Electronic Journal of Applied Statistical Analysis
EJASA, Electron. J. App. Stat. Anal.
http://siba-ese.unisalento.it/index.php/ejasa/index
e-ISSN: 2070-5948
DOI: DOI–10.1285/i20705948v15n3p606

Multivariate analysis of mixed data. The R Package PCAmixdata

By Chavent, Kuentz, Labenne et al.

Published: November 20, 2022

Electronic Journal of Applied Statistical Analysis
Vol. 15, Issue 03, November 2022, 606-645
DOI: DOI–10.1285/i20705948v15n3p606

Multivariate analysis of mixed data. The R Package PCAmixdata

Marie Chavent*[a], Vanessa Kuentz[b], Amaury Labenne[b], and Jérôme Saracco[a]

[a] *Univ. Bordeaux, CNRS, INRIA, Bordeaux INP, IMB, UMR 5251, F-33400 Talence*
[b] *INRAE, ETTIS, F-33612 Cestas, France*

Published: November 20, 2022

Mixed data arise when observations are described by a mixture of numerical and categorical variables. The `R` package **PCAmixdata** extends to this type of data standard multivariate analysis methods which allow description, exploration and visualization of the data. The key techniques/methods included in the package are principal component analysis for mixed data (`PCAmix`), varimax-like orthogonal rotation for `PCAmix`, and multiple factor analysis for mixed multi-table data. This paper proposes a unified mathematical presentation of the different methods with common notations, as well as providing a summarised presentation of the three algorithms, with details to help the user understand graphical and numerical outputs of the corresponding `R` functions. This then allows the user to easily provide relevant interpretations of the results obtained. The three main methods are illustrated on a real dataset composed of four data tables characterizing living conditions in different municipalities in the Gironde region of southwest France.

keywords: mixture of numerical and categorical data, PCA, multiple correspondence analysis, multiple factor analysis, varimax rotation, `R`.

1. Introduction

Multivariate data analysis refers to descriptive statistical methods used to analyze data arising from more than one variable. The common goal of these methods is to provide

*Corresponding author: marie.chavent@u-bordeaux.fr

ISSN: 2070-5948
http://siba-ese.unisalento.it/index.php/ejasa/index

description, exploration and visualization of the data, via data reduction and graphical display. These variables can be either numerical or categorical. For example, principal component analysis (PCA) and varimax rotation handle numerical variables, whereas multiple correspondence analysis (MCA) handles categorical variables. Multiple factor analysis (MFA) works with multi-table data, where the type of the variables can vary from one data table to the other but the variables should be of the same type within a given data table (Escofier and Pagès, 1994; Bécue-Bertaut and Pagès, 2008; Abdi et al., 2013).

While PCA, varimax and MFA for mixed data have been already described elsewhere, our paper provides a unified mathematical presentation of all the different methods with common notations, which greatly enhances the user experience. A synthetic presentation of the corresponding algorithms is given, with details to help the user understand all the graphical and numerical outputs of the `R` package **PCAmixdata** (Chavent et al., 2017). The way the methods are presented is clearly in the French tradition of data analysis ("analyse des données" in French), following the words of Jean-Paul Benzecri "all in all, doing a data analysis, in good mathematics, is simply searching for eigenvectors; all the science (or the art) of it is in finding the right matrix to diagonalize". More precisely, the underlying theory involves reducing data dimensionality, via generalized singular value decomposition, to provide a subspace that best represents the data in the sense of maximizing the variability of the projected points. Because of this, great importance is attached to relevant graphical representations of both rows and columns of the data matrices.

Several existing `R` (`R` Core Team, 2017) packages use standard multivariate analysis methods. These include **ade4** (Dray and Dufour, 2007; Dray et al., 2017), **FactoMineR** (Lê et al., 2008; Husson et al., 2017), **ExPosition** (Beaton et al., 2014) or **Gifi** (Mair et al., 2019; de Leeuw and Mair, 2009). Most of them propose a function to perform PCA with a mixture of numerical and categorical data. For instance, the function `dudi.mix` in the package **ade4** implements the method developed by Hill and Smith (1976) and the function `FAMD` of the package **FactoMineR** implements that developed by Pagès (2004) in the spirit of the French school. Note that these methods are also equivalent to PCAMIX, a method proposed independently by de Leeuw and van Rijckevorsel (1980) and extended by Kiers (1991), in the spirit of the Dutch school. The `R` package **PCAmixdata** presented in this paper is dedicated to mixed data and provides three main functions:

- `PCAmix` (PCA of a mixture of numerical and categorical variables),
- `PCArot` (rotation after `PCAmix`),
- and `MFAmix` (multiple factor analysis of mixed multi-table data).

The `PCAmix` function gives similar results to `dudi.mix` and `FAMD`. The procedure `PCArot` (Chavent et al., 2012) is not implemented elsewhere and allows in particular to make rotation in MCA for categorical data. The function `MFAmix` allows mixed single data table (groups with both numerical and categorical variables) and differs from the function `MFA` of the package **FactoMineR** where only groups of numerical and groups of categorical variables are allowed.

In addition, the package **PCAmixdata** naturally offers functions to plot graphical outputs. One particularly useful feature of the package is the possibility to predict scores for new observations of the principal components of `PCAmix`, `PCArot` and `MFAmix`, and to project supplementary variables or levels (resp. supplementary groups of variables) on the maps of `PCAmix` (resp. `MFAmix`). These functions are implemented in the `R` package as S3 methods with generic names `plot`, `predict` and `suppvar` associated with the objects of class `PCAmix`, `PCArot` and `MFAmix`.

A real dataset called `gironde` is available in the package to illustrate the functions and the outputs with simple examples. This will allow the user to easily understand the numerical and graphical outputs, and thus draw fine and relevant interpretations from the results obtained. This dataset is made up of four data tables, each characterizing living conditions in 542 municipalities in the Gironde region in southwest France, see Appendix A for details. This dataset was taken from the 2009 census database[1] of the French national institute of statistics and economic studies and from a topographic database[2] of the French national institute of geographic and forestry information. The first data table describes the 542 municipalities with 9 numerical variables relating to employment conditions. The second data table describes those municipalities with 5 variables (2 categorical and 3 numerical) relating to housing conditions, the third one with 9 categorical variables relating to services (restaurants, doctors, post offices,...) and the last one with 4 numerical variables relating to environmental conditions. A complete description of the 27 variables, divided into 4 groups (Employment, Housing, Services, Environment) is given in Appendix A.

The rest of the paper is organized as follows. Section 2 details the link between standard PCA and MCA via Generalized Singular Value Decomposition (GSVD). It demonstrates how MCA can be obtained from a single PCA with metrics, the cornerstone for merging standard PCA and MCA in `PCAmix`. Sections 3, 4 and 5 present respectively the `PCAmix`, `PCArot` and `MFAmix` methods with details for the interpretation of the associated graphical and numerical outputs. Some technical proofs have been provided in Appendices B and D. In each of these sections, the corresponding method is illustrated with the `gironde` dataset and the associated `R` code is given. Finally, concluding remarks are provided in Section 6.

2. PCA with metrics

PCA with metrics is a generalization of the standard PCA method where metrics are used to introduce weights into rows (observations) and columns (variables) within a data matrix. Standard PCA for numerical data and standard MCA for categorical data can be presented within this general framework, so that the unique PCAmix procedure for a mixture of numerical and categorical data can be easily defined.

[1] http://www.insee.fr/fr/bases-de-donnees/

[2] http://professionnels.ign.fr/bdtopo

2.1. The general framework

Let $\mathbf{Z}$ be a real matrix of dimension $n \times p$. Let $\mathbf{N}$ (resp. $\mathbf{M}$) be the diagonal matrix of the weights of the n rows (resp. the weights of the p columns).

Generalized Singular Value Decomposition. The GSVD of $\mathbf{Z}$ with metrics $\mathbf{N}$ on $\mathbb{R}^n$ and $\mathbf{M}$ on $\mathbb{R}^p$ gives the following decomposition:

$$\mathbf{Z} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top, \tag{1}$$

where

- $\mathbf{\Lambda} = \text{diag}(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_r})$ is the $r \times r$ diagonal matrix of the singular values of $\mathbf{ZMZ}^\top\mathbf{N}$ and $\mathbf{Z}^\top\mathbf{NZM}$, and r denotes the rank of $\mathbf{Z}$;
- $\mathbf{U}$ is the $n \times r$ matrix of the first r eigenvectors of $\mathbf{ZMZ}^\top\mathbf{N}$ such that $\mathbf{U}^\top\mathbf{NU} = \mathbb{I}_r$, with $\mathbb{I}_r$ the identity matrix of size $n \times r$;
- $\mathbf{V}$ is the $p \times r$ matrix of the first r eigenvectors of $\mathbf{Z}^\top\mathbf{NZM}$ such that $\mathbf{V}^\top\mathbf{MV} = \mathbb{I}_r$.

Remark 1. *The GSVD of* $\mathbf{Z}$ *can be obtained by performing the standard SVD of the matrix* $\tilde{\mathbf{Z}} = \mathbf{N}^{1/2}\mathbf{Z}\mathbf{M}^{1/2}$, *that is a GSVD with metrics* $\mathbb{I}_n$ *on* $\mathbb{R}^n$ *and* $\mathbb{I}_p$ *on* $\mathbb{R}^p$. *It gives:*

$$\tilde{\mathbf{Z}} = \tilde{\mathbf{U}}\tilde{\mathbf{\Lambda}}\tilde{\mathbf{V}}^\top \tag{2}$$

and transformation back to the original scale gives:

$$\mathbf{\Lambda} = \tilde{\mathbf{\Lambda}}, \quad \mathbf{U} = \mathbf{N}^{-1/2}\tilde{\mathbf{U}}, \quad \mathbf{V} = \mathbf{M}^{-1/2}\tilde{\mathbf{V}}. \tag{3}$$

Principal Components. The n rows of $\mathbf{Z}$ are projected with respect to the inner product matrix $\mathbf{M}$ onto the axes spanned by the vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$ of $\mathbb{R}^p$ (columns of $\mathbf{V}$) found by solving the sequence (indexed by i) of optimization problems:

$$\begin{array}{ll} \text{maximize} & \|\mathbf{ZMv}_i\|^2_{\mathbf{N}} \\ \text{subject to} & \mathbf{v}_i^\top\mathbf{Mv}_j = 0 \quad \forall 1 \le j < i, \\ & \mathbf{v}_i^\top\mathbf{Mv}_i = 1. \end{array} \tag{4}$$

The solutions $\mathbf{v}_1, \ldots, \mathbf{v}_r$ are the eigenvectors of $\mathbf{Z}^\top\mathbf{NZM}$, i.e., the right-singular vectors in (1).

The principal component scores (also called factor coordinates of the rows hereafter) are the coordinates of the projections of the n rows onto these r axes. Let $\mathbf{F}$ denote the $n \times r$ matrix of the factor coordinates of the rows. By definition

$$\mathbf{F} = \mathbf{ZMV}, \tag{5}$$

and we deduce from (1) that:

$$\mathbf{F} = \mathbf{U}\Lambda. \tag{6}$$

Let $\mathbf{f}_i = \mathbf{ZMv}_i$ denote a column of $\mathbf{F}$. The vector $\mathbf{f}_i \in \mathbb{R}^n$ is called the ith principal component (PC) and the solution of (4) gives $\|\mathbf{f}_i\|^2_{\mathbf{N}} = \lambda_i$.

Loadings. The p columns of $\mathbf{Z}$ are projected with respect to the inner product matrix $\mathbf{N}$ onto the axes spanned by the vectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$ of $\mathbb{R}^n$ (columns of $\mathbf{U}$) found by solving the sequence (indexed by i) of optimization problems:

$$\begin{array}{ll} \text{maximize} & \|\mathbf{Z}^\top \mathbf{N}\mathbf{u}_i\|^2_{\mathbf{M}} \\ \text{subject to} & \mathbf{u}_i^\top \mathbf{N}\mathbf{u}_j = 0 \quad \forall 1 \leq j < i, \\ & \mathbf{u}_i^\top \mathbf{N}\mathbf{u}_i = 1. \end{array} \tag{7}$$

The solutions $\mathbf{u}_1, \ldots, \mathbf{u}_r$ are the eigenvectors of $\mathbf{Z}\mathbf{M}\mathbf{Z}^\top\mathbf{N}$, i.e., the left-singular vectors in (1).

The loadings (also called factor coordinates of the columns hereafter) are the coordinates of the projections of the p columns onto these r axes. Let $\mathbf{A}$ denote the $p \times r$ matrix of the factor coordinates of the columns. By definition

$$\mathbf{A} = \mathbf{Z}^\top \mathbf{N}\mathbf{U}, \tag{8}$$

and we deduce from (1) that:

$$\mathbf{A} = \mathbf{V}\Lambda. \tag{9}$$

Let us denote $\mathbf{a}_i = \mathbf{Z}^\top\mathbf{N}\mathbf{u}_i$ a column of $\mathbf{A}$. The vector $\mathbf{a}_i \in \mathbb{R}^p$ is called the ith loadings vector and the solution of (7) gives $\|\mathbf{a}_i\|^2_{\mathbf{M}} = \lambda_i$.

Remark 2. *The previous definitions give the following entries of* $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^\top$*:*

- *The first writing is* $\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{A}^\top$ *where*
  - $\boldsymbol{U}$ *is the matrix of the standardized principal components* ($\boldsymbol{U} = \boldsymbol{F}\boldsymbol{\Lambda}^{-1}$),
  - $\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}$ *is then the matrix of the loadings of the standardized principal components.*
- *The second writing is* $\boldsymbol{Z} = \boldsymbol{F}\boldsymbol{V}^\top$ *where*
  - $\boldsymbol{F}$ *is the matrix of the principal components,*
  - $\boldsymbol{V}$ *is then the matrix of the loadings of the principal components.*

Reduced rank matrix approximation. PCA is often seen as a method of least square approximation of the matrix $\mathbf{Z}$ by a matrix $\hat{\mathbf{Z}}$ of rank $q < r$. This optimal low-rank approximation is defined by:

$$\hat{\mathbf{Z}} = \mathbf{U}_q\boldsymbol{\Lambda}_q\mathbf{V}_q^\top \tag{10}$$

where $\mathbf{U}_q$ (resp. $\mathbf{V}_q$) denote the matrix of the first q columns of $\mathbf{U}$ (resp. $\mathbf{V}$) and $\boldsymbol{\Lambda}$ is the diagonal matrix of the first q singular values.

