Open Access Journal of Behavioural Science & Psychology
ISSN: 2642-0856

Academic Strive
Researcher's perception is our reality

Research Article *Volume 1; Issue 2*

Supplementary Material

Contents

The s-Methods section contains some methodological details of the different methods used in the paper. The s-Code section includes an R pseudo code that can be used to reproduce the analyses presented in the manuscript.

s-Methods

Multiple Correspondence Analysis (MCA)

Multiple correspondence analyses are a generalization of correspondence analysis. It is a multivariate technique used when there are more than two categorical variables, with the purpose to study the association between the different categories (e.g. male and female for the variable Sex) of all the variables involved in the study to identify individuals with similar profiles (i.e. with the highest number of common categories). The final outcome is a plot that shows the relationships among categories, among subjects and among categories and subjects in a two-dimensional space in order to display the geometric configuration of the variable categories. Categories that are in the same quadrant or that are close enough suggest an association [1]. Substantially, the aim of the MCA is to obtain a measure of the association in terms of geometric distance so that associated categories are closely displayed in the output plot. The geometric distances are based on the definition of row and column profiles provided below.

	Variables							
	Sex		**Alcohol abuse**		**...**	**Qth variable**		**Total**
Subjects	Male	Female	Yes	No	...	$J-1$	J	
1	0	1	1	0	...	0	1	Q
2	0	1	0	1	...	1	0	Q
...	...	...	...	...	...	...	...	Q
i	1	0	0	1	...	1	0	Q
...	...	...	...	...	...	...	...	Q
I	1	0	1	0	...	1	0	Q
Total	Z_{+1}	Z_{+2}	Z_{+3}	Z_{+4}		Z_{+J-1}	Z_{+J}	Z_{++}

Table 1: Multiple correspondence analysis.

In details, based on the following data table of I subjects, Q categorical variables and J categories (Jq for each variable, $\Sigma JqQq=1=J$), it is possible to define the following quantities:
- $r_{ij}=\{0\ if\ subject\ i\ has\ not\ the\ j_{th}\ category\ 1_Q\ if\ subject\ i\ presents\ the\ j_{th}\ category$ with $\Sigma r_{ijIi=1}=1$ and $j=1,...,J$
- $r_i=(r_{i1},r_{i2},...,r_{ij},...,r_{iJ})'$ *row profile* and

Citation: Clarissa Ferrari, et al. Multivariate Statistical Techniques to Manage Multiple Data in Psychology. OA J Behavioural Sci Psych 2018, 1(2): 180006.

- cij={0 *if the jthcategory has not been chosen by the ithsubject*1/z+j *if the jthcategory has been chosen by the ithsubject - column profile* cj=($c1j$,$c2j$,...,cij,...,cIj)′.

Since MCA involves individuals and variable, two kind of distances can be evaluated (between row profiles, i.e. between individuals, and between column profiles, i.e. between categories of variables):

a. The distance between two row profiles *ri and ri′* of two different subjects is defined as:

$dD2(ri,ri')=\Sigma(rij-ri'j)2\bar{rj}Jj=1=IQ\Sigma(rij-ri'j)2z+j\ Jj=1.$

This distance will be equal to zero if the individuals have the same categories and it will increase when the number of distinct categories presented by the two subjects increases.

b. In a similar way the distance between two column profiles *cij−cij′* is defined as follows:

$dD2(cj,cj')=\Sigma(cij-cij')2\bar{ci}Ii=1=I\Sigma(cij-cij')2\ Ii=1$

Consequently, the profiles of the two categories j and j' will be the same when these are shown by the same subjects and the distance will increase with the number of individuals that show different categories. These distances will be displayed in a common unique plot (named Biplot) [2] such that the distance between any row profile or column profile gives the measure of their similarity (or dissimilarity).

Cluster Analysis

Cluster analysis is a multivariate technique used when all the variables of interest are continuous. It allows to aggregate n subjects in different groups, named *clusters,* on the basis of their individual data. The main goal is to find an optimal grouping so that the observations within the same cluster are similar (minimizing distance within clusters) and dissimilar from the observations in the other clusters (maximizing distance between clusters) [1].

