Factor PD-Clustering

Mireille Gettler Summa, Francesco Palumbo, Cristina Tortora

Factor PD-Clustering

Mireille Gettler Summa[1], Francesco Palumbo[2], Cristina Tortora[1,2].
[1]Université Paris Dauphine CEREMADE CNRS
[2]Università di Napoli Federico II

Abstract

Factorial clustering methods have been developed in recent years thanks to the improving of computational power. These methods perform a linear transformation of data and a clustering on transformed data optimizing a common criterion. Factor PD-clustering is based on Probabilistic Distance clustering (PD-clustering). PD-clustering is an iterative, distribution free, probabilistic, clustering method. Factor PD-clustering makes a linear transformation of original variables into a reduced number of orthogonal ones using a common criterion with PD-Clustering. This paper demonstrates that Tucker3 decomposition permits to obtain this transformation. Factor PD-clustering exploits alternatively Tucker3 decomposition and PD-clustering on transformed data until convergence is achieved. This method can significantly improve the algorithm performance; large datasets can thus be partitioned into clusters with increasing stability and robustness of the results.

keyword

Multivariate analysis, Exploratory data analysis, Clustering, Factorial clustering, Non hierarchical iterative clustering.

1 Introduction

In a wide definition Cluster Analysis is a multivariate analysis technique that seeks to organize information about variables in order to discover homogeneous groups, or "clusters" into data. The presence of groups in data depends on the association structure over the data. Clustering algorithms aim at finding homogeneous groups with respect to their association structure among variables. Proximity measures or distances can be properly used to separate homogeneous groups. A measure of the homogeneity of a group is the variance. Dealing with numerical linearly independent variables, clustering problem consists in minimising the sum of the squared Euclidean distances within classes: within groups deviance.

"The term cluster analysis refers to an entire process where clustering maybe only a step" [Gordon, 1999]. According to Gordon's definition cluster analysis can be sketched in three main stages:

- transformation of data into a similarity/dissimilarity matrix;
- clustering;
- validation.

Transformation of data into similarity/dissimilarity measures depends on data type. On the transformed matrix a clustering method can be applied. Clustering methods can be divided into three main types: hierarchical, non hierarchical and fuzzy [Wedel and Kamakura, 1999]. Non hierarchical clustering methods are considered in this paper. Among them the most well-known and used method is k-means. It is an iterative method that starts with a random initial partition of units and keeps reassigning the units into clusters based on the squared distances between the units and the clusters' centers until the convergence is reached. Interested readers can refer to [Gordon, 1999]. Major k-means issues are that clusters can be sensitive to the choice of the initial centers and that the algorithm could converge to local minima.

The choice of the number of clusters is a well known problem of non hierarchical methods, this problem will not be dealt with in this paper where the number of clusters is assumed as a priori known.

Non hierarchical clustering methods performance can be strongly affected by the dimensionality. Let us consider an $n \times J$ data matrix X, with n number of units and J number of variables. Non hierarchical methods easily deal with large n, however they can fail when J becomes large or very large and when the variables are correlated. They do not converge or they converge into a different solution at each iteration. To cope with these issues, French school of data analysis [Lebart et al., 1984] suggested a strategy to improve the overall quality of clustering that consists in two phases: variables transformation through a factorial method and clustering method on transformed variables. Arabie and Hubert in 1996 [Arabie et al., 1996] fourthly formalized the method and called it *tandem analysis*.

The choice of the factorial method is an important and tricky phase because it will affect the results. Principal factorial methods are [Le Roux and Rouanet, 2004]:

- quantitative data;
  - Principal Component Analysis (PCA);
- binary data;
  - Principal Component Analysis (PCA);
  - Correspondence Analysis (CA);
  - Multiple Correspondence Analysis (MCA);
- nominal data;
  - Multiple Correspondence Analysis (MCA);

The second phase of the *tandem analysis* consists in applying clustering methods.

Tandem analysis exploits the factor analysis capabilities that consist in obtaining a reduced number of uncorrelated variables which are linear transformation of the original ones. This method gives more stability to the results and makes the procedure faster. However tandem analysis minimises two different functions that can be in contrast and the first factorial step can in part obscure or mask the clustering structure.

This technique has the advantage of working with a reduced number of variables that are orthogonal and ordered with respect to the borrowed information. Moreover dimensionality reduction permits to visualize the cluster structure in two or three dimensional factorial space [Palumbo et al., 2008].

To cope with these issues Vichi and Kiers [Vichi and Kiers, 2001] proposed Factorial k-means analysis for two-way data. The aim of this method is to identify the best partition of the objects and to find a subset of factors that best describe the classification according to the least squares criterion. Two steps Alternating Least Squares algorithm (ALS) based on solves this problem. The advantage of Factorial k-means is that the two steps optimize a single objective function. However the k-means algorithm itself, and as a consequence the tandem analysis and Factorial k-means, is based on the arithmetic mean that gives rise to unsatisfactory solutions when clusters have not spherical shape.

Probabilistic clustering methods may allow us to obtain better results under this condition because they assign a statistical unit to a cluster according to a probability function that can be independently defined with respect to the arithmetic mean.
Probabilistic Distance clustering (PD-clustering) [Ben-Israel and Iyigun, 2008] is an iterative, distribution free, probabilistic, clustering method. PD-clustering assigns units to a cluster according to their probability of belonging to the cluster, under the constraint that the product between the probability and the distance of each point to any cluster center is a constant.
When the number of variables is large and variables are correlated, PD-clustering becomes unstable and the correlation between variables can hide the real number of clusters. A linear transformation of original variables into a reduced number of orthogonal ones using common criteria with PD-clustering can significantly improve the algorithm performance. The objective of this paper is to introduce an improved version of PD-clustering called Factor PD-clustering (FPDC).

The paper has the following structure: section 2: detailed presentation of PD-clustering method; section 3: presentation of our suggestion for a Factor PD-clustering method; section 4: application of Factor PD-clustering on a simulated case study and comparison with k-means.

