# Unsupervised Learning with R

Luca Paoletti

June 2022

## Contents

# 1 Introduction

With the following project, I mean to run different unsupervised learning algorithms in order to analyze and classify several observations of a dataset, each one representing a country in the world. The dataset describes the countries through health and economic features and the purpose is to create aggregates where to see differences in features' values. As far as I use unsupervised techniques, the dataset hasn't got the response variable $y$, only independent variables are used.
During the project I will show two main techniques, the first one is the $K$-means algorithm, used to classify each country, and the second one is the Principal Component Analysis, used to reduce the number of variables. With PCA I can't predict any type of value, but I can see the importance of each feature and consequently have a better idea of how each feature influences the country's cluster (assigned with $K$-means).

# 2 State of Art

## 2.1 $K$-Mean

$K$-Mean clustering is one of the most used unsupervised learning algorithms in the field of Machine Learning and Statistical Learning. Here below I will explain in brief the aim of the method.

Let $C_1, \dots, C_K$ denote sets containing the indices of the observations in each cluster. These sets satisfy two properties:

1. $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \dots, n\}$. In other words, each observation belongs to at least one of the $K$ clusters.

2. $C_k \cap C_{k'} = 0 \forall k \neq k'$. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster.

The idea behind $K$-means clustering is that a good clustering is one for which the **within-cluster variation** is as small as possible.
The within-cluster variation for cluster $C_k$ is a measure $WCV(C_k)$ of the amount by which the observations within a cluster differ from each other. Hence we want to solve the problem

$$\min_{C_1 \dots, C_K} \{\sum_{k=1}^{K} WCV(C_k)\}.$$

This formula says that we want to partition the observations into $K$ clusters such that the total within-cluster variation, summed over all $K$ clusters, is as small as possible.
In order the define the within cluster variation, we typically use Euclidean distance

$$WCV(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

where $|C_k|$ denotes the number of observations in the $k$th cluster.
In brief, the algorithm works like this:

1. Randomly assign a number, from 1 to $K$, to each of the observations. These serve as initial cluster assignments for the observations.

2. Iterate until the cluster assignments stop changing:

   (a) For each of the $K$ clusters, compute the cluster *centroid.* The $k$th cluster centroid is the vector of the $p$ feature means for the observations in the $k$th cluster.

   (b) Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

## 2.2 Hierarchical Clustering

Hierarchical clustering is an alternative approach to $K$-means which does not require any commitment to a particular choice of $K$.
The most common type of hierarchical algorithm is the *bottom-up* (or *agglomerative*) clustering, and refers to the fact that a dendogram is built starting from the leaves and combining clusters up to the trunk.
The approach is the following:

1. Start with each point in its own cluster
2. Identify the closest two clusters and merge them
3. Repeat
4. The process will end

In order to create the dendogram, there are several types of *linkage* differently describing the link between observations. The main ones are:

- Complete: Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster $A$ and the observations in cluster $B$,and record the *largest* of the dissimilarities.
- Single: Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster $A$ and the observations in cluster $B$, and record the *smallest* of the dissimilarities.
- Average: meaning inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster $A$ and the observations in cluster $B$, and record the *average* of the dissimilarities.
- Centroid: Dissimilarity between the centroid for cluster $A$ (a mean vector of length $p$) and the centroid for cluster $B$. Centroid linkage can result in undesirable *inversions.*

## 2.3 Principal Component Analysis (PCA)

PCA is not a clustering algorithm but, it produces a low-dimensional representation of a dataset. It finds a sequence of linear combinations of the variables that have maximal variance, and are mutually uncorrelated. In addition, PCA also serves as a tool for data visualization.

The first principal component of a set of features $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \cdots + \phi_{p1}X_p$$

that has the largest variance. By normalized, we mean that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.
We refer to the elements $\phi_{11}, \ldots, \phi_{p1}$ as the loadings of the first principal component; together, the loadings make up the principal component loading vectors.

$$\phi_1 = (\phi_{11}, \phi_{21}, \ldots, \phi_{p1})^T$$

We constrain the loadings so that their sum of squares is equal to one, since otherwise setting these elements to be arbitrarily large in absolute value could result in an arbitrarily large variance.

The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with $Z_1$.
The second princiapl scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12}x_{i1} + \phi_{22}x_{i2} + \cdots + \phi_{p2}x_{ip}$$

where $\phi_2$ is the second principal component loading vector with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

The loading vector $\phi_1$ defines a direction in feature space along which the data varies the most. If we project the $n$ data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$ themselves.

To understand the strength of each component, we are interested in knowing the proportion of variance explained (PVE) by each one. The *total variance* present in a data set is defined as

$$\sum_{j=1}^{p} Var(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2$$

and the variance explained by the $m$th principal component is

$$Var(Z_m) = \frac{1}{n} \sum_{i=1}^{n} z_{im}^2$$

# 3 The Data

## 3.1 Description of the Dataset

The dataset choice describes different countries all over the world on the basis of socio-economics features. The aim of the project is to categorise the countries using some socio-economic and health factors that determine the overall development of the country. Of course, as far as this is an unsupervised learning project, I will not have the response variable. The available features can be divided into groups:
Social features

- child mort: Death of children under 5 years of age per 1000 live birth
- health: Total health expense per capita. Given as %age of GDP per capita
- life expec: The average number of years a new born child would live if the current mortality patterns are to remain the same
- total fer: The number of children that would be born to each woman if the current age-fertility rates remain the same.

Economic features:

- exports: Exports of goods and services per capita. Given as %age of the GDP per capita
- imports: Imports of goods and services per capita. Given as %age of the GDP per capita
- income: Net income per person
- inflation: The measurement of the annual growth rate of the Total GDP
- gdpp: The GDP per capita. Calculated as the Total GDP divided by the total population.

My purpose is to cluster countries basing my research on the socio economic features mentioned before. I expect to obtain clusters with high differences among them but, at the same time, very similar inside. For example, I can imagine to obtain 3 different clusters describing developed, in development and under developed countries. Of course the most important attributes could be GDPP or children mortality rate and life expectation.

## 3.2 Pre-Processing

First of all, I have considered the class of the variables and, as far as they were all continuous I did not have to remove or change any one of them. In addition no NAs have been found.
Next, I have plotted some boxplot of social and economics features in order to check for possible outliers and also to check the features' distribution.

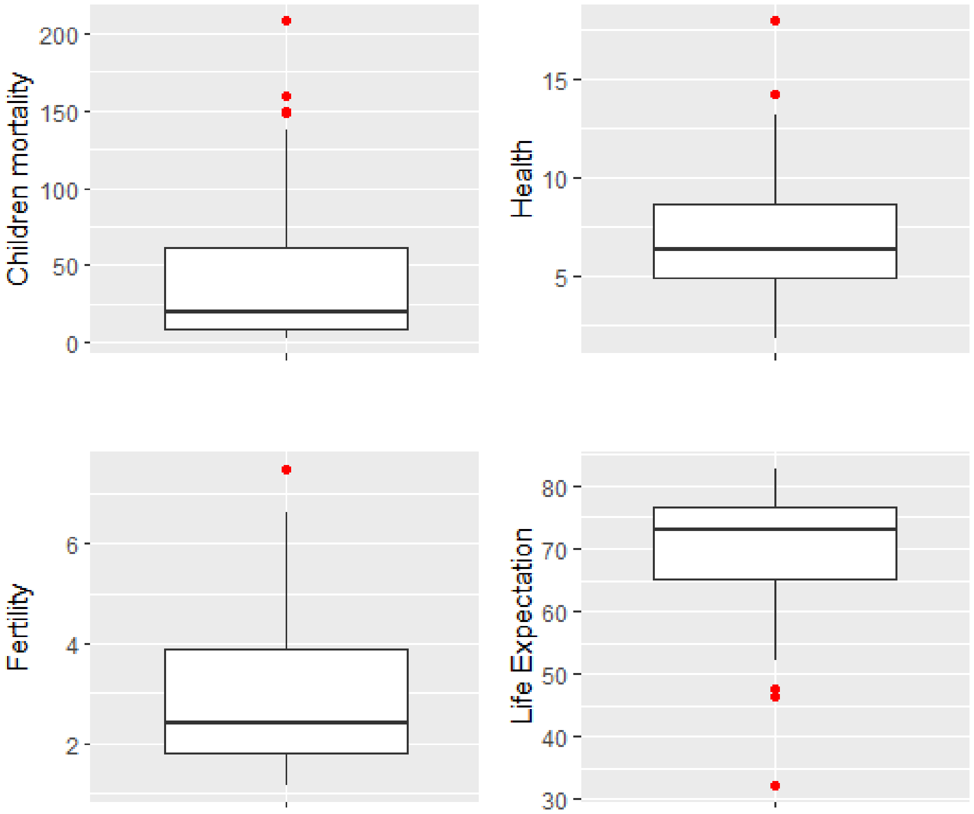


Figure 1: Social features distribution

As we can clearly see, there aren't many outliers (red points) and consequently I have decided to stick to these observations. The last decision is also due to the fact that one observation represents one country, removing one observation from the dataset will mean removing one entire country from the project and, in my opinion, it is interesting to see how the algorithms will cluster all the countries worldwide.
After the social distribution, I have analyzed the economic features too; here are some boxplots representing the attributes mentioned before.

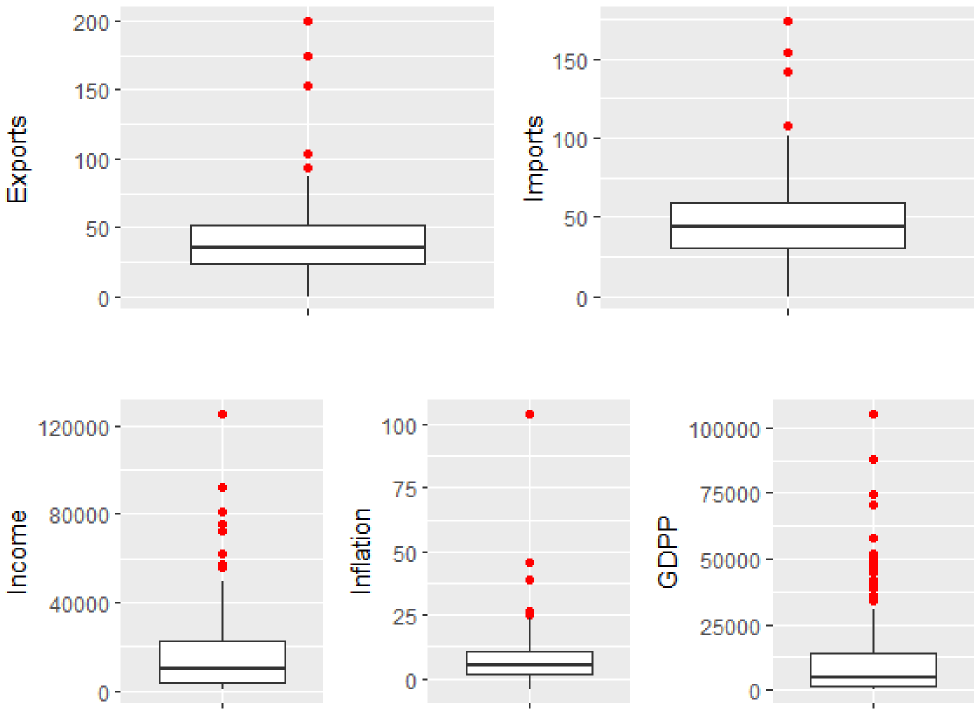


Figure 2: Economic features distribution

Through these boxplots it is possible to understand that, different countries, show high differences in economic measures. Let's consider for example the GDPP; I expected to notice that the GDPP has low values as main distribution (the plot confirms) because of the higher number of underdeveloped / in development countries with respect to the most developed. In this case, the red dots that should be outliers, simply are developed countries representing the minority and consequently they results as outliers.
This fact shows why I do not remove outliers in this dataset.

Now let's consider the correlation analysis. High correlation among the variables could return multicollinearity issues. Despite the fact that some attributes could be highly correlated with each other, it is also important to understand when the correlation is spurious or not. Sometimes correlation is really high between two variables but this does not necessarily mean that the presence of one feature influences the value of the other one.

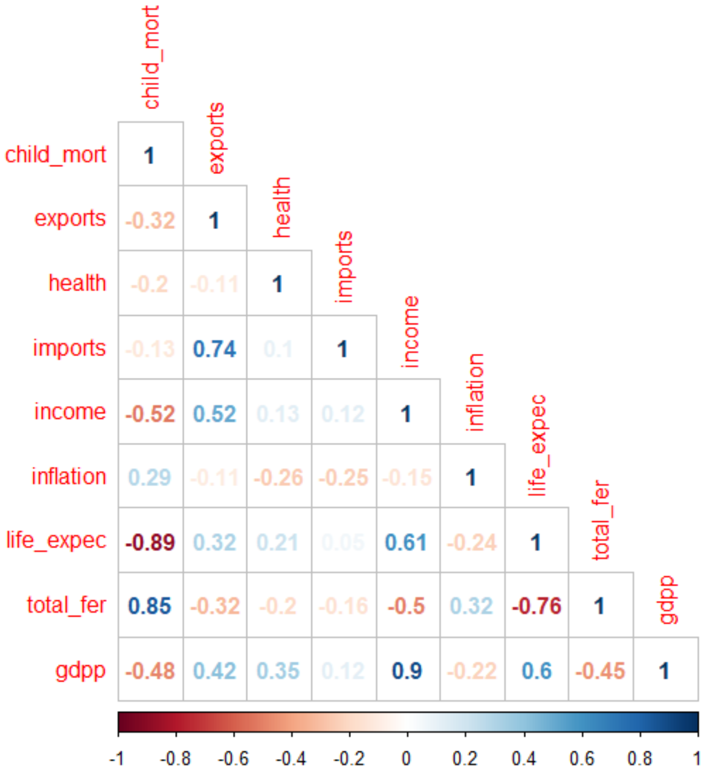


Figure 3: Correlation plot

As we can see, there is a really high correlation between life expectation and children mortality, between GDPP and life expectation, also between total fertility and children mortality or between GDPP and Income. Focusing on the meaning of these correlations, I reached the conclusion that, for example, a country with high incomes, will also have high GDPP or countries with high fertility rate, will induce a higher probability of children mortality and so on.

As a consequence, I have decided to remove from the dataset the following features:

- Income (GDPP more important than Income)
- Children Mortality (high correlation with several features)

## 3.3 Descriptive Analysis

In the previous section I have speculated on the possible cause of the correlations. In this part, I want to plot some descriptive analysis in order to confirm my previous thesis.

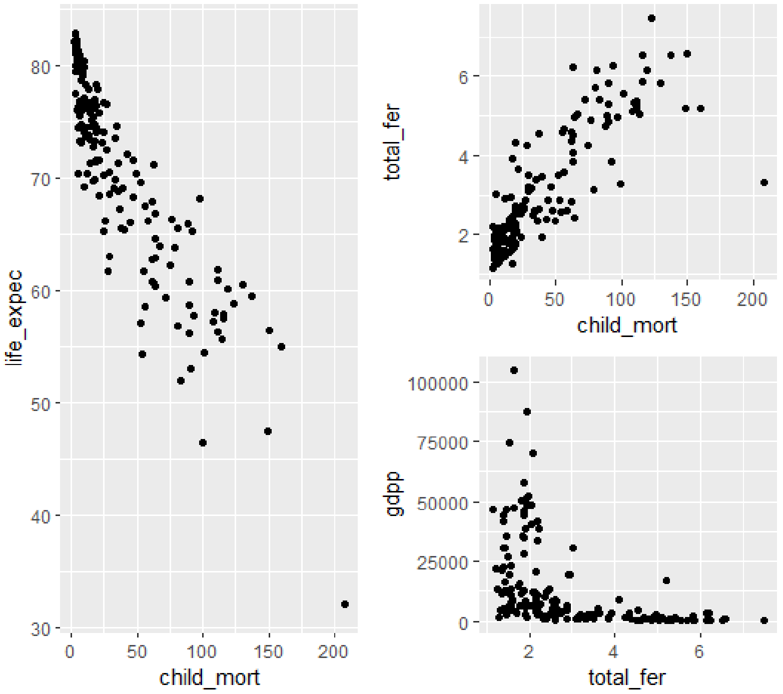


Figure 4: Health relations

The high correlations previously explained in the correlation matrix, are now clearly underlined by these graphs.

Children mortality is correctly negative correlated; where health care is more developed we expect to have lower children mortality. The dots extremely allocated on the $x$ axis are not outliers, they are just countries where life expectation is really low and consequently children mortality is high.

This also applies for children mortality and total fertility, although in this case the correlation is positive. Where the children mortality rate is high, it is obvious to see that fertility rate increases and on the other hand, where the fertility rate is low, it is less probable to see high children mortality.

This last example could also apply to the last plot, fertility and GDPP. Developed countries rarely have high fertility rate and the third plot is the proof of that.

With the clustering algorithms I expected to find basically 3 clusters based on development of the countries. One of the main attributes could be GDPP and I want to anticipate the results plotting a barplot of the country with the highest GDPP. This plot will be helpful in the next sections to prove the correctness of the clustering criteria.

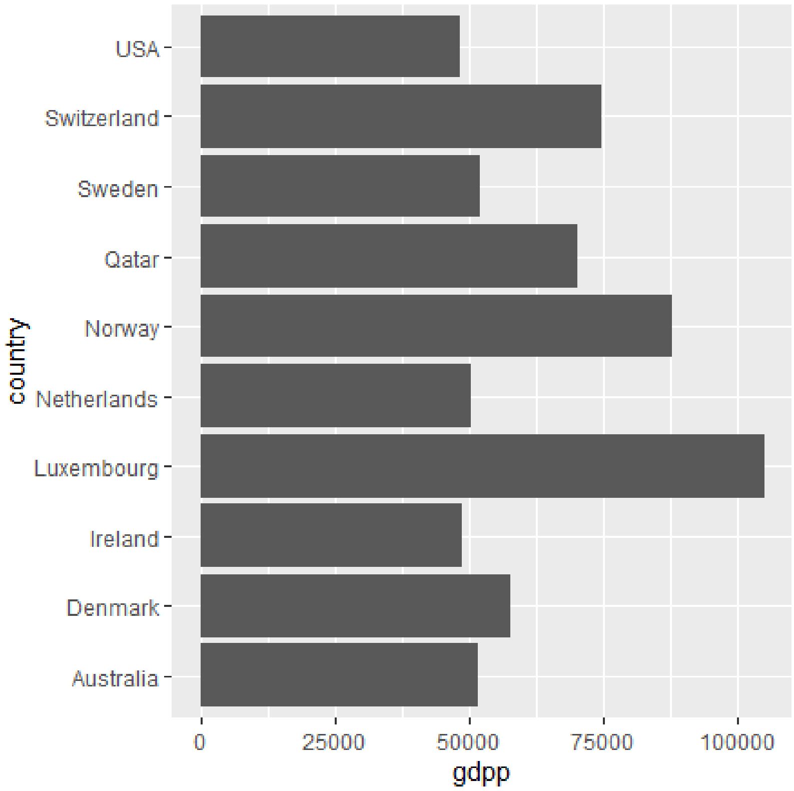


Figure 5: Highest GDPP

# 4 Models

## 4.1 K-Means

Kmeans takes as input one fundamental parameter, $K$, and two less important ones, *max iteration* that is the maximum number of iteration that the algorithm should run and finally *nstart*, the number of random sets that should be chosen. In my case I opted for $maxiteration = 100$ and $nstart = 1$; the key parameter is of course the number of clusters $K$ but, which is the best?

### 4.1.1 TWSS Method

I have run the algorithm several times each time with different values of $K$, storing each time the TWSS value and obtaining the following results

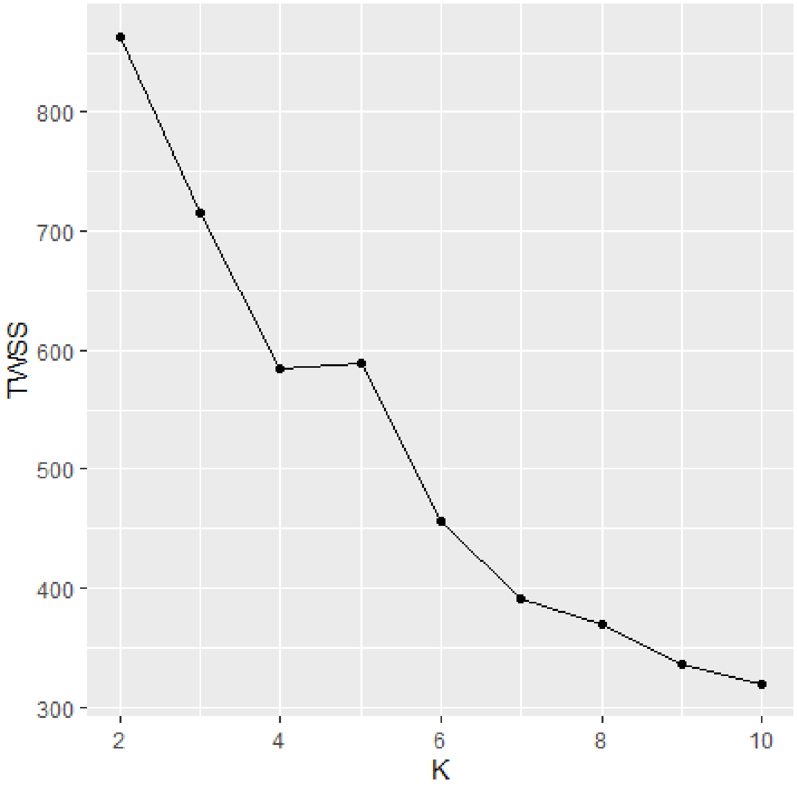
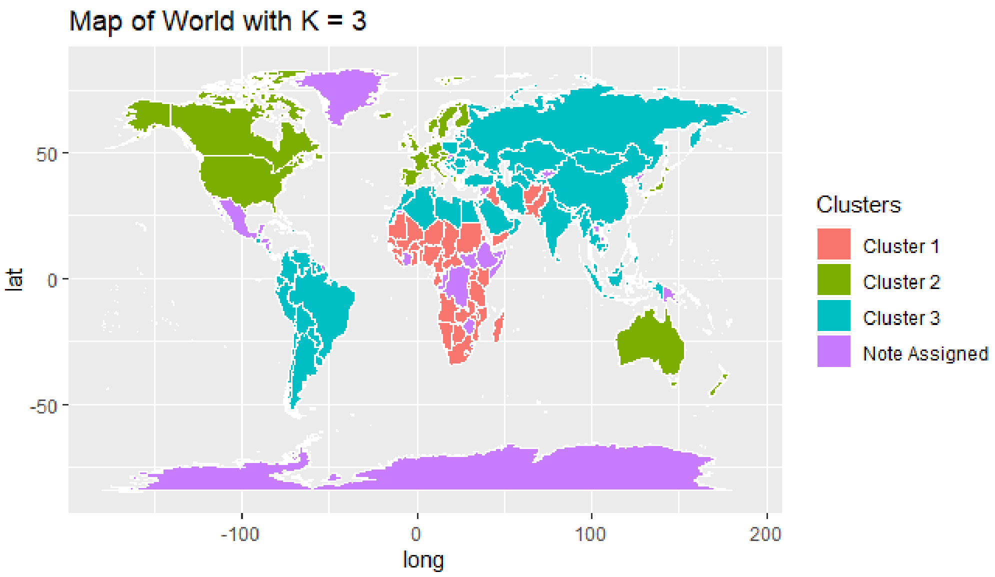


Figure 6: $K$ and TWSS

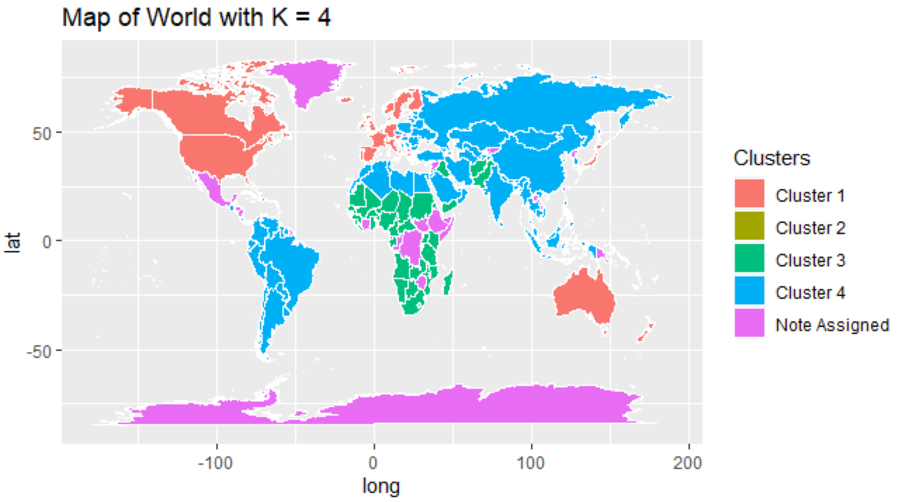
In this plot the TWSS represents the total within sum of squares, the optimal $K$ is between 3 and 4 (Knee-Elbow Method). I have tried to run the algorithm with both the $K$ values to see which is the best one.



From this plot we can clearly distinguish developed country (Green), developing ones (Blue) and under developed ones (Red). Of course these clusters are really different and the average values inside each cluster are the following:

| Cluster | Exports | Health | Imports | Inflation | Life Expectation | Total Fertility | GDPP |
|---|---|---|---|---|---|---|---|
| 1 | 29.32313 | 6.636250 | 43.79375 | 11.798646 | 59.27292 | 4.998542 | 1945.042 |
| 2 | 57.54516 | 9.550000 | 52.72903 | 1.456935 | 80.53871 | 1.735161 | 44458.065 |
| 3 | 41.74760 | 5.950341 | 46.52234 | 7.818932 | 73.19318 | 2.256705 | 7880.136 |

The table above clearly explains the differences between clusters. Cluster number 2 is the most developed (highest GDPP, low fertility rate and low inflation); cluster number 1 is the less developed, with high inflation, really low GDPP and also low life expectation; finally cluster number 3 is the middle one, it is the biggest one in terms of records and in fact we can see how, sometimes, the average value doesn't work so well (Health of cluster 3 is lower than cluster 2 despite the fact that cluster 2 is the less developed one).
As I stated before, I have decided to run the same algorithm with $K = 4$ but, as we will see in the next picture, the new cluster is not significant.



#### 4.1.2 Silhouette Method

The second method used to chose the right $K$ is the silhouette method. Silhouette score is used to evaluate the quality of clusters created using clustering algorithms in terms of how well the samples are clustered with other similar samples. In this case, the optimal $K$ is given by the highest value of the average silhouette. As for the TWSS method, in this case too, it seems that best $K$ should equal to 3 or 4. We have already run the algorithm with these two values and we already know the results.

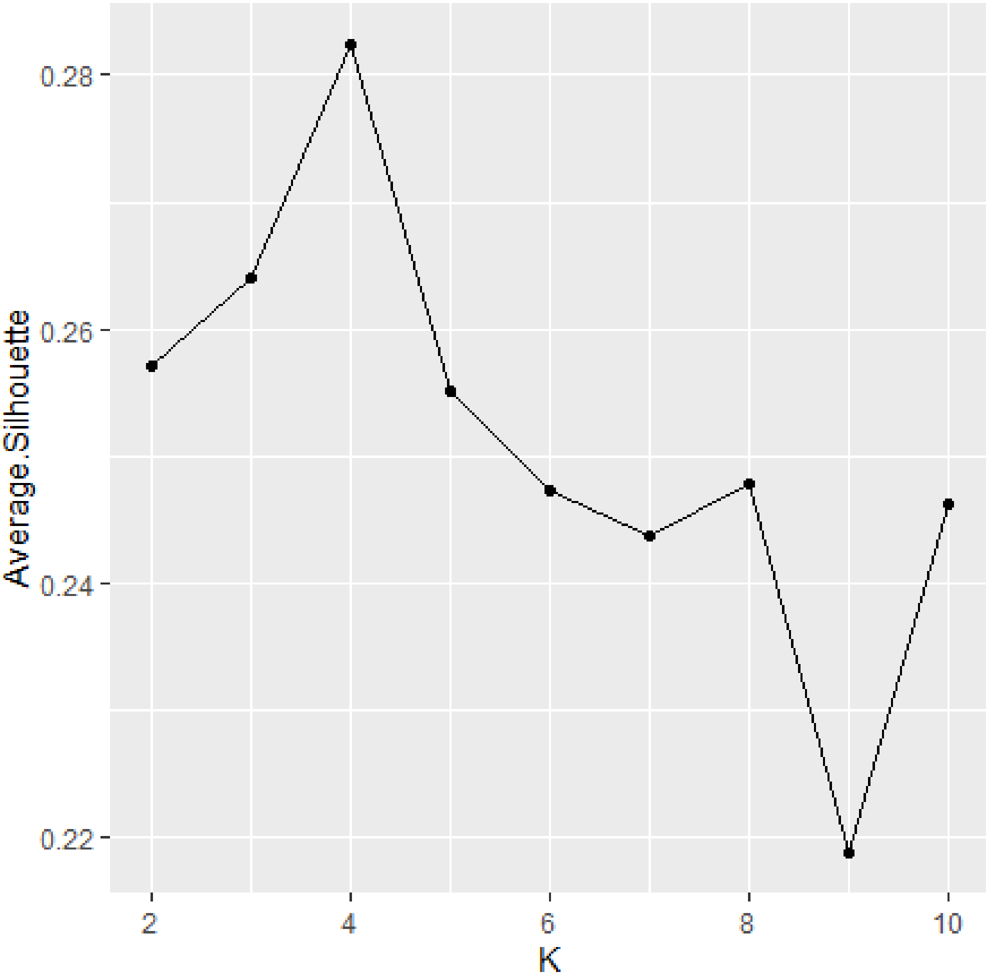


Figure 7: Silhouette Method

The silhouette plot displays a measure of how close each point in one cluster is to other points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually. This measure has a range of $[-1, 1]$.

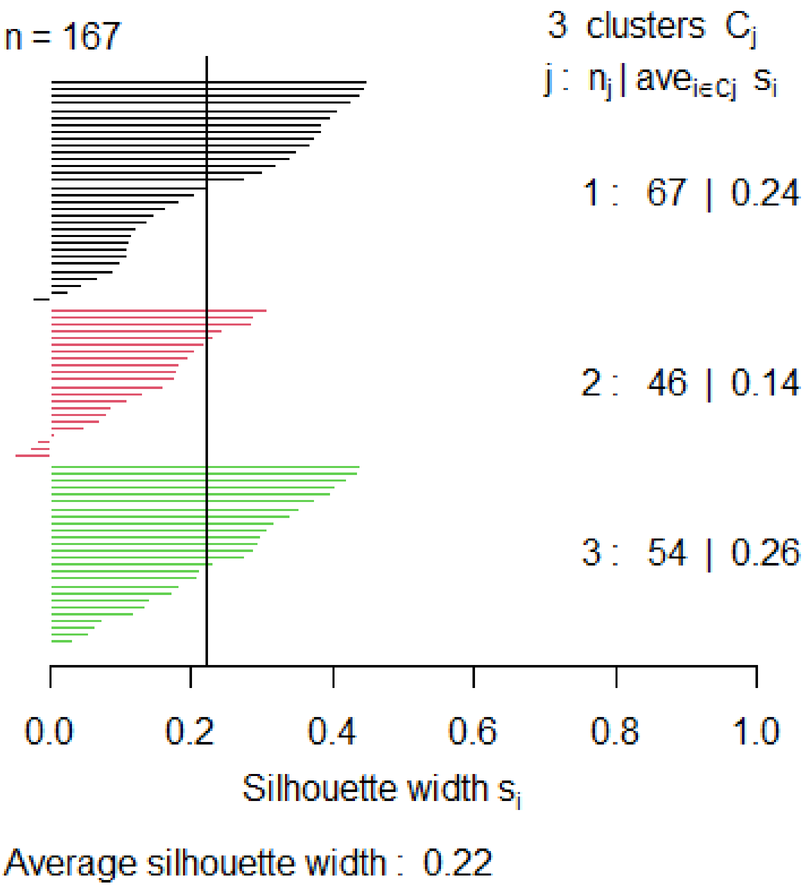


Figure 8: Silhouette Plot withj $K = 3$

Silhouette coefficients (as these values are referred to as) near +1 indicate that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters and negative values indicate that those samples might have been assigned to the wrong cluster.
In this case we have a few records wrongly classified but the 3 clusters are really 'near' to each other, as shown by the low value of average silhouette.

## 4.2 Hierarchical clustering

With the hierarchical clustering I have decided to calculate the distances with the *Euclidean* distance and to create the dendogram with the *Ward* method. Given that that the best $K$ is equal to 3, I have cut the dendogram in order to create 3 clusters. The result is the following

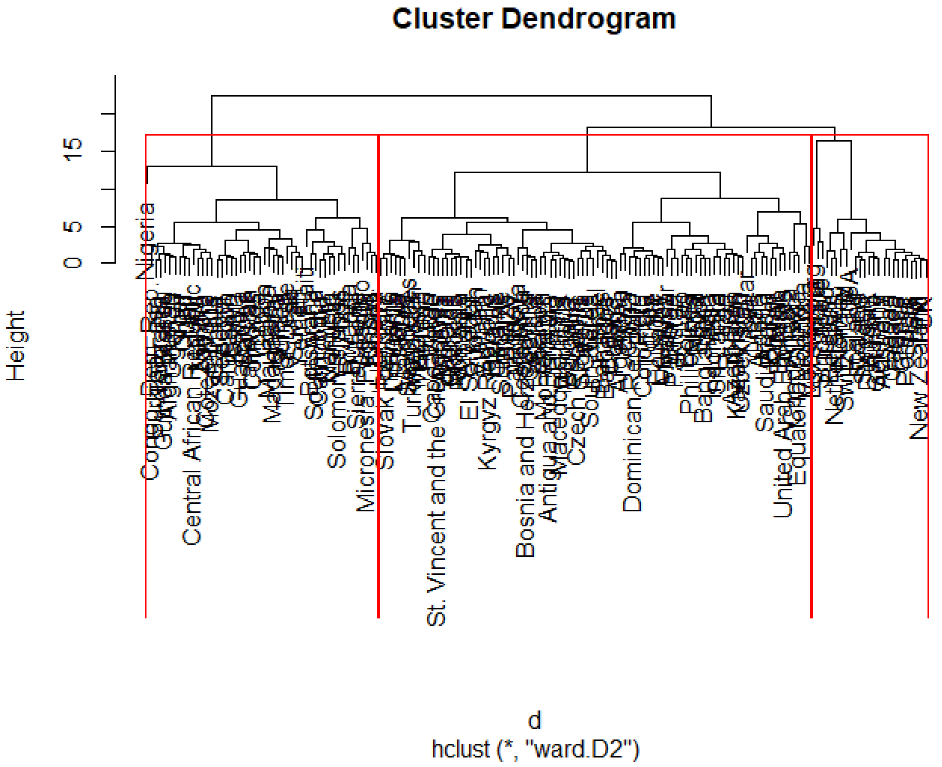
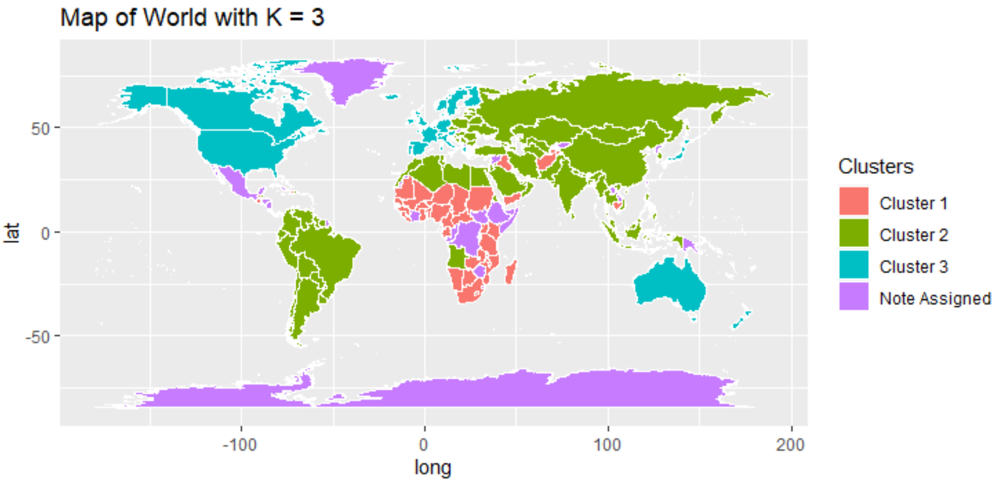


Figure 9: hierarchical clustering

Of course through this plot it is hard to read the label of each observation, so I have displayed the clusters on the world map and I have noticed how these results are very similar to the ones obtained with the kmeans.



Again, the clusters created represents the 3 level of development of each country. In fact the mean values of each feature inside each cluster are similar to the ones previously discussed.

| Cluster | Exports | Health | Imports | Inflation | Life Expectation | Total Fertility | GDPP |
|---|---|---|---|---|---|---|---|
| 1 | 27.9364 | 6.881000 | 45.74600 | 10.176900 | 60.26800 | 4.786800 | 1728.880 |
| 2 | 43.8976 | 5.865326 | 45.94093 | 8.199826 | 73.32609 | 2.284891 | 9694.239 |
| 3 | 57.1920 | 10.182400 | 52.67200 | 1.453480 | 80.93600 | 1.710400 | 47468.000 |

## 4.3 Principal Component Analysis

In this project, I am going to use the PCA in order to understand the importance of each feature. The chosen dataset does not need to reduce features, but, through PCA, I can easily understand the importance of each independent variable. In addition, using the first two principal components I can check whether the clusters obtained with $K$-means are valid or not.

First of all, I run the PCA on the entire scaled dataset with the aim of reading and understanding the proportion of variance explained by each principal component. The plot displayed has the aim of understanding how many PCs I need to work with.

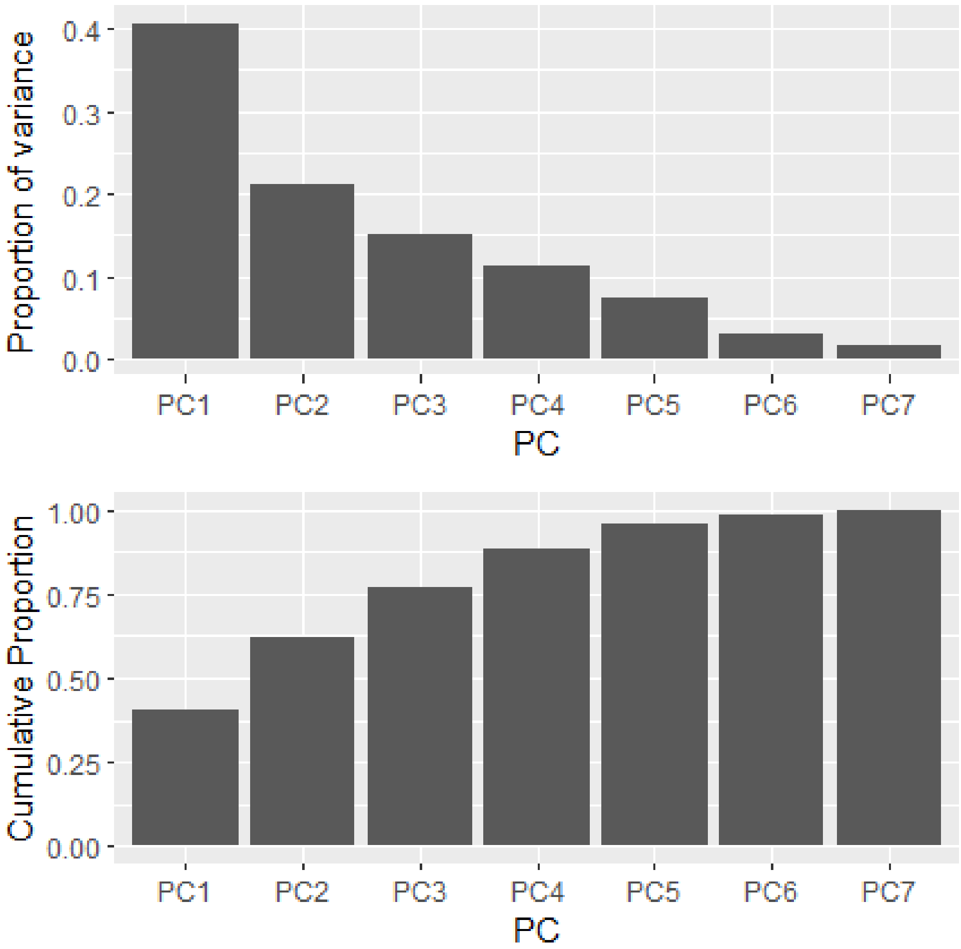


Figure 10: Scree Plot

As we can clearly see, the first 3 components together explain 75% of the variance, and of course the first one is the most explanatory (≈40%).
In addition, adding more PCs to the dataset will increase the proportion of variance explained but, as far as PCA has the objective of reducing the features to deal with, in my opinion 3 PCs could be a great deal.

In order to better understand how the PC are created and the importance of each feature inside the PC, I will display the following plot.

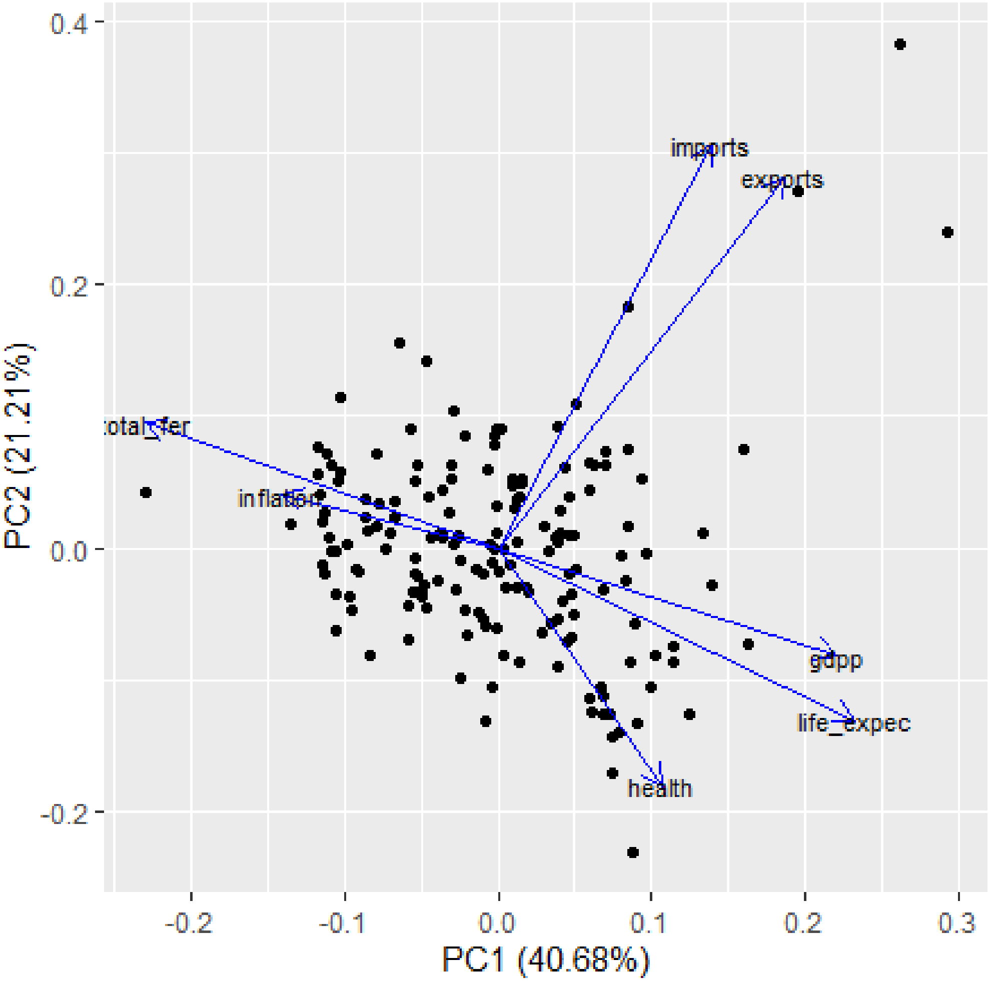


Figure 11: Biplot

From the Biplot we can understand how much a single feature contributes to the PC displayed; the more the vector is parallel to the PC axis, the more that feature contributes to the constitution of that PC. In this case we have that import and export are really significant for PC1 while total fer, inflation, gdpp and life exp are more important for PC2.
Moreover, the length of the vector told us how much the 2 components displayed are explained by the feature. For example, in this case imports and exports are the longest vector for these two components which means, these two features are explained by the the components displayed; on the other hand, inflation is not really significant for the PCs displayed in the current graph. Lastly, when two or more components have the same direction, this means that they are correlated, in particular, positive correlated if the direction is the same (import and export), negatively correlated when the direction is the opposite (inflation and gdpp).

To conclude this section, I will display on the same space of the previous graph, the points representing the records but this time I will add the colour of each cluster obtained with $K$-means. Of course the dimensions are only two but the result is satisfactory. It seems that the countries have been correctly classified.

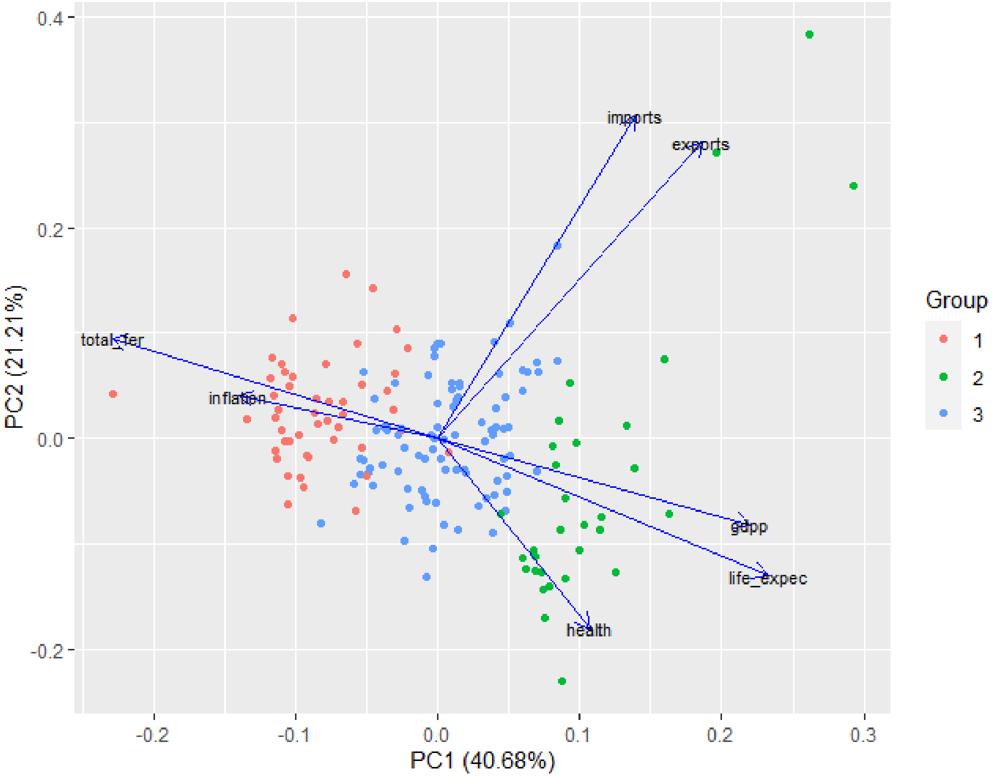


Figure 12: Clustering displayed with PCA

## 5 Findings and conclusion

Results and findings of this project have been already mentioned in the world map plot above. First of all, I have earned that running a clustering algorithm on this type of dataset, with socio economic features, results in clustering the records in basically 3 clusters: developed countries, developing countries and underdeveloped countries. Adding more clusters to this segmentation, creates more confusion and less distance between clusters. I have tried with 4 clusters, but the results weren't significant.

The second point I want to underline is the similarity of the learning algorithm. I have run both $K$-means and Hierarchical clustering on the same dataset and the results were very similar. This type of algorithm has few hyperparameters and consequently the hyperparamters tuning is simple. Once I found out that $K = 3$ was the best $K$, the two clustering algorithms reported the same results. Of course, running the hierarchical clustering with different distance measure or different linkage type could result in different clusters dimensions, but the aim of the project was also to get reasonable results, and the process done was the most suitable.

Last but not least, the feature importance. Through PCA I was able to get the importance of each feature in each cluster. I have seen how GDPP, Life expectation and Health were really important in order to separate cluster developed from cluster underdeveloped where other features like inflation and total fertility were more important. In addition, in Figure 12, it is also possible to see how the blue cluster, developing countries, is in the middle between under developed and developed countries; I can say that the more a dot (country) is near the green dot (developed countries) the more that country is about the pass from one cluster to the other. The same results apply for a 'blue' country near a red one (under developed).