Multidimensional scaling (MDS)

Enviroment Setup and Import data files

```
library(cluster)
library(MASS)
library(smacof)
library(magrittr)
library(dplyr)
library(ggpubr)
library(psych)statistics  <-  read.csv("UN_Statistics.csv")
X <- as.matrix(statistics[,-1])
rownames(X) <- statistics[,1]
```

#data exploration

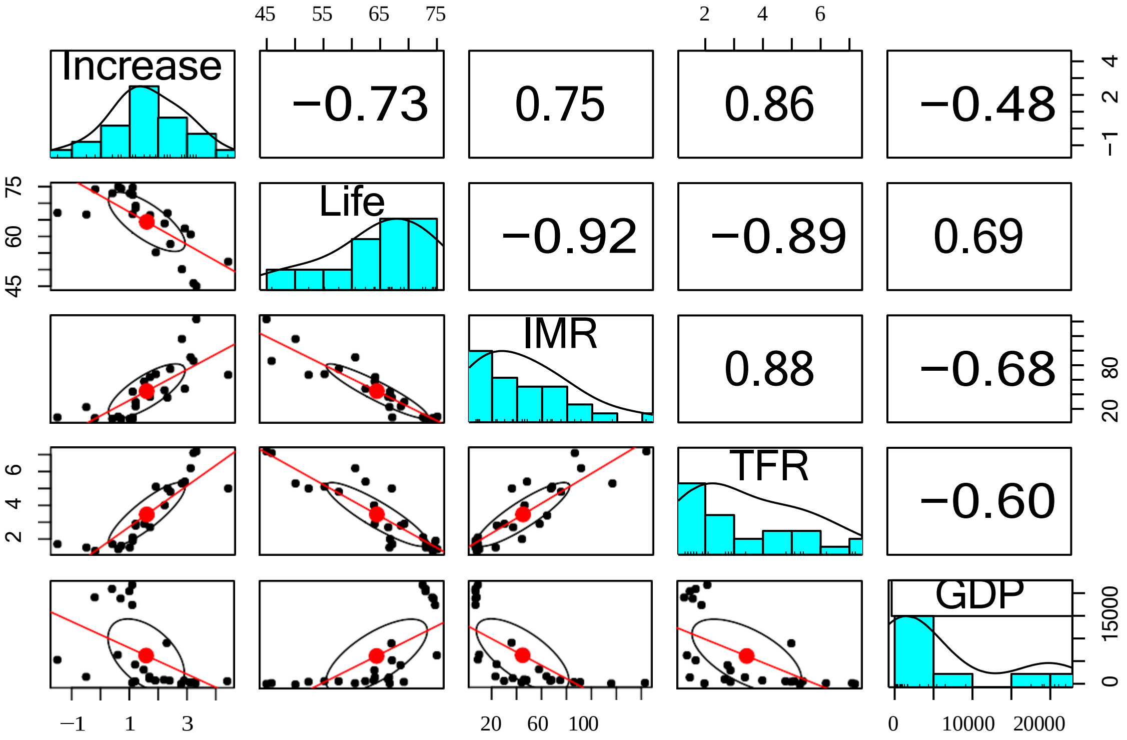
```
rownames(X)
```

```
##   [1]  "Albania"            "Argentina"          "Australia"
##   [4]  "Austria"            "Benin"              "Boliva"
##   [7]  "Brazil"             "Cambodia"           "China"
##  [10]  "Colombia"           "Croatia"            "El  Salvador"
##  [13]  "France"             "Greece"             "Guatemala"
##  [16]  "Iran"               "Italy"              "Malawi"
##  [19]  "Netherlands"        "Pakistan"           "Papua  new  Guinea"
##  [22]  "Peru"               "Romania"            "USA"
##  [25]  "Zimbabwe"
```

```
sapply(statistics,class)
```

```
##     Country   Increase            Life           IMR          TFR
     GDP ##       "factor"  "numeric"  "numeric"  "integer"   "numeric"
"numeric"
```

```
pairs.panels(statistics[,-1],cex=1,lm=TRUE)
```

