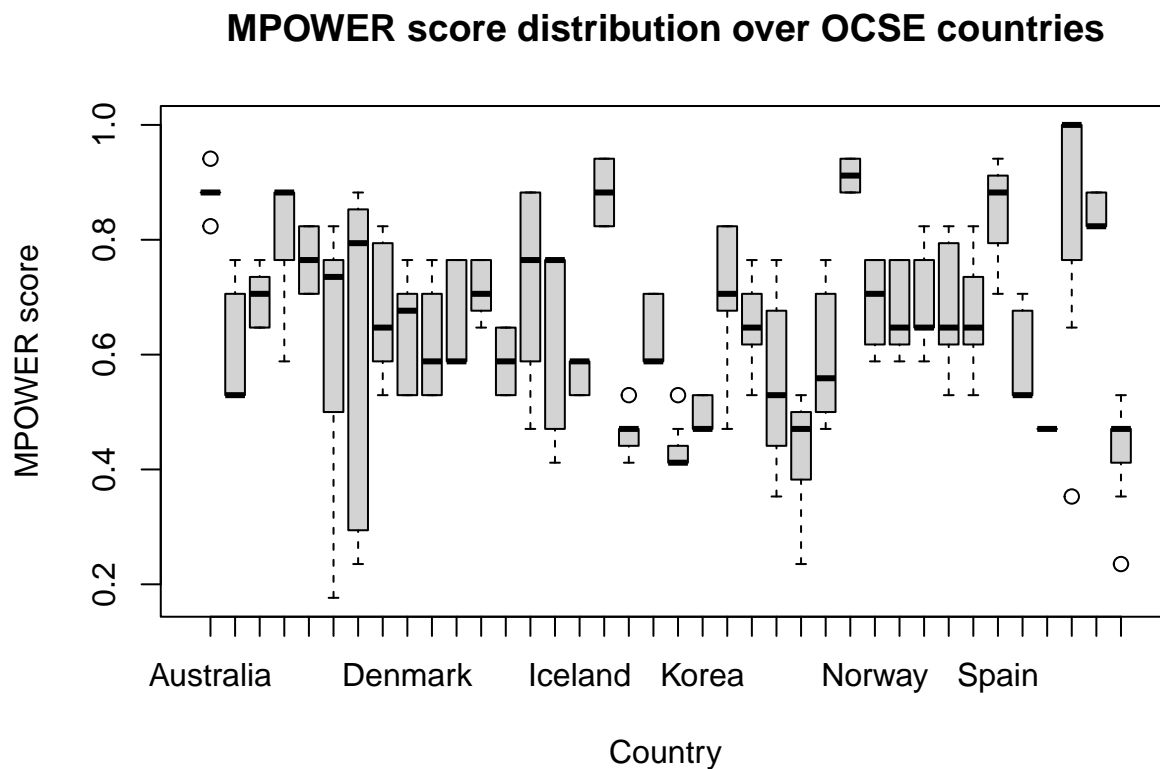# Mpower index analysis

2023-07-05

Plotting mPOWER distributions over years

```r
par(mfrow=c(1,1))
boxplot(data$mpower_all~data$Country,
        main="MPOWER score distribution over OCSE countries",
        xlab="Country",
        ylab="MPOWER score")
```

MPOWER score distribution over OCSE countries

Plotting mPOWER distributions across countries

```r
ggplot(data, aes(x = as.factor(Year), y = mpower_all)) +
  geom_boxplot(fill = "steelblue") +
  labs(title = "MPOWER Score Distribution in 2008-2020",
       x = "Year",
       y = "MPOWER Score") +
  theme_minimal() +
  theme(plot.title = element_text(size = 16, face = "bold"),
        axis.title = element_text(size = 14),
        axis.text = element_text(size = 14))
```