There exist two main approaches of clustering: hierarchical and non-hierarchical:

The first approach is a "data-driven" process that may be *aggregative* (if it starts from n different clusters, one for each subject, and ends with only one: a single cluster containing all the observations) or *divisive (*if it does the opposite: starting from an unique group and ending with n different groups) [1]. On the contrary, the non-hierarchical approach may be defined "hypothesis-driven" since the number of clusters is defined *a* priori by the researcher. Among these two approaches there are different techniques that can be applied. In this paper we present the *Ward's method* (hierarchical aggregative, known also as *incremental sum of squares method*) and the *k-means* (non-hierarchical). Ward's method uses the within and between clusters squared distances: the two clusters that minimize the increase in the sum of squared errors (i.e. minimize the between-cluster distance) are melted together. The *k-means* method is based on the number of clusters, fixed a priori, and on the distances of each subject from the cluster means' vector (*centroids*). It is an iterative procedure that allows subjects reallocation in different clusters (not possible with hierarchical methods) and it ends when no subject is reallocated.

The main aim of cluster analysis is to identify the observations (subjects) that are similar and to group them into clusters. A numerical measure that is generally used to evaluate proximity between subjects (and so if two observations are similar or not) is the distance. One of the most used distances is the Euclidean one defined as:

$d(xr,xs)=[(xr-xs)'(xr-xs)]1/2=[\Sigma(xrj-xsj)2pj=1]1/2$

Where p is the number of variables, xr, are the p-dimensional observation vectors for the subjects r and s and xrj and xsj are the values of the jth variable for the subjects r and s.

Each cluster is specified by its *centroid*, i.e. the cluster mean vector: $CG=(XG\bar{1},XG\bar{2},\dots,XG\bar{p})$

Where $XG\bar{i}$ is the mean of the ith variable in cluster G, with $1\leq i\leq p$.

The algorithms of the two (hierarchical and non-hierarchical) approaches, with a particular attention to the specific methods used in the manuscript (Ward's method and k-means) are reported below.

Hierarchical approach Preliminar steps:

a. Choice of the dissimilarity measure of the clustering method.
b. Calculation of the dissimilarity between each pair of subjects or clusters.

After the preliminary steps, an agglomerative or a divisive algorithm is chosen. Here only the agglomerative one is presented (since it is the one used in the paper analysis.

Hierarchical agglomerative clustering algorithm:

a. Start with n clusters, one for each object;
b. The two most similar clusters are searched and are melt into a new cluster;
c. Evaluation of similarity (or dissimilarity) between the new clusters and other clusters;
d. Repeat until you have only one cluster with n objects.

A possible choice for the measuring the similarity between clusters is given by the Ward's method that minimizes the sum of the squared distances of points from their cluster mean vector (centroid). In details, it uses the within and between-clusters distances and join the two clusters A (of size nA) and B (of size nB) that minimize the increase IAB

in the sum of squared errors (SSE): $IAB=SSEAB-(SSEA+SSEB)$ where $SSEA=\Sigma(yi-y\bar{A})'(yi-y\bar{A})nAi=1$, $SSEB=\Sigma(yi-y\bar{B})'(yi-y\bar{B})nBi=1$ and $SSEAB=\Sigma(yi-yA\bar{B})'(yi-yA\bar{B})nABi=1$; with $y\bar{A}, y\bar{B}$ and $yA\bar{B}$ being the mean vectors of cluster A, B and the new joint cluster AB. [1]

The result of the hierarchical approach is displayed in a plot called *dendrogram*: a tree diagram representing all the procedure steps including the distances at which clusters are joined. The leaves of the tree represent the subjects while the y-axis (height of the dendrogram) is simply the value of the distance metric between clusters. The number of clusters is defined by cutting (through a horizontal line) the dendrogram at a specific height.

Non-hierarchical approach K-means algorithm:

a. Choice of the dissimilarity measure.
b. Choice of the number of clusters.
c. First random partition of the objects in the k clusters.
d. Calculation of the k centroids.
e. Evaluation of the distances between each object and each centroid.
f. Reallocation of every object in the cluster with the nearest centroid
g. If at least one object has been moved, go back to point 4; else go to point 8.
h. Stop.