The reconstruction matrix $\hat{\mathbf{Z}}$ is said to be optimal for matrices of rank q because it satisfies the following condition :

$$\|\mathbf{Z} - \hat{\mathbf{Z}}\|^2_{\mathbf{N},\mathbf{M}} = \min_{\mathbf{X}} \|\mathbf{Z} - \mathbf{X}\|^2_{\mathbf{N},\mathbf{M}} \tag{11}$$

where $\|\mathbf{X}\|^2_{\mathbf{N},\mathbf{M}} = \mathrm{tr}(\mathbf{NXMX}^\top)$ is a generalization of the Froebonius norm to the context of generalized SVD with metrics $\mathbf{N}$ and $\mathbf{M}$ (Abdi, 2007). Moreover it can be shown that:

$$\|\mathbf{Z} - \hat{\mathbf{Z}}\|^2_{\mathbf{N},\mathbf{M}} = \sum_{i=q+1}^{r} \lambda_i \tag{12}$$

and accordingly the quality of the reconstruction defined by

$$\frac{\lambda_1 + \ldots + \lambda_q}{\sum_{i=1}^{r} \lambda_i} \tag{13}$$

is maximized. The quantity (13) is interpreted as the proportion of the variance of the data explained by the principal components or as the reconstructed proportion.

2.2. Standard PCA and standard MCA

This section presents how standard PCA (for numerical data) and standard MCA (for categorical data) can be obtained from the GSVD of specific matrices $\mathbf{Z}$, $\mathbf{N}$, $\mathbf{M}$. In both cases, the numerical matrix $\mathbf{Z}$ is obtained by pre-processing the original data matrix $\mathbf{X}$ and the matrix $\mathbf{N}$ (resp. $\mathbf{M}$) is the diagonal matrix of the weights of the rows (resp. the columns) of $\mathbf{Z}$.

Standard PCA. The data table to be analyzed by PCA comprises n observations described by p numerical variables, and is represented by the $n \times p$ quantitative matrix $\mathbf{X}$. In the pre-processing step, the columns of $\mathbf{X}$ are centered and normalized to construct the standardized matrix $\mathbf{Z}$ (defined such that $\frac{1}{n}\mathbf{Z}^\top\mathbf{Z}$ is the linear correlation matrix). The n rows (observations) are usually weighted by $\frac{1}{n}$ and the p columns (variables) are weighted by 1. It gives $\mathbf{N} = \frac{1}{n}\mathbb{I}_n$ and $\mathbf{M} = \mathbb{I}_p$. The metric $\mathbf{M}$ indicates that the distance between two observations is the standard euclidean distance between two rows of $\mathbf{Z}$. The total inertia of $\mathbf{Z}$ is then equal to p. The matrix $\mathbf{F}$ of the factor coordinates of the observations (principal components) and the matrix $\mathbf{A}$ of the factor coordinates of the variables (loadings) are calculated directly from (6) and (1). The well-known properties of PCA are the following:

- Each loading a_{ji} (element of $\mathbf{A}$) is the linear correlation between the numerical variable $\mathbf{x}_j$ (the jth column of $\mathbf{X}$) and the ith principal component $\mathbf{f}_i$ (the ith column of $\mathbf{F}$):

$$a_{ji} = \mathbf{z}_j^\top \mathbf{N}\mathbf{u}_i = r(\mathbf{x}_j, \mathbf{f}_i), \tag{14}$$

  where $\mathbf{u}_i = \frac{\mathbf{f}_i}{\sqrt{\lambda_i}}$ is the ith standardized principal component and $\mathbf{z}_j$ (resp. $\mathbf{x}_j$) is the jth column of $\mathbf{Z}$ (resp. $\mathbf{X}$).

- Each eigenvalue λ_i is the variance of the ith principal component:

$$\lambda_i = \|\mathbf{f}_i\|^2_{\mathbf{N}} = \mathrm{Var}(\mathbf{f}_i). \tag{15}$$

- Each eigenvalue λ_i is also the sum of the squared correlations between the p numerical variables and the ith principal component:

$$\lambda_i = \|\mathbf{a}_i\|^2_{\mathbf{M}} = \sum_{j=1}^{p} r^2(\mathbf{x}_j, \mathbf{f}_i). \tag{16}$$

- The contribution of the variable $\mathbf{x}_j$ to the variance of the ith principal component interprets as a squared loading i.e. squared correlation here:

$$c_{ji} = a^2_{ji} = r^2(\mathbf{x}_j, \mathbf{f}_i). \tag{17}$$

- The total variance of the data matrix $\mathbf{Z}$ is equal to p. The proportion of variance explained by the ith principal component is then:

$$\frac{\lambda_i}{p}.$$

Standard MCA. The data table to be analyzed by MCA comprises n observations described by p categorical variables and it is represented by the $n \times p$ matrix $\mathbf{X}$. Each categorical variable has m_j levels and the sum of the m_j's is equal to m. In the preprocessing step, each level is coded as a binary variable and the $n \times m$ indicator matrix $\mathbf{G}$ is constructed. Usually MCA is performed by applying standard Correspondence Analysis (CA) to this indicator matrix. Here, we provide different ways to calculate the factor coordinates of MCA by applying a single PCA with metrics to the indicator matrix $\mathbf{G}$.

Let $\mathbf{Z}$ now denote the centered indicator matrix $\mathbf{G}$. The n rows (observations) are usually weighted by $\frac{1}{n}$ and the m columns (levels) are weighted by $\frac{n}{n_s}$, the inverse of the frequency of the level s, where n_s denotes the number of observations that belong to the sth level. It gives $\mathbf{N} = \frac{1}{n}\mathbb{I}_n$ and $\mathbf{M} = \text{diag}(\frac{n}{n_s}, s = 1\ldots, m)$. This metric $\mathbf{M}$ indicates that the distance between two observations is a weighted euclidean distance similar to the χ^2 distance in CA. This distance gives more importance to rare levels. The total inertia of $\mathbf{Z}$ with this distance and the weights $\frac{1}{n}$ is equal to $m-p$. The GSVD of $\mathbf{Z}$ with these metrics allow a direct calculation using (6) the matrix $\mathbf{F}$ of the factor coordinates of the observations (the principal components).The factor coordinates of the levels however are not obtained directly from the loading matrix $\mathbf{A}$ defined in (1) but from:

$$\mathbf{A}^* = \mathbf{MA}. \tag{18}$$

to get back the barycentric property recalled in (19) which is central to the interpretation of the results in MCA. The usual properties in MCA are:

- Each coordinate a^*_{si} (element of $\mathbf{A}^*$) is the mean value of the (standardized) factor coordinates of the observations that belong to level s:

$$a^*_{si} = \frac{n}{n_s} a_{si} = \frac{n}{n_s} \mathbf{z}_s^\top \mathbf{N} \mathbf{u}_i = \bar{u}^s_i, \tag{19}$$

where $\mathbf{z}_s$ is the sth column of $\mathbf{Z}$, $\mathbf{u}_i = \frac{\mathbf{f}_i}{\sqrt{\lambda_i}}$ is the ith standardized principal component and $\bar{u}_i^s$ is the mean value of the coordinates of $\mathbf{u}_i$ associated with the observations that belong to level s.

- Each eigenvalue λ_i is the variance of the ith principal component:

$$\lambda_i = \|\mathbf{f}_i\|^2_{\mathbf{N}} = \mathrm{Var}(\mathbf{f}_i). \tag{20}$$

- Each eigenvalue λ_i is the sum of the correlation ratios between the p categorical variables and the ith principal component (which is numerical):

$$\lambda_i = \|\mathbf{a}_i\|^2_{\mathbf{M}} = \|\mathbf{a}_i^*\|^2_{\mathbf{M}^{-1}} = \sum_{j=1}^{p} \eta^2(\mathbf{f}_i|\mathbf{x}_j). \tag{21}$$

where:

$$\eta^2(\mathbf{f}_i|\mathbf{x}_j) = \frac{\sum_{s\in I_j} \frac{n_s}{n}(\bar{\mathbf{f}}_i^s - \bar{\mathbf{f}}_i)^2}{\mathrm{Var}(\mathbf{f}_i)} \tag{22}$$

where I_j is the set of indices of the levels of the categorical variable j and $\bar{\mathbf{f}}_i^s$ is the mean value of the coordinates of $\mathbf{f}_i$ associated with the observations that belong to level s. Here $\bar{\mathbf{f}}_i = 0$ because the principal components $\mathbf{f}_i$ are all centered as linear combinaisons of the centered columms of $\mathbf{Z}$.

The correlation ratio $\eta^2(\mathbf{f}_i|\mathbf{x}_j)$ measures the link between the categorical variable $\mathbf{x}_j$ and the numerical principal component $\mathbf{f}_i$ and interprets as the part of the variance of $\mathbf{f}_i$ explained by $\mathbf{x}_j$.

- The contribution of the variable $\mathbf{x}_j$ to the variance of the ith principal component is:

$$c_{ji} = \sum_{s\in I_j} \frac{n}{n_s} a_{si}^2 = \sum_{s\in I_j} \frac{n_s}{n} a_{si}^{*2} = \eta^2(\mathbf{f}_i|\mathbf{x}_j) \tag{23}$$

The contribution c_{ij} in (23) is also called a squared loading to mimic PCA where squared loadings are squared correlations (see equation (17)).

- The total variance of $\mathbf{Z}$ is equal to $m - p$. The proportion of variance explained by the ith principal component is then:

$$\frac{\lambda_i}{m-p}.$$

Remark 3. *Compared to standard MCA method where correspondence analysis (CA) is applied to the indicator matrix, we can note that:*

- *the total inertia of* $\boldsymbol{Z}$ *(based on the metrics* $\boldsymbol{M}$ *and* $\boldsymbol{N}$*) is equal to* $m - p$*, whereas the total inertia in standard MCA is multiplied by* p *and is equal to* $p(m - p)$*. This property will be useful to define PCA for mixed data right after. It will allow to balance the inertia of the numerical data (equal to the number of numerical variables) and the inertia of the categorical data (equal now to the number of levels minus the number of categorical variables),*

- *the factor coordinates of the levels are the same. However, the eigenvalues are multiplied by p and factor coordinates of the observations are then multiplied by $\sqrt{p}$. This property has no impact since results are identical to within one multiplier coefficient.*

3. PCA of a mixture of numerical and categorical data

Principal Component Analysis (PCA) methods dealing with a mixture of numerical and categorical variables already exist and have been implemented in functions like `FAMD` of the package **FactoMineR** or `dudi.mix` of the package **ade4**. In the `R` package **PCAmixdata**, the function `PCAmix` implements an algorithm presented hereafter as a single PCA with metrics, i.e., based on a Generalized Singular Value Decomposition (GSVD) of pre-processed data. This algorithm includes naturally standard PCA and standard MCA as special cases. Note that `FAMD`, `dudi.mix` and `PCAmix` are three different implementations that give identical results (sometimes up to a constant factor) [3].

3.1. The `PCAmix` algorithm

The data table to be analyzed by `PCAmix` comprises n observations described by p_1 numerical variables and p_2 categorical variables. It is represented by the $n \times p_1$ numerical data matrix $\mathbf{X}_1$ and the $n \times p_2$ categorical data matrix $\mathbf{X}_2$. Let m denote the total number of levels of the p_2 categorical variables. The `PCAmix` algorithm merges PCA and MCA thanks to the general framework given in Section 2 . The two steps of `PCAmix` (pre-processing and factor coordinates processing) mimic this general framework with the numerical data matrix $\mathbf{X}_1$ and the categorical data matrix $\mathbf{X}_2$ as inputs.

Step 1: pre-processing.

1. Build the real matrix $\mathbf{Z} = [\mathbf{Z}_1, \mathbf{Z}_2]$ of dimension $n \times (p_1 + m)$ where:
   - ↪ $\mathbf{Z}_1$ is the standardized version of $\mathbf{X}_1$ (as in standard PCA),
   - ↪ $\mathbf{Z}_2$ is the centered version of the indicator matrix $\mathbf{G}$ of $\mathbf{X}_2$ (as in standard MCA).
2. Build the diagonal matrix $\mathbf{N}$ of the weights of the rows of $\mathbf{Z}$. The n rows are often weighted by $\frac{1}{n}$, such that $\mathbf{N} = \frac{1}{n}\mathbb{I}_n$.
3. Build the diagonal matrix $\mathbf{M}$ of the weights of the columns of $\mathbf{Z}$:
   - ↪ The first p_1 columns (corresponding to the numerical variables) are weighted by 1 (as in standard PCA).
   - ↪ The last m columns (corresponding to the levels of the categorical variables) are weighted by $\frac{n}{n_s}$ (as in standard MCA), where $n_s, s = 1, \ldots, m$ denotes the number of observations that belong to the sth level.

[3] https://chavent.github.io/PCAmixdata/PCAmixcompare.html

The metric

$$\mathbf{M} = \text{diag}(1, \ldots, 1, \frac{n}{n_1}, \ldots, \frac{n}{n_m}) \tag{24}$$

indicates that the distance between two rows of $\mathbf{Z}$ is a mixture of the simple euclidean distance used in PCA (for the first p_1 columns) and the weighted distance in the spirit of the χ^2 distance used in MCA (for the last m columns). The total inertia of $\mathbf{Z}$ with this distance and the weights $\frac{1}{n}$ is equal to $p_1 + m - p_2$.

Step 2: factor coordinates processing.

1. The GSVD of $\mathbf{Z}$ with metrics $\mathbf{N}$ and $\mathbf{M}$ gives the decomposition:

   $$\mathbf{Z} = \mathbf{U\Lambda V}^\top$$

   as defined in (1). Let r denote the rank of $\mathbf{Z}$.

2. The matrix of dimension $n \times r$ of the factor coordinates of the n observations is:

   $$\mathbf{F} = \mathbf{ZMV}, \tag{25}$$

   or directly computed from the GSVD decomposition as:

   $$\mathbf{F} = \mathbf{U\Lambda}. \tag{26}$$

   The columns $\mathbf{f}_i$ of the matrix $\mathbf{F}$ are the principal components and the columns $\mathbf{u}_i = \frac{\mathbf{f}_i}{\sqrt{\lambda_i}}$ of the matrix $\mathbf{U}$ are the standardized principal components.

3. The matrix of dimension $(p_1+m) \times r$ of the factor coordinates of the p_1 quantitative variables and the m levels of the p_2 categorical variables is:

   $$\mathbf{A}^* = \mathbf{MV\Lambda} = \mathbf{MA}, \tag{27}$$

   where $\mathbf{A} = \mathbf{V\Lambda}$ is the matrix of the loadings of the standardized principal components (see Remark 2).