MPOWER Score Distribution in 2008–2020
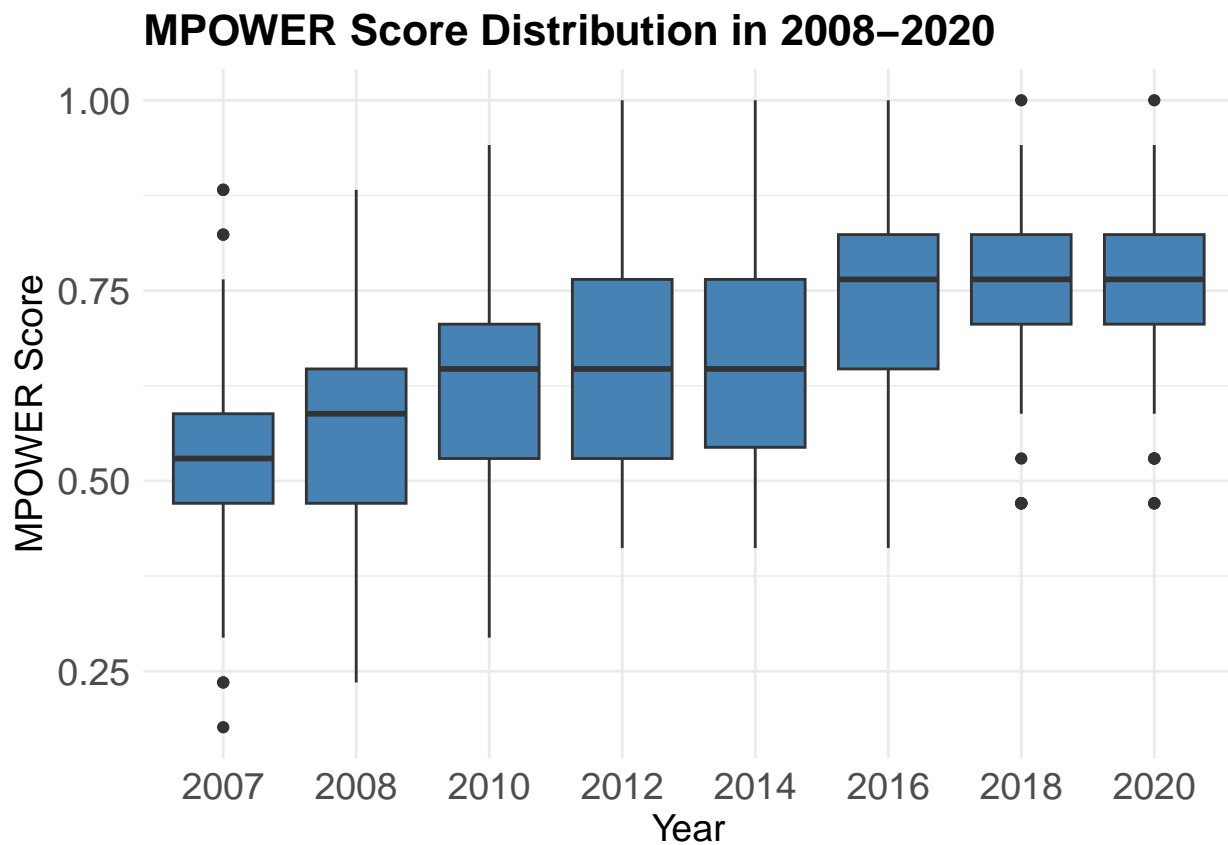
Index seems to have increased

```r
#Can we say that we have increased over time?---------
#We use two approaches:

dataset_mpower<-data[,c(1,2,24)]
#Both
mpower_table <- dataset_mpower %>%
  pivot_wider(names_from = Country,
              values_from = mpower_all)

mpower_table<-as.data.frame(mpower_table)

df<-t(mpower_table[8,-1]-mpower_table[2,-1])
```

```r
#First we check that the difference between  2020 and 2008 is 0

compute_t_stat=function(df,med0){
  med_diff<-median((df)-med0)
  return (abs(med_diff))
}

med0<-0
T0<-compute_t_stat(df,med0)
T0
```

```
## [1] 0.1764706
```

```r
symmetric_perm_wrapper = function(df, med0) {
  n = dim(df)[1]
  p =  dim(df)[2] #2 #I am confronting two things

  med0matrix <- matrix(med0,
                       nrow = n,
                       ncol = p,
                       byrow = TRUE)

  # In this case we use changes of signs in place of permutations
  signs.perm <- rbinom(n, 1, 0.5) * 2 - 1
  df_perm <- med0matrix + (as.numeric(df) - med0) * matrix(signs.perm,
                                                           nrow = n,
                                                           ncol = p,
                                                           byrow = FALSE)
  compute_t_stat(df_perm, med0matrix)
}

# parallel
n_cores <- detectCores()
cl = makeCluster(n_cores)
clusterExport(cl, varlist = list("symmetric_perm_wrapper", "df",
                                 "med0",
                                 "compute_t_stat","seed"))
set.seed(seed)
T2 <- pbreplicate(1e3, symmetric_perm_wrapper(df, med0), cl = cl)
stopCluster(cl)


diagnostic_permutation <- function(T20, T2) {
  B <- length(T2)
  # Compare real test statistic with the ones given by the permuted data
 # hist(T2, xlim = range(c(T2, T20)))
#  abline(v = T20, col = 3, lwd = 4)
  # Empirical cumulative distribution function
 # plot(ecdf(T2))
#  abline(v = T20, col = 3, lwd = 4)
  # P-value
  p_val <- sum(T2 >= T20) / B
  cat("p-value: ", p_val)
}
```

```
## p-value:  0
```

