Journal of Statistical Software

July 2018, Volume 85, Issue 10. doi: 10.18637/jss.v085.i10

ContaminatedMixt: An R Package for Fitting Parsimonious Mixtures of Multivariate Contaminated Normal Distributions

Antonio Punzo
University of Catania

Angelo Mazza
University of Catania

Paul D. McNicholas
McMaster University

Abstract

We introduce the R package **ContaminatedMixt**, conceived to disseminate the use of mixtures of multivariate contaminated normal distributions as a tool for robust clustering and classification under the common assumption of elliptically contoured groups. Thirteen variants of the model are also implemented to introduce parsimony. The expectation-conditional maximization algorithm is adopted to obtain maximum likelihood parameter estimates, and likelihood-based model selection criteria are used to select the model and the number of groups. Parallel computation can be used on multicore PCs and computer clusters, when several models have to be fitted. Differently from the more popular mixtures of multivariate normal and t distributions, this approach also allows for automatic detection of mild outliers via the maximum *a posteriori* probabilities procedure. To exemplify the use of the package, applications to artificial and real data are presented.

Keywords: mixture models, EM algorithm, contaminated normal distribution, outlier detection, robust clustering, robust estimates.

1. Introduction

Finite mixtures of distributions are commonly used in statistical modeling as a powerful device for clustering and classification by often assuming that each mixture component represents a cluster (or group or class) in the original data (see McLachlan and Basford 1988, Fraley and Raftery 1998, Böhning 2000 and McNicholas 2016).

For continuous multivariate random variables, attention is commonly focused on mixtures of multivariate normal distributions because of their computational and theoretical convenience. However, real data are often "contaminated" by outliers, i.e., observations that do not comply with the model assumed and affect the estimation of the component means and

covariance matrices (see, e.g., Barnett and Lewis 1994, Becker and Gather 1999, Bock 2002, and Gallegos and Ritter 2009). Outliers are "mild" (also referred to as bad points herein, in analogy with Aitkin and Wilson 1980) when they can be modeled by means of more flexible distributions, usually elliptically symmetric and endowed with heavy tails (Ritter 2015, p. 79). These distributions offer the flexibility needed for achieving mild outlier robustness, whereas the multivariate normal distribution, often used as the reference distribution for the good observations, lacks sufficient fit; for a discussion about the concept of reference distribution, see Davies and Gather (1993) and Hennig (2002). In this context, the multivariate t distribution (see, e.g., Lange, Little, and Taylor 1989), the heavy-tailed versions of the multivariate power exponential distribution (see, e.g., Gómez-Villegas, Gómez-Sánchez-Manzano, Maín, and Navarro 2011), and the multivariate leptokurtic-normal distribution (Bagnato, Punzo, and Zoia 2017), represent possible alternatives. When used as mixture components, these distributions respectively yield mixtures of multivariate t distributions (McLachlan and Peel 1998 and Peel and McLachlan 2000), mixtures of multivariate power exponential distributions (Zhang and Liang 2010 and Dang, Browne, and McNicholas 2015), and mixtures of multivariate leptokurtic-normal distributions (Bagnato *et al.* 2017). Although these methods robustify the estimation of the component means and covariance matrices with respect to mixtures of multivariate normal distributions, they do not allow for automatic detection of bad points. To overcome this problem, Punzo and McNicholas (2016) introduce mixtures of multivariate contaminated normal distributions. The multivariate contaminated normal distribution, which dates back to the seminal work of Tukey (1960), is a further common and simple elliptically symmetric generalization of the multivariate normal distribution having heavier tails for the occurrence of bad points; it is a two-component normal mixture in which one of the components, with a large prior probability, represents the good observations (reference distribution), and the other, with a small prior probability, the same mean, and an inflated covariance matrix, represents the bad observations (Aitkin and Wilson 1980). For further recent uses of this distribution in model-based clustering, see Punzo, Blostein, and McNicholas (2017); Punzo and McNicholas (2017), Punzo and Maruotti (2016), and Maruotti and Punzo (2017).

In this paper we present the R (R Core Team 2018) package **ContaminatedMixt** (Punzo, Mazza, and McNicholas 2018), which is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=ContaminatedMixt`. The package allows for model-based clustering and classification by means of a family, proposed by Punzo and McNicholas (2016), of fourteen parsimonious variants of mixtures of multivariate contaminated normal distributions. Parsimony is attained by applying the eigen-decomposition of the component scale matrices, in the fashion of Banfield and Raftery (1993) and Celeux and Govaert (1995). Fitting is performed via the expectation-conditional maximization (ECM) algorithm (Meng and Rubin 1993) and likelihood-based model selection criteria are adopted to select both the number of mixture components and the parsimonious model.

Several CRAN packages are available supporting model-based clustering and classification via mixtures of elliptically contoured distributions. A list of them may be found in the CRAN Task View "Cluster Analysis & Finite Mixture Models" (Leisch and Grün 2018). One of the most flexible packages for clustering via mixtures of multivariate normal distributions is package **mclust** (Fraley and Raftery 2007 and Fraley, Raftery, Scrucca, Murphy, and Fop 2017); from version 5.0.0, it provides all of the fourteen parsimonious mixtures of multivariate normal distributions of Celeux and Govaert (1995), obtained via a slightly different

normalization of the component eigenvalues matrices used by Banfield and Raftery (1993), it implements an EM algorithm for model fitting, and it uses the Bayesian information criterion (BIC, Schwarz 1978) to determine the number of components. Instead, the packages **Rmixmod** (Lebret, Iovleff, Langrognet, Biernacki, Celeux, and Govaert 2015) and **mixture** (Browne, ElSherbiny, and McNicholas 2018) fit the fourteen parsimonious models of Celeux and Govaert (1995). Mixtures of multivariate normal distributions, with alternative parsimonious covariance structures, are also implemented by the packages **bgmm** (Biecek, Szczurek, Vingron, and Tiuryn 2012) and **pgmm** (McNicholas, ElSherbiny, McDaid, and Murphy 2018). The **teigen** package (Andrews, Wickins, Boers, and McNicholas 2018) allows to fit a family of fourteen parsimonious mixtures of multivariate t distributions (with eigen-decomposed component scale matrices as in Celeux and Govaert 1995) from a clustering or classification point of view (see Andrews, McNicholas, and Subedi 2011 and Andrews and McNicholas 2012 for details). Finally, although not available on CRAN, the **MPE** package, available at `http://onlinelibrary.wiley.com/doi/10.1111/biom.12351/suppinfo`, allows to fit a family, introduced by Dang *et al.* (2015), of eight parsimonious variants of mixtures of multivariate power exponential distributions (with eigen-decomposed component scale matrices as in Celeux and Govaert 1995).

The paper is organized as follows. Section 2 retraces the models implemented in the