Unsupervised learning project

Machine learning, statistical learning, deep learning and artificial intelligence

Martina Corsini ID 944506

MSc Data science and economics
Università degli studi di Milano

July 2021

Abstract: In recent years, there has been much discussion about the possibility of a multi-speed Europe, namely the idea of a differentiated integration that could take into account the diversity that European countries show. Therefore, it is crucial to understand how we can group the European countries according to some variables, which are not related to the geographic location, but to their socio-demographic and economic situation. In order to perform this task, the research exploits two different unsupervised algorithms, the hierarchical clustering and the k-means clustering. Finally, the PCA method is used in order to understand how the variables relate to each other and to have in conclusion a visual representation of the data in a two dimensional space, onto which most of the variance is captured.

Keywords: Hierachical clustering, K-means clustering, PCA

Contents

List of Figures

List of Tables

1 Introduction

Over the decades, the number of Member States of the European Union has increased from 12 to 28, with a rise of the heterogeneity in internal conditions and in the economic parameters. For this reason, there has been much discussion about the possibility of a multi-speed Europe, namely the idea of a differentiated integration that could take into account the diversity that European countries show [1]. Therefore, it is crucial to understand how we can group the European countries according to some variables, which are not related to the geographic location, but to their socio-demographic and economic situation. In order to achieve this goal, I will use different unsupervised methods: two clustering algorithms, the hierarchical clustering and the k-means clustering, will be exploited so to show the similarity between some states; then, the PCA method will help us to understand the direction of the variables considered and the relationship between them and to have in conclusion a visual representation of the data in a two dimensional space.

2 Theoretical background [2]

2.1 Clustering methods

Clustering methods are a set of unsupervised learning techniques that aims to find homogeneous subgroups among the whole set of data points according to a similarity criteria. The goal is therefore to identify distinct clusters so that the variance within is minimized while variance between is maximized. In this research, I will apply the two best-known clustering approaches: K-Means clustering and hierarchical clustering. These unsupervised learning techniques are highly affected by the unit of measures of the data. Data must be therefore standardized in order to make them comparable.

2.1.1 Hierarchical clustering

In the hierarchical clustering approach it is not necessary to specify the number of clusters in advance. In fact, it will result in a dendrogram, that allows us to visualize at once the clustering obtained for each possible number of clusters. The similarity of two observations can be retrieved looking at the point on the y-axis of the dendrogram where branches containing those two observations first are fused: observations that are grouped together at a lower level tend to be more similar than others grouped at a higher one. It is suggested to apply this kind of cluster analysis using several distance measures and linkage methods in order to check for the robustness of the analysis. In the Table 1, we can see a summary of the types of linkage I used in my analysis.

Linkage	Description
Complete	Distance is based on similarity of the furthest pair of data points
Average	Distance is equal to the average distance from any member of one cluster to any member of the other cluster
Single	Distance is based on similarity of the closest pair of data points
Ward	Minimize the total within-cluster variance

Table 1: Linkages

With respect to the distance, I chose to use two type of distances, the euclidean distance, $d(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$, and the Manhattan distance, $d(p,q) = \sqrt{\sum_{i=1}^{n} |q_i - p_i|}$. We can use the Manhattan distance instead of the Euclidean one,the default one in this kind of analysis, because it deals better the presence of some extreme observations.

2.1.2 K-means clustering

In K-means clustering method, we seek to partition the observations in a pre-specified number of non-overlapping clusters. The aim of this algorithm is to minimize the total within-cluster variation. Note that the algorithm finds a local optimum, so it is important to run the algorithm multiple times from different random initial configuration and then select the best solution. In order to find the optimal number of clusters for this algorithm, the context of the problem at hand could be considered, for instance if you know that there is a specific number of groups in your data. However, some approaches could be used in order to automatically identify the number of clusters. I used two approaches: the elbow method, that looks at the total within-cluster sum of square (WSS) as a function of the number of clusters and the silhouette method, that measures the quality of a clustering and determines how well each point lies within its cluster [3].

2.2 PCA

Principal Component Analysis could be used to produce a low-dimensional representation of the data in a way that captures as much of the information as possible. This method finds a sequence of linear normalized combinations of variables that show maximal variance and that are mutually correlated. The amount of variance explained by each principal component could be visualized through the scree plot. This tool could be exploited for the choice of the optimal number of principal components to use, looking for an "elbow" in the plot, namely a point in which the subsequent variance explained drops off. With PCA, variables can be grouped in new derived variables, so to understand how the variable relate to each other. In order to achieve the latter goal, it could be useful to look also to the correlation matrix. Furthermore, to have a visual interpretation of the original n-datapoints in a two-dimensional space, we could project them on the first two principal components previously found.