So the difference is significant! I can reject the null hypothesis Now I want to compute a bootstrap confidence interval on this difference An alternative approach is to use bootstrap on the differences

We now compute the bootstrap confidence interval on the differences

```r
#Utils
diagnostic_bootstrap <- function(distro, obs, alpha) {
  variance_pred <- var(distro)
  print(paste("Variance: ", variance_pred))
```

```r
  sd_pred <- sd(distro)
  print(paste("Standard deviation: ", sd_pred))
  bias_pred <- mean(distro) - obs
  print(paste("Bias: ", bias_pred))
  MSE_pred <- variance_pred + bias_pred^2
  print(paste("MSE: ", MSE_pred))
  # computing quantiles
  right.quantile <- quantile(distro, 1 - alpha / 2)
  left.quantile <- quantile(distro, alpha / 2)
  CI <- c(
    obs - (right.quantile - obs),
    obs,
    obs - (left.quantile - obs)
  )
  names(CI) <- c("lower", "center", "upper")
  print(CI)
  plot(ecdf(distro), main = "Bootstrap distribution")
  abline(v = CI[2], col = 3, lwd = 2)
  abline(v = CI[c(1, 3)], lty = 3)
}
```

```r
T0=median(df)
T0
```

```
## [1] 0.1764706
```

```r
compute_bootstrap_sample <- function(df) {
  permutation <- sample(1:nrow(df), replace = T)
  df.boot <- df[permutation, ]
  #Using the median
  med <- median(df.boot)
  return(med)
}

n_cores <- detectCores()
cl <- makeCluster(n_cores)
invisible(clusterEvalQ(cl, library(np)))


clusterExport(cl, varlist = list("df",
                                  "compute_bootstrap_sample",
                                  "seed"))
set.seed(seed)
T.boot <- pbreplicate(B, compute_bootstrap_sample(df), cl = cl)
stopCluster(cl)

myalpha=0.05
diagnostic_bootstrap(T.boot, T0, alpha = myalpha)
```
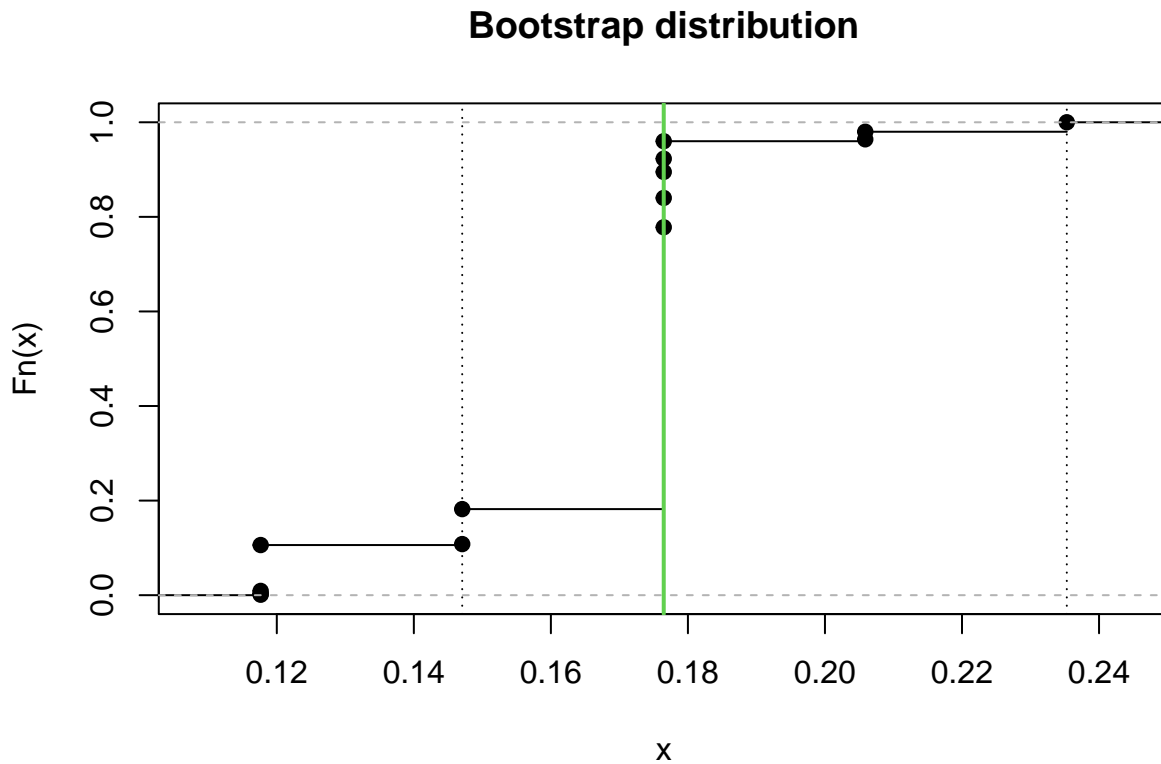
```
## [1] "Variance:  0.000474536820557582"
## [1] "Standard deviation:  0.0217838660608622"
## [1] "Bias:  -0.00670588235294114"
```

```
## [1] "MSE:  0.00051950567868907"
##    lower    center    upper
## 0.1470588 0.1764706 0.2352941
```