Principal Component Analysis (PCA)

Principal component analysis is a data-reduction technique that can be used to reduce a large set of correlated continuous variables $X=(X1,X2,...,Xp)$ into a smaller set of uncorrelated ones, i.e. the principal components $PCj, j<p$, that still contain most of the information of the original set of variables. This technique is based on a constrained optimization problem that aims to find a linear combination of variables that maximizes the amount of data variability explained. Each PC is derived in decreasing order for the amount of data variability explained (therefore the first one has the highest amount of explained variance). The main aim of PCA is to find a reduced number of linear combinations of the observed variables which explain most of the variance in the data [3].

Let's suppose to have $X=(X1,X2,...,Xp)$, a p-dimensional vector of continuous variables with mean μ and covariance matrix Σ. The jth principal component is defined as a linear combination of the variables:
$PCj=aj1X1+aj2X2+\cdots+ajpXp=aj'X$,
Where $aj=(aj1,...,ajp)$ is a p-dimensional vector of loadings.

The purpose is to find the loading vector aj that maximizes the variance of the linear combination $(PCj)=aj'\Sigma aj$ under the constraint that $aj'aj=1$, i.e. to solve the following constrained maximum problem
$\{\max a\ aj'\Sigma aj\ \ aj'aj=1 aj'aj-1=aj-1'aj=0$ with $j=2,...,$ (for $j=1$ the second constraint disappear) by using Lagrange multipliers method [4]. The Constrains ensure orthogonality among the PCs.

A good way to represent graphically PCA results is the *biplot* that shows simultaneously the variables (represented by arrows) and the subjects (represented by points) on a two (or three) axes plot. The axes are given by the first two (or three) PCs. The accuracy of the representation is proportional to the total variance extracted by the depicted components [5]. As in the MCA plot, also for the PCA the association is detected in terms of geometrical distance, so that the variables (arrows) are associated to the closer subject's subgroup.

Partial Least Square Discriminant Analysis (PLS-DA)

Partial Least Square Discriminant Analysis is a classification technique that combines a Partial Least Square Regression (PLS-R) and Linear Discriminant Analysis (LDA); in particular, the PLS-DA is a PLS-R in which the dependent variable is categorical. PLS-R is a method that at the same time allows dimension reduction and the fit of a regression model. It is an approach similar to PCA but, instead of a computation of a new variable (the principal component), a categorical response variable is used [6]. LDA is a method whose purpose is to find a linear combination that allows discriminating two or more groups maximizing their separation.

Therefore, PLS-DA provides a dimension reduction in a discriminant application maximizing among-groups variability [7]. Formally, PLS-R is based on a regression model between the data matrix X and the vector of categories (the group variable) c. The fundamental equations of PLS-DA are the following: $X=TP+E\ \ c=Tq+f$ where T is the score matrix, E and f are the residuals and P and q are respectively the loadings of X and of c (for more details see [8]).

Once the model is built the class membership can be predicted through the equation
$\hat{c}=x\beta$, where β is the regression coefficient vector.

Therefore, a generic subject will be classified i.e. assigned to a category on the basis of the estimated value of c (and it will be assigned to the category with the nearest value). The main advantage of PLS-DA, compared to linear

discriminant analysis, is that it can provide also variables loadings (represented by a bar plot) that allow to identify not only the group a subject belongs to, but also which variables are more helpful to discriminate the subjects in the different classes [8].

Classification Trees (CART)

Classification trees are one of the most popular machine learning algorithms that belong to the family of decision trees, that can be used for both classification and regression purpose. CART are models in which the dependent variable (variable that has to be predicted) is categorical and the independent ones (covariates) can be categorical or quantitative. Classification trees are directed graphs in which there is an initial node that branches too many. Each node represents an independent variable, each edge corresponds to a decision rule and each leaf represents an outcome (a value of the predicted variable). The top node contains the entire sample that is consequently divided into different subsets. If the covariates are quantitative splits are created on the basis of some cut-offs on a scale; if the covariates are categorical, splits are based on the different categories [9]. After computing the entire tree (with all the independent variables) some techniques have to be used to reduce tree dimension and to improve the tree predictive power, reducing over fitting. Among these, one of the most used is p*runing* [10], a method that allows to remove the variables that do not contribute (are not significantly associated) to the final outcome, considering a penalty for the increase of parameters in the model].

