clustering

Ata COLAK

2023-07-28

Clustering

Introduction

This report demonstrates a clustering analysis on the dataset “ssi_2016_indicators”.

The code is divided into several parts:

Part 1 involves importing the dataset and displaying the first few rows to ensure successful reading of the data.

Part 2 applies the K-means clustering algorithm with 4 clusters to the dataset.

Part 3 plots the clusters on a world map using the “rworldmap” library. The map provides a visual representation of the distribution of the clusters, indicating which countries are grouped together based on similarity.

Part 4 generates a scatterplot matrix to visualize the relationships between the variables in the dataset. Each data point is represented by a shape and color corresponding to its cluster, and a legend is added to provide a key for interpreting the scatterplot.

Part 5 computes the average values of the three dimensions, education, income distribution and GDP, within each cluster and displays the results.

Part 6 performs computations using the complete linkage clustering algorithm. It calculates the distance matrix, performs complete linkage and generates a map and scatterplot matrix for the clusters. Finally, it computes the average values of the dimensions within each cluster.

The dataset can be accessed using the link below:

https://raw.githubusercontent.com/gagolews/teaching-data/master/marek/ssi_2016_indicators.csv

Part 1 - Importing the Dataset

In the first chunk below, the csv file is loaded and read by my device, and head of the dataset is printed to show that they have been read successfully.

```
ssi <- read.csv(paste0("https://raw.githubusercontent.com/gagolews/teaching-data/master/marek/ssi_2016_
comment.char="#")
ssi <- ssi[, c("Country", "Education",
"IncomeDistribution", "GDP")]
head(ssi, 2)
```

```
##   Country Education IncomeDistribution      GDP
## 1 Albania  8.844812           8.381965 5.462496
## 2 Algeria  8.096907           5.545405 6.292930
```

Part 2 - Applying the K-means Algorithm

> Second part below sets the number of clusters (k) to 4 for the K-means algorithm, and it the seed value to 123 to make sure that clustering results remain consistent across different runs.
>
> kmeans() function is used on the dataset, excluding the first column, and k-means clustering is done with 4 clusters. The nstart parameter controls the number of random starts for the algorithm.

```
k <- 4
set.seed(123)
clusters <- kmeans(ssi[, -1], centers = k, nstart = 100)$cluster
head(clusters)
```

```
## [1] 4 4 1 2 3 4
```

#Part 3 - Plotting of Clustering on a World Map

> Third part below plots the clusters on a world map, using “rworldmap” library.

```
library(rworldmap)
```

```
## Loading required package: sp
```

```
## The legacy packages maptools, rgdal, and rgeos, underpinning the sp package,
## which was just loaded, will retire in October 2023.
## Please refer to R-spatial evolution reports for details, especially
## https://r-spatial.org/r/2023/05/15/evolution4.html.
## It may be desirable to make the sf package available;
## package maintainers should consider adding sf to Suggests:.
## The sp package is now running under evolution status 2
##      (status 2 uses the sf package in place of rgdal)
```