Bootstrap distribution

The bootstrap interval does no contain 0, so we can say that the median difference between MPOWER value at 2008 and at 2020 is greater than 0

There has been an increase of around 14.6-0.23 in the MPOWER compound score

Now we perform bootstrap on 2008

```
df<-as.numeric(mpower_table[2,-1])
T0=median(df)
T0
```
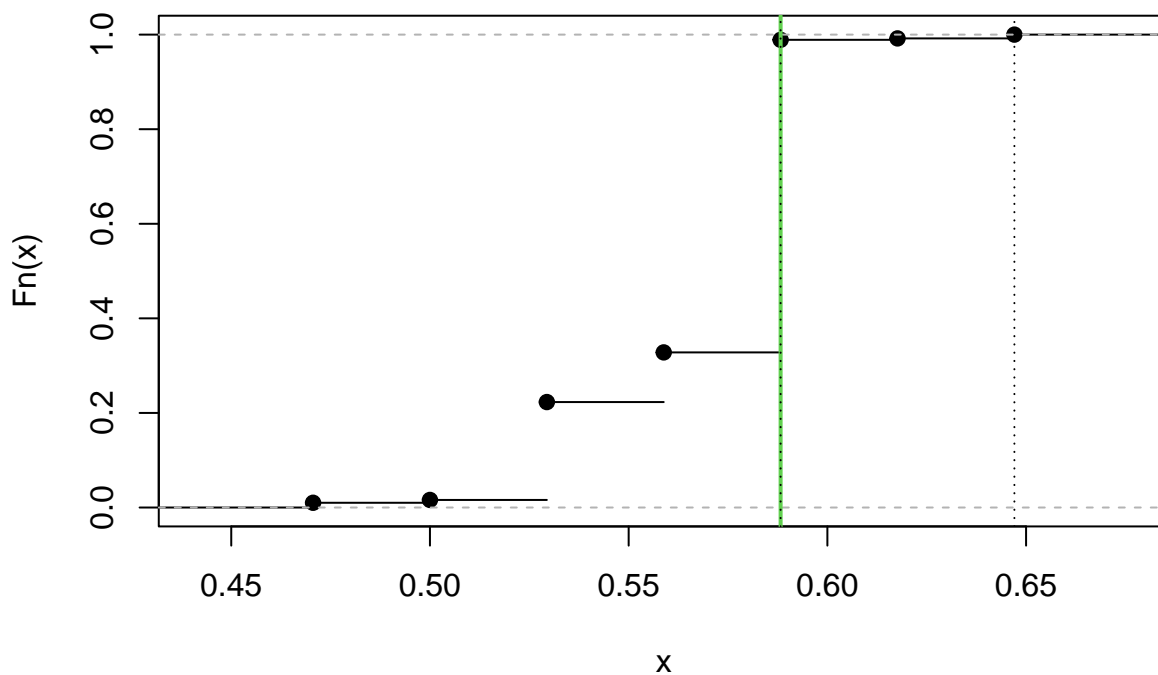
```
## [1] 0.5882353
```

```
compute_bootstrap_sample <- function(df) {
  permutation <- sample(1:length(df), replace = T)
  df.boot <- df[permutation]
  #Using the median
  med <- median(df.boot)
  return(med)
}
n_cores <- detectCores()
cl <- makeCluster(n_cores)
invisible(clusterEvalQ(cl, library(np)))
clusterExport(cl, varlist = list("df",
                                 "compute_bootstrap_sample",
                                 "seed"))
```

```
set.seed(seed)
T.boot <- pbreplicate(B, compute_bootstrap_sample(df), cl = cl)
stopCluster(cl)

myalpha=0.05
diagnostic_bootstrap(T.boot, T0, alpha = myalpha)
```

```
## [1] "Variance:  0.000753899227947662"
## [1] "Standard deviation:  0.0274572254233319"
## [1] "Bias:  -0.0164117647058823"
## [1] "MSE:  0.00102324524870891"
##     lower    center    upper
## 0.5882353 0.5882353 0.6470588
```