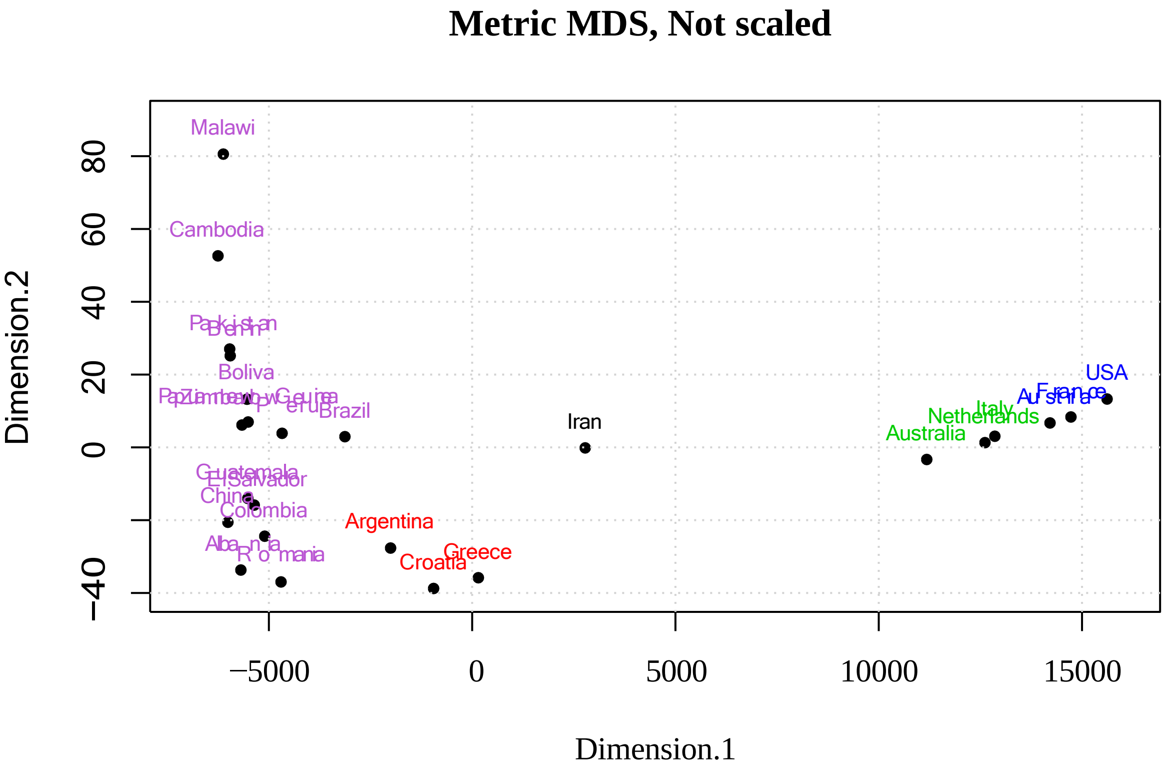


Classical MDS Scaling

First let’s plot the data before we normalize it.