   The matrix $\mathbf{A}^*$ of factor coordinates splits as follows: $\mathbf{A}^* = \begin{bmatrix} \mathbf{A}_1^* \\ \mathbf{A}_2^* \end{bmatrix} \begin{matrix} \}\, p_1 \\ \}\, m \end{matrix}$ where

   $\hookrightarrow$ $\mathbf{A}_1^*$ contains the factor coordinates of the p_1 numerical variables,

   $\hookrightarrow$ $\mathbf{A}_2^*$ contains the factor coordinates of the m levels.

   The matrix $\mathbf{A}^*$ differs from the matrix $\mathbf{A}$ of the loadings so that $\mathbf{A}_2^*$ verifies the MCA's barycentic property (29).

3.2. Properties of PCAmix

The PCAmix procedure shares and generalizes the properties of PCA and MCA:

- For $j = 1, \dots, p_1$:
$$a^*_{ji} = a_{ji} = r(\mathbf{x}_j, \mathbf{f}_i), \tag{28}$$
The factor coordinates in $\mathbf{A}^*_1$ give the correlations between the p_1 quantitative variables and the principal components.

- For $s = p_1 + 1, \dots, p_1 + m$:
$$a^*_{si} = \bar{u}^s_i, \tag{29}$$
The factor coordinates of the m levels in $\mathbf{A}^*_2$ give the mean values of the standardized principal components $\mathbf{u}_i$ for the observations that belong to level s.

- Each eigenvalue λ_i is the variance of the ith principal component:
$$\lambda_i = \|\mathbf{f}_i\|^2_{\mathbf{N}} = \mathrm{Var}(\mathbf{f}_i). \tag{30}$$

- Each eigenvalue is the sum of squared correlations (resp. the correlation ratios) between the p_1 numerical (resp. p_2 categorical) variables and the ith principal component (which is numerical):
$$\begin{aligned}\lambda_i &= \|\mathbf{a}_i\|^2_{\mathbf{M}} = \|\mathbf{a}^*_i\|^2_{\mathbf{M}^{-1}},\\ &= \sum_{j=1}^{p_1} a^2_{ji} + \sum_{j=p_1+1}^{p_2} \sum_{s \in I_j} \frac{n}{n_s} a^2_{si},\\ &= \sum_{j=1}^{p_1} r^2(\mathbf{x}_j, \mathbf{f}_i) + \sum_{j=p_1+1}^{p_2} \eta^2(\mathbf{f}_i|\mathbf{x}_j). \end{aligned} \tag{31}$$
where $\eta^2(\mathbf{f}_i|\mathbf{x}_j)$ is the correlation ratio defined in (22).

- The contribution of a variable $\mathbf{x}_j$ to the variance of the ith principal component is:
$$\begin{cases} c_{ji} = a^2_{ji} = r^2(\mathbf{x}_j, \mathbf{f}_i) & \text{if the variable } \mathbf{x}_j \text{ is numerical,}\\ c_{ji} = \sum_{s \in I_j} \frac{n}{n_s} a^2_{si} = \eta^2(\mathbf{f}_i|\mathbf{x}_j) & \text{if the variable } \mathbf{x}_j \text{ is categorical,} \end{cases} \tag{32}$$
The contributions are called **squared loadings**. Note that the term squared loadings for categorical variables draws an analogy with squared loadings in PCA. Squared loadings are defined as squared correlations for numerical variables and correlation ratios for categorical variables.

- The total variance of $\mathbf{Z}$ is equal to p_1+m-p_2. The proportion of variance explained by the ith principal component is then:

$$\frac{\lambda_i}{p_1+m-p_2}.$$

Remark 4. *PCAmix computes $q \leq r$ new numerical variables (the principal components) that will "explain" or "extract" the largest part of the inertia of the matrix $\boldsymbol{Z}$ built from the original data tables $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$. The principal components (columns of $\boldsymbol{F}$) are by construction non correlated linear combinations of the columns of $\boldsymbol{Z}$ and can be viewed as new synthetic numerical variables with maximum dispersion (30) and maximum link with the original variables (31).*

3.3. Graphical outputs of `PCAmix`

The function `plot.PCAmix` plots the observations, the numerical variables and the levels of the categorical variables according to their factor coordinates.

Correlation circle. The map of the quantitative variables, called the correlation circle, gives an idea of the pattern of linear links between the quantitative variables. If two columns $\mathbf{z}_j$ and $\mathbf{z}_{j'}$ of $\mathbf{Z}_1$ corresponding to two quantitative variables $\mathbf{x}_j$ and $\mathbf{x}_{j'}$ (two columns of $\mathbf{X}_1$) are well projected on the map i.e. with squared cosine close to 1), the cosine of their angle in projection gives an idea of their correlation in $\mathbb{R}^n$ defined by

$$r(\mathbf{x}_j, \mathbf{x}_{j'}) = \mathbf{z}_j^\top \mathbf{N} \mathbf{z}_{j'}$$

with $\mathbf{N} = \frac{1}{n}\mathbb{I}_n$ in the usual case of observations weighted by $\frac{1}{n}$.

Levels map. The levels map gives an idea of the pattern of proximities between the levels of (different) categorical variables. If two levels $\mathbf{z}_s$ and $\mathbf{z}_{s'}$ (two columns of the centered indicator matrix $\mathbf{Z}_2$) are well projected on the map, the distance when projected gives an idea of their distance in $\mathbb{R}^n$ given by

$$d^2_{\mathbf{N}}(\mathbf{z}_s, \mathbf{z}_{s'}) = (\mathbf{z}_s - \mathbf{z}_{s'})^\top \mathbf{N} (\mathbf{z}_s - \mathbf{z}_{s'})$$

which can be interpreted as 1 minus the proportion of observations having both levels s and s'. With this distance two levels are similar if they are owned by the same observations.

Squared loadings plot. Another graphical output available in `plot.PCAmix` is the plot of the variables (numerical and categorical) according to their squared loadings. The map of all the variables gives an idea of the pattern of links between the variables regardless of their type (quantitative or categorical). More precisely, it is easy to verify that the squared loading c_{ji} defined in (32) is equal to:

- the squared correlation $r^2(\mathbf{f}_i, \mathbf{x}_j)$ if the variable $\mathbf{x}_j$ is numerical,

- the correlation ratio $\eta^2(\mathbf{f}_i|\mathbf{x}_j)$ if the variable $\mathbf{x}_j$ is categorical.

Coordinates (between 0 and 1) of the variables on this plot measure the intensity of the links between variables and principal components and can be used to interpret principal component maps.

Observations map. The map of the observations (also called principal component map) gives an idea of the pattern of similarities between the observations. If two observations $\mathbf{z}_k$ and $\mathbf{z}_{k'}$ (two rows of $\mathbf{Z}$) are well projected on the map, their distance in projection gives an idea of their distance in $\mathbb{R}^{p_1+m}$ defined by

$$d^2_{\mathbf{M}}(\mathbf{z}_k, \mathbf{z}_{k'}) = (\mathbf{z}_k - \mathbf{z}_{k'})^\top \mathbf{M}(\mathbf{z}_k - \mathbf{z}_{k'})$$

where $\mathbf{M}$ is defined in (24). This squared distance can be interpreted as the squared euclidean distance calculated on the standardized numerical variables plus the squared χ^2 distance calculated on the levels of the categorical variables. Moreover the position (left, right, up, bottom) of the observations on the PC's map can be interpreted in terms of:

- numerical variables using the property indicating that coordinates on the correlation circle give correlations with PCs,
- levels of categorical variables using the property indicating that coordinates on the level map are means of PC scores.

3.4. Prediction of PC scores with `predict.PCAmix`

A function to predict scores for new observations on the principal components can be helpful. For example:

- projecting new observations onto the principal component map of `PCAmix`,
- when the PCs are used as synthetic numerical variables replacing the original variables (quantitative or categorical) in a predictive model (regression or classification for instance).

The ith principal component of `PCAmix` can be written as a linear combination of the vectors $\mathbf{z}_1, \ldots, \mathbf{z}_{p_1+m}$ (columns of $\mathbf{Z}$):

$$\mathbf{f}_i = \mathbf{Z}\mathbf{M}\mathbf{v}_i = \sum_{\ell=1}^{p_1} v_{\ell i}\mathbf{z}_\ell + \sum_{\ell=p_1+1}^{p_1+m} \frac{n}{n_\ell} v_{\ell i}\mathbf{z}_\ell.$$

It is then easy to write $\mathbf{f}_i$ as a linear combination of the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{p_1+m}$ (columns of $\mathbf{X} = (\mathbf{X}_1|\mathbf{G})$):

$$\mathbf{f}_i = \beta_{0i} + \sum_{\ell=1}^{p_1+m} \beta_{\ell i}\mathbf{x}_\ell, \tag{33}$$

with the coefficients defined as follows:

$$
\begin{aligned}
\beta_{0i} &= -\sum_{\ell=1}^{p_1} v_{\ell i}\frac{\bar{\mathbf{x}}_\ell}{\hat{\sigma}_\ell} - \sum_{\ell=p_1+1}^{p_1+m} v_{\ell i},\\
\beta_{\ell i} &= v_{\ell i}\frac{1}{\hat{\sigma}_\ell},\ \text{for } \ell = 1,\dots,p_1,\\
\beta_{\ell i} &= v_{\ell i}\frac{n}{n_\ell},\ \text{for } \ell = p_1+1,\dots,p_1+m,
\end{aligned}
$$

where $\bar{\mathbf{x}}_\ell$ and $\hat{\sigma}_\ell$ are respectively the empirical mean and the standard deviation of the column $\mathbf{x}_\ell$.

The principal components are thereby written in (33) as a linear combination of the original numerical variables and of the original indicator vectors of the levels of the categorical variables. The function `predict.PCAmix` uses these coefficients to predict the scores (coordinates) of new observations on the $q \leq r$ first principal components (q is chosen by the user) of `PCAmix`.

3.5. Illustration of `PCAmix`

Let us now illustrate the procedure `PCAmix` with the data table `housing` of the dataset `gironde`. This data table contains $n = 542$ municipalities described on $p_1 = 3$ numerical variables and $p_2 = 2$ categorical with a total of $m = 4$ levels (see Appendix A for the description of the variables).

```
R> library("PCAmixdata")
R> data("gironde")
R> head(gironde$housing)
                   density primaryres   houses owners council
ABZAC               131.70      88.77  inf 90%  64.23  sup 5%
AILLAS               21.21      87.52  sup 90%  77.12  inf 5%
AMBARES-ET-LAGRAVE  531.99      94.90  inf 90%  65.74  sup 5%
AMBES               101.21      93.79  sup 90%  66.54  sup 5%
ANDERNOS-LES-BAINS  551.87      62.14  inf 90%  71.54  inf 5%
ANGLADE              63.82      81.02  sup 90%  80.54  inf 5%
```

In order to explore the mixed data table `housing`, a principal component analysis is performed using the function `PCAmix`.

```
R> split <- splitmix(gironde$housing)
R> X1 <- split$X.quanti
R> X2 <- split$X.quali
R> res.pcamix <- PCAmix(X.quanti = X1, X.quali = X2, rename.level = TRUE, graph = FALSE)
R> res.pcamix$eig
      Eigenvalue Proportion Cumulative
dim 1  2.5268771  50.537541   50.53754
dim 2  1.0692777  21.385553   71.92309
dim 3  0.6303253  12.606505   84.52960
dim 4  0.4230216   8.460432   92.99003
dim 5  0.3504984   7.009968  100.00000
```

Note that the function splitmix splits a mixed data matrix into two datasets: one with the numerical variables and one with the categorical variables.

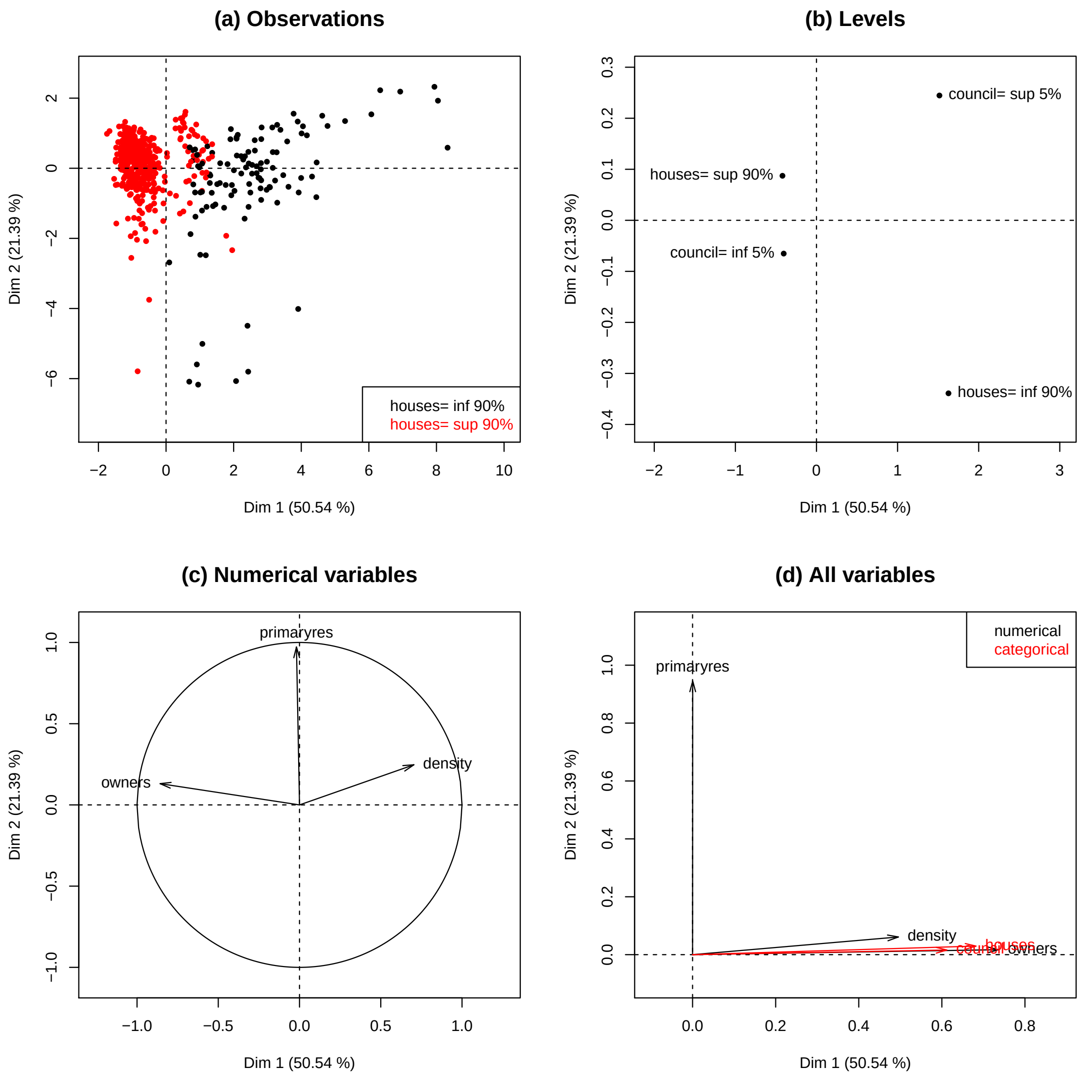
The sum of the eigenvalues is equal to the total inertia $p_1+m-p_2 = 5$ and the first two dimensions retrieve 71% of the total inertia. Let us visualize on these two dimensions the 4 different plots presented in Section 3.3.

```
R> plot(res.pcamix, choice = "ind", coloring.ind = X2$houses, label = FALSE,
       posleg = "bottomright", main = "(a) Observations")
R> plot(res.pcamix, choice = "levels", xlim = c(-1.5,2.5), main = "(b) Levels")
R> plot(res.pcamix,choice = "cor", main = "(c) Numerical variables")
R> plot(res.pcamix, choice = "sqload", coloring.var = T, leg = TRUE,
      posleg = "topright", main = "(d) All variables")
```

Figure 1(a) shows the principal component map where the municipalities (the observations) are colored by their percentage of houses (less than 90%, more than 90%). The first dimension (left hand side) highlights municipalities with large proportions of privately-owned properties. The level map in Figure 1(b) confirms this interpretation and suggests that municipalities with a high proportion of houses (on the left) have a low percentage of council housing. The correlation circle in Figure 1(c) indicates that population density is negatively correlated with the percentage of home owners and that these two variables discriminate the municipalities on the first dimension.