Bootstrap distribution

Bootstrap on 2020

```
df<-as.numeric(mpower_table[8,-1])
T0=median(df)
T0
```

```
## [1] 0.7647059
```

```
compute_bootstrap_sample <- function(df) {
  permutation <- sample(1:length(df), replace = T)
  df.boot <- df[permutation]
  #Using the median
  med <- median(df.boot)
  return(med)
```

```
}
n_cores <- detectCores()
cl <- makeCluster(n_cores)
invisible(clusterEvalQ(cl, library(np)))
clusterExport(cl, varlist = list("df",
                                 "compute_bootstrap_sample",
                                 "seed"))
set.seed(seed)
T.boot <- pbreplicate(B, compute_bootstrap_sample(df), cl = cl)
stopCluster(cl)

myalpha=0.05
diagnostic_bootstrap(T.boot, T0, alpha = myalpha)
```
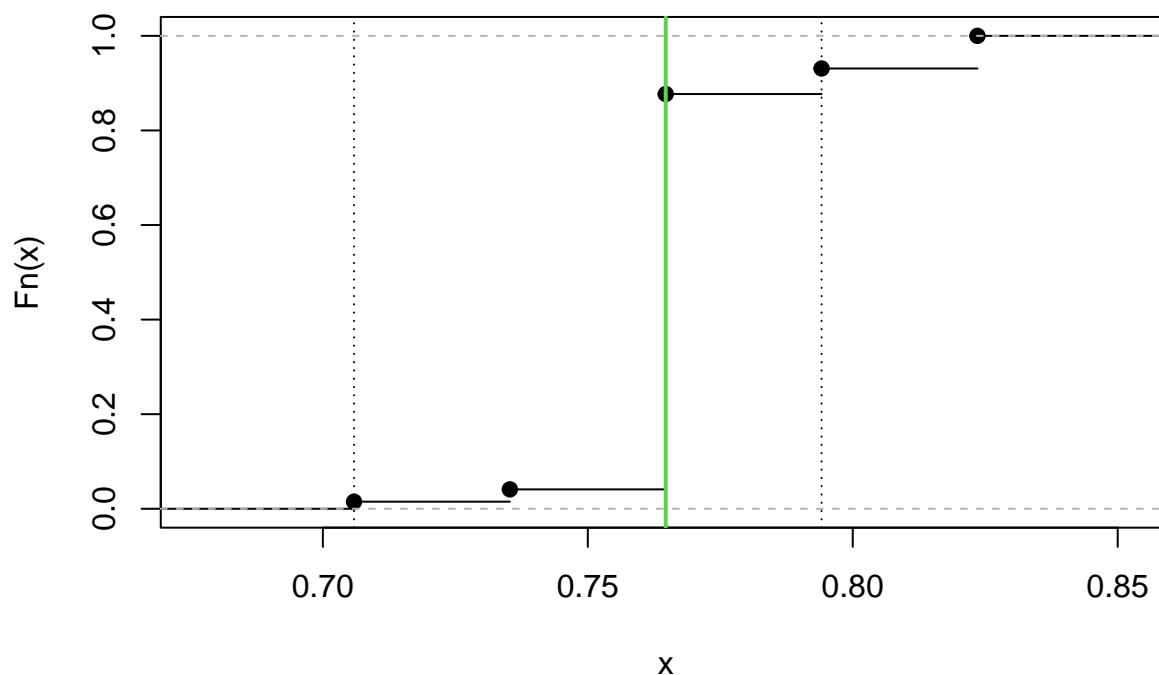
```
## [1] "Variance:  0.000344205797492994"
## [1] "Standard deviation:  0.0185527840900764"
## [1] "Bias:  0.004"
## [1] "MSE:  0.000360205797492994"
##     lower    center     upper
## 0.7058824 0.7647059 0.7941176
```

