dcurran3\_8

2022-10-28

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

x<- matrix(c(150,400,320,520,350,320))  
y <- matrix(c(14000,14000,42000,28000,19000,14000,35000,21000,10500,42000,25000,15000), ncol = 2)  
colnames(y) <- c("Reimbursed Patient-Days","Privately Paid Patient-Days")  
colnames(x) <- c("Staffed Hours per Day")  
dea(x,y,RTS = "crs")

## [1] 1.0000 0.3269 1.0000 0.5028 0.4932 0.3828

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

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

lambda(c)

## L1 L3  
## [1,] 1.0000000 0.0000000  
## [2,] 0.5555556 0.1481481  
## [3,] 0.0000000 1.0000000  
## [4,] 1.1111111 0.2962963  
## [5,] 0.6428571 0.2380952  
## [6,] 0.3650794 0.2116402

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

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

lambda(f)

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

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

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

lambda(v)

## L1 L3 L4  
## [1,] 1.0000000 0.0000000 0  
## [2,] 1.0000000 0.0000000 0  
## [3,] 0.0000000 1.0000000 0  
## [4,] 0.0000000 0.0000000 1  
## [5,] 0.8214286 0.1785714 0  
## [6,] 1.0000000 0.0000000 0

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

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

lambda(i)

## L1 L3  
## [1,] 1.0000000 0.0000000  
## [2,] 1.0000000 0.0000000  
## [3,] 0.0000000 1.0000000  
## [4,] 1.1111111 0.2962963  
## [5,] 0.8214286 0.1785714  
## [6,] 1.0000000 0.0000000

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

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

lambda(d)

## L1 L3 L4  
## [1,] 1.0000000 0.0000000 0  
## [2,] 0.5555556 0.1481481 0  
## [3,] 0.0000000 1.0000000 0  
## [4,] 0.0000000 0.0000000 1  
## [5,] 0.6428571 0.2380952 0  
## [6,] 0.3650794 0.2116402 0

After running all these test it is clear to see that everyone is slightly different in the outcome. However overall each one shows similar things. First facility three seems to be the most efficient across the board. One difference that I noticed is that in some test facility for performs well and others it does not.