Figure 1(d) plots the variables (categorical or numerical) using squared loadings as coordinates. For numerical variables, squared loadings are squared correlations and for categorical variables squared loadings are correlation ratios. In both cases, they measure the link between the variables and the principal components. It can be observed that the two numerical variables density and owners and the two categorical variables houses and council are linked to the first component. On the contrary, the variable primaryres is clearly orthogonal to these variables and associated with the second component. Note that these links show neither a positive nor a negative association, and the maps Figure 1(b) and Figure 1(c) are necessary for a more precise interpretation.

In summary, municipalities on the right of the principal component map have a relatively high proportion of council housing and a small percentage of privately-owned houses, with most accommodation being rented. On the other hand, municipalities on the left hand side are mostly composed of home owners living in their primary residence. The percentage of primary residences also has a structuring role in the characterization of municipalities in this region of France by defining clearly the second dimension. Indeed the municipalities at the bottom of the map (those with small values on the second dimension) are sea resorts with many secondary residences. For instance the 10 municipalities with the smallest coordinates in the second dimension are well-known resorts on France's Atlantic coast:



(a) Observations

(b) Levels

(c) Numerical variables

(d) All variables

Figure 1: Graphical outputs of PCAmix applied to the data table housing

```
R> sort(res.pcamix$ind$coord[,2])[1:10]
 VENDAYS-MONTALIVET             CARCANS             LACANAU
          -6.171971           -6.087304           -6.070451
     SOULAC-SUR-MER GRAYAN-ET-L’HOPITAL     LEGE-CAP-FERRET
          -5.802359           -5.791642           -5.596315
     VERDON-SUR-MER             HOURTIN            ARCACHON
          -5.008545           -4.493259           -4.013374
              PORGE
          -3.751233
```

Prediction and plot of scores for new observations. We will now illustrate how the function `predict.PCAmix` can be helpful in predicting the coordinates (scores) of observations not used in `PCAmix`. Here, 100 municipalities are sampled at random (test set) and the 442 remaining municipalities (training set) are used to perform `PCAmix`. The following `R` code shows how to predict the scores of the municipalities of the test set on the two first PCs obtained with the training set.

```
R> set.seed(10)
R> test <- sample(1:nrow(gironde$housing), 100)
R> train.pcamix <- PCAmix(X1[-test,], X2[-test,], ndim = 2, graph = FALSE)
R> pred <- predict(train.pcamix, X1[test,], X2[test,])
R> head(pred)
                                dim1        dim2
MAZION                    -0.4120140  0.03905247
FLAUJAGUES                -0.6881160 -0.33163728
LATRESNE                   0.7447583  0.65305517
SAINT-CHRISTOLY-DE-BLAYE -0.7006372 -0.33216807
BERSON                    -1.1426625  0.33607088
CHAMADELLE                -1.3781919  0.24609791
```

These predicted coordinates can be used to plot the 100 supplementary municipalities on the principal component map of the other 442 municipalities (see Figure 2).

```
R> plot(train.pcamix, axes = c(1,2), label = FALSE, main = "Observations map")
R> points(pred, col = 2, pch = 16)
R> legend("bottomright", legend = c("train","test"), fill = 1:2, col = 1:2)
```

Supplementary variables. The function `supvar.PCAmix` calculates the coordinates of supplementary variables (numerical or categorical) on the maps of `PCAmix`. More precisely this function builds an `R` object of class `PCAmix` including the supplementary coordinates. For instance let us consider the numerical variable `building` of the dataset `environment` and the categorical variable `doctor` of the dataset `services` as supplementary variables (see Appendix A for description of these two variables).

```
R>  X1sup <- gironde$environment[ , 1, drop = FALSE]
R>  X2sup <- gironde$services[ , 7, drop = FALSE]
R>  res.sup <- supvar(res.pcamix, X1sup, X2sup, rename.level = TRUE)
R>  res.sup$quanti.sup$coord[ , 1:2, drop = FALSE]
              dim1      dim2
building 0.6945295 0.1884711
R>  res.sup$levels.sup$coord[ ,1:2]
                     dim1         dim2
doctor=0      -0.44403187 -0.006224754
doctor=1 to 2  0.07592759 -0.112352412
doctor=3 or +  1.11104073  0.099723319
```

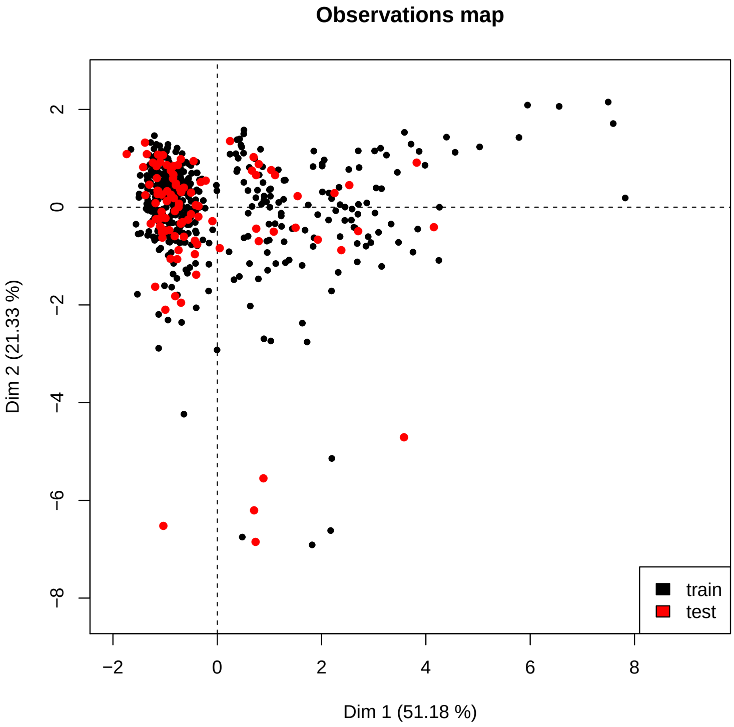


Figure 2: Projection of 100 supplementary municipalities (in red) on the PC map of the other 442 municipalities (in black)

The coordinates of the supplementary numerical variables `building` are still correlations. For instance, the correlation between `building` and the first PC is equal to 0.69. The coordinates of the levels of the supplementary categorical variables are still mean values. For instance the coordinate -0.44 of the level `doctor=0` is the mean value of the municipalities with 0 doctors on the first standardized PC. They are probably mostly left of the PC map. Graphical outputs including these supplementary variables and the original ones can be obtained as previously with the function `plot.PCAmix`, see Figure 3.

```
R> plot(res.sup, choice = "cor", main = "Numerical variables")
R> plot(res.sup, choice = "levels", main = "Levels", xlim = c(-2,2.5))
```

4. Orthogonal rotation in PCA of mixed data

It is common practice in PCA to apply a rotation procedure to loadings to simplify interpretation of the principal components. The well known varimax rotation procedure (Kaiser, 1958) is implemented in the `R` function `varimax` of the **stats** package but this

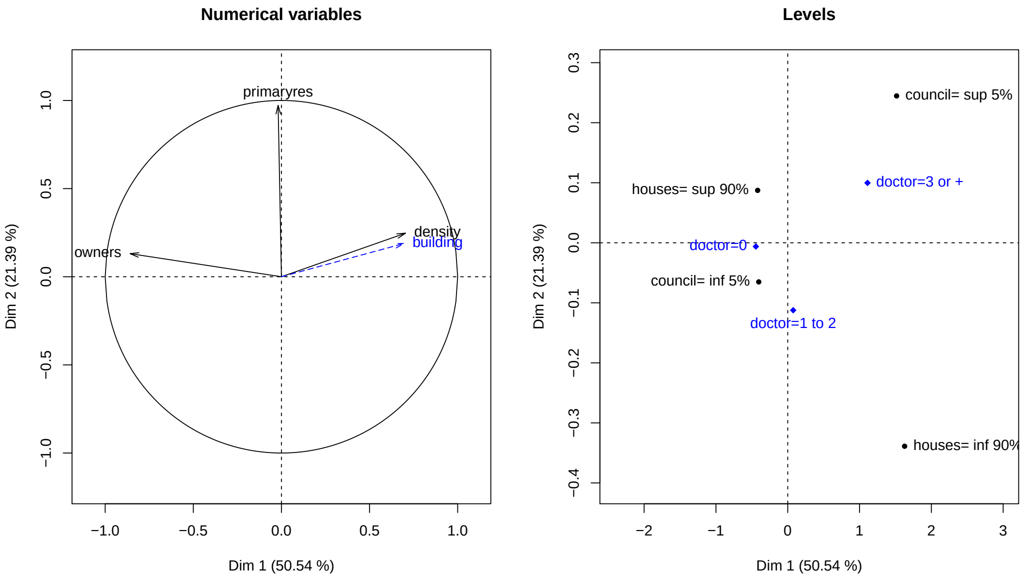


Figure 3: In blue, projection of the supplementary numerical variable building (left) and projection of the levels of the supplementary categorical variable doctor (right)

procedure fits only for numerical data. The function `PCArot` of the package **PCAmixdata** implements a generalization of the varimax procedure to the case of mixed data (Chavent et al., 2012). The rotation procedure `PCArot` applies to the (standardized) principal components of `PCAmix` to get either large (close to 1) or small (close to 0) squared loadings. Indeed in `PCAmix` the squared loadings are squared correlations for numerical variables and correlation ratios for categorical variables measuring then the link between the variables (numerical or categorical) and the principal components. The rotation procedure `PCArot` is therefore applied to the first q principal components of the procedure `PCAmix` where q is chosen by the user.

4.1. The `PCArot` algorithm

We have seen that `PCAmix` is essentially a GSVD that gives the decomposition:

$$\mathbf{Z} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$$

defined in (1). The columns of $\mathbf{U}$ are the standardized principal components (PCs) and the columns of $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}$ are the loading vectors of the standardized principal components. The `PCArot` procedure rotates the matrix $\mathbf{U}_q$ of the first q standardized PCs and the matrix $\mathbf{A}_q$ of the first q loading vectors. Rotating the loadings $\mathbf{A}_q$ of the standardized PCs (rather than loadings $\mathbf{V}_q$ of the PCs) provides rotated scores that continue to be uncorrelated. The `PCArot` procedure does so by following Kiers (1991) proposal to maximize Kaiser's varimax function applied to squared loadings for quantitative variables and 'pseudo' squared loadings for categorical variables. Here the `PCArot` procedure uses the alternative algorithm proposed by Chavent et al. (2012) which expressed the squared loadings for categorical variables somewhat differently, using all elements of $\mathbf{A}_q$, as follows.

Let $\mathbf{T}$ be a $q \times q$ orthonormal rotation matrix. Let $\mathbf{U}_{\text{rot}} = \mathbf{U}_q\mathbf{T}$ denote the matrix of the rotated standardized PCs and $\mathbf{A}_{\text{rot}} = \mathbf{A}_q\mathbf{T}$ denote the matrix of the rotated loading vectors. In varimax rotation, the matrix $\mathbf{T}$ is computed by maximizing the variance of the contributions of the variables which interprets as squared loadings. The squared loadings defined in (32) write after rotation:

$$\begin{cases} c_{ji,\text{rot}} = a^2_{ji,\text{rot}} & \text{if the variable } \mathbf{x}_j \text{ is numerical,} \\ c_{ji,\text{rot}} = \sum_{s \in I_j} \frac{n}{n_s} a^2_{si,\text{rot}} & \text{if the variable } \mathbf{x}_j \text{ is categorical,} \end{cases} \tag{34}$$

They measure the links (squared correlations or correlation ratios) between the principal components after rotation and the variables. Maximizing the variance of the squared loadings after rotation leads to high values of squared loadings for several variables and low for the remainder while leaving the quality of the matrix reconstruction unchanged. The varimax rotation problem is then rephrased as

$$\max_{\mathbf{T}} \quad \{f(\mathbf{T}) | \mathbf{T}\mathbf{T}^\top = \mathbf{T}^\top\mathbf{T} = \mathbb{I}_q\}, \tag{35}$$

where

$$f(\mathbf{T}) = \sum_{i=1}^{q}\sum_{j=1}^{p}(c_{ji,\text{rot}})^2 - \frac{1}{p}\sum_{i=1}^{q}\left(\sum_{j=1}^{p} c_{ji,\text{rot}}\right)^2. \tag{36}$$

The objective function (36) also writes (see Appendix C):

$$f(\mathbf{T}) = \sum_{i=1}^{q}\sum_{j=1}^{p}(\tilde{c}_{ji,\text{rot}})^2 - \frac{1}{p}\sum_{i=1}^{q}\left(\sum_{j=1}^{p} \tilde{c}_{ji,\text{rot}}\right)^2. \tag{37}$$

where $\tilde{c}_{ji,\text{rot}}$ are the squared loadings after rotation obtained with the standard SVD

$$\tilde{\mathbf{Z}} = \tilde{\mathbf{U}}\tilde{\mathbf{\Lambda}}\tilde{\mathbf{V}}^\top,$$

of the matrix $\tilde{\mathbf{Z}} = \mathbf{N}^{1/2}\mathbf{Z}\mathbf{M}^{1/2}$ (see Remark 1). The rotation procedure proposed by (Chavent et al., 2012) uses the standard SVD of $\tilde{\mathbf{Z}}$ to optimize the objective function (37). This procedure summarized in Appendix B finds an optimal rotation matrix $\mathbf{T}$ and gives:

$$\tilde{\mathbf{U}}_{\text{rot}} = \tilde{\mathbf{U}}_q\mathbf{T} \tag{38}$$

$$\tilde{\mathbf{A}}_{\text{rot}} = \tilde{\mathbf{A}}_q\mathbf{T} \tag{39}$$

Rotated factor coordinates processing.