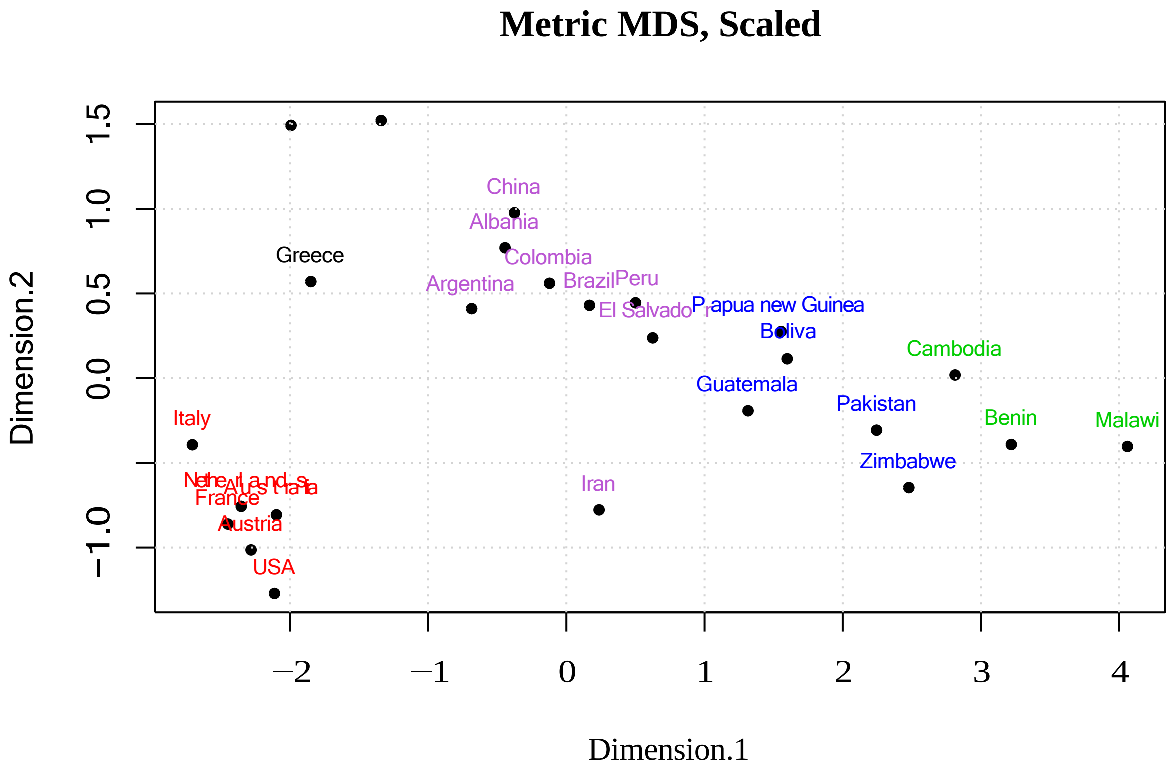
```r
d <- dist(X) # euclidean distances between the rows

#preform clustering so we can allocate colours to the plot
cluster <- hclust(d,method="complete")
clusvec <- cutree(cluster, k=5)

scaledDistances <- cmdscale(d) # preform the multidimensional scalling

# create empty plot and then add text and colours. Colours added based on the clustor groups
plot (scaledDistances, xlab="Dimension.1", ylab="Dimension.2",
   main="Metric MDS, Not scaled", type="p", pch=20,ylim = c(-40,90), xlim = c(-7000,16000))
grid()
#ensure you list enough colours for the number of clusters
colvec <- c("mediumorchid",
            "red",
            "green3",
            "blue",
            "black",
            "gold",
            "indianred",
            "moccasin",
            "lightcyan",
            "skyblue")
```

```
for (i in 1:length(scaledDistances[,1]))
  text (scaledDistances[i,1],
        scaledDistances[i,2],
        rownames(X)[i],
        col=colvec[clusvec[i]],
        cex=0.7,
        pos = 3)
```