2 Probabilistic Distance Clustering

PD-clustering is a non hierarchical algorithm that assigns units to clusters according to their belonging probability to the cluster. According to Ben-Israel and Iyigun [Ben-Israel and Iyigun, 2008] notation we introduce PD-clustering. Given some random centers, the probability of any point to belong to each class is assumed to be inversely proportional to the distance from the centers of the clusters. Given an X data matrix with n units and J variables, given K clusters that are assumed not empty, PD-Clustering is based on two quantities: the distance of each data point x_i from the K cluster centers c_k, $d(x_i,c_k)$, and the probabilities for each point to belong to a cluster, $p(x_i,c_k)$ with $k=1,\ldots,K$ and $i=1,\ldots,n$. The relation between them is the basic assumption of the method. Let us consider the general term x_{ij} of X and a center matrix C, of elements c_{kj} with $k=1,\ldots,K$, $i=1,\ldots,n$ and $j=1,\ldots,J$, their distance can be computed according to different criteria, the squared norm is one of the most commonly used. The generic distance $d(x_i,c_k)$ represents the distance of the generic point i to the generic center k. The probability $p(x_i,c_k)$ of each point to belong to a cluster can be computed according to the following assumption: the product between the distances and the probabilities is a constant depending on x_i: $F(x_i)$.
For short we use $p_{ik}=p(x_i,c_k)$ and $d_k(x_i)=d(x_i,c_k)$; PD-clustering basic assumption is expressed as:

$$p_{ik}d_k(x_i)=F(x_i). \tag{1}$$

for a given value of x_i and for all $k=1,\ldots,K$.

At the decreasing of the point closeness from the cluster center the belonging probability of the point to the cluster decreases. The constant depends only on the point and does not depend on the cluster k.
Starting from the 1 it is possible to compute p_{ik}:

$$p_{im}d_m(x_i)=p_{ik}d_k(x_i);\ p_{im}=\frac{p_{ik}d_k(x_i)}{d_m(x_i)};\ \forall m=1,\ldots,K \tag{2}$$

The term p_{ik} is a probability so, under the constraint $\sum_{m=1}^{K} p_{im} = 1$, the sum over m of 2 becomes:

$$p_{ik} \sum_{m=1}^{K} \left(\frac{d_k(x_i)}{d_m(x_i)} \right) = 1,$$

$$p_{ik} = \left(\sum_{m=1}^{K} \left(\frac{d_k(x_i)}{d_m(x_i)} \right) \right)^{-1} = \frac{\prod_{m \neq k} d_m(x_i)}{\sum_{m=1}^{K} \prod_{k \neq m} d_k(x_i)}, k = 1, \dots K. \tag{3}$$

Starting from the 1 and using 3 it is possible to define the value of the constant $F(x_i)$:

$$F(x_i) = p_{ik} d_k(x_i), k = 1, \dots K,$$

$$F(x_i) = \frac{\prod_{m=1}^{K} d_m(x_i)}{\sum_{m=1}^{K} \prod_{k \neq m} d_k(x_i)}. \tag{4}$$

The quantity $F(x_i)$, also called *Joint Distance Function* (JDF), is a measure of the closeness of x_i from all clusters' centers. The JDF measures the classificability of the point x_i with respect to the centers c_k with $k = 1, \dots, K$. If it is equal to zero, the point coincides with one of the clusters' centers, in this case the point belongs to the class with probability 1. If all the distances between the point x_i and the centers of the classes are equal to d_i, $F(x_i) = d_i/k$ and all the belonging probabilities to each class are equal: $p_{ik} = 1/K$. The smaller the JDF value, the higher the probability for the point to belong to one cluster.

The whole clustering problem consists in the identification of the centers that minimises the JDF. Without loss of generality the PD-Clustering optimality criterium can be demonstrated according to $k = 2$.

$$\begin{aligned} \min & \left(d_1(x_i) p_{i1}^2 + d_2(x_i) p_{i2}^2 \right) \\ \text{s.t.} & \quad p_{i1} + p_{i2} = 1 \\ & \quad p_{i1}, p_{i2} \geq 0 \end{aligned} \tag{5}$$

The probabilities are squared because it is a smoothed version of the original function. The Lagrangian of this problem is:

$$\mathscr{L}(p_{i1}, p_{i2}, \lambda) = d_1(x_i) p_{i1}^2 + d_2(x_i) p_{i2}^2 - \lambda (p_{i1} + p_{i2} - 1) \tag{6}$$

Setting to zero the partial derivates with respect to p_{i1} and p_{i2}, substituting the probabilities 3 and considering the principle $p_{i1} d_1(x_i) = p_{i2} d_2(x_i)$ we obtain the optimal value of the Lagrangian.

$$\mathscr{L}(p_{i1}, p_{i2}, \lambda) = \frac{d_1(x_i) d_2(x_i)}{d_1(x_i) + d_2(x_i)}. \tag{7}$$

This value coincides with the JDF, the matrix of centers that minimises this principle minimises the JDF too. Substituting the generic value $d_k(x_i)$ with $\|x_i - c_k\|$, we can find the equations of the centers that minimise the JDF (and maximize the probability of each point to belong to only one cluster).

$$c_k = \sum_{i=1,\dots,N} \left(\frac{u_k(x_i)}{\sum_{j=1,\dots,N} u_k(x_j)} \right) x_i, \tag{8}$$

where

$$u_k(x_i) = \frac{p_{ik}^2}{d_k(x_i)}. \tag{9}$$

As showed before, the value of JDF at all centers k is equal to zero and it is necessarily positive elsewhere. So the centers are the global minimiser of the JDF. Other stationary points may exist because the function is not convex neither quasi-convex, but they are saddle points.

There are alternative ways for modeling the relation between probabilities and distances, for example the probabilities can decay exponentially as distances increase. In this case the probabilities p_{ik} and the distances $d_k(x_i)$ are related by:

$$p_{ik}e^{d_k(x_i)} = E(x_i), \tag{10}$$

where $E(x_i)$ is a constant depending on x_i.
Many results of the previous case can be extended to this case by replacing the distance $d_k(x_i)$ with $e^{d_k(x_i)}$. Interested readers are referred to Ben-Israel and Iyigun [Ben-Israel and Iyigun, 2008].