1. The matrix of dimension $(p_1 + m) \times q$ of the rotated factor coordinates of the p_1 quantitative variables and the m levels of the p_2 categorical variables is (see 58):
$$\mathbf{A}^*_{\text{rot}} = \mathbf{M}\mathbf{A}_{\text{rot}} = \mathbf{M}^{1/2}\tilde{\mathbf{A}}_{\text{rot}}. \tag{40}$$
$\mathbf{A}^*_{\text{rot}}$ is split as follows: $\mathbf{A}^*_{\text{rot}} = \left[\begin{array}{c} \mathbf{A}^*_{1,\text{rot}} \\ \mathbf{A}^*_{2,\text{rot}} \end{array}\right] \begin{array}{l} \}\, p_1 \\ \}\, m \end{array}$ where
   - $\hookrightarrow$ $\mathbf{A}^*_{1,\text{rot}}$ contains the rotated factor coordinates of the p_1 numerical variables,
   - $\hookrightarrow$ $\mathbf{A}^*_{2,\text{rot}}$ contains the rotated factor coordinates of the m levels.
2. The variance $\lambda_{i,\text{rot}}$ of the ith rotated principal component is calculated as:
$$\lambda_{i,\text{rot}} = \|\mathbf{a}_{i,\text{rot}}\|^2_{\mathbf{M}} = \|\tilde{\mathbf{a}}_{i,\text{rot}}\|^2_{\mathbb{I}_{p_1+m}}, \tag{41}$$
where $\mathbf{a}_{i,\text{rot}}$ (resp.$\tilde{\mathbf{a}}_{i,\text{rot}}$) is the ith column of $\mathbf{A}_{\text{rot}}$ (resp. $\tilde{\mathbf{A}}_{\text{rot}}$).
Let $\Lambda_{\text{rot}} = \text{diag}(\sqrt{\lambda_{1,\text{rot}}}, \ldots, \sqrt{\lambda_{q,\text{rot}}})$ denote the diagonal matrix of the standard deviations of the q rotated principal components.
3. The matrix of dimension $n \times q$ of the rotated factor coordinates of the n observations is:
$$\mathbf{F}_{\text{rot}} = \mathbf{U}_{\text{rot}}\Lambda_{\text{rot}} = \mathbf{N}^{-1/2}\tilde{\mathbf{U}}_{\text{rot}}\Lambda_{\text{rot}}. \tag{42}$$

Remark 5. *For numerical data,* `PCArot` *is the standard varimax procedure defined by Kaiser (1958) for rotation in PCA. For categorical data,* `PCArot` *is an orthogonal rotation procedure for Multiple Correspondence Analysis (MCA).*

4.2. Properties of `PCArot`

The properties used to interpret the graphical outputs of `PCAmix` remain true after rotation:

- the rotated factor coordinates of the p_1 numerical variables (the first p_1 rows of $\mathbf{A}^*_{\text{rot}}$) are correlations with the rotated principal components (the columns of $\mathbf{F}_{\text{rot}}$),
- the rotated factor scores of the m levels (the m last rows of $\mathbf{A}^*_{\text{rot}}$) are mean values of the (standardized) rotated factor coordinates of the observations that belong these levels.

The contribution (squared loading) of the variable $\mathbf{x}_j$ to the variance of the rotated principal component $\mathbf{f}_{i,\text{rot}}$ is calculated directly from the matrix $\tilde{\mathbf{A}}_{\text{rot}}$ with:

$$\left\{\begin{array}{ll} c_{ji,\text{rot}} = \tilde{a}^2_{ji,\text{rot}} = r^2(\mathbf{f}_{i,\text{rot}}, \mathbf{x}_j) & \text{if the variable } \mathbf{x}_j \text{ is numerical,} \\ c_{ji,\text{rot}} = \sum_{s\in I_j} \tilde{a}^2_{si,\text{rot}} = \eta^2(\mathbf{f}_{i,\text{rot}}|\mathbf{x}_j) & \text{if the variable } \mathbf{x}_j \text{ is categorical.} \end{array}\right. \tag{43}$$

The squared loadings after rotation are then the squared correlation or correlation ratio between the variables and the rotated principal components.

The function `plot.PCAmix` presented in Section 3.5 plots the observations, the numerical variables and the levels of the categorical variables according to their factor coordinates after rotation. It also plots variables according to their squared loadings after rotation. The interpretation rules given in Section 3.3 remain true.

4.3. Prediction of rotated PC scores with `predict.PCAmix`

`PCArot` computes q new non correlated numerical variables called rotated principal components that will explain the same part of inertia than `PCAmix` but with simpler interpretation. Let us show that the rotated principal components (columns of $\mathbf{F}_{\text{rot}}$) are linear combination of the columns of $\mathbf{Z}$.

First it can be showed (see Appendix D) that:

$$\mathbf{F}_{\text{rot}} = \mathbf{Z}\mathbf{V}_{\text{rot}}, \tag{44}$$

with

$$\mathbf{V}_{\text{rot}} = \mathbf{M}^{1/2}\tilde{\mathbf{V}}_q\tilde{\mathbf{\Lambda}}_q^{-1}\mathbf{T}\mathbf{\Lambda}_{\text{rot}}, \tag{45}$$

and

$$\mathbf{T} = \tilde{\mathbf{U}}_q^{\top}\tilde{\mathbf{U}}_{\text{rot}}. \tag{46}$$

It follows that the ith rotated principal component $\mathbf{f}_{i,\text{rot}}$ of `PCArot` writes as a linear combination of the vectors $\mathbf{z}_1, \ldots, \mathbf{z}_{p_1+m}$ (columns of $\mathbf{Z}$):

$$\mathbf{f}_{i,\text{rot}} = \mathbf{Z}\mathbf{v}_{i,\text{rot}} = \sum_{\ell=1}^{p_1+m} v_{\ell i,\text{rot}}\mathbf{z}_\ell. \tag{47}$$

It is then easy to write $\mathbf{f}_{i,\text{rot}}$ as a linear combination of the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{p_1+m}$ (columns of $\mathbf{X} = (\mathbf{X}_1|\mathbf{G})$):

$$\mathbf{f}_{i,\text{rot}} = \beta_{0i,\text{rot}} + \sum_{\ell=1}^{p_1+m} \beta_{\ell i,\text{rot}}\mathbf{x}_\ell, \tag{48}$$

with the coefficients

$$\begin{aligned}
\beta_{0i,\text{rot}} &= -\sum_{\ell=1}^{p_1} v_{\ell i,\text{rot}}\frac{\bar{\mathbf{x}}_\ell}{\hat{\sigma}_\ell} - \sum_{\ell=p_1+1}^{p_1+m} v_{\ell i,\text{rot}}\frac{n}{n_\ell}\bar{\mathbf{x}}_\ell,\\
\beta_{\ell i,\text{rot}} &= v_{\ell i,\text{rot}}\frac{1}{\hat{\sigma}_\ell}, \text{ for } \ell = 1, \ldots, p_1,\\
\beta_{\ell i,\text{rot}} &= v_{\ell i,\text{rot}}\frac{n}{n_\ell}, \text{ for } \ell = p_1+1, \ldots, p_1+m,
\end{aligned}$$

where $\bar{\mathbf{x}}_\ell$ and $\hat{\sigma}_\ell$ are respectively the empirical mean and the standard deviation of the column $\mathbf{x}_\ell$.

The rotated principal components are thereby in (48) linear combinations of the original numerical variables and of the original indicator vectors of the levels of the categorical variables. The function `predict.PCAmix` uses these coefficients to predict the scores (coordinates) of new observations on the q rotated principal components of `PCArot`.

4.4. Illustration of `PCArot`

Let us now illustrate the procedure `PCArot` with the mixed data table `housing` already used in Section3.5. Let us first create a data frame without the first ten municipalities (used later for prediction purposes).

```
R> library("PCAmixdata")
R> data("gironde")
R> train <- gironde$housing[-c(1:10), ]
R> split <- splitmix(train)
R> X1 <- split$X.quanti
R> X2 <- split$X.quali
R> res.pcamix <- PCAmix(X.quanti=X1, X.quali = X2, rename.level = TRUE, graph = FALSE)
R> res.pcamix$eig
      Eigenvalue Proportion Cumulative
dim 1  2.5189342  50.378685   50.37868
dim 2  1.0781913  21.563825   71.94251
dim 3  0.6290897  12.581794   84.52430
dim 4  0.4269180   8.538361   93.06267
dim 5  0.3468667   6.937335  100.00000
```

The first $q = 3$ principal components of `PCAmix` retrieve 84.5% of the total inertia. In order to improve the interpretation of these 3 components without adversely affecting the proportion of explained inertia we perform a rotation using the function `PCArot`.

```
R> res.pcarot<-PCArot(res.pcamix, dim = 3, graph = FALSE)
R> res.pcarot$eig #variance of the rotated PCs
         Variance Proportion
dim1.rot 1.919546   38.39092
dim2.rot 1.057868   21.15737
dim3.rot 1.248801   24.97601
```

The spread of the proportion of variance in the three dimensions is modified but the rotated principal components still contain 84.5% of the total inertia:

```
R> sum(res.pcarot$eig[ ,2])
[1] 84.5243
```

The rotation also modifies squared loadings with more clear association after rotation between the third principal component and the variable density. Indeed the squared correlation between `density` and the third PC is equal to 0.39 before rotation and increases to 0.9 after rotation.

```
R> res.pcamix$sqload[ ,1:3]
           dim 1 dim 2 dim 3
density     0.49  0.07  0.39
primaryres  0.00  0.94  0.02
owners      0.73  0.02  0.00
houses      0.68  0.03  0.03
council     0.61  0.01  0.18
```

```
R> res.pcarot$sqload
           dim1.rot dim2.rot dim3.rot
density        0.04     0.01     0.90
primaryres     0.00     0.96     0.01
owners         0.48     0.03     0.25
houses         0.63     0.03     0.08
council        0.76     0.03     0.01
```

Because the rotation improves the interpretation of the third principal component while the second component hardly changed, we plot the observations and the variables on the dimensions 1 and 3.

```
R> plot(res.pcamix, choice = "ind", axes = c(1,3), label = FALSE,
      main = "Observations before rotation")
R> plot(res.pcarot, choice = "ind", axes = c(1,3), label = FALSE,
      main = "Observations after rotation")
R> plot(res.pcamix, choice = "sqload", axes = c(1,3),
      main="Variables before rotation", coloring.var = TRUE, leg = TRUE)
R> plot(res.pcarot, choice = "sqload", axes = c(1,3),
      main="Variables after rotation", coloring.var = TRUE, leg = TRUE)
```

Figure 4 shows how the variable `density` is more clearly linked after rotation to the third principal component. Indeed, after rotation, the coordinates of the variable `density` on the y-axis is equal to 0.9 (the squared correlation between `density` and the 3rd rotated principal component). The municipalities at the top of the plot of the observations after rotation are then characterized by their population density. Note that the benefit of using rotation on this dataset is limited.

Prediction after rotation. Let us now predict the scores of the 10 first municipalities of the data table `housing` on the rotated principal components of `PCArot`.