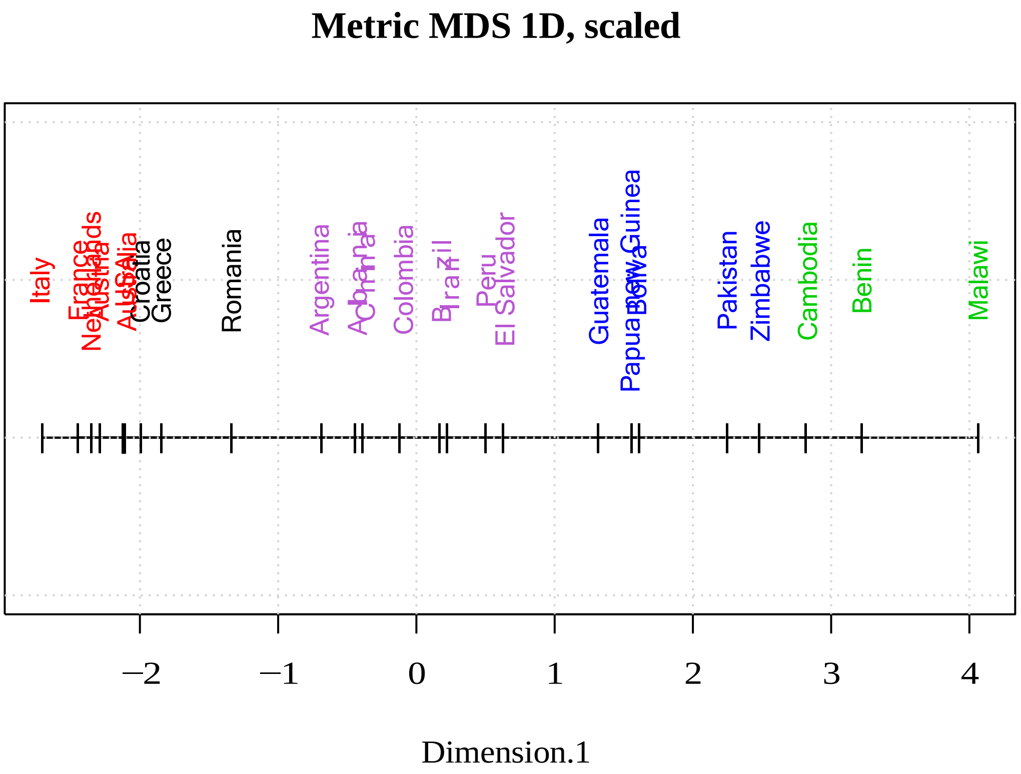
#Normalized MDS Next we can repeat the process but this time normalize the data scaling

```
# euclidean distances between the rows
d <- dist(scale(X, center=TRUE, scale=TRUE), method="euclidean")

#preform clustering so we can allocate colours to the plot
cluster <- hclust(d,method="complete")
clusvec <- cutree(cluster, k=5)

scaledDistances <- cmdscale(d, k = 2) # preform the multidimensional scalling

# create empty plot and then add text and colours. Colours added based on the clustor groups
plot (scaledDistances, xlab="Dimension.1", ylab="Dimension.2",
  main="Metric MDS, Scaled", type="p", pch=20)
grid()
for (i in 1:length(scaledDistances[,1]))
  text (scaledDistances[i,1],
        scaledDistances[i,2],
        rownames(X)[i],
        col=colvec[clusvec[i]],
        cex=0.7,
        pos = 3)
```



#One Dimentional Plot Generate a one dimentional representation of the data

```
d <- dist(scale(X)) # euclidean distances between the rows

#preform clustering so we can allocate colours to the plot
cluster <- hclust(d,method="complete")
clusvec <- cutree(cluster, k=5)


scaledDistances <- cmdscale(d, eig=TRUE, k=1) # preform the multidimensional scalling

x <- data.frame(scaledDistances$points[,1],1)

# create empty plot and then add text and colours. Colours added based on the clustor groups
plot (x, xlab="Dimension.1",
  main="Metric MDS 1D, scaled",
  type = 'o',
  pch = '|',
  ylab = '',
  yaxt='n',
  xlim=c(min(x),
           max(x)),
  ylim=c(0.95,1.1))
grid()

for (i in 1:length(x[,1]))
  text (x[i,1],
          1.05,
          rownames(X)[i],
          col=colvec[clusvec[i]],
          cex=0.8,
          srt=90)
```