The optimization problem presented in 5 is the original version proposed by Ben-Israel and Iyigun. Notice that in the optimization problem the probabilities p_k are considered in squared form. The Authors affirm that it is possible to consider d_k as well d_k^2. Both choices have some advantages and drawbacks. Squared distances offer analytical advantages due to linear derivates. Using simple distances endures more robust results and the optimization problem can be reconducted to a Fermat-Weber location problem. The Fermat-Weber location problem aims at finding a point that minimises the sum of the Euclidean distances from a set of given points. This problem can be solved with the Weiszfeld method [Weiszfeld, 1937]. Convergence of this method was established by modifying the gradient so that it is always defined [Khun, 1973]. The modification is not carried out in practice. The global solution is guaranteed only in case of one cluster. Dealing with more than one cluster, in practice, the method converges only for a limited number of centers depending on the data.

In this paper we consider the squared form:

$$d_k(x_i) = \sum_{j=1}^{J}(x_{ij} - c_{kj})^2, \tag{11}$$

where $k = 1,\dots,K$ and $i = 1,\dots,N$. Starting from the 11 the distance matrix D of order $n \times K$ is defined, where the general element is $d_k(x_i)$. The final solution $J\hat{D}F$ is obtained minimising the quantity:

$$JDF = \sum_{i=1}^{n}\sum_{k=1}^{K} d_k(x_i)p_{ik}^2 = \sum_{i=1}^{n}\sum_{j=1}^{J}\sum_{k=1}^{K}(x_{ij} - c_{kj})^2 p_{ik}^2, \tag{12}$$

$$J\hat{D}F = \arg\min_{C;P} \sum_{i=1}^{n}\sum_{j=1}^{J}\sum_{k=1}^{K}(x_{ij} - c_{kj})^2 p_{ik}^2. \tag{13}$$

Where c_k is the generic center and $d_k(x_i)$ is defined in 11.

The solution of PD-clustering problem can be obtained through an iterative algorithm.

The algorithm convergence is demonstrated in [Iyigun, 2007].
Each unit is then assigned to the k^{th} cluster according to the highest probability that is computed a posteriori using the formula in equation 3.

Algorithm 1 Probabilistic Distance Clustering Function

```
function PDC($X,K$)
    $C \leftarrow \text{rand}(K,J)$                         ▷ Matrix $C_{K,J}$ is randomly initialised
    $JDF \leftarrow 1/\text{eps}$                           ▷ $JDF$ is initialised to the maximum
    $D \leftarrow 0$                                        ▷ Initialise the array $D$, of dimension $n \times K$, to 0
    $p \leftarrow \frac{1}{k}$                              ▷ Initialise to $\frac{1}{k}$ the probability vector $p$ of $n$ elements
    repeat
        for $k = 1,K$ do
            $D_k \leftarrow distance(X,C(k))$               ▷ $D_k$ distances of all units from the centre $k$ according to formula 11
        end for
        $JDF0 \leftarrow JDF$                               ▷ Current $JDF$ is stored in $JDF0$
        $C \leftarrow C^*$                                  ▷ Centres are updated according to formula 8
        $JDF \leftarrow jdf(D)$                             ▷ $\leftarrow jdf(D)$ implements the formula 4
    until $JDF0 > JDF$
    $P \leftarrow compp(D)$                                 ▷ function $compp$ implements the formula 3
     return $C,P,JDF$
end function
```

3 Factor PD-Clustering

When the number of variables is large and variables are correlated, PD-Clustering becomes very unstable and the correlation between variables can hide the real number of clusters. A linear transformation of original variables into a reduced number of orthogonal ones can significantly improve the algorithm performance. Combination of PD-Clustering and variables linear transformation implies a common criterion.

This section shows how the Tucker3 method [Kroonenberg, 2008] can be properly adopted for the transformation into the Factor PD-Clustering; an algorithm is then proposed to perform the method.

3.1 Theoretical approach to Factor PD-clustering

Firstly we demonstrate that the minimization problem in 12 corresponds to the Tucker3 decomposition of the distance matrix G of general elements $g_{ijk} = |x_{ij} - c_{kj}|$. It is a 3-way matrix $n \times J \times K$ where n is the number of units, J the number of variables and K the occasions. For any c_k with $k = 1,\dots,K$, a G_k $n \times J$ distances matrix is defined. In matrix notation:

$$G_k = X - hc_k \tag{14}$$

where h is an $n \times 1$ column vector with all terms equal to 1; X and c_k $(k = 1,\dots,K)$ have been already defined in section 2.

Tucker3 method decomposes the matrix G in three components, one for each mode, in a full core array Λ and in an error term E.

$$g_{ijk} = \sum_{r=1}^{R}\sum_{q=1}^{Q}\sum_{s=1}^{S}\lambda_{rqs}(u_{ir}b_{jq}v_{ks}) + e_{ijk}, \tag{15}$$

where λ_{rqs} and e_{ijk} are respectively the general terms of the three way matrix Λ of order $R \times S \times Q$ and E of order $n \times J \times K$;

u_{ir}, b_{jq} and v_{ks} are respectively the general terms of the matrix U of order $n \times R$, B of order $J \times Q$ and V of order $K \times S$, with $i = 1,\dots,n$, $j = 1,\dots,J$, $k = 1,\dots,K$.

As in all factorial methods, factorial axes in Tucker3 model are sorted according to explained variability. The first factorial axes explain the greatest part of the variability, latest factors are influenced by anomalous data or represent the ground noise. For this reason the choice of a

number of factors lower than the number of variables makes the method externally robust. According to [Kiers and Kinderen, 2003] the choice of the parameters R, Q and S is a ticklish problem because they define the overall explained variability. The interested readers are referred to [Kroonenberg, 2008] for the theoretical aspects concerning this choice. We use an heuristic approach to cope with this crucial issue: we choose the minimum number of factors that corresponds to a significant value of the explained variability.
The coordinates x^*_{iq} of the generic unit x_i into the space of variables obtained through Tucker3 decomposition are obtained by the following expression:

$$x^*_{iq} = \sum_{j=1}^{J} x_{ij} b_{jq}. \tag{16}$$

Finally on these x^*_{iq} coordinates a PD-Clustering is applied in order to solve the clustering problem.