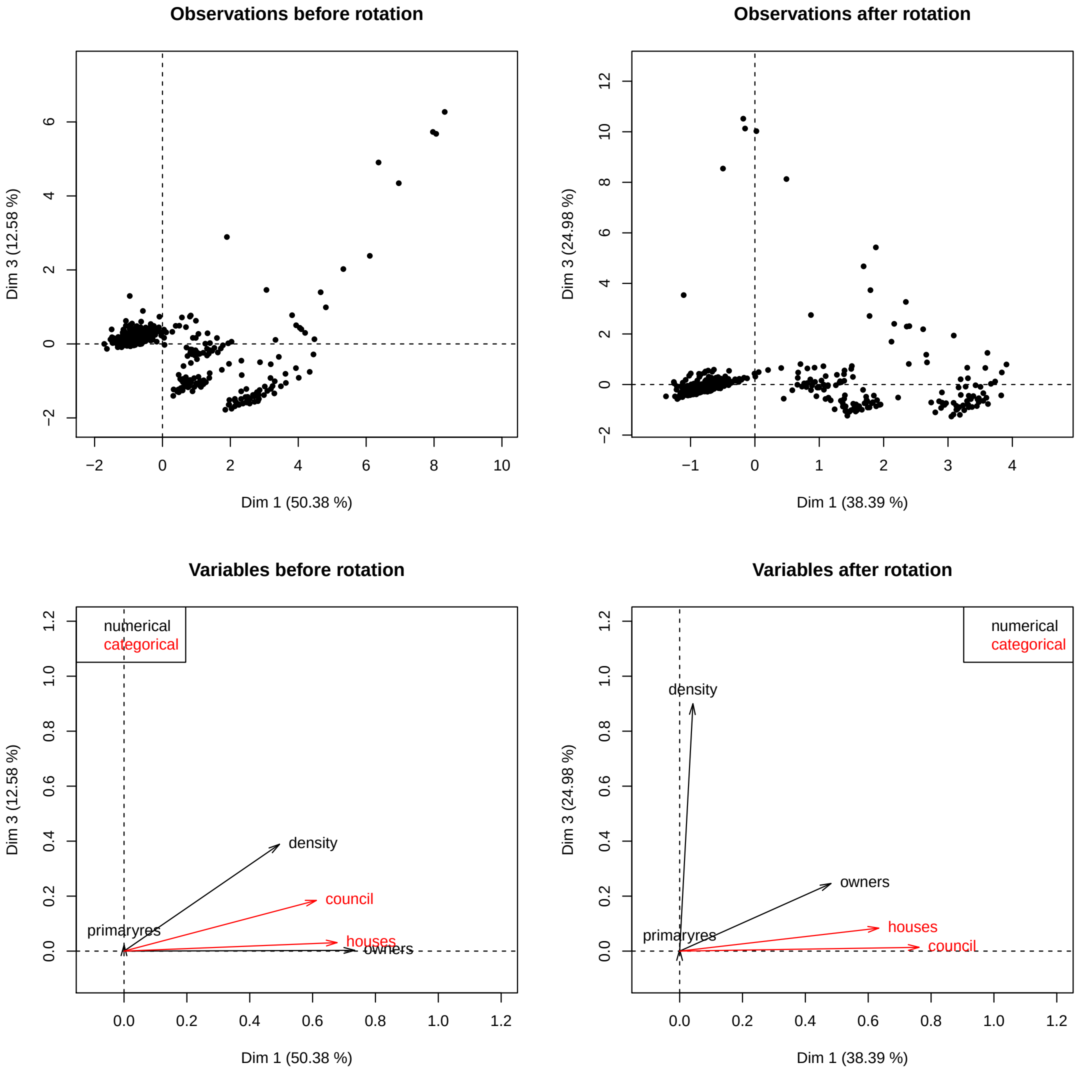


Figure 4: Graphical outputs of PCAmix applied to the data table housing (deprived of the 10 first rows) before rotation (left) and after rotation with PCArot (right).

```
R> test <- gironde$housing[1:10, ]
R> splitnew <- splitmix(test)
R> X1new <- splitnew$X.quanti
R> X2new<-splitnew$X.quali
R> pred.rot <- predict(object = res.pcarot, X.quanti = X1new, X.quali = X2new)
```

```
R> pred.rot
```

```
                    dim1.rot   dim2.rot    dim3.rot
ABZAC               3.2685436  0.3494533 -0.85177749
AILLAS             -0.7235629  0.1200285 -0.22254455
AMBARES-ET-LAGRAVE  2.8852451  0.9823515 -0.03451571
AMBES               1.7220716  1.1590890 -0.78227835
ANDERNOS-LES-BAINS  0.3423361 -2.6886415  0.90574890
ANGLADE            -0.9131248 -0.4514258 -0.20108349
ARBANATS           -0.6653760  0.4217893  0.13105217
ARBIS              -0.7668742  0.3099338 -0.23304721
ARCACHON            1.8825083 -4.4533014  2.36935740
ARCINS             -0.6896492  0.2060403 -0.09049882
```

These predicted coordinates can be used to plot the 10 supplementary municipalities on the rotated principal component map of the other 532 municipalities (Figure 5).

```
R> plot(res.pcarot, axes = c(1,3), label = FALSE, main = "Observations map after rotatio
R> points(pred.rot[ ,c(1,3)], col = 2, pch = 16)
R> legend("topright", legend = c("train","test"), fill = 1:2, col = 1:2)
```

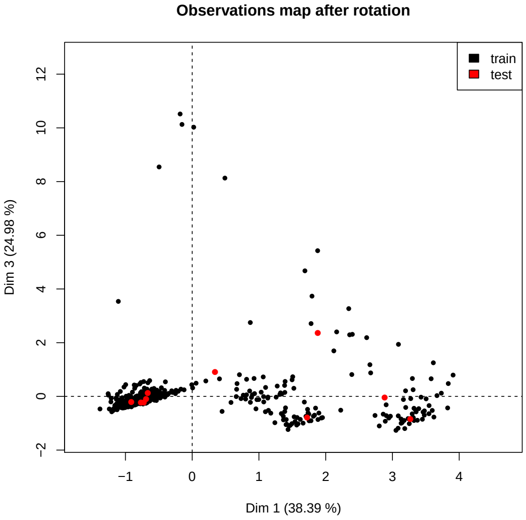


Figure 5: Projection of 10 supplementary municipalities (in red) on the map after rotation.

5. Multiple factor analysis of mixed data

Multiple factor analysis (Escofier and Pagès, 1994; Abdi et al., 2013) is a multivariate analysis method for multi-table data where observations are described by several groups of variables. The straightforward analysis obtained by concatenating all variables in a single data table has the drawback of giving more importance to groups with strong structure. The main idea in Multiple Factor Analysis (MFA) is therefore to give the same importance to each group by weighting each variable by the inverse of the variance of the first principal component of its group. In standard MFA, the nature of the variables (categorical or numerical) can vary from one group to another but the variables within a group must be of the same nature. The `MFAmix` procedure proposed in this paper works with mixed data even within a group.

5.1. The `MFAmix` algorithm

Here the p variables are separated into G groups. The types of variables within a group can be mixed. Each group is represented by a data matrix $\mathbf{X}^{(g)} = [\mathbf{X}_1^{(g)}, \mathbf{X}_2^{(g)}]$ where $\mathbf{X}_1^{(g)}$ (resp. $\mathbf{X}_2^{(g)}$) contains the numerical (resp. categorical) variables of group $g = 1, \ldots, G$. The numerical columns (resp. the categorical columns) of the matrices $\mathbf{X}^{(g)}$ are concatenated in a global numerical data matrix $\mathbf{X}_1 = [\mathbf{X}_1^{(1)}, \ldots, \mathbf{X}_1^{(G)}]$ (resp. a global categorical data matrix $\mathbf{X}_2 = [\mathbf{X}_2^{(1)}, \ldots, \mathbf{X}_2^{(G)}]$). Let $\mathbf{Z}$ denote the matrix constructed with $\mathbf{X}_1$ and $\mathbf{X}_2$ as described in the pre-processing step of `PCAmix` in Section 3.1. The matrix $\mathbf{Z}$ has then n rows and $p_1 + m$ columns where $p_1 = p_1^{(1)} + \ldots + p_1^{(G)}$ and $m = m^{(1)} + \ldots + m^{(G)}$. Each column of $\mathbf{Z}$ is either a numerical variable (standardized) or the indicator vector of a level (centered). Let $\mathbf{N} = \frac{1}{n}\mathbb{I}_n$ and $\mathbf{M} = \text{diag}(1, \ldots, 1, \frac{n}{n_1}, \ldots, \frac{n}{n_m})$ be the diagonal matrices of the weights of the rows and columns of $\mathbf{Z}$.

The `MFAmix` algorithm is a procedure where the first step modifies the weights of the columns of $\mathbf{Z}$ to equilibrate the importance of the groups in a global `PCAmix` analysis.

Step 1: weighting step.

1. For $g = 1, \ldots, G$, compute the first eigenvalue $\lambda_1^{(g)}$ of `PCAmix` applied to $\mathbf{X}^{(g)}$.
2. Build the diagonal matrix $\mathbf{P}$ of the weights $\frac{1}{\lambda_1^{(t_k)}}$ where $t_k \in \{1, \ldots, g, \ldots, G\}$ denote the group of the kth column of $\mathbf{Z}$.
3. Build the diagonal matrix $\mathbf{MP}$ of the new weights of the column of $\mathbf{Z}$.

Step 2: re-weighted global `PCAmix` step.

1. The GSVD of $\mathbf{Z}$ with metrics $\mathbf{N}$ on $\mathbb{R}^n$ and $\mathbf{MP}$ on $\mathbb{R}^{p_1+m}$ gives:

$$\mathbf{Z} = \mathbf{U}_{\text{mfa}} \mathbf{\Lambda}_{\text{mfa}} \mathbf{V}_{\text{mfa}}^\top,$$

   as defined in (1). Let r denote the rank of $\mathbf{Z}$.

2. The matrix of dimension $n \times r$ of the factor coordinates of the n observations is:

$$\mathbf{F}_{\text{mfa}} = \mathbf{U}_{\text{mfa}} \mathbf{\Lambda}_{\text{mfa}}. \tag{49}$$

3. The matrix of dimension $(p_1+m)\times r$ of the factor coordinates of the p_1 quantitative variables and the m levels is:

$$\mathbf{A}^*_{\text{mfa}} = \mathbf{M}\mathbf{V}_{\text{mfa}}\mathbf{\Lambda}_{\text{mfa}}. \tag{50}$$

   The first p_1 rows contain the factor coordinates of the numerical variables and the following m rows contain the factor coordinates of the levels.

Step 3: squared loading processing. The squared loadings are the contributions of the p variables to the variance of the r principal components (columns of $\mathbf{F}_{\text{mfa}}$). It comes from Section 2.1 that the variance of the ith principal component $\mathbf{f}_{i,\text{mfa}}$ is $\text{Var}(\mathbf{f}_{i,\text{mfa}}) = \|\mathbf{a}_{i,\text{mfa}}\|^2_{\mathbf{MP}}$ where $\mathbf{a}_{i,\text{mfa}}$ is the ith loadings vector (column of $\mathbf{A}_{\text{mfa}} = \mathbf{V}_{\text{mfa}}\mathbf{\Lambda}_{\text{mfa}}$). The contribution $c_{ji,\text{mfa}}$ of the variable $\mathbf{x}_j$ to the variance of the principal component $\mathbf{f}_{i,\text{mfa}}$ is then:

$$\begin{cases} c_{ji,\text{mfa}} = \dfrac{1}{\lambda_1^{(t_j)}} a^2_{ji,\text{mfa}} = \dfrac{1}{\lambda_1^{(t_j)}} a^{*2}_{ji,\text{mfa}} & \text{if the variable } \mathbf{x}_j \text{ is numerical,} \\ c_{ji,\text{mfa}} = \displaystyle\sum_{s\in I_j} \frac{1}{\lambda_1^{(t_s)}} \frac{n}{n_s} a^2_{si,\text{mfa}} = \sum_{s\in I_j} \frac{1}{\lambda_1^{(t_s)}} \frac{n_s}{n} a^{*2}_{si,\text{mfa}} & \text{if the variable } \mathbf{x}_j \text{ is categorical,} \end{cases} \tag{51}$$

where I_j is the set of indices of the levels of the categorical variable $\mathbf{x}_j$. Note that the contributions are no longer squared correlation or correlation ratios as previously in `PCArot` and `PCAmix`.

Remark 6. *In general $q \leq r$ dimensions are required by the user in `MFAmix`.*

5.2. Graphical outputs of `MFAmix`

The graphical outputs of `MFAmix` are obtained with the function `plot.MFAmix`. The standard plots (observations, numerical variables and levels according to their factor coordinates) are interpreted with the same rules as in `PCAmix` (see Section 3.3) which remain true in `MFAmix`. The interpretation of the plot of the variables according to their squared loadings is however slightly different. Indeed, in `MFAmix`, squared loadings need to be interpreted as contributions and no longer as squared correlations or correlation ratios. The group structure of the variables allows to build in `MFAmix` new graphical outputs: plot of the groups, plot of the partial observations and plot of the partial axes.

Contribution of a group. The contribution of a variable is defined in (51). The contribution of a group g is therefore the sum of the contributions of all the variables of the group. The groups can then be plotted as points on a map using their contribution to the variance of the principal components.

Partial observations. The principal component map of the observations reveals the structure common to the groups, but it is not possible to see how each group relates to the principal component space. The visualization of an observation according to a specific group (called a partial observation) can be achieved by projecting the dataset of each group onto this space. This is done as follows:

1. For $g = 1, \dots, G$, construct the matrix $\mathbf{Z}^{(g)}_{\text{part}}$ by equating to zero in $\mathbf{Z}$ the values of the columns k such that $t_k \neq g$. The rows of $\mathbf{Z}^{(g)}_{\text{part}}$ are the partial observations for the group g.

2. For $g = 1, \dots, G$, the factor coordinates of the partial observations are computed as:
$$\mathbf{F}^{(g)}_{\text{part}} = G\,\mathbf{Z}^{(g)}_{\text{part}}\mathbf{MPV}. \tag{52}$$
This matrix contains the coordinates of the orthogonal projections (with respect to the adjusted metric matrix $\mathbf{MP}$) of the n rows of $\mathbf{Z}^{(g)}_{\text{part}}$ onto the axes spanned by the columns of $\mathbf{V}$ (with the number of groups G as multiplying factor). This multiplying factor comes to get the factor coordinates of an observation at the barycenter of the coordinates of its G partial observations.

The partial observations can then be plotted as supplementary points on the principal component map of the observations. To facilitate interpretation, lines linking an observation with its G partial observations are drawn on the map.

Partial axes. The `PCAmix` procedure is applied first to the G separated data tables $\mathbf{X}^{(g)}$. The principal components $\mathbf{f}^{(g)}_i, i = 1 \dots q$ of these separate analyses are called the partial axes. Let $\mathbf{f}_{i,\text{mfa}}$ denote the ith principal component of the global analysis. The link between the separated analysis and the global analysis is explored by computing correlations between the principal components of each separated study and the principal components of the global study. The correlations $r(\mathbf{f}^{(g)}_i, \mathbf{f}_{i,\text{mfa}})$ are used as coordinates to plot the partial axes on a map.

5.3. Prediction of PC scores with `predict.MFAmix`

The $q \leq r$ principal components (PCs) are new numerical variables defined as a linear combination of the vectors $\mathbf{z}_1, \dots, \mathbf{z}_{p_1+m}$ (columns of $\mathbf{Z}$). For $i = 1, \dots, q$:

$$\mathbf{f}_{i,\text{mfa}} = \mathbf{ZMPv}_{i,\text{mfa}} = \sum_{\ell=1}^{p_1} \frac{1}{\lambda_1^{(t_\ell)}} v_{\ell i,\text{mfa}} \mathbf{z}_j + \sum_{\ell=p_1+1}^{p_1+m} \frac{1}{\lambda_1^{(t_\ell)}} \frac{n}{n_\ell} v_{\ell i,\text{mfa}} \mathbf{z}_\ell.$$

It is then easy to write $\mathbf{f}_{i,\text{mfa}}$ as a linear combination of the vectors $\mathbf{x}_1, \dots, \mathbf{x}_{p_1+m}$ (columns of $\mathbf{X} = (\mathbf{X}_1|\mathbf{G})$) where $\mathbf{G}$ is the indicator matrix of the m levels:

$$\mathbf{f}_{i,\text{mfa}} = \beta_{0i,\text{mfa}} + \sum_{\ell=1}^{p_1+m} \beta_{\ell i,\text{mfa}} \mathbf{x}_\ell, \tag{53}$$

with the coefficients

$$\beta_{0i,\text{mfa}} = -\sum_{\ell=1}^{p_1} \frac{1}{\lambda_1^{(t_\ell)}} v_{\ell i,\text{mfa}} \frac{\bar{\mathbf{x}}_\ell}{\hat{\sigma}_\ell} - \sum_{\ell=p_1+1}^{p_1+m} \frac{1}{\lambda_1^{(t_\ell)}} \frac{n}{n_\ell} v_{\ell i,\text{mfa}} \bar{\mathbf{x}},$$

$$\beta_{\ell i,\text{mfa}} = \frac{1}{\lambda_1^{(t_\ell)}} v_{\ell i,\text{mfa}} \frac{1}{\hat{\sigma}_\ell}, \text{ for } \ell = 1, \ldots, p_1,$$

$$\beta_{\ell i,\text{mfa}} = \frac{1}{\lambda_1^{(t_\ell)}} \frac{n}{n_\ell} v_{\ell i,\text{mfa}}, \text{ for } \ell = p_1 + 1, \ldots, p_1 + m,$$

where $\bar{\mathbf{x}}_\ell$ and $\hat{\sigma}_\ell$ are respectively the empirical mean and the standard deviation of the column $\mathbf{x}_\ell$.