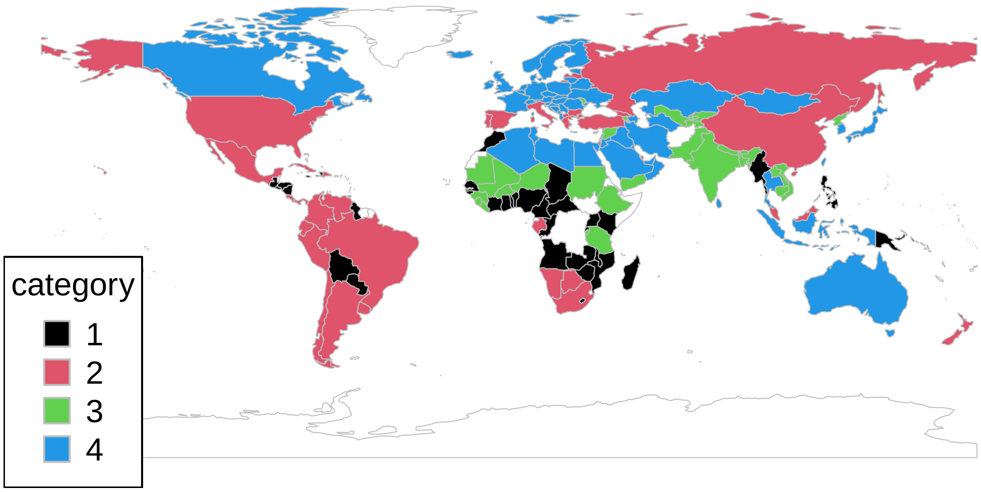
```
## Please note that 'maptools' will be retired during October 2023,
## plan transition at your earliest convenience (see
## https://r-spatial.org/r/2023/05/15/evolution4.html and earlier blogs
## for guidance);some functionality will be moved to 'sp'.
##  Checking rgeos availability: FALSE
## ### Welcome to rworldmap ###
```

```
## For a short introduction type :   vignette('rworldmap')
```

```
mapdata <- data.frame(Country=ssi[["Country"]], Cluster=clusters)
mapdata <- joinCountryData2Map(mapdata, joinCode="NAME",
nameJoinColumn="Country")
```

```
## 153 codes from your data successfully matched countries in the map
## 1 codes from your data failed to match with a country code in the map
## 90 codes from the map weren't represented in your data
```

```
mapCountryData(mapdata, nameColumnToPlot="Cluster",
catMethod="categorical", missingCountryCol="white",
colourPalette=palette.colors(k, "R4"), mapTitle="")
```

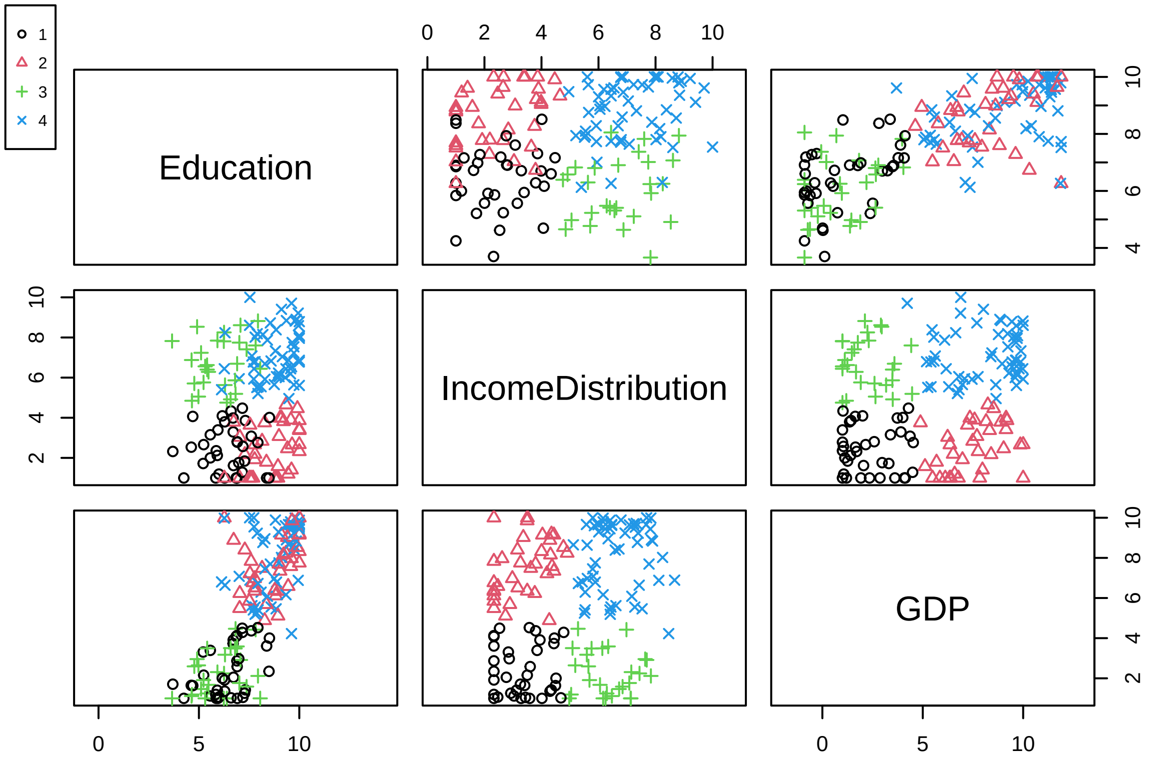


> The map provides a visual representation of the distribution of the clusters, showing which countries are grouped together on similar patterns.

Part 4 - Scatterplot Matrix of Data

Fourth part below focuses on generating a scatterplot matrix in order to visualize the relationship in the dataset. The first line creates a a scatterplot, and second line creates the legend for the scatterplot.

```
pairs(ssi[-1], col=clusters, pch=clusters, asp=1)
legend("topleft", col=1:k, pch=1:k, legend=1:k, xpd=NA, bg="white", cex = 0.5)
```



In the scatterplot above, every shape or color represents one cluster. We can see that there is a slight correlation between education and GDP, other than that, there is close to none relationships between variables.

Part 5 - Average Computation of the Three Dimensions in Each Cluster

Fifth part below calculates the average values of the three dimensions within each cluster in the dataset and displays the results.

```
ave_dim <- sapply(split(ssi[, -1], clusters), colMeans)
print(ave_dim)
```

```
##                           1        2        3        4
## Education          6.453550 8.616167 6.038627 8.793434
## IncomeDistribution 2.441276 2.532283 6.714405 7.152726
## GDP                2.406336 7.517458 2.254238 8.091242
```

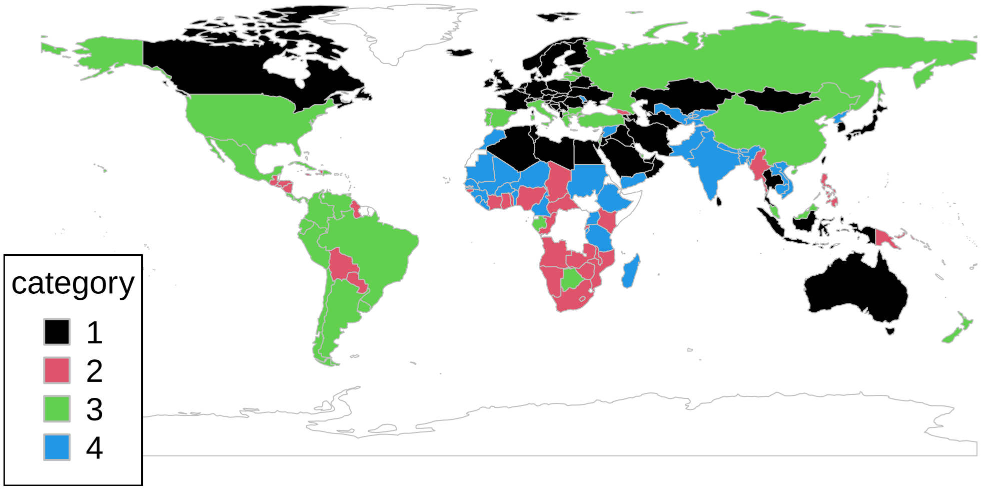
Part 6 - Map, Scatterplot Matrix and Cluster Centers Computations to a Complete Linkage Cluster

Sixth part below performs computations to a complete linkage clustering algorithm such as mapping the clusters, creating a scatterplot matrix, and calculating the average values of the dimensions in each cluster.