Let us start considering the expression 12; it is worth noting that minimising the quantity:

$$JDF = \textstyle\sum_{i=1}^{n} \sum_{j=1}^{J} \sum_{k=1}^{K} (x_{ij} - c_{kj})^2 p^2_{ik} \qquad \text{s.t.} \quad \sum_{i=1}^{n} \sum_{k=1}^{K} p^2_{ik} \leq n, \tag{17}$$

is equivalent to compute the maximum of $-\sum_{i=1}^{n} \sum_{j=1}^{J} \sum_{k=1}^{K} (x_{ij} - c_{kj})^2 p^2_{ik}$, under the same constraints.

Taking into account the Proposition 1 (proof in A.1) and the following lemma, we demonstrate that the Tucker3 decomposition is a consistent linear variable transformation that determines the best subspace according to the PD-clustering criterion.
Proposition 1 *Given an unknown matrix B of generic element b_{im} and a set of coefficients $0 \leq \psi_{im} \leq 1$, with $m = 1, \ldots, M$ and $i = 1, \ldots, n$. Maximising*

$$-\sum_{m=1}^{M} \sum_{i=1}^{n} b_{im} \psi^2_{im},$$

s.t. $\sum_{m=1}^{M} \sum_{i=1}^{n} \psi^2_{im} \leq n$ *is equivalent to solve the equation*

$$\sum_{m=1}^{M} \sum_{i=1}^{n} b_{im} \psi_{im} = \mu \sum_{m=1}^{M} \sum_{i=1}^{n} \psi_{im},$$

where $\mu \geq 0$.

Lemma. *Tucker3 decomposition permits to define the best subspace for the PD-clustering.*

We consider the proposition of the *Proposition 1* where:

$$\begin{aligned} M &= K \\ b_{ik} &= \sum_{j=1}^{J} (x_{ij} - c_{kj})^2 \\ \text{and} & \\ \psi_{ik} &= p_{ik}, \qquad \text{with } i = 1, \ldots, n: \; k = 1, \ldots, K \end{aligned} \tag{18}$$

Let us assume that c_{kj} and p_{ik} are known, replacing $(x_{ij} - c_{kj})$ with g_{ijk} in 17 we develop the following squared form:

$$\begin{aligned} \max \quad & \left(-\textstyle\sum_{k=1}^{K} \sum_{i=1}^{n} \left(\sum_{j=1}^{J} g^2_{ijk} \right) p^2_{ik} \right) \\ \text{s.t.} \quad & \textstyle\sum_{k=1}^{K} \sum_{i=1}^{n} p^2_{ik} \leq n \end{aligned}$$

according to the Theorem 1 we obtain:

$$\sum_{k=1}^{K}\sum_{i=1}^{n}\left(\sum_{j=1}^{J} g_{ijk}^{2}\right) p_{ik} = \mu \sum_{k=1}^{K}\sum_{i=1}^{n} p_{ik} \tag{19}$$

The value of μ that optimize the 19 can be find trough the singular value decomposition of the matrix G, which is equivalent to the following Tucker3 decomposition:

$$g_{ijk} = \sum_{r=1}^{R}\sum_{q=1}^{Q}\sum_{s=1}^{S} \lambda_{rqs}(u_{ir} b_{jq} v_{ks}) + e_{ijk},$$

with $i = 1,\dots,n$, $j = 1,\dots,J$, $k = 1,\dots,K$.
Defining with: R number of components of U, Q number of components of B and S number of components of V.
In matrix notation:

$$G = U\Lambda(V' \otimes B') + E \tag{20}$$

It can be verified that the second derivate is not positive, see section A.2.
■

The Proposition 1 and the Lemma 1 demonstrate that the Tucker3 transformation of the distance matrix G minimises the JDF. The following subsection presents an iterative algorithm to alternatively calculate c_{kj} and p_{ik} on one hand, and b_{jq} on the other hand, until the convergence is reached. In A.2 we empirically demonstrate that the minimisation of the quantity in the formula 17 converges at least to local minima.

3.2 Factor PD-clustering iterative algorithm

Let us start considering the equation 13, where we apply the linear transformation $x_{ij}b_{jq}$ to x_{ij} according to 16:

$$J\hat{D}F = \arg\min_{C;B} \sum_{i=1}^{n}\sum_{q=1}^{Q}\sum_{k=1}^{K} (x_{iq}^{*} - c_{kq})^{2} p_{ik}^{2}. \tag{21}$$

Let us note that in formula 21:
x_{ij} and b_{jq} are the general elements of the matrices X and B that have been already defined in section 3.1;
c_{kq} is the general element of the matrix C, (see eq. 8).

It is worth to note that C and B are unknown matrices and p_{ik} is determined as C and B are fixed. The problem does not admit a direct solution and an iterative two steps procedure is required. The two alternative steps are:

- Linear transformation of original data;
- PD-Clustering on transformed data.

The procedure starts with a pseudorandomly defined centre matrix C of elements c_{kj} with $k = 1,\dots,K$ and $j = 1,\dots,J$. Then a first solution for probabilities and distance matrices is computed according to 14. Given the initial C and X, the matrix B is calculated; once B is fixed the matrix C is updated (and the values p_{ik} are consequently updated). Last two steps are iterated until the convergence is reached: $J\hat{D}F^{(t)} - J\hat{D}F^{(t-1)} > 0$, where t indicates the number of iterations.

Here under the procedure is presented according to the usual flow diagram notation:

Remark that the Tucker3 function is in MatLab Toolbox N-way [Chen, 2010].

Algorithm 2 Factor Probabilistic Distance Clustering

```
function FPDC($X,K$)
    $JDF \leftarrow 1/\text{eps}$                        ▷ $JDF$ is initialised to the maximum
    $G \leftarrow 0$                                     ▷ Initialise the array $G$, of dimension $n \times J \times K$, to 0
    $P \leftarrow \frac{1}{k}$                           ▷ Initialise to $\frac{1}{k}$ the probability vector $p$ of $n$ elements
    $C \leftarrow \text{rand}(K,J)$                      ▷ Matrix $C_{K,J}$ is randomly initialised
    repeat
        for $k = 1,K$ do
            $G_k \leftarrow distance(X,C(k))$            ▷ $G_k$ distances of all units from the centre $k$
        end for
        $B \leftarrow Tucker3(G)$                        ▷ Tucker3 fun. in MatLab Toolbox N-way [Chen, 2010]
        $X^* \leftarrow XB$
        $JDF0 \leftarrow JDF$                            ▷ Current $JDF$ is stored in $JDF0$
        $(C,P,JDF) \leftarrow PDC(X^*,K)$                ▷ PDC() function is defined by the algorithm 1
    until $JDF0 > JDF$
     return $C,P$
end function
```

4 Application on a simulated dataset

In order to evaluate the performance of FPDC it has been applied on a simulated dataset. The dataset has been created according to Maronna and Zamar [Maronna and Zamar, 2002] procedure and notations.