The principal components are thereby written in (53) as a linear combination of the original numerical variables and of the original indicator vectors of the levels of the categorical variables. The function `predict.MFAmix` uses these coefficients to predict the scores (coordinates) of new observations on the first $q \leq r$ principal component of `MFAmix` (where q is chosen by the user).

5.4. Illustration of `MFAmix`

Let us now illustrate the procedure `MFAmix` with the 4 mixed data tables available in the dataset `gironde`. As introduced previously, this dataset describes 542 municipalities on 27 variables separated into 4 groups (Employment, Housing, Services, Environment). The dataset `gironde` is then a list of 4 data tables (one data table by group).

```
R> library("PCAmixdata")
R> data("gironde")
R> names(gironde)
[1] "employment"  "housing"     "services"    "environment"
```

The four groups contain respectively 9, 5, 9 and 4 variables and the description of the variables of each data table is available in Appendix A.

The function `MFAmix` uses three main input arguments:

- `data`: the global data frame obtained by concatenation of the separated data tables,
- `group`: a vector of integer with the index of the group of each variable,
- `name.group`: a vector of character with the name of each group.

```
R> dat <- cbind(gironde$employment, gironde$housing, gironde$services,
gironde$environment)
R> index <- c(rep(1,9), rep(2,5), rep(3,9), rep(4,4))
R> names <- c("employment", "housing", "services", "environment")
R> res.mfamix <- MFAmix(data = dat, groups = index, name.groups = names,
    ndim = 3, rename.level = TRUE, graph = FALSE)
```

The function MFAmix builds an object (of class MFAmix) which is a list with many numerical results described shortly with the print function. Here, the number of dimensions kept in the results is equal to 3. The group structure of the variables gives specific graphical outputs like the four maps of Figure 6.

```
R> plot(res.mfamix, choice = "cor", coloring.var = "groups", leg = TRUE,
     main = "(a) Numerical variables")
R> plot(res.mfamix, choice = "ind", partial = c("SAINTE-FOY-LA-GRANDE"), label = TRUE,
     posleg = "topright", main = "(b) Observations")
R> plot(res.mfamix, choice = "sqload", coloring.var = "groups",
     posleg = "topright", main="(c) All variables")
R> plot(res.mfamix, choice = "groups", coloring.var = "groups", main = "(d) Groups")
```

Figure 6(a) is the correlation circle of the 16 numerical variables, colored according to their group membership. The coordinates of the variables on this map are correlations with the principal components of MFAmix. Because this map can be difficult to read due to multiple overlaying of the names of some variables, it can be useful to look at the numerical values of the coordinates available in the object res.MFAmix.

```
R> coord.var <- res.mfamix$quanti$coord[ , 1:2]
```

Table 1 highlights four numerical variables that are correlated (in absolute value) with the first principal component: density, buildings, owners and agricul. The municipalities on the right hand side of the principal component map in Figure 6(b) have then higher values for variables density and buildings, whereas municipalities to the left have higher values of the variables owners and agric.

To interpret the position of the municipalities at the top and bottom of Figure 6(b), the coordinates of the variables in the second dimension are useful. Table 1 highlights four numerical variables that are correlated with the second principal component: managers, middleempl, employrate, income and vegetation. The position (top or bottom) of the municipalities on the principal component map can then be interpreted with these variables.

For example, Figure 6(b) shows the municipality of SAINTE-FOY-LA-GRANDE plotted with its 4 partial representations (the four colored points linked to it with a line). The position of this municipality on the right of the map suggests a municipality with higher density of population, higher proportion of buildings, less owners and less agricultural land. Its position at the bottom of the map suggests smaller values on 4 variables of the group employment (managers, middleempl,employrate,income) and smaller values on the variable vegetation of the group environment.

Now we come to the 9 categorical variables of the group services. These variables naturally do not appear in the correlation circle, but do appear in Figure 6(c) where all the variables are plotted according to their contributions to the principal components. This map shows that all the variables of the group services (dentist, dentist, nursery,...) contribute strongly to the first principal component. However it is not possible to know in which way. For instance does the municipality SAINTE-FOY-LA-GRANDE

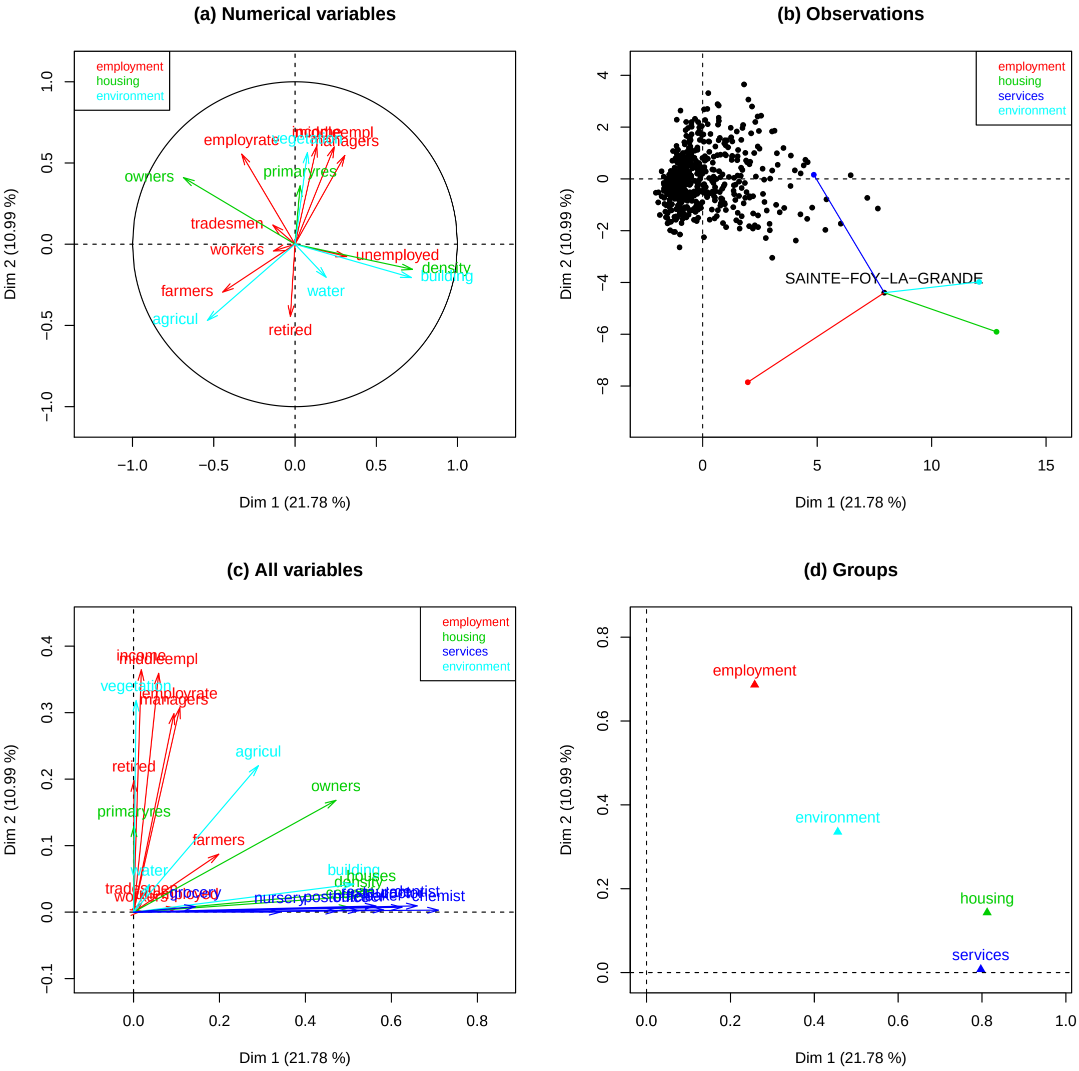


Figure 6: Some graphical outputs of MFAmix applied to the four data table of the dataset gironde.

which has a high score on the first principal component have more or less services than others? This information is given in Figure 7 where the levels of the categorical variables are plotted.

```
R> plot(res.mfamix, choice = "levels", coloring.var = "groups",
     posleg = "bottomleft", main = "Levels", cex = 1.3, cex.leg = 1.3, xlim = c(-2,4))
```

The level map can be used with the barycentric property to interpret the map of the municipalities given Figure 6(b): the municipalities on the right are provided with more

Table 1: Factor coordinates of the variables obtained with `MFAmix`

	dim 1	dim 2
farmers	-0.45	-0.30
tradesmen	-0.14	0.12
managers	0.31	**0.55**
workers	-0.13	-0.04
unemployed	0.32	-0.08
middleempl	0.24	**0.60**
retired	-0.03	-0.44
employrate	-0.33	0.55
income	0.13	**0.60**
density	**0.72**	-0.15
primaryres	0.03	0.36
owners	**-0.69**	0.41
building	**0.72**	-0.21
water	0.19	-0.20
vegetation	0.08	0.56
agricul	**-0.54**	-0.47

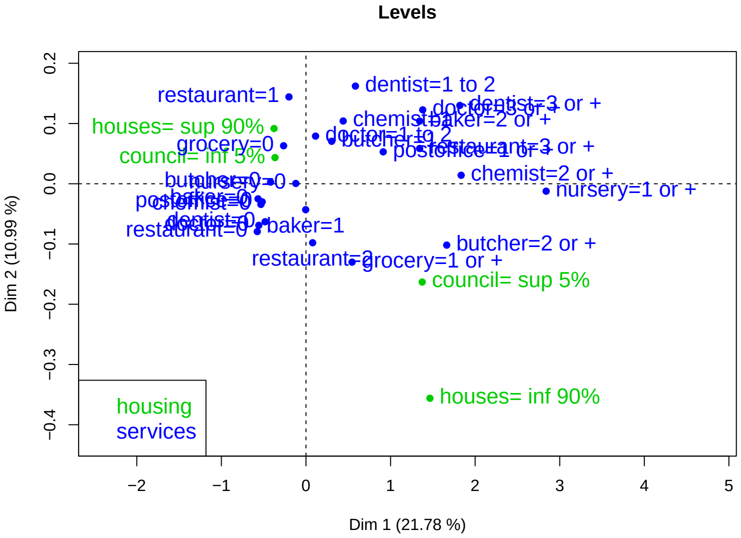


Figure 7: Plot of the levels of the 10 categorical variables after applying MFAmix.

services than those on the left. The municipalities to the bottom right of the map (like `SAINTE-FOY-LA-GRANDE`) have more likely a smaller proportion of houses.

In summary, the municipality SAINTE-FOY-LA-GRANDE is a municipality with a good level of services, but with a fairly stagnant employment market and whose inhabitants are more likely to live in apartments than in other municipalities.

The last map Figure 6(d) is the plot of the groups according to their contributions to the first two principal components. This map confirms the previous interpretations of the principal components of MFAmix and the impact of the groups services and housing on the first dimension as well as the impact of the group employment on the second dimension.

Predicted scores for new observations. The scores of new observations can be obtained with the predict.MFAmix function. The municipality SAINTE-FOY-LA-GRANDE for instance can be considered as supplementary and plotted as an illustrative observation (test sample) on the map given in Figure 8 obtained with the $n-1$ remaining municipalities (training sample).

```
R> sel <- which(rownames(dat) == "SAINTE-FOY-LA-GRANDE")
R> res.mfamix <- MFAmix(data = dat[-sel,], groups = index,
                        name.groups = names, rename.level = TRUE, graph = FALSE)
R> pred <- predict(res.mfamix, dat[sel, , drop=FALSE])
```

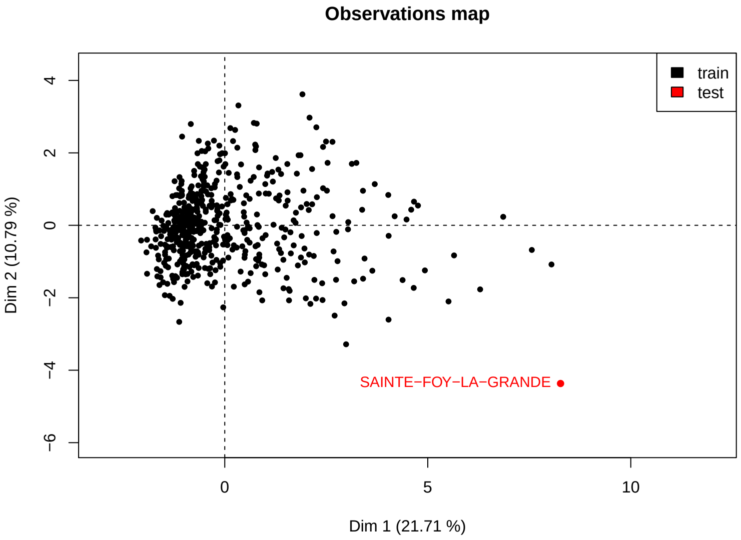


Figure 8: The municipality SAINTE-FOY-LA-GRANDE is plotted in supplementary in the graphical output of MFAmix.