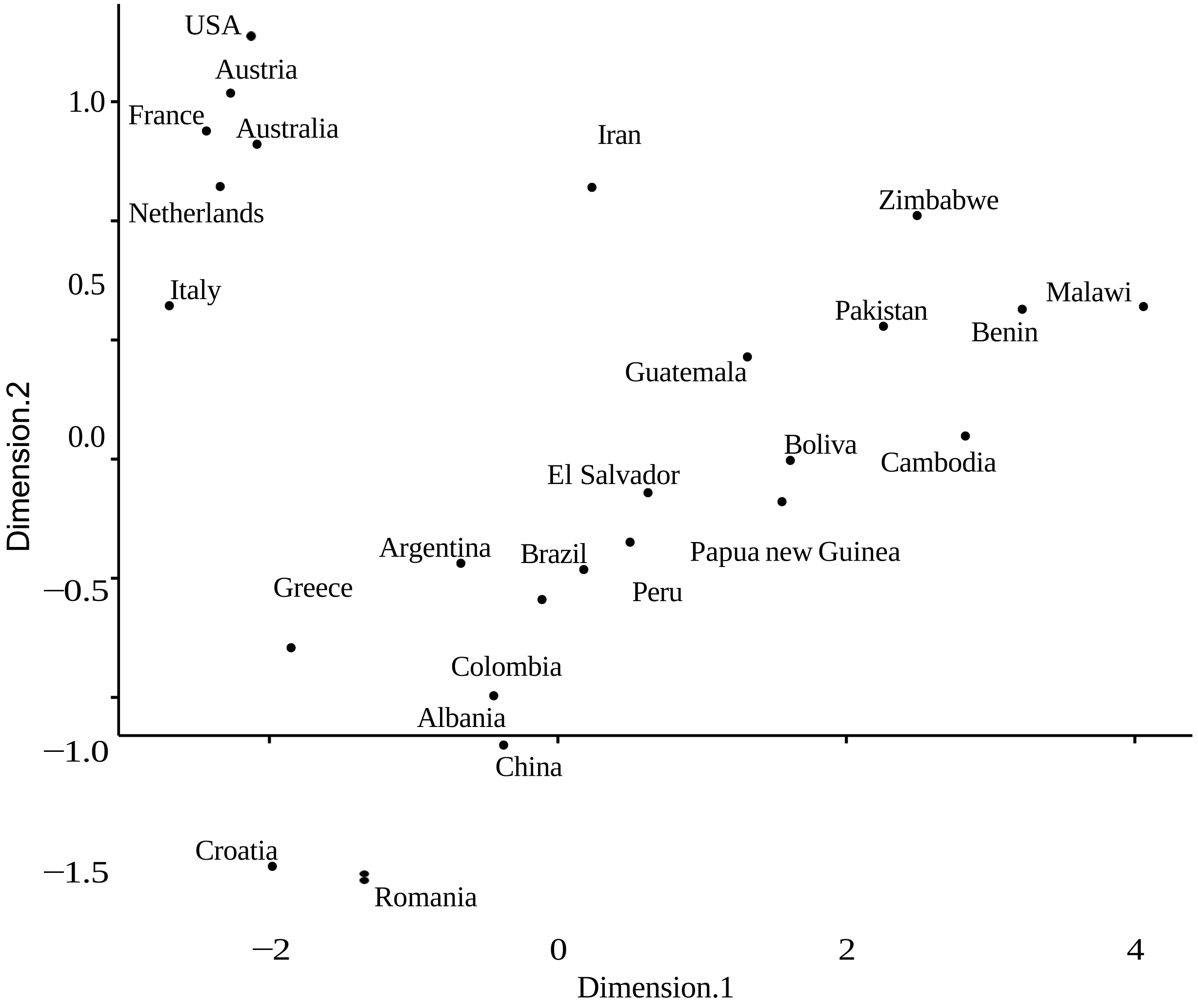


Classical MDS

```
# Cmpute MDS
mds <- scale(X) %>%
  dist() %>%
  cmdscale() %>%
  as_tibble()
```

