POLEN
Política Externa em Números

Dynamic DEA Analysis
#Below we show the steps taken to estimate Brazil's diplomatic efficiency using a dynamic DEA model#

#Step 1. Prepare data for estimation#

#Import the Excel data available on the POLEN portal, on the page: https://sites.usp.br/pen/analise-de-eficiencia/, called Disaggregated Data of the Diplomatic Efficiency Index - Edit the path to locate the downloaded file on the computer - We call the database "dados"#

```
library(readxl)
dados <- read_excel("~/Your Path/Dados_Indice de Eficiencia_site.xlsx")
```

#Balance the panel, excluding countries that are not 10 years old - DEA analysis does not accept missing data - we call the new object "dados2"#

```
dados2<- dados[dados$country %in% names(which(table(dados$country) == 10)), ]
```

#Normalize data with the min_max method - DEA analysis requires normalized data, remembering that the fpi is already normalized#

```
library(tidyr)
library(dplyr)

cols_to_normalize <- c("funcionarios", "salario", "embaixada")

min_max_rank_normalize <- function(x) {
  ranked <- rank(x)
  normalized <- (ranked - min(ranked)) / (max(ranked) - min(ranked))
  return(normalized)
}

normalized_data <- as.data.frame(lapply(dados2[cols_to_normalize],
min_max_rank_normalize))
```

#Change variable names to match databases#

```
names(normalized_data)[names(normalized_data) == "funcionarios"] <- "func_norm"
names(normalized_data)[names(normalized_data) == "salario"] <- "sal_norm"
names(normalized_data)[names(normalized_data) == "embaixada"] <- "emb_norm"
```

#Combine the normalized databases with the already modified data ("data2") - creating the "data" object#

```
data <- cbind(dados2, normalized_data)
```

#Remove variables that will not be used, creating a new object, "data2"#

```
data2 <- subset(data, select = -c(funcionarios, salario, embaixada))
```

#There are few values with zero in the inputs, but in order not to affect the DEA estimation, it is recommended to exclude these cases, creating the "data_clean" object, eliminating the zero cases#

```
data_clean <- subset(data2, func_norm != 0 & sal_norm != 0 & emb_norm != 0 & fpi != 0)
```

#Remove any remaining missing items#

```
data_clean <- na.omit(data_clean)
```

#Step 2. Estimate the Dynamic DEA model#

#Import Benchmarking package for estimation of dynamic DEA#

```
library(Benchmarking)
```

 # Space reserved for storing efficiency results#

```
efficiency_results_dynamic <- data.frame()
```

 # Loop through each specific year of the sample to run the DEA, selecting func_norm, sal_norm and emb_norm as inputs and the fpi as output#

```
for (current_year in unique(data_clean$year)) {
  data_current_year <- subset(data_clean, year == current_year)
  inputs <- as.matrix(data_current_year[, c("func_norm", "sal_norm", "emb_norm")])
  outputs <- as.matrix(data_current_year[, c("fpi")])
  dea_model <- dea(X = inputs, Y = outputs, RTS = "vrs")
  efficiency_scores <- eff(dea_model)
  results_current_year <- data.frame(
    country = data_current_year$country,
    year = current_year,
    Efficiency_Score = efficiency_scores
  )
  efficiency_results_dynamic <- rbind(efficiency_results_dynamic, results_current_year)
}
```

POLEN
Política Externa em Números

Returns the results of the DEA estimation#

print(efficiency_results_dynamic)

POLEN – Política Externa em Números