Every cluster has been obtained generating uncorrelated normal data $x_i \sim N(0,I)$ where I is a $J \times J$ identity matrix. Each element x_i has been transformed into $y_i = \Sigma x_i$ where Σ^2 is a covariance matrix with $\Sigma_{jj} = 1$ and $\Sigma_{jr} = \rho$ for $r \neq j$. For every cluster 100 vectors y_i with 7 variables have been generated. Every cluster has been centered on points which are uniformly distributed on a hypersphere. Each cluster has been contaminated at a level $\varepsilon = 20\%$, cluster contamination is generated according to a normal distribution $y_i \sim N(ra_0\sqrt{J}, \Sigma_k)$ where a_0 is a unitary vector generated orthogonal to $(1,1,\dots,1)^T$. The parameter r measures the distance between the outliers and the cluster center. To avoid that outliers overlap the elements of the clusters the minimum value of r is $r_{min} = \frac{\left(1.2\sqrt{\chi^2_{J,1-\alpha}} + \sqrt{\chi^2_{J,1-\alpha}}\right)}{\sqrt{J}}$. In this case we have chosen $r = 4$ that verifies $r > r_{min}$. In order to evaluate the stability of the results each method has been iterated 100 times, JDF has been measured at each iteration; results are represented in fig. 2.

The modal percentile is obtained in 59% of cases, the JDF is included in the interval $[975, 983]$. In this percentile the maximum variation in clustering structure is 1% that corresponds to six units. In 59% of cases the error term is in the interval $[0,21\%, 1,5\%]$. The clustering structure on the first three variables is represented in fig. 3.

A well known problem in cluster analysis is the validation of clustering structure. There is no index that measures clustering results because each clustering method optimizes a different function. In order to evaluate the cluster partition a density based silhouette plot (dbs) can be used. According to this method the dbs index is measured for all the observations x_i, all the clusters are sorted in a decreasing order with respect to dbs and plotted on a bar graph, fig. 4. Usually euclidean distance is used to measure the distance between clusters center and each datapoint; however Euclidean distance is not suitable dealing with probabilistic clustering. A measure of dbs for probabilistic clustering method is proposed in Menardi [Menardi, 2011]. An adaptation of this measure for FPDC is the following one:

$$dbs_i = \frac{log\left(\frac{p_{im_k}}{p_{im_1}}\right)}{\max_{i=1,\dots,n} |log\left(\frac{p_{im_k}}{p_{im_1}}\right)|}, \tag{22}$$

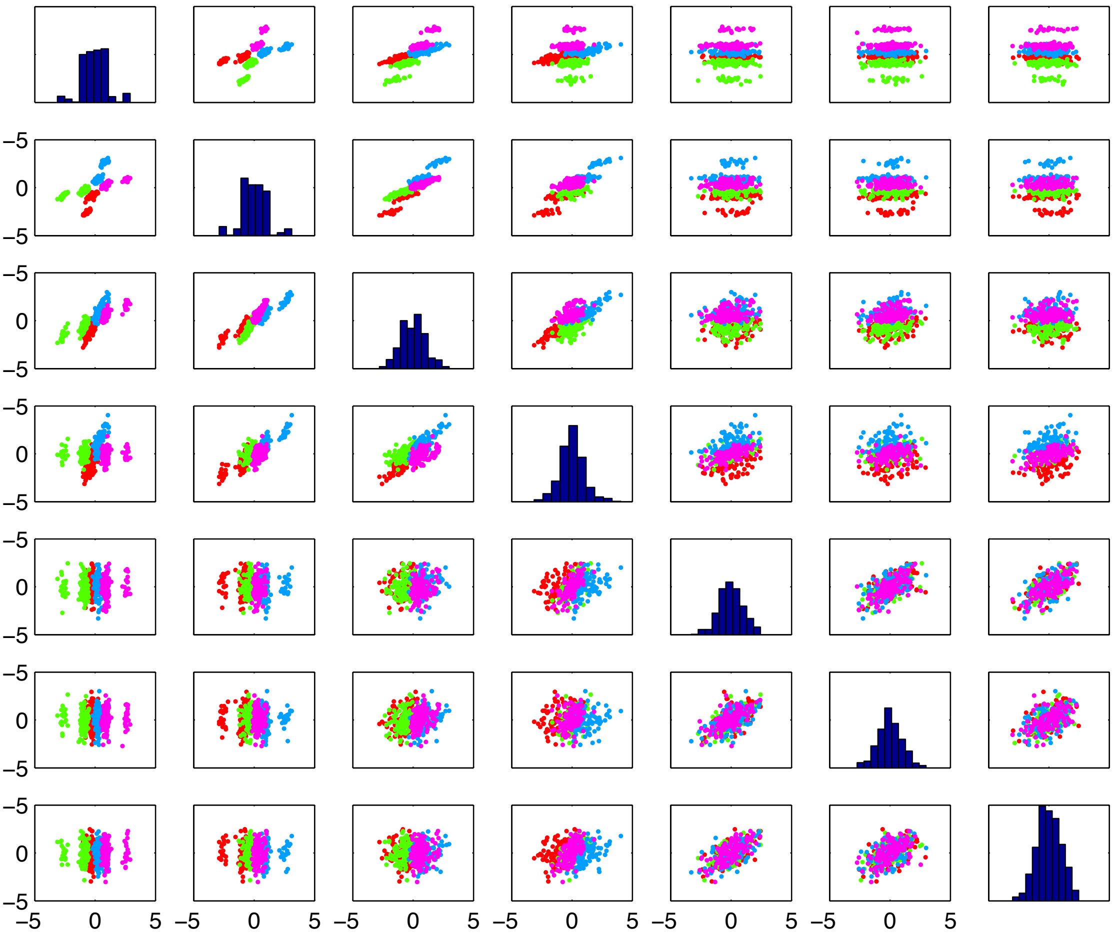

Figure 1: Scatter plot matrix of the simulated dataset. The dataset represents 4 normally generated cluster with a level of contamination of 20% and correlated according to the scheme in the section 4. Displayed data have been standardized.

where m_k is such that x_i belongs to cluster k and m_1 is such that p_{im_1} is maximum for $m \neq m_k$. The graphic shows that the clustering structure is correct.

