675
## 2     1.0000
## 3     0.8963
## 4     0.8963
## 5     0.8675
## 6     1.0000
```

Question 4 Compare and Contrast the Results

We looked at the six classical DEA models: namely,the original constant return to scale (CRS) model; the decreasing, increasing and varying return to scale (DRS, IRS and VRS) models; and the free disposability and the replicability hull (FDH, FRH) models. Lets start with CRS and DRS. When looking at the results of the model under these assumptions we will see that facilities 1, 2, 3, and 4 are efficient however facility 5 is only 97.75% efficient and 86.75% for facility 6. Under FDH and FRH all facilities are efficient. With VRS and IRS, all firms are efficient except for 6 which is at 89.63% efficient.

The purpose of running the peers is to see for the inefficient facilities, which efficient facility they should reference to lead to improvements. As we saw above, under CRS and DRS firms 5 and 6 are not efficient. Both firms should reference facilities 1, 2, and 4 for improvements. Under VRS and IRS facility 6 is inefficient and when looking at the peers, we will see that this facility should reference facilities 1, 2, and 5 to make improvements to its efficiency.The lambda results help us determine deeper insights into the peer by providing weights for the reference unit. For example, under CRS and DRS, we will notice that the weights for facilities 1, 2, and 4 are 34%, 39%, and 13% respectively. Under VRS and IRS, firms 1, 2, and 5, are weighted as 40%, 34%, and 25%.