Therefore, the final tree shows only the independent variables that are significant predictors of the dependent one (outcome) and, differently from the traditional regression models, those that are not predictors do not influence the final result. The classification trees can be built by a recursive partitioning program using a two-stage procedure [11].

a. The variable which best splits the data into groups (i.e. with the greatest association with the dependent variable) is found. The subjects are divided and this process is repeated separately to each subject subgroup recursively until the subgroups either reach a minimum size or until no improvement (in terms of predictive performance) can be made.
b. A cross-validation (*pruning*) will be performed to trim the full tree, since the full model is quite certainly too complex and over fitted.

s-Code

Multiple Correspondence Analysis

```
> data<- read.csv ("file.csv", sep=";", header=T) # read the data
> data_mca<-data [ , c(a, b, c,...)] #It takes all the rows of data and the columns a, b, c,... that are the columns of the categorical variables you want to analyze. Be careful: data_mca has to contain only categorical variables.
> library (FactoMineR) #If you don't have this package, install it with install.packages("FactoMineR") .
> mcadat<-MCA(data_mca, graph=F, na.method = "Average") #It performs a Multiple Correspondence Analysis.
> cats <- apply(data_mca, 2, function(x) nlevels(as.factor(x)))
> mca_vars_df <- data.frame(mcadat$var$coord, Variable = rep(names(cats), cats))
> mca_obs_df <- data.frame(mcadat$ind$coord)
#To display the MCA results you can use ggplot function (of the library ggplot2) applied to the mcadat object.
library(ggplot2) #If you don't have this package install it with install.packages("ggplot2")
> ggplot(data = mca_obs_df, aes(x = Dim.1, y = Dim.2)) + geom_hline (yintercept = 0, colour = "gray70") +
geom_vline (xintercept = 0, colour = "gray70") + geom_point (colour = "gray50", alpha = 0.7,size=4) +
geom_text (data = mca_vars_df, aes(x = Dim.1, y = Dim.2, label = rownames(mca_vars_df), colour = Variable), size=5.5) + ggtitle ("MCA plot - Variables and Subjects") + scale_colour_discrete (name = "Variables") + theme(plot.title=element_text (size=20,face="bold"), legend.text=element_text (size=15), legend. title=element_text (size=17), axis.title=element_text (size=15), axis.line=element_line (size=1, colour= "black"), panel.grid.major=element_line (colour = "#d3d3d3"), panel.grid.minor=element_blank (), panel.border= element_blank (), panel.background = element_blank ()).
```

Cluster Analysis

```
> data_clus<-data [ , c(a,b,c,...)] #data_clus must contain only continuous variables and the group variable (in our case, diagnosis) and must not contain missing values.
> data_clus<-as.matrix (data_clus).

#Hierarchical approach – Ward's method:
> hc=hclust (dist (data_clus[ ,-1]), method="ward.D") #it performs hierarchical clustering. The object data_clus [ , -1] is a matrix containing only continuous variables (i.e. the group variable in the first column has been excluded)
> dhc=as.dendrogram(hc)
```

#To assign the labels of dendrogram object with new colors:
> data_clus$Group[which(data_clus$Group==1)]<-"A" #(In our case A and B are BPD and BD patients)
> data_clus$Group[which(data_clus$Group ==2)]<-"B"
> colorCodes<- c(A="red", B="blue")
> library(dendextend) #If you don't have this package, install it with *install.packages("dendextend")*
> labels_colors(dhc)<-colorCodes[data_clus$Group][order.dendrogram(dhc)]
> plot (dhc, main="Hierarchical clustering approach – Ward's method", ylab="Height", size= 2)
> legend ("topright", legend = c("A" , "B"), col = c("red", "blue"), pch = c(20,20,4,4,4), bty = "n", pt.cex = 1.5, cex = 0.8, text.col = "black", horiz = FALSE, inset = c(0.1, 0.1)).