Bootstrap distribution
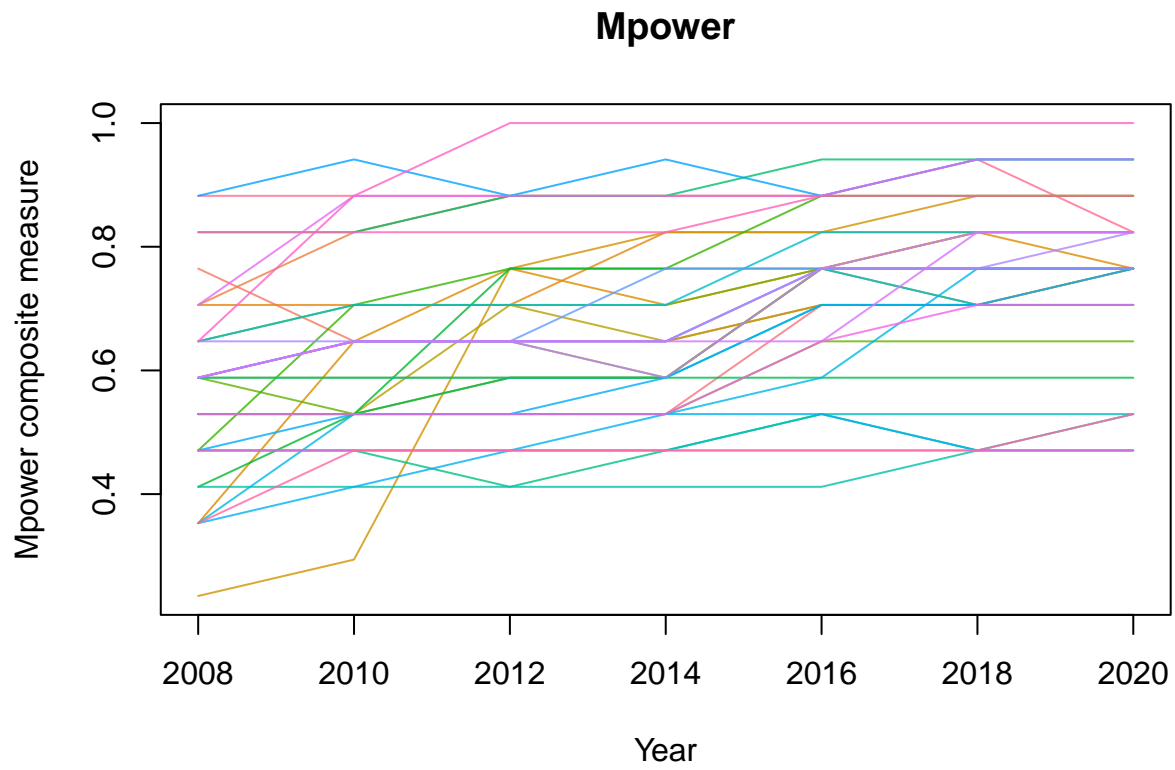

Can we move to a functional setting?

```
mpower_table_f<-as.data.frame(mpower_table[-1,-1])

grid <- seq(2008, 2020, by=2)

#Create functional data
f_data <- roahd::fData(grid,t(mpower_table_f))
```
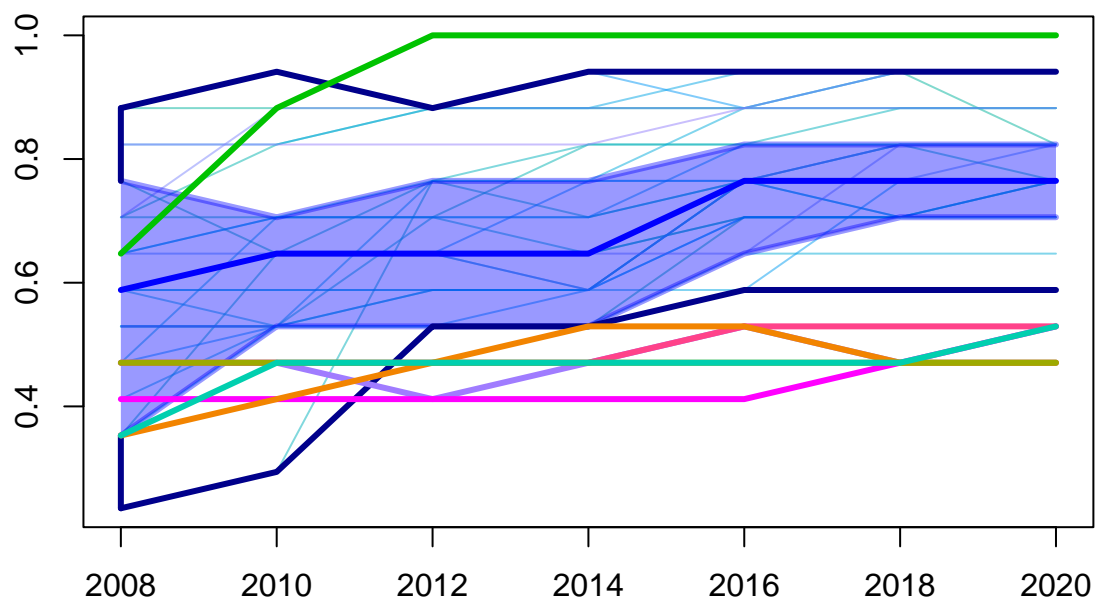
```
plot(f_data,
     xlab="Year",
     ylab="Mpower composite measure",
     main="Mpower")
```

Mpower


```
fb_plot = roahd::fbplot(f_data, main="Magnitude outliers - MBD", Depths = "MBD")  # Functional Box Plot
```

Magnitude outliers – MBD


```
fb_plot$ID_outliers
```

```
##        Israel         Japan          Korea         Mexico   Switzerland
##            18            20             21             25            35
##       Türkiye United States
##            36            38
```

```
MBD_f<-as.data.frame(t(MBD(f_data)))
MBD_f[fb_plot$ID_outliers]
```

```
##      Israel    Japan    Korea   Mexico Switzerland   Türkiye United States
## 1 0.2031599 0.143416 0.261075 0.1958443   0.2072749 0.1199451      0.193101
```

The following seem to have increased less than others their MPOWER composite measure

```
mpower_table$Israel
```

```
## [1] 0.4117647 0.4705882 0.4705882 0.4117647 0.4705882 0.5294118 0.4705882
## [8] 0.4705882
```

```
mpower_table$Japan
```

```
## [1] 0.4117647 0.4117647 0.4117647 0.4117647 0.4117647 0.4117647 0.4705882
## [8] 0.5294118
```