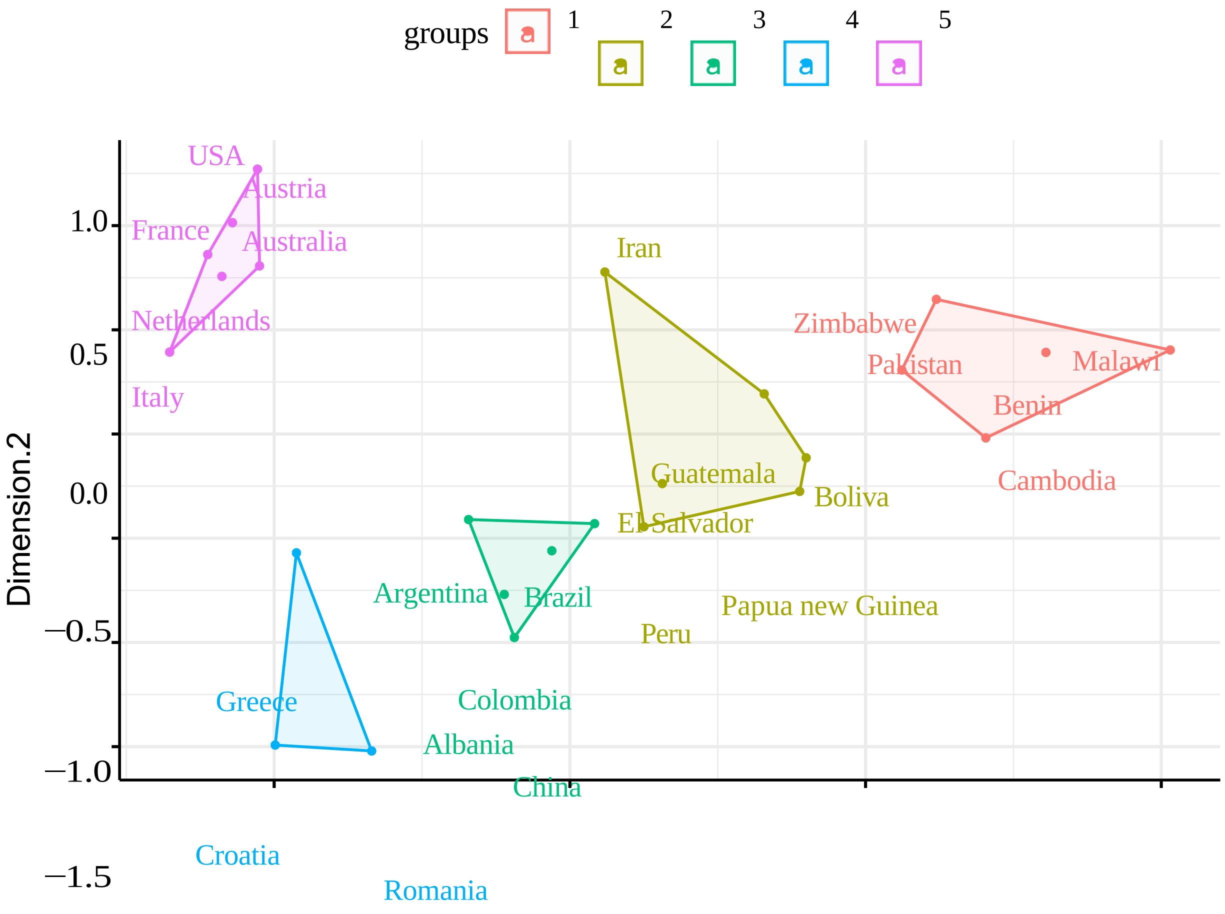
```
## Warning: `as_tibble.matrix()` requires a matrix with column names or a `.name_repair` argument. Usin
## This warning is displayed once per session.
```

```
colnames(mds) <- c("Dimension.1", "Dimension.2")
# Plot MDS
mds[,2] <- -1 * mds[,2]
ggscatter(mds, x = "Dimension.1", y = "Dimension.2",
          label = rownames(X),
          size = 1,
          repel = TRUE)
```

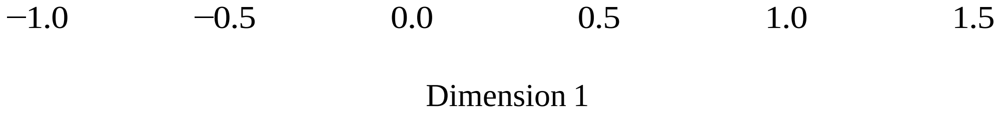


We can add colours to the plots while clustering them together

```
# K-means clustering
clust <- kmeans(mds, 5)$cluster %>%
  as.factor()
mds <- mds %>%
  mutate(groups = clust)
# Plot and color by groups
ggscatter(mds, x = "Dimension.1", y = "Dimension.2",
          label = rownames(X),
          color = "groups",
          palette = "pal3",
          size = 1,
          ellipse = TRUE,
          ellipse.type = "convex",
          repel = TRUE) +
grids(axis = c("xy", "x", "y"), color = "grey92", size = NULL,
  linetype = NULL)
```


groups
1
2
3
4
5
Dimension.2
1.0
0.5
0.0
−0.5
−1.0
−1.5
USA
Austria
France
Australia
Netherlands
Italy
Iran
Zimbabwe
Pakistan
Malawi
Benin
Cambodia
Guatemala
Boliva
El Salvador
Argentina
Brazil
Papua new Guinea
Peru
Greece
Colombia
Albania
China
Croatia
Romania

−2
0
2
4
Dimension.1


−1.0
−0.5
0.0
0.5
1.0
1.5
Dimension 1