#Non hierarchical approach- k-means method:
> library(cluster) #If you don't have this package, install it with *install.packages("cluster")*
> set.seed (123) #It fixes the seed of the random number generator in order to have reproducible results.
> clus <- kmeans(data_clus[.-1], 2, iter.max = 50)
#It performs k-means method
> clus$centers #It provides clusters centroids (variables mean in each cluster)
> clus$size #It provides clusters size (number of subjects in each cluster)
> table (clus$cluster, data_clus$Group) #It provides a frequency table in which rows represent cluster allocation and columns represent the group categories

#To plot the results:
>clusplot (data_clus, clus$cluster, color=T, shade=F, labels=5, lines=2, col.p=as.character(data_clus$Group), col.clus = c("red", "blue"), main="Non-hierarchical clustering approach – k-means method")
> legend ("topleft", bty = "n", legend=c("Cluster 1", "Cluster 2"), pch=c(1,2), cex=0.8).

Principal Component Analysis

> pca<-prcomp (data_clus[, -1], center=T, scale=T) #It performs a principal component analysis.

#To plot the results of PCA:
> Library (pca3d) #If you don't have this package install it with *install.packages ("pca3d")*
> Library (rgl) #If you don't have this package install it with *install.packages ("rgl")*
>pca3d (pca, group=as.factor (data_clus$Group), biplot=TRUE, biplot.vars=3, show.ellipses=F, palette=c ("red, "blue"))
> legend3d ("right", legend=c("A","B"),pch=c(17,19), col=c("red, "blue"), cex=1.5).

Partial Least Squares Discriminant Analysis

> library(mixOmics) #If you don't have this package, install it with *install.packages("mixOmics")*
> dis= splsda(data_clus[,-1], data_clus$Group, ncomp = 3)
> plotLoadings (dis, comp = 1, method = 'median', size.name=1.5, legend.title="Group", legend.color= c("red", "blue"), size.legend=1.2, contrib='max', xlim=c(-0.6,0.6), title="Loading vectors – PLS-DA").

Classification Trees

> Library (rpart) #If you don't have this package, install it with *install.packages ("rpart")*
> Library (rpart.plot) #If you don't have this package, install it with *install.packages ("rpart.plot")*
> Tree=rpart(Group~V1+V2+… +Vj, method ="class", data=data) #V1, V2, …, Vj are the names of the independent variables; Group is the dependent one.
> rpart.plot(tree, extra=4, type=5, box.palette=c("red", "blue"), main="Decision tree").

s-References

1. Rencher AC (2003) Methods of Multivariate Analysis. (2nd edn), Wiley Series in Probability and Statistics, John Wiley & Sons, Inc, USA.

2. Greenacre MJ (1993) Biplot in correspondence analysis. Journal of Applied Statistics 20(2): 251-269.

3. Joliffe IT (2002) Principal Component Analysis. (2nd edn), Springer series in Statistics, Springer-Verlag, New York.

4. Bishop CM (2006) Appendix E of Pattern Recognition and Machine Learning. Springer Science with Business Media, Heidelberg, Germany.

5. Flom PL (2008) An introduction to biplots. NDRI Statistics Support Group.

6. Boulesteix AL, Strimmer K (2006) Partial least squares: a versatile tool for the analysis of high-dimensional genomic data. Brief Bioinform 8(1): 32-44.

7. Barker M, Rayens W (2003) Partial least squares for discrimination. Journal of Chemometrics 17(3): 166-173.

8. Brereton RG, Lloyd GR (2014) Partial least squares discriminant analysis: taking the magic away. Journal of Chemometrics 28(4): 213-225.

9. Wilkinson L (1992) Tree Structured Data Analysis: AID, CHAID and CART. Sawtooth/SYSTAT Joint Software Conference, Sun Valley, ID, USA.

10. Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and Regression Trees. Wadsworth International Group, Chapman & Hall, Belmont, USA.

11. Therneau TM, Atkinson E (2018) An introduction to Recursvie Partitioning Using the RPART Routines. Mayo Foundation, New York.