Supplementary groups. The supvar.MFAmix function calculates the coordinates of supplementary groups of variables on the maps of MFAmix. Let us for instance apply

`MFAmix` with three groups (`employment`, `services`, `environment`) and add the group `housing` as a supplement.

```
R> dat <- cbind(gironde$employment, gironde$services, gironde$environment)
R> names <- c("employment", "services", "environment")
R> mfa <-MFAmix(data = dat, groups = c(rep(1,9), rep(2,9), rep(3,4)),
                     name.groups = names, rename.level =T RUE, graph = FALSE)
R> mfa.sup <- supvar(mfa, data.sup = gironde$housing, groups.sup = rep(1,5),
                name.groups.sup = "housing.sup", rename.level = TRUE)
```

The group `housing` is then plotted as supplementary on the maps of `MFAmix`, see Figure 9.

```
R> plot(mfa.sup, choice = "groups", coloring.var = "groups",
      col.groups = c(2,4,5), col.groups.sup = 3)
R> plot(mfa.sup,choice="cor", coloring.var = "groups",
      col.groups = c(2,4,5), col.groups.sup = 3)
```

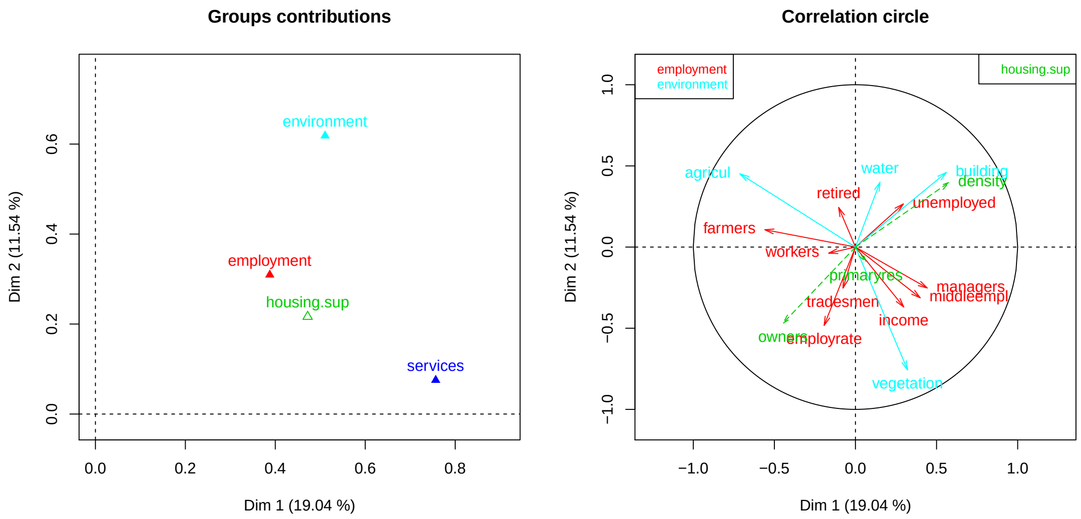


Figure 9: The group housing is plotted as supplementary in the graphical outputs of MFAmix.

6. Concluding remarks

In this paper, the multivariate analysis methods implemented in the R package `PCAmixdata` are presented in such a way that the theoretical details can be read separately from the R examples. Therefore, users interested in the practical aspects of the methods `PCAmix`, PCArot and MFAmix can reproduce the R code provided after each theoretical section, either with the dataset gironde (available in the package) or with their own data. Keys are also provided for the interpretation of most numerical results and graphical outputs.

The definition of multivariate analysis methods to mixed data is important in practice and is sometimes neglected in statistical literature and software. Research and implementation work remain to be done in this sense. For instance, the development of a method of linear discriminant analysis compatible with mixed data is currently under investigation. Moreover, extension of orthogonal rotation to the principal component of MFAmix could be done in the same spirit as PCArot, because MFAmix is a re-weighted general PCAmix analysis, this implementation should not require too many theoretical developments.

The package PCAmixdata handles missing data with a very simple approach where missing values are replaced by mean values for numerical variables and by zeros in the indicator matrix for the categorical variables. Of course more relevant methods like the method proposed by Audigier et al. (2016) and implemented in the R package **missMDA** could be used to complete the missing values.

Acknowledgment

The authors warmly thank the anonymous reviewers who contribute to greatly improve the manuscript.

Appendices

A. The dataset gironde

Table 2 provides the description of all the numerical and categorical variables of the gironde dataset.

Table 2: Description of variables of the gironde dataset

R_Names	Description	Group	Data type
farmers	Percentage of farmers	employment	Num
tradesmen	Percentage of tradesmen and shopkeepers	employment	Num
managers	Percentage of managers and executives	employment	Num
workers	Percentage of workers and employees	employment	Num
unemployed	Percentage of unemployed workers	employment	Num
middleemp	Percentage of middle-range employees	employment	Num
retired	Percentage of retired people	employment	Num
employrate	employment rate	employment	Num
income	Average income	employment	Num
density	Population density	housing	Num
primaryres	Percentage of primary residences	housing	Num
houses	Percentage of houses	housing	Categ
owners	Percentage of home owners living in their primary residence	housing	Num
council	Percentage of council housing	housing	Categ
butcher	Number of butchers	services	Categ
baker	Number of bakers	services	Categ
postoffice	Number of post offices	services	Categ
dentist	Number of dentists	services	Categ
grocery	Number of grocery stores	services	Categ
nursery	Number of child care day nurseries	services	Categ
doctor	Number of doctors	services	Categ
chemist	Number of chemists	services	Categ
restaurant	Number of restaurants	services	Categ
building	Percentage of buildings	environment	Num
water	Percentage of water	environment	Num
vegetation	Percentage of vegetation	environment	Num
agricul	Percentage of agricultural land	environment	Num

B. The iterative optimization step of PCArot

Let $\tilde{\mathbf{U}}_q$ (resp. $\tilde{\mathbf{A}}_q$) denote the matrix of the first q columns of $\tilde{\mathbf{U}}$ (resp. $\tilde{\mathbf{A}} = \tilde{\mathbf{\Lambda}}\tilde{\mathbf{V}}$).

1. Initialization: $\tilde{\mathbf{U}}_{\text{rot}} = \tilde{\mathbf{U}}_q$ and $\tilde{\mathbf{A}}_{\text{rot}} = \tilde{\mathbf{A}}_q$.

2. For each pair of dimensions (l,t), i.e., for $l = 1,\dots,q-1$ and $t = (l+1),\dots,q$:

   $\hookrightarrow$ calculate the angle of rotation $\theta = \psi/4$ with:

$$\psi = \begin{cases} \arccos\left(\dfrac{h}{\sqrt{g^2+h^2}}\right) & \text{if} \quad g \geq 0, \\ -\arccos\left(\dfrac{b}{\sqrt{g^2+h^2}}\right) & \text{if} \quad g \leq 0, \end{cases} \tag{54}$$

   where g and h are given by:

$$g = 2p\sum_{j=1}^{p}\alpha_j\beta_j - 2\sum_{j=1}^{p}\alpha_j\sum_{j=1}^{p}\beta_j, \tag{55}$$

$$h = p\sum_{j=1}^{p}({\alpha_j}^2 - {\beta_j}^2) - \left(\sum_{j=1}^{p}\alpha_j\right)^2 + \left(\sum_{j=1}^{p}\beta_j\right)^2, \tag{56}$$

   with p the total number of variables, and α_j and β_j defined by:

$$\alpha_j = \sum_{s\in I_j}(\tilde{a}^2_{sl,\text{rot}} - \tilde{a}^2_{st,\text{rot}}) \quad \text{and} \quad \beta_j = 2\sum_{s\in I_j}\tilde{a}_{sl,\text{rot}}\tilde{a}_{st,\text{rot}}\ . \tag{57}$$

   Here, I_j is the set of row indices of $\tilde{\mathbf{A}}_{\text{rot}}$ associated with the levels of the variable j in the categorical case and $I_j = \{j\}$ in the numerical case.

   $\hookrightarrow$ calculate the corresponding matrix of planar rotation $\mathbf{T}_2 = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$,

   $\hookrightarrow$ update the matrices $\tilde{\mathbf{U}}_{\text{rot}}$ and $\tilde{\mathbf{A}}_{\text{rot}}$ by rotation of their l-th and t-th columns.

3. Repeat the previous step until the $q(q-1)/2$ successive rotations provide an angle of rotation θ equal to zero.

C. Equivalence between (36) and (37)

We know from (27) that:

$$\mathbf{A}^* = \mathbf{MA} = \mathbf{MV\Lambda}.$$

Moreover we know from (3) that $\mathbf{V} = \mathbf{M}^{-1/2}\tilde{\mathbf{V}}$ and that $\mathbf{\Lambda} = \tilde{\mathbf{\Lambda}}$. It gives then:

$$\mathbf{A}^* = \mathbf{M}^{1/2}\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}} = \mathbf{M}^{1/2}\tilde{\mathbf{A}}. \tag{58}$$

Note that in Chavent et al. (2012) we wrote $\mathbf{A}^* = \mathbf{M}^{-1/2}\tilde{\mathbf{A}}$ which was a mistake. It gives also

$$\mathbf{A} = \mathbf{M}^{-1/2}\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}} = \mathbf{M}^{-1/2}\tilde{\mathbf{A}}. \tag{59}$$

We deduce easily from (59) that:

$$\begin{cases} c_{ji} = a_{ji}^2 = \tilde{a}_{ji}^2 = \tilde{c}_{ji} & \text{if the variable } \mathbf{x}_j \text{ is numerical,} \\ c_{ji} = \sum_{s \in I_j} \frac{n}{n_s} a_{si}^2 = \sum_{s \in I_j} \tilde{a}_{si}^2 = \tilde{c}_{ji} & \text{if the variable } \mathbf{x}_j \text{ is categorical,} \end{cases} \tag{60}$$

D. Proof of (44)

The $q \times q$ rotation matrix $\mathbf{T}$ is such that

$$\tilde{\mathbf{U}}_{\text{rot}} = \tilde{\mathbf{U}}_q \mathbf{T}. \tag{61}$$

By definition of $\tilde{\mathbf{U}}_q$, we have $\tilde{\mathbf{U}}_q^\top \tilde{\mathbf{U}}_q = \mathbb{I}_q$. It gives (46). By definition, $\tilde{\mathbf{F}}_{\text{rot}} = \tilde{\mathbf{U}}_{\text{rot}} \boldsymbol{\Lambda}_{\text{rot}}$. It gives $\tilde{\mathbf{F}}_{\text{rot}} = \tilde{\mathbf{U}}_q \mathbf{T} \boldsymbol{\Lambda}_{\text{rot}}$. The SVD decomposition $\tilde{\mathbf{Z}} = \tilde{\mathbf{U}} \tilde{\boldsymbol{\Lambda}} \tilde{\mathbf{V}}^\top$ gives $\tilde{\mathbf{U}}_q = \tilde{\mathbf{Z}} \tilde{\mathbf{V}}_q \tilde{\boldsymbol{\Lambda}}_q^{-1}$. Then $\tilde{\mathbf{F}}_{\text{rot}} = \tilde{\mathbf{Z}} \tilde{\mathbf{V}}_q \tilde{\boldsymbol{\Lambda}}_q^{-1} \mathbf{T} \boldsymbol{\Lambda}_{\text{rot}}$. With $\tilde{\mathbf{F}}_{\text{rot}} = \mathbf{N}^{1/2} \mathbf{F}_{\text{rot}}$ and $\tilde{\mathbf{Z}} = \mathbf{N}^{1/2} \mathbf{Z} \mathbf{M}^{1/2}$, it gives (44) and (45).

References

Abdi, H. (2007). Singular value decomposition (SVD) and generalized singular value decomposition. *Encyclopedia of measurement and statistics*, pages 907–912.

Abdi, H., Williams, L. J., and Valentin, D. (2013). Multiple Factor Analysis: Principal Component Analysis for Multitable and Multiblock Data Sets. *Wiley Interdisciplinary Reviews: Computational Statistics*, 5(2):149–179.

Audigier, V., Husson, F., and Josse, J. (2016). A principal component method to impute missing values for mixed data. *Advances in Data Analysis and Classification*, 10(1):5–26.

Beaton, D., Fatt, C. R. C., and Abdi, H. (2014). An ExPosition of multivariate analysis with the singular value decomposition in R. *Computational Statistics and Data Analysis*, 72(0):176 – 189.

Bécue-Bertaut, M. and Pagès, J. (2008). Multiple factor analysis and clustering of a mixture of quantitative, categorical and frequency data. *Computational Statistics and Data Analysis*, 52(6):3255–3268.

Chavent, M., Kuentz-Simonet, V., Labenne, A., Liquet, B., and Saracco, J. (2017). ***PCAmixdata**: Multivariate Analysis of Mixed Data.* R package version 3-1.

Chavent, M., Kuentz-Simonet, V., and Saracco, J. (2012). Orthogonal Rotation in PCAMIX. *Advances in Data Analysis and Classification*, 6(2):131–146.

de Leeuw, J. and Mair, P. (2009). Gifi methods for Optimal scaling in R: The Package homals. *Journal of Statistical Software*, 31(4):1–20.

de Leeuw, J. and van Rijckevorsel, J. L. A. (1980). HOMALS and PRINCALS, some generalizations of principal components analysis. In Diday, E. and al., editors, *Data analysis and informatics II*, pages 231–242. Elsevier Science Publishers, Amsterdam.

Dray, S. and Dufour, A.-B. (2007). The **ade4** Package: Implementing the Duality Diagram for Ecologists. *Journal of Statistical Software*, 22(4):1–20.

Dray, S., Dufour, A.-B., Thioulouse, J., et al. (2017). ***ade4**: Analysis of Ecological Data: Exploratory and Euclidean Methods in Environmental Sciences.* R package version 1.7-8.

Escofier, B. and Pagès, J. (1994). Multiple Factor Analysis (**AFMULT** Package). *Computational Statistics & Data Analysis*, 18(1):121–140.

Hill, M. and Smith, A. (1976). Principal Component Analysis of Taxonomic Data with Multi-State Discrete Characters. *Taxon*, 25(2/3):249–255.

Husson, F., Josse, J., Lê, S., and Mazet, J. (2017). ***FactoMineR**: Multivariate Exploratory Data Analysis and Data Mining.* R package version 1.38.

Kaiser, H. F. (1958). The Varimax Criterion for Analytic Rotation in Factor Analysis. *Psychometrika*, 23(3):187–200.

Kiers, H. A. (1991). Simple Structure in Component Analysis Techniques for Mixtures of Qualitative and Quantitative Variables. *Psychometrika*, 56(2):197–212.

Lê, S., Josse, J., and Husson, F. (2008). **FactoMineR**: an R Package for Multivariate Analysis. *Journal of Statistical Software*, 25(1):1–18.

Mair, P., Leeuw, J. D., and Groenen, P. J. F. (2019). ***Gifi**: Multivariate Analysis with Optimal Scaling.* R package version 0.3-9.

Pagès, J. (2004). Analyse Factorielle de Données Mixtes. *Revue de Statistique Appliquée*, 52(4):93–111.

R Core Team (2017). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria.