MVA - Project 3

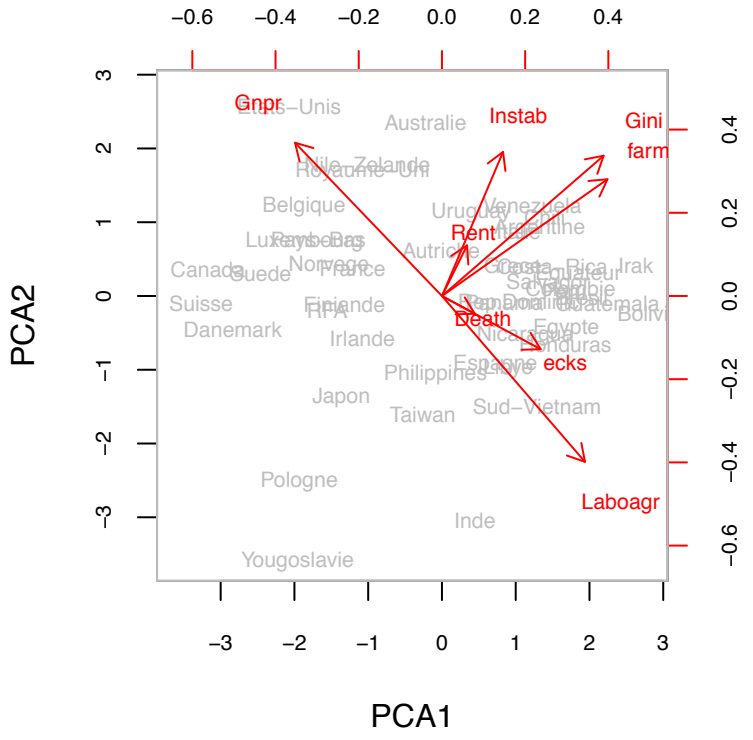
Carles Garriga Estrade i Balbina Virgili Rocosa

04/03/2018

All the code created for us to develop this assignment can be found on LAB03.R file. This file is located in the same folder of this document.

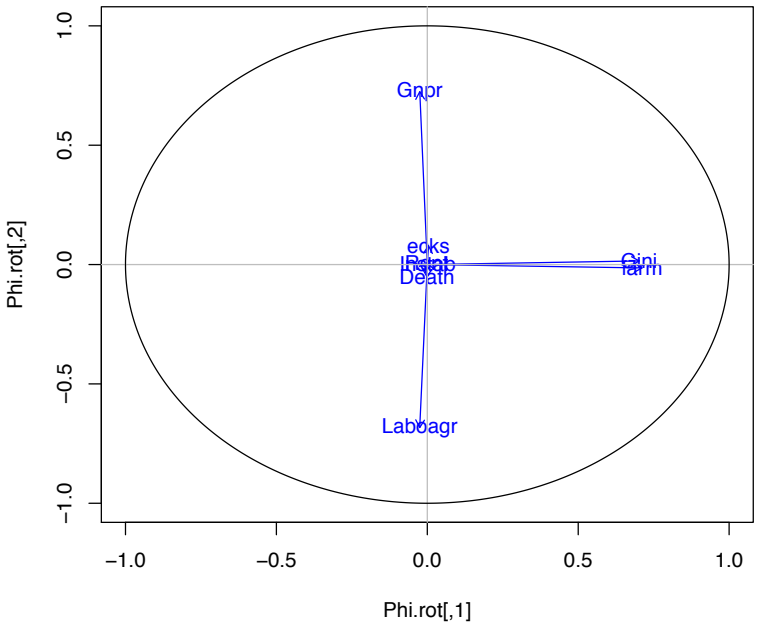
Exercise 1. First of all, the Russet dataset has been read and the completed matrix X has been defined with the standardized continuous data. To do it, missing values have been filled with KNN method and all data of the dataset except the one related with 'demo' variables. Also, each value of the matrix has been centered and standardized, by subtracting the centroid and dividing the result by the standardized deviation of each individual.

Exercise 2 and 3. After obtaining the standardized continuous matrix, the Nonlinear Iterative Partial Least Squares (NIPALS) algorithm can be applied on the dataset. NIPALS can be described as an iterative algorithm based on simple least squares regressions for calculating principal components, one by one on each iteration. To implement it, we have used the function *nipals* of the R package *chemometrics* and, as we did on the last assignment, we treat Cuba as a supplementary variable and we just want to calculate the first five significant principal components. To execute it, we have specified the number of iterations because, otherwise, the number of default ones were not sufficient to let the matrix converge and, as a consequence, the result was not accurate. Below, a biplot is showed with the results obtained.



The biplot joint the representation of the individuals and variables in the same display, to be able to do it, both projections have been represented in *Rp* space. With the biplot we can see both results PCA1 and PCA2 calculated with NIPALS. We can see that the results obtained are very similar as the ones obtained on the last assignment. As data is standarized and all data is represented in a 2D plot, we cannot assure an accurate relation between variables and individuals but we can have a general idea of the relation between them. For example, *Estats-Units* seems to be very impacted by *Gnpr*, while other individuals seems to be impacted by the combination of other variables.

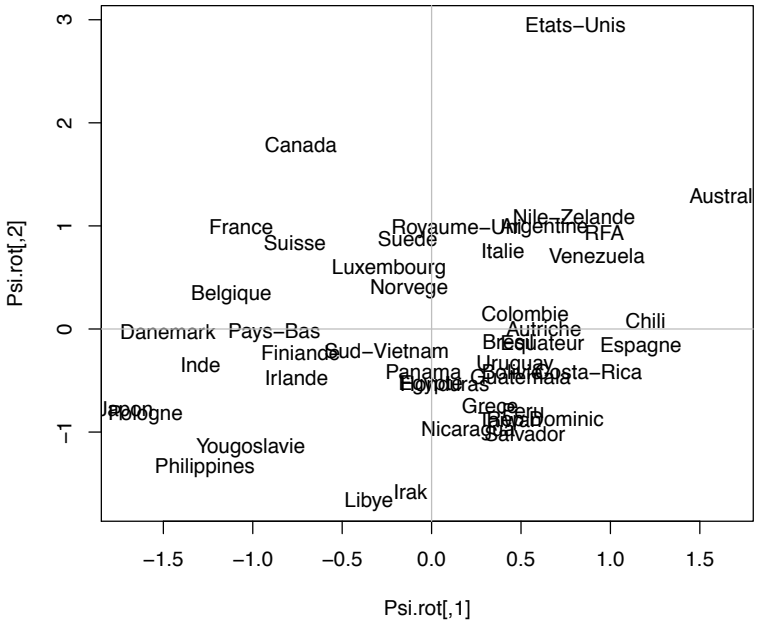
Exercise 4. Once the principal components are obtained, we are able to find a rotation matrix *Rotmat*. To do so, we need to define the new rotated axes, which are the ones where variables tend to be very correlated with one of them and zero correlated with the other. To implement it, we have used the function *varimax* of the R package *stats* and the results obtained are showed with the plot below.



With the plot obtained, the first two principal components calculated on *Exercise 2* are still represented, so the projected variance is the same as before. With this new visualization, it can be determined that *Gini* and *farm* are very correlated but both of them have no correlation with *Gnpr* and *Laboagr*, which at the same time, have a negative impact between them. With this rotated result, we have lost information of the other variables but we have a clear new view of the mentioned ones.

Exercise 5.

After computing the loadings of variables, the same has been done for the scores of individuals. They are shown on the plot below.



With *dimdesc* function, the variables that have more impact on each dimension have been found and they are represented on the table below.

Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
farm (+)	Gnpr (+)	Rent (+)	Instab (-)	ecks (+)
Gini (+)	Laboagr (-)			

We can realize that the results obtained for the first two dimensions are the same as the ones explained on *Exercise 4*. So, individuals are represented on the same way, depending on the influence of each variable they are located on the plot. For example, it can be easily interpretable that *Inde* and *Japo* seem to have a huge impact of *Laboagr*.

Exercise 6 - 7.

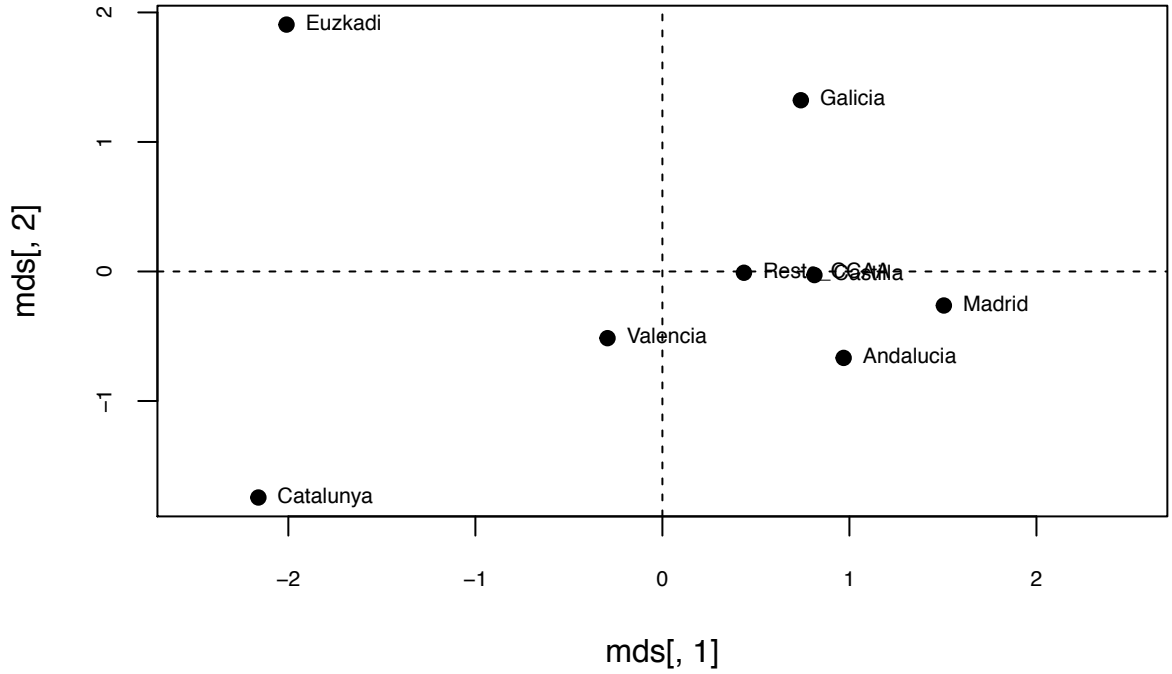
In order to symmetrize the data matrix previously read, we need to compute the joint feeling between different CCAA. Therefore, the feelings between two CCAA are averaged in order to compute the joint feeling.

Exercise 8.

After symmetrizing the data matrix, we can compute the dissimilarity matrix. As specified in the exercise statement, no feeling has a value greater than 10, which will the value for the max similarity. In order to extract the dissimilarity matrix, for each cell, we can simply substract the value of cell of the simililarity matrix to the max. Similiarity.

Exercise 9 - 10.

Once we have extracted the dissimilarity matrix, if we want to observe the dissimilarites between two CCAA we need to increase the number of dimensions. There's no better option than using a multidimensional scaling function together with PCA as the underlying metric to be able to compute the distances of the dissimilarity matrix. The resulting distances will represent the difference of the joint feelings between CCAA.



As seen in the plot of the first two components, both Catalunya and Euskadi are distant to the other interior CCAA, such as the Castilla's or Madrid. It can be interpreted that Catalunya and Euzkadi have completely different joint feelings to each other and to the rest of CCAA. And the are followed by Galcia and Valencia,

that are more closer. The closest ones are Castilla and the rest of CCAA (Resto_CCAA) and they are the ones that have similar feelings towards the other CCAA.