Although an index that compares clustering structure does not exist, in order to point out the quality of FPDC the dataset has been partitioned using k-means method too. The method has been iterated 100 times, the within variance has been measured at each iteration, results are represented in fig. 5.

The results have an high variability, the modal case is obtained 15% of times, the first percentile is obtained 24% of times. In all resulting clustering structures there is high percentage of error due to outliers. Results obtained in the modal case are represented in fig. 6.

5 Conclusion and perspectives

In this paper a new factorial two-step clustering method has been brought up: Factor PD-clustering. This method can be inlaid into a new field of clustering techniques which has been developed in recent years: iterative clustering methods. Two-step clustering methods were proposed by French school of data analysis in order to cope with some clustering issues. Thanks to computer developing, recently, iterative clustering methods have been introduced. These methods optimize

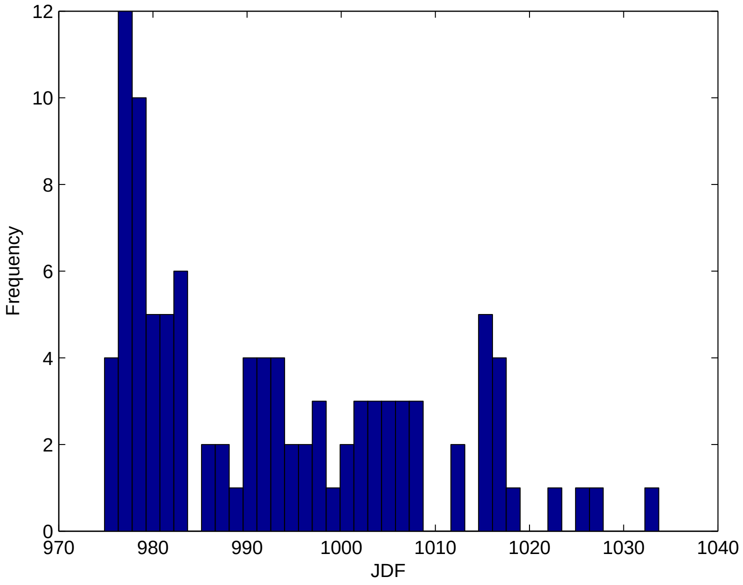

Figure 2: The bar-graph represents the distribution of JDF obtained through 100 FPDC iterations on the simulated dataset. The picture shows the stability of the results. The modal percentile is $[975,983]$ and corresponds to 59% of cases.

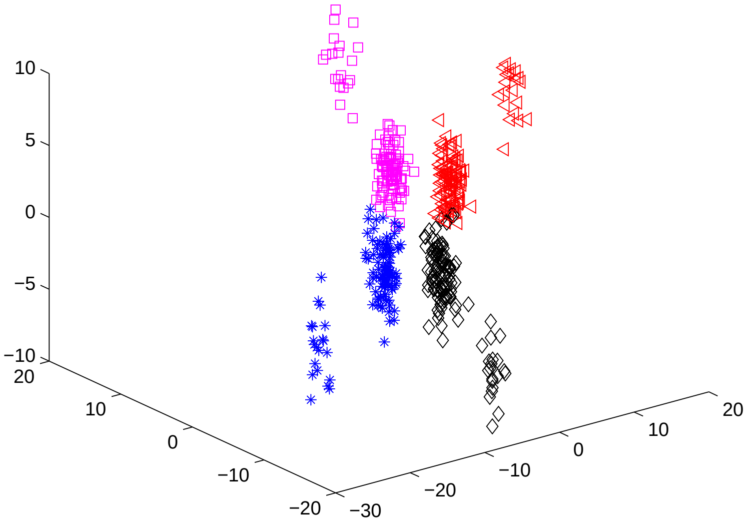

Figure 3: The figure shows the FPDC results of the simulated dataset composed by 4 clusters. The axes correspond to the first 3 simulated variables see also 1. Colors and symbols are referred to FPDC results. The misclassification error rate is $[0,21\%,1,5\%]$.

a common criterion iteratively performing a linear transformation of data and a clustering optimizing a common criterion. Factor PD-clustering performs a linear transformation of data and Probabilistic D-clustering iteratively. Probabilistic D-clustering is an iterative, distribution free, probabilistic, clustering method. When the number of variables is large and variables are cor-

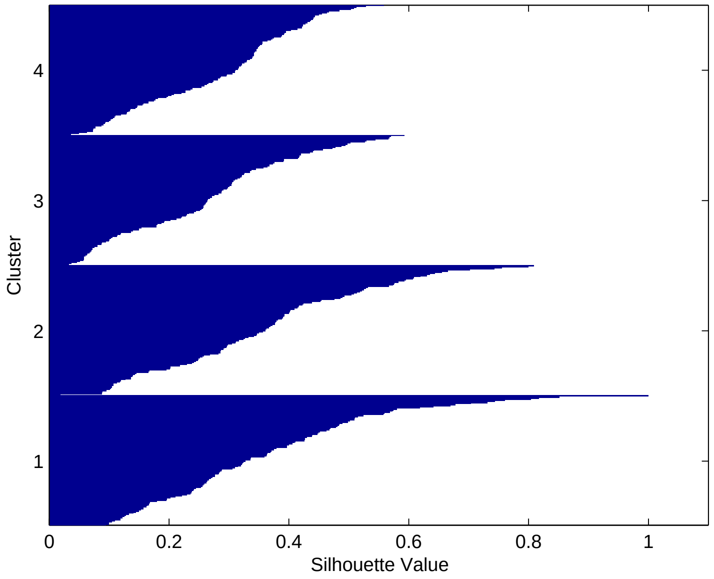

Figure 4: The figure represents density based silhouette plot on clusters obtained in the modal value of the JDF on 100 FPDC iterations. The graphic shows that points have been rightly classified.