3 Dataset

The European Social Survey (ESS) is an academically driven cross-national survey that has been conducted across Europe since its establishment in 2001[4]. Every two years, face-to-face interviews are conducted with newly selected, cross-sectional samples.We have therefore 9 rounds until now. I decided not to split the data into rounds because otherwise I should have to eliminate some countries from my analysis, since in some years questionnaires of some countries didn't contain the questions that I chose as variables. I am aware that this choice could lead to bias due to the fact that in the years the conditions have changed, but nevertheless I decided to operate this choice for sake of completeness. The problem could be solved integrating the data with national sources or performing a complete interview in each of the considered countries. In my analysis, I selected several variables from the ESS and I also added some new variables in order to make the analysis more complete. The following schema explains which variables I used in my research.

Variable	Description	Source
trstprl	Trust in country's parliament	ESS
trstlgl	Trust in the legal system	ESS
trstplc	Trust in the police	ESS
trstplt	Trust in politicians	ESS
trstprt	Trust in political parties	ESS
trstep	Trust in the European Parliament	ESS
trstun	Trust in the United Nations	ESS
relig	Level of religiosity	ESS
income	Income scale	ESS
freedom	Total freedom score	Freedom House [5]
density	Density pop/km^2	Index mundi [6]
TAI	Technology achievement index	Desai et al. [7]

Table 2: Variables

The last step in the creation of my dataset is the process of grouping by countries. In this way I obtained a dataset with countries as cases and the previous selected measurements as variables. The countries that I took into consideration are: Austria, Belgium, Switzerland, Germany, Denmark, Spain, Finland, France, UK, Hungary, Ireland, Italy, Netherlands, Norway, Poland, Portugal, Sweden, Slovenia.

4 Findings

4.1 Clustering

4.1.1 Hierarchical clustering

First of all, I proceeded with the creation of groups of states that show similar patterns regarding to the selected variables. The first method I used was the hierarchical clustering. I did it on my standardized variables considering two distance measures (Euclidean and Manhattan distances) and four agglomeration methods (complete, average, single, Ward linkages) in order to check for the robustness of my cluster analysis. What I found is represented in the Fig 1 - 8. The results don't change so much, so I could conclude that the clusters are quite well defined: Norway, Finland, Denmark and Sweden make up an homogeneous group of states with all the linkage methods and distance measures, as Belgium, Switzerland and Netherlands; Austria, Germany and UK are always considered as quite similar to each other; the remaining countries are often grouped in one cluster. Note that Hungary is sometimes put in a separate group using the Euclidean distance, probably due to the fact that it is an outlier in the Europe scenario concerning to the level of freedom and this kind of distance does not perform very well in the presence of extreme observations. In Fig. 9 we can see the mean of each variable for each cluster. It could be useful for understanding how the clustering performed on our data.

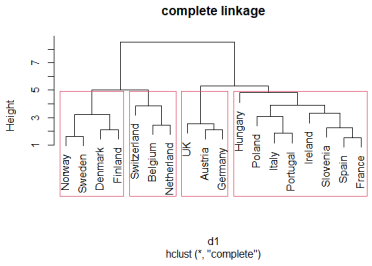

Figure 1: Complete, Euclidean

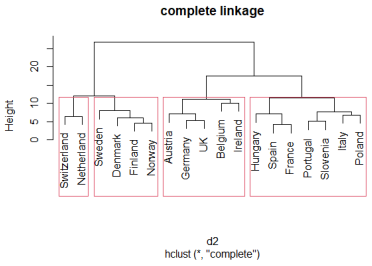

Figure 2: Complete, Manhattan

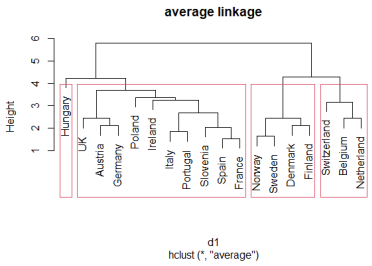

Figure 3: Average, Euclidean

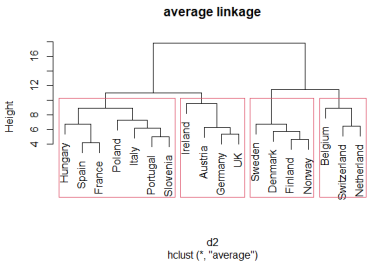

Figure 4: Average, Manhattan

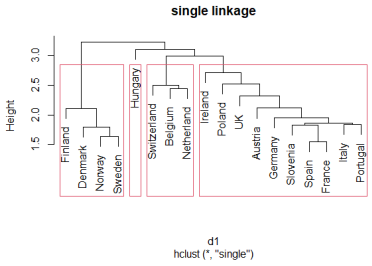

Figure 5: Single, Euclidean

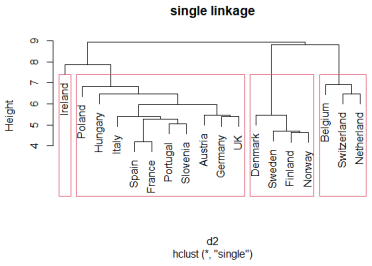

Figure 6: Single, Manhattan

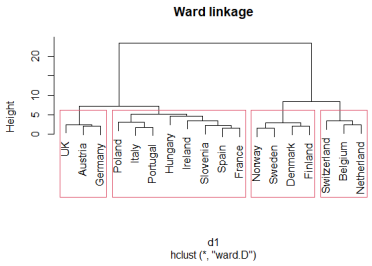

Figure 7: Ward, Euclidean

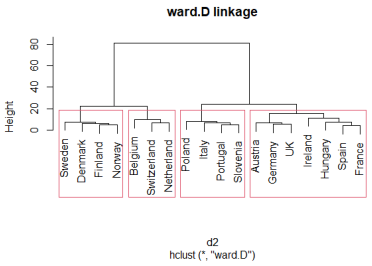

Figure 8: Ward, Manhattan

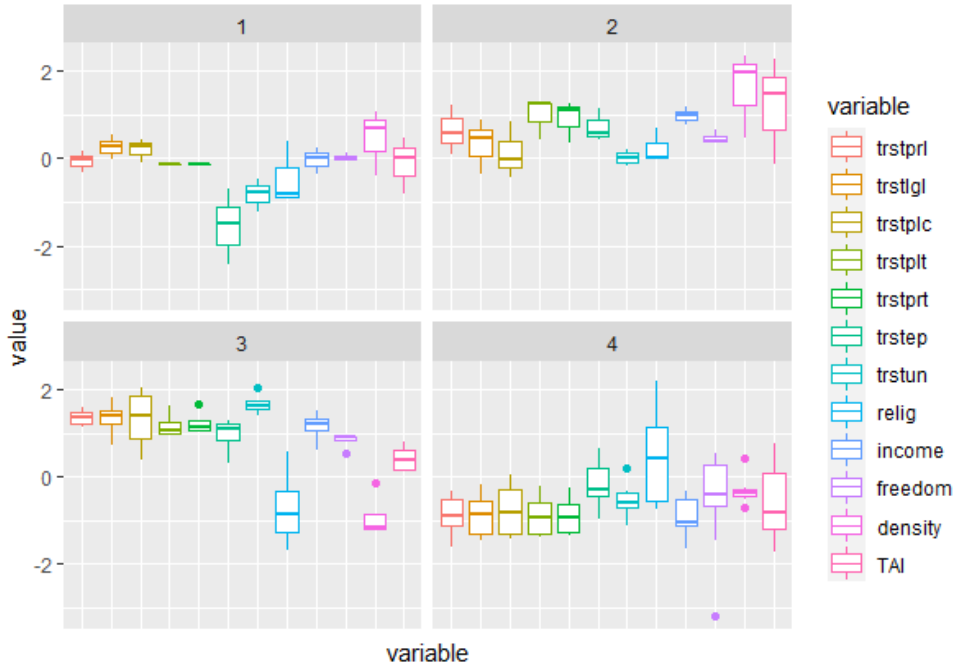

Figure 9: Hierarchical cluster interpretation

4.1.2 K-means clustering

The second method I used for the identifications of homogeneous group is the K-means cluster. I used the Silhouette method (Fig 10) and the Elbow method (Fig 11) in order to identify the number of clusters that minimizes the within variance, in this case 2. Then, I procedeed with the creation of the clusters, that show a similar behavior to the ones of the hierarchical approaches, as shown in Fig 12. Note that procedure has been iterated 15 times, since otherwise an undesirable local optimum may be obtained.

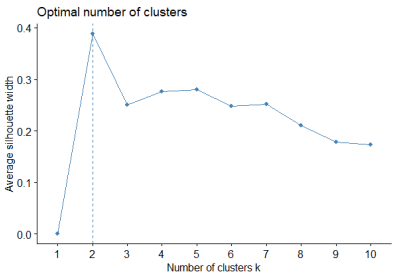

Figure 10: Silhouette method

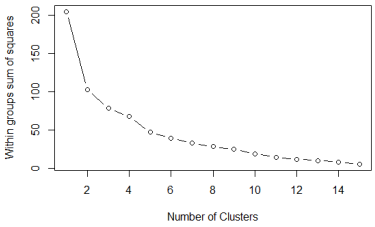

Figure 11: Elbow method

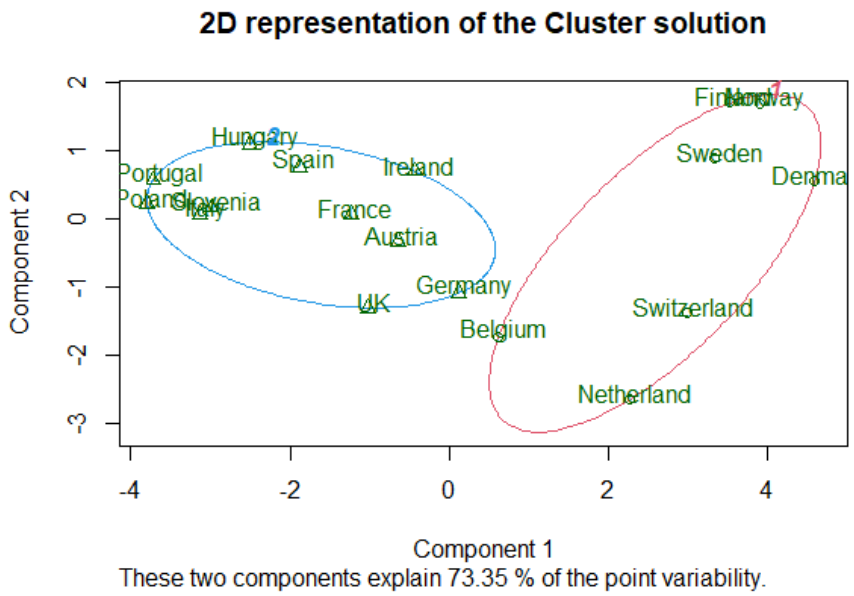

Figure 12: K-mean clustering

4.2 PCA

The second unsupervised technique that I exploited in my analysis is the PCA, through which I grouped the variables in new principal components and subsequently I plotted the dataset considering the first two variables. But first, it is important to detect if they are correlated. In order to gain these results, I plotted the correlation matrix, as we can see in Fig 13. The correlation matrix, as expected, shows us that the levels of trust in the national institutions are strongly positive correlated with each other and slightly less strongly correlated with the trust in international institutions, such as EU and UN, and the level of freedom. Also the average income and the level of technology (expressed by the TAI index) is positively correlated with the trust in the institutions (national and international) and with the level of freedom of a country. The density measure is not strongly correlated with the other variables. It is important to note that the level of density is aggregated for country: the lack of correlation between this variable and the other ones in terms of countries should not be confused with a lack of correlation in terms of individuals: if it is likely that highly populated areas show higher level of trust with respect to underpopulated ones, it is also true that countries with low level of density, such as the North ones, are traditionally characterized by high level of trust in national and international institutions. Finally, the level of religiosity is negatively correlated with the other variables.

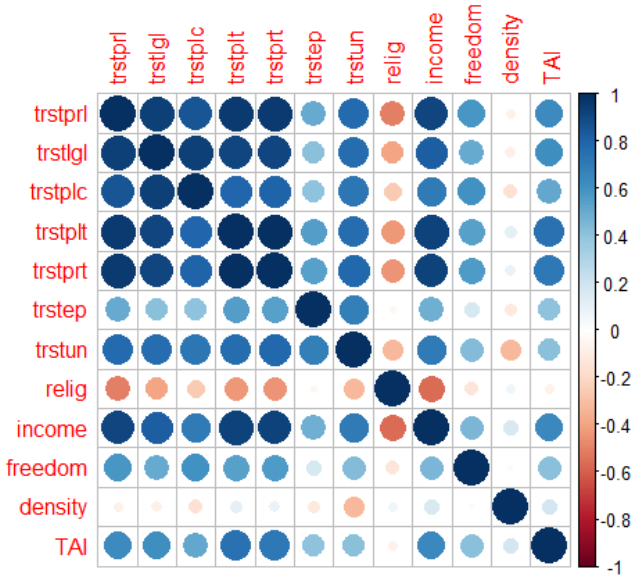

Figure 13: Correlation matrix

Then, I procedeed with the principal component analysis on the scaled data. The amount of variance explained by each principal component could be visualized through the scree plot of Fig 15. Using the first two components, the 67,3 variance is captured

(Fig 14), so the information lost is not very high.

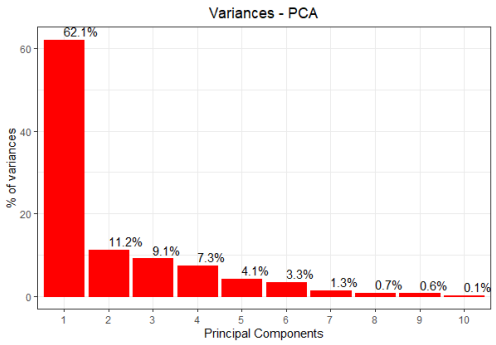

Figure 14: Variance explained by PCs

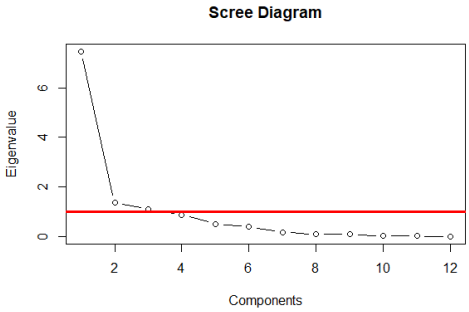

Figure 15: Scree plot

I later plotted the variables in the first two principal components. As we can see in Fig 16, all the variables related to the trust goes in the same direction as regards to the first dimension, as the income, the technology achievement index and the level of freedom of a country. The level of religiosity, instead, goes in the opposite direction. The density level doesn't affect the first principal component. Regarding to the second principal component, the most important variables is the one related to the density,that dominates the dimension, and subsequently the economic indexes. Trust in the international institutions goes in the opposite direction.

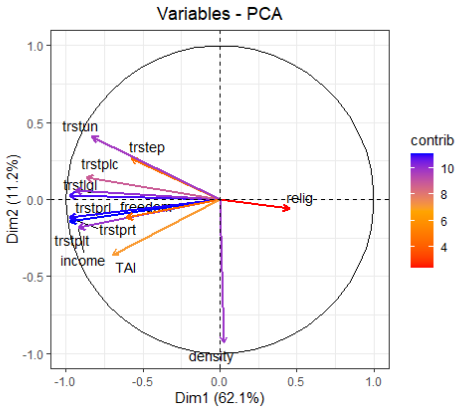

Figure 16: Variables plot

Variable	Comp1	Comp2
trstprl	0.979	0.027
trstlgl	0.940	0.056
trstplc	0.870	0.142
trstplt	0.978	-0.144
trstprt	0.980	-0.117
trstep	0.578	0.267
trstun	0.832	0.409
relig	-0.450	-0.059
income	0.920	-0.183
freedom	0.606	-0.117
density	-0.024	-0.934
TAI	0.701	-0.359

Figure 17: First two PCs

Finally, we can analyze how the observations are placed in the diagram through Fig 18. Plotting the datapoints onto the first two principal components contributed to identify four main sectors, two for each dimension, which could be interpreted by looking at the directions of the loadings of each variable. We can see that the states that are in

the first quadrant show low level of trust in the national and international institutions and lower level of freedom compared to the other European countries, high religiosity, low income and TAI and low density. The second quadrant is the one composed by countries characterized by high level of trust, high level of freedom, low religiosity, high income and TAI and low density. The third one is the one of the states that show low level of trust in institutions (national or international), low level of freedom, high religiosity, low income and TAI and high density. The last one has high level of trust, high level of freedom, low religiosity, high income and TAI and high density.

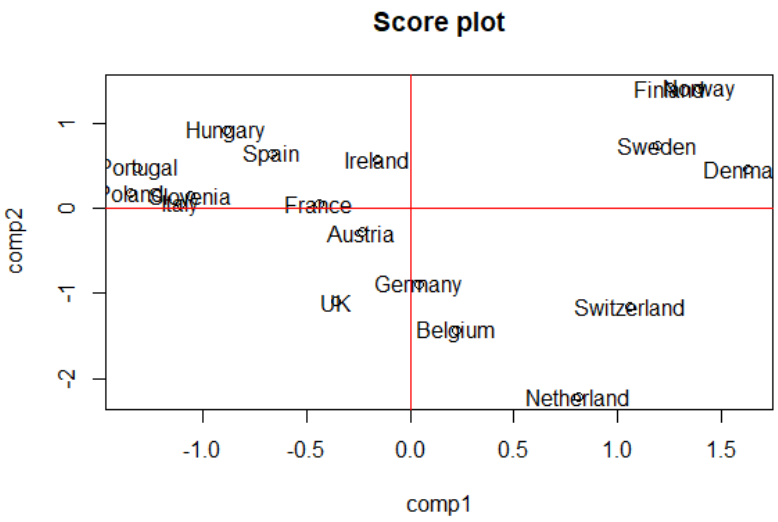

Figure 18: Score plot

5 Conclusions

The aim of this research was to understand how the European countries could be grouped according to some socio-demographic and economic variables. The variables that I used in my analysis were mostly retrieved by aggregating by state the individual answers to the European Social Survey, an academically driven cross-national survey conducted across Europe since its establishment: I decided to take into consideration the levels of trust on several internal and international institutions, the level of religiosity and the level of income in a scale from 0 to 10. I also added some other variables: the total freedom score from Freedom House, the population density and the Technology achievement index.
In order to achieve my goal, I used different unsupervised methods: two clustering algorithms, the hierarchical clustering and the k-means clustering, were exploited so to show the similarity and dissimilarity between some states; then, the PCA method was used in order to have a visual representation of the data in a two dimensional space, onto which most of the variance is captured.
With regards to the hierarchical clustering, I performed it using several distance measures and agglomeration methods in order to check for the robustness of my cluster analysis. What I found was four well-defined clusters: Norway, Finland, Denmark and Sweden made up an homogeneous group of states with all the linkage methods and distance measures, as Belgium, Switzerland and Netherlands; Austria, Germany and UK were always considered as quite similar to each other; the remaining countries were often grouped in one cluster. Furthermore, I could note that Hungary was often put in a separate group using the Euclidean distance, probably due to the fact that it is an outlier in the Europe scenario concerning to the level of freedom and this kind of distance does not perform very well in the presence of extreme observations.
The results that I found using the K-means clustering were consistent with the findings of the hierarchical clustering: grouping the observation into two groups, the optimal number of clusters in this analysis, I obtained a group characterized by high level of income, TAI and trust in national and international institutions and low level of religiosity and a group for which the opposite is true.
The last unsupervised technique that I exploited in my analysis was the PCA. Plotting the datapoints onto the first two principal components contributed to identify four main sectors, two for each dimension, which could be interpreted by looking at the directions of the loadings of each variable: the states that were in the first quadrant showed low level of trust in the national and international institution sand lower level of freedom compared to the other European countries, high religiosity, low income and TAI and low density. The second quadrant was the one composed by countries characterized by high level of trust, high level of freedom, low religiosity,high income and TAI and low density. The third one was the one of the states that showed low level of trust in institutions (national or international), low level of freedom, high religiosity, low income and TAI and high density. The last one had high level of trust, high level of freedom, low religiosity, high income and TAI and high density.

6 R codes

```r
#call libraries
library(factoextra)
library(reshape2)
library(cluster)
library(ggplot2)
library(FactoMineR)
library(corrplot)

#choose variables from my original dataset
mydata <- read.csv2(file ="C:\\Users\\DELL\\Downloads\\ESS1.csv")
df <- mydata [1:10]
df <- na.omit(df)

#create dataset grouping by countries and add new variables
a<- aggregate(df[-1], by = list(df$  ..country), FUN =mean)
a$freedom <- c(93, 96, 96, 94, 97, 90, 100, 90, 93, 69, 97, 90, 98, 100,
    82, 96, 100, 95)
a$density <- c(106,376,206,232, 135,92, 16,118, 272, 105, 69, 200,421,
    14, 123,112,23,102)
a$TAI <- c (0.617, 0.604, 0.813, 0.658, 0.666, 0.534, 0.633, 0.622,
    0.546, 0.516, 0.682, 0.507, 0.745, 0.626, 0.626, 0.467, 0.685, 0.556)
dati1<-a[,-1]

#normalize variables
dati1<-scale(dati1[,1:12])
rownames(dati1)<-c("Austria", "Belgium", "Switzerland", "Germany", "
    Denmark", "Spain", "Finland", "France", "UK", "Hungary", "Ireland", "
    Italy", "Netherland", "Norway", "Poland", "Portugal", "Sweden", "
    Slovenia")

#choose the distances
d1 <- dist(dati1, method="euclidean", diag=F, upper=F)
d2<- dist (dati1, method = "manhattan", diag=F, upper= F)

#function to generate the agglomeration program
agglo <- function(hc){
  data.frame(row.names=paste0("Cluster",seq_along(hc$height)),
             height=hc$height,
             components=ifelse(hc$merge<0,
                               hc$labels[abs(hc$merge)], paste0("Cluster"
    ,hc$merge)),
             stringsAsFactors=FALSE) }

#with complete linkage
h1 <- hclust(d1, method="complete"); h1
agglo(h1)
plot(h1, main="complete linkage")
complete <- cutree(h1, k=4)
rect.hclust(h1, 4)
h1cluster <- cutree(h1, k=4)
h1cluster

#with manhattan distance
```

```
h12 <- hclust(d2, method="complete"); h12
agglo(h12)
plot(h12, main="complete linkage")
complete <- cutree(h12, k=4)
rect.hclust(h12, 4)
h12cluster <- cutree(h12, k=4)
h12cluster

#with avg linkage
h2<-hclust(d1,method="average");h2
agglo(h2)
plot(h2, main="average linkage")
average <- cutree(h2, k=4)
rect.hclust(h2, 4)
h2cluster <- cutree(h2, k=4)
h2cluster

#with manhattan distance
h22 <- hclust(d2, method="average"); h22
agglo(h22)
plot(h22, main="average linkage")
complete <- cutree(h22, k=4)
rect.hclust(h22, 4)
h22cluster <- cutree(h22, k=4)
h22cluster

#with single linkage
h3<-hclust(d1,method="single");h3
agglo(h3)
plot(h3, main="single linkage")
single<- cutree(h3, k=4)
rect.hclust(h3, 4)
h3cluster <- cutree(h3, k=4)
h3cluster

#with manhattan distance
h32 <- hclust(d2, method="single"); h32
agglo(h32)
plot(h32, main="single linkage")
complete <- cutree(h32, k=4)
rect.hclust(h32, 4)
h32cluster <- cutree(h32, k=4)
h32cluster

#with ward linkage
h4<-hclust(d1,method="ward.D");h4
agglo(h4)
plot(h4, main="Ward linkage")
ward<- cutree(h4, k=4)
rect.hclust(h4, 4)
h4cluster <- cutree(h4, k=4)
h4cluster
plot(a[-1], col=h4cluster, main="Ward Methods")

#with manhattan distance
h42 <- hclust(d2, method="ward.D"); h42
```

```
agglo(h42)
plot(h42, main="ward.D linkage")
complete <- cutree(h42, k=4)
rect.hclust(h42, 4)
h42cluster <- cutree(h42, k=4)
h42cluster

#add the cluster to the dataset
h4cluster<- cutree(h4, k=4)
dati1 <- as.data.frame(dati1)
a$clu<-h4cluster
dati1$clu<-h4cluster

#means for variables (non normalized)
medie<-aggregate(a[,2:13], list(h4cluster), mean)
medie

#calculus of R^2 for each variables
mydata<-dati1
R2 <- rep(NA, (ncol(mydata)-1))
for(i in 1:(ncol(mydata)-1))
  R2[i] <- anova(aov(mydata[,i] ~ mydata[,ncol(mydata)]))[1,2]/(anova(aov
    (mydata[,i] ~ mydata[,ncol(mydata)]))[1,2]+anova(aov(mydata[,i] ~
    mydata[,ncol(mydata)]))[2,2])
 R2
col<-colnames(mydata[-13])
finali<-cbind(col,round(R2,2))

#plots for cluster interpretation
col<-colnames(mydata)
mydataz<-data.frame(scale(mydata))
mydataz$clu<-h4cluster
dati <- melt(mydataz, measure.vars=col)
ggplot(dati[1:216,], aes(x = variable, y = value, color=variable)) +
  geom_boxplot() +
  facet_wrap(~ mydataz$clu) +
  theme(axis.text.x=element_blank(),
        axis.ticks.x=element_blank())

#K-Means
dati1.stand <- dati1[,-13]

#select number of K (Elbow method)
set.seed(123)
wssplot <- function(data, nc=15, seed=1234){
  wss <- (nrow(data)-1)*sum(apply(data,2,var))
  for (i in 2:nc){
    set.seed(seed)
    wss[i] <- sum(kmeans(data, centers=i)$withinss)}
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares")
  wss
}
wssplot(dati1.stand)

#select number of K (Silhouette method)
```

```r
fviz_nbclust(dati1.stand, kmeans,nstart=15, method = "silhouette")

#k = 2
k.means.fit <- kmeans(dati1.stand, 2)
str(k.means.fit)
clusplot(dati1.stand, k.means.fit$cluster,
         main='2D representation of the Cluster solution',
         color=TRUE,
         labels=2, lines=0)

set.seed(1233)
final <- kmeans(dati1.stand, 2, nstart = 15)
print(final)
fviz_cluster(final, data = dati1.stand)

#correlation matrix
corrplot(cor(dati1[-13]))

#PCA
datipuliti<- dati1.stand
n <- nrow(datipuliti)
p <- ncol(datipuliti)

#PCA from correlation
rho <- cor(datipuliti)
eigen(rho)
autoval <- eigen(rho)$values
autovec <- eigen(rho)$vectors

#select components
pvarsp = autoval/p
pvarspcum = cumsum(pvarsp)
pvarsp

#scree Diagram
plot(autoval, type="b", main="Scree Diagram", xlab="Components", ylab="
    Eigenvalue")
abline(h=1, lwd=3, col="red")

#number of components
world.pca<-prcomp(dati1[-13], center = TRUE,scale. = TRUE)
fviz_eig(world.pca,barcolor = "red",
         barfill = "red",geom = c("bar"),addlabels= TRUE ) +labs(title =
    "Variances - PCA",
                                                                 x = "
    Principal Components", y = "% of variances") +theme_bw()+theme(plot.
    title = element_text(hjust = 0.5))
summary(world.pca)

#interpret the components
eigen(rho)$vectors[,1:2]

#component matrix
comp<-round(cbind(-eigen(rho)$vectors[,1]*sqrt(autoval[1]),-eigen(rho)$
    vectors[,2]*sqrt(autoval[2])),3)
colnames(comp)<-c("Comp1","Comp2")
```

```
comp
rownames

#commonality
comunalita<-comp[,1]^2+comp[,2]^2
comp<-cbind(comp,comunalita)
comp<- as.data.frame(comp)
rownames(comp)<-colnames(datipuliti)

#scores
datipuliti.scale <- scale(datipuliti, T, T)
punteggi <- datipuliti.scale%*%autovec[,1:2]

#standardized scores
punteggiz<-round(cbind(-punteggi[,1]/sqrt(autoval[1]),-punteggi[,2]/sqrt(
    autoval[2])),2)
plot(punteggiz, main="Score plot",
     xlab="comp1",ylab="comp2")
text(punteggiz, rownames(dati1))
abline(v=0,h=0,col="red")

#loadinngs
plot(comp[,1:2], main="Loadings plot",
     xlab="comp1",ylab="comp2", xlim=range(-1,1))
text(comp, rownames(comp))
abline(v=0,h=0,col="red")

#princomp
acp<-princomp(datipuliti, cor=T)
summary(princomp(datipuliti, cor=T))

#biplot
biplot(acp)
fviz_pca_var(world.pca,col.var = "contrib",
             gradient.cols = c("red","orange","blue"),
             repel = TRUE,col.circle = "black",arrowsize = 1,
             labelsize = 0.5,jitter = list(what = "both", width = 1,
    height = 1) )+theme_bw()+theme(plot.title = element_text(hjust = 0.5)
    )
```

Listing 1: R Codes

References

[1] http://europa.eu/scadplus/glossary/multispeed_europe_en.htm

[2] This chapter is based on the book "An introduction to statistical learning: with application in R" (2013), James G., Witten D., Hatie T.,Tibshirani R., Springer

[3] https://statsandr.com/blog/clustering-analysis-k-means-and-hierarchical-clustering-by-hand-and-in-r/

[4] https://www.europeansocialsurvey.org/data/

[5] https://freedomhouse.org/countries/freedom-world/scores

[6] https://www.indexmundi.com/factbook/fields/population

[7] "Measuring the technology achievement index: comparison and ranking of countries"(2017), Incekara A., Guz T., Sengun G, Journal of Economics, Finance and Accounting (JEFA), V.4, Iss.2, p.164-174