```
mpower_table$Korea
```

```
## [1] 0.4705882 0.4705882 0.4705882 0.4705882 0.4705882 0.5294118 0.5294118
## [8] 0.5294118
```

```
mpower_table$Mexico
```

```
## [1] 0.2352941 0.3529412 0.4117647 0.4705882 0.5294118 0.5294118 0.4705882
## [8] 0.4705882
```

```
mpower_table$Switzerland
```

```
## [1] 0.4705882 0.4705882 0.4705882 0.4705882 0.4705882 0.4705882 0.4705882
## [8] 0.4705882
```

```
mpower_table$`United States`
```

```
## [1] 0.2352941 0.3529412 0.4705882 0.4705882 0.4705882 0.4705882 0.4705882
## [8] 0.5294118
```

Turkiye has had a very steep increase

```
mpower_table$Türkiye
```

```
## [1] 0.3529412 0.6470588 0.8823529 1.0000000 1.0000000 1.0000000 1.0000000
## [8] 1.0000000
```

Frow WHO: "In Turkey, the adoption of MPOWER measures in 2008 led to a significant reduction in smoking. By 2012 data showed that smoking decreased by 13.4% and exposure to second-hand smoke was on the decline too. Turkey became the third country in Europe to go 100% smoke-free indoors and the first country to achieve all six MPOWER measures at the highest level.

The MPOWER measures have been life-changing for Turkey's people, reducing the likelihood of heart disease, lung cancer, and other chronic diseases. Progress is evident, with studies showing a 20% decline in 2012 in the number of citizens admitted to hospital for smoking-related diseases."