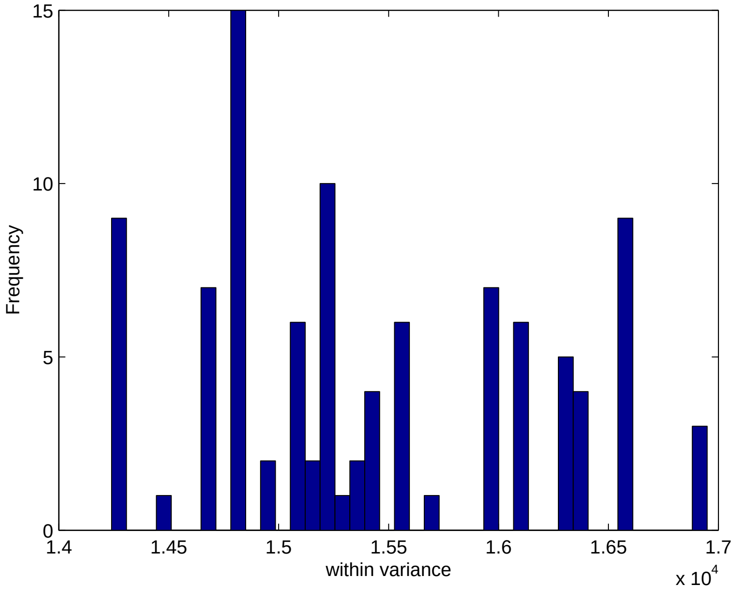

Figure 5: The bar-graph represents the distribution of within variance obtained through 100 k-means iterations on the simulated dataset. The picture shows the stability of the results. The modal percentile corresponds to 24% of cases.

related PD-Clustering becomes unstable and the correlation between variables can hide the real number of clusters. A linear transformation of original variables into a reduced number of orthogonal ones using common criteria with PD-Clustering can significantly improve the algorithm performance. Factor PD-clustering allows to work with large dataset improving the stability and

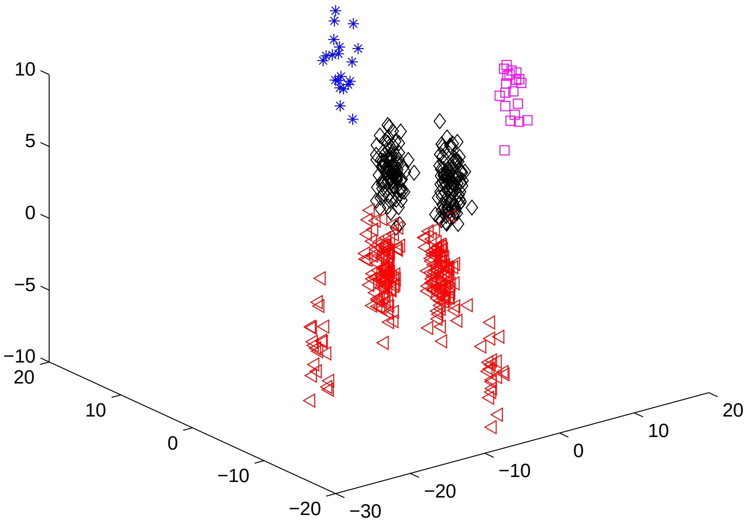

Figure 6: The figure shows the k-means results of the simulated dataset composed by 4 clusters. The axes correspond to the first 3 simulated variables (see also 1). Colors and symbols are referred to k-means results. The method does not find the right clustering structure in the dataset.

the robustness of the method.

An important issue in the future research is the FPDC generalization to the case of categorical data. Dealing with big nominal and binary data matrices, the sparseness of data and the non-linearity in the association can be more prejudicial to the overall cluster stability. In this context, Factor clustering represents a suitable solution. Some methods have been already presented, it is worth mentioning the contributions of Hwang *et al.* [Hwang et al., 2006] and of Palumbo and Iodice D'Enza [Iodice D'Enza and Palumbo, 2010], in the case of nominal data and of binary data, respectively.

A Appendix

This appendix contains two proves: the first for proposition 1 and the second for second order optimality condition demonstration.

A.1 Proof of the Proposition 1

Proposition 1 *Given an unknown matrix B of generic element b_{im} and a set of coefficients* $0 \leq \psi_{im} \leq 1$*, with* $m = 1, \ldots, M$ *and* $i = 1, \ldots, n$*.* Maximising

$$-\sum_{m=1}^{M}\sum_{i=1}^{n} b_{im}\psi_{im}^2,$$

s.t. $\sum_{m=1}^{M}\sum_{i=1}^{n}\psi_{im}^2 \leq n$ *is equivalent to solve the equation*

$$\sum_{m=1}^{M}\sum_{i=1}^{n} b_{im}\psi_{im} = \mu\sum_{m=1}^{M}\sum_{i=1}^{n}\psi_{im},$$

where $\mu \geq 0$.

Proof (Proposition 1)**.** *To prove the proposition we introduce the Lagrangian function:*

$$\mathscr{L} = -\sum_{m=1}^{M}\sum_{i=1}^{n} b_{im}\psi_{im}^{2} + \mu\left(\sum_{m=1}^{M}\sum_{i=1}^{n}\psi_{im}^{2} - n\right)$$

where μ *is the Lagrange multiplier. Let us consider the first derivative of* $\mathscr{L}$ *w.r.t.* ψ_{im} *equal to 0 :*

$$\frac{\partial \mathscr{L}}{\partial \psi_{im}} = -2\sum_{m=1}^{M}\sum_{i=1}^{n} b_{im}\psi_{im} + 2\mu\sum_{m=1}^{M}\sum_{i=1}^{n}\psi_{im} = 0$$

which is equivalent to

$$\sum_{m=1}^{M}\sum_{i=1}^{n} b_{im}\psi_{im} = \mu\sum_{m=1}^{M}\sum_{i=1}^{n}\psi_{im}$$

■

A.2 FPD-Clustering algorithm convergence

In general the proof of the algorithm convergence requires the demonstration of the convexity of the objective function. Dealing with multivariate data, the analytical proof of the convexity becomes a complex issue. In most multivariate situations the empirical evidence is a satisfactory approach to verify the algorithm convergence. Moreover the high capacity of modern CPU permits to get the minimum, avoiding local minima, through the *multiple starts* of the algorithm. This section aims at empirically showing the procedure convergence whereas a simulation study has been conducted by [Tortora and Marino, 2011]. The proposition states that the convergence to a global or to a local maximum is guaranteed. Two data sets are generated; the first one is the one used in section 4. The second one is a simulated 450 × 2 four clusters dataset where variables are independent (see fig. 8). The four clusters have been generated according to four normal distributions with different number of elements.