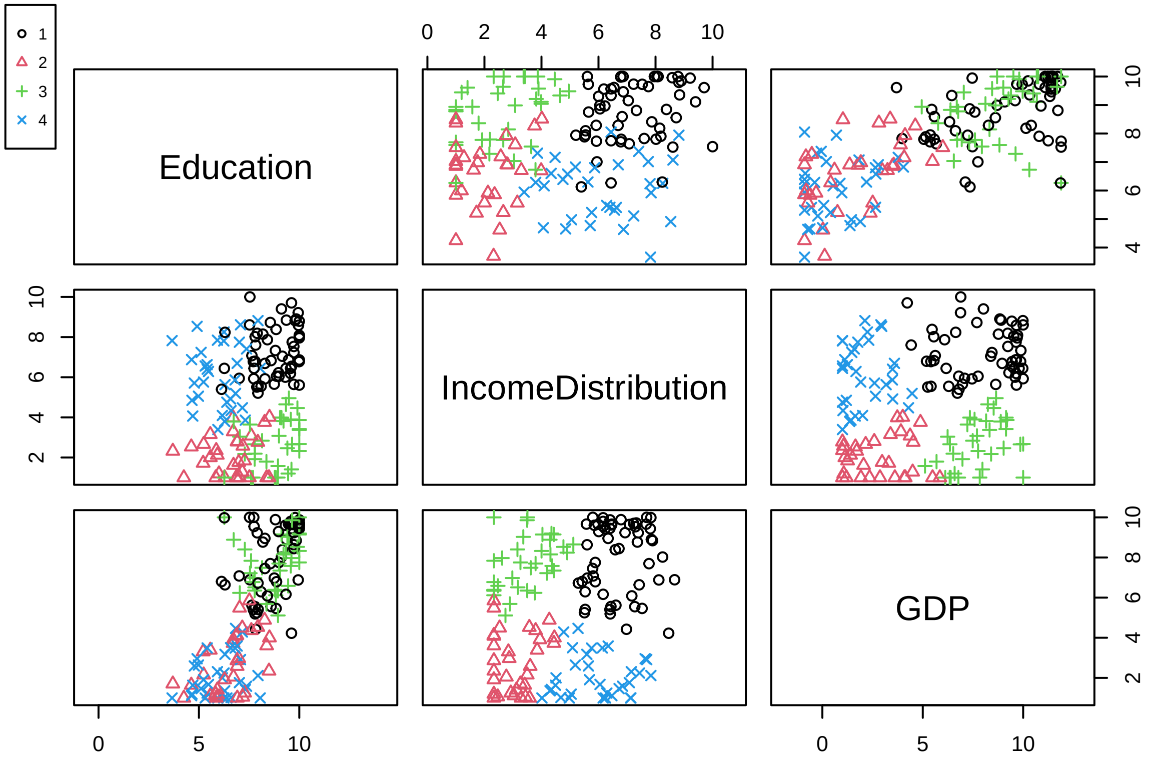
```
dist_matrix <- dist(ssi[, -1])
hc <- hclust(dist_matrix, method = "complete")
clusters <- cutree(hc, k)

mapdata <- data.frame(Country = ssi[["Country"]], Cluster = clusters)
mapdata <- joinCountryData2Map(mapdata, joinCode = "NAME", nameJoinColumn = "Country")
```

```
## 153 codes from your data successfully matched countries in the map
## 1 codes from your data failed to match with a country code in the map
## 90 codes from the map weren't represented in your data
```

```
mapCountryData(mapdata,
nameColumnToPlot="Cluster",
catMethod="categorical",
missingCountryCol="white",
colourPalette=palette.colors(k,"R4"),
mapTitle="")
```



```
pairs(ssi[-1], col = clusters, pch = clusters, asp = 1)
legend("topleft", col=1:k, pch=1:k, legend=1:k, xpd=NA, bg="white", cex = 0.5)
```



```
ave_dim <- sapply(split(ssi[, -1], clusters), colMeans)
print(ave_dim)
```

```
##                            1        2        3        4
## Education           8.763851 6.593011 8.734436 6.041834
## IncomeDistribution 7.199984 2.050633 2.661461 6.112421
## GDP                 8.015755 2.816098 7.743619 2.096251
```

As we can see in the scatterplot of this report, the correlation between education and GDP is slightly more correlated, and we can see different clusters for each country in the world map, with each country categorized by color shown in the legend of the map. We can also observe the difference in education, income distribution and GDP average values of 3 dimensions compared to the previous part.

Conclusion

In conclusion, the analysis on the dataset showed which countries have similar rankings based on education, income distribution and GDP. This analysis has shown usage of cluster center computations, scatterplots and plotting clusters on a world map.