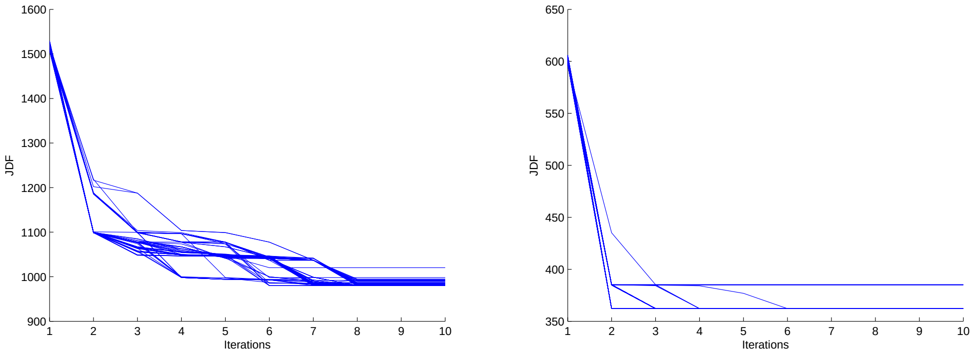

Figure 7: The displays represent the JDF behavior at each iteration of FPDC algorithm obtained along 100 iterations on two simulated datasets.

Figure 7 represents the following results: on the left-hand side the convergence of the dataset one, the right-hand side of the dataset two. The horizontal axis represents the number of iterations, the vertical refers to the value of JDF. Each broken line represents the value of the criterion at each iteration. When convergence is reached the line is straight and parallel to the horizontal axis. In both cases the procedure converges in a limited number of iterations. It is worth to note that the first iteration is not counted.

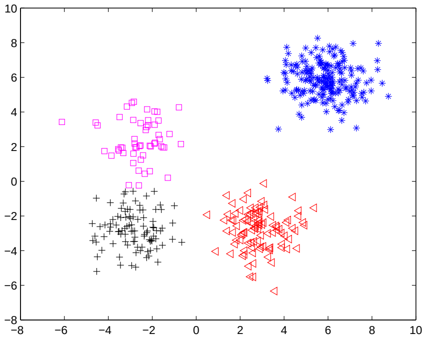

Figure 8: The figure represent the simulated 450 × 2 four clusters dataset.

References

[Arabie et al., 1996] Arabie, P., Hubert, L. J., and De Soete, G. (1996). *Clustering and Classification*. Word Scientific.

[Ben-Israel and Iyigun, 2008] Ben-Israel, A. and Iyigun, C. (2008). Probabilistic d-clustering. *Journal of Classification*, 25(1):5–26.

[Chen, 2010] Chen, H. (2010). N-way toolbox for matlab. *http://www.models.life.ku.dk/algorithms*, Accessed 20 June 2011.

[Gordon, 1999] Gordon, A. D. (1999). *Classification*. Chapman and Hall/CRC, 2nd edition.

[Hwang et al., 2006] Hwang, H., Dillon, W. R., and Takane, Y. (2006). An extension of multiple correspondence analysis for identifying heterogenous subgroups of respondents. *Psychometrika*, 71:161–171.

[Iodice D'Enza and Palumbo, 2010] Iodice D'Enza, A. and Palumbo, F. (2010). Clustering and dimensionality reduction to discover interesting patterns in binary data. *Advances in Data Analysis, Data Handling and Business Intelligence*, pages 45–55.

[Iyigun, 2007] Iyigun, C. (2007). *Probabilistic Distance Clustering*. Ph.D. thesis at, New Brunswick Rutgers, The State University of New Jersey.

[Khun, 1973] Khun, H. W. (1973). *A note on Fermat's problem*, volume 4 of *Mathematical programming*. Spinger.

[Kiers and Kinderen, 2003] Kiers, H. and Kinderen, A. (2003). A fast method for choosing the numbers of components in tucker3 analysis. *British Journal of Mathematical and Statistical Psychology*, 56(1):119–125.

[Kroonenberg, 2008] Kroonenberg, P. (2008). *Applied multiway data analysis*. Ebooks Corporation.

[Le Roux and Rouanet, 2004] Le Roux , B. and Rouanet, H. (2004). *Geometric data analysis*. Kluwer Academic Publishers.

[Lebart et al., 1984] Lebart, A., Morineau, A., and Warwick, K. (1984). Multivariate statistical descriptive analysis. *New York: Wiley*.

[Maronna and Zamar, 2002] Maronna, R. A. and Zamar, R. H. (2002). Robust estimates of location and dispersion for high-dimensional datasets. *Technometrics*, 44(4):307–317.

[Menardi, 2011] Menardi, G. (2011). Density-based Silhouette diagnostics for clustering methods. *Statistics and Computing*, pages 295–308.

[Palumbo et al., 2008] Palumbo, F., Vistocco, D., and Morineau, A. (2008). *Huge Multidimensional Data Visualization: Back to the Virtue of Principal Coordinates and Dendrograms in the New Computer Age*, volume Handbook of Data Visualization, pages 349–387. Springer.

[Tortora and Marino, 2011] Tortora, C. and Marino, M. (2011). A simulation study on Factor PD-clustering convergence. *submitted*.

[Vichi and Kiers, 2001] Vichi, M. and Kiers, H. (2001). Factorial k-means analysis for two way data. *Computational Statistics and Data Analysis*, 37:29–64.

[Wedel and Kamakura, 1999] Wedel, M. and Kamakura, W. A. (1999). *Market segmentation*. Kluwer Academic Publishers.

[Weiszfeld, 1937] Weiszfeld, E. (1937). Sur le point par lequel la somme des distances de n points donnés est minimum. *Tohoku Mathematics journal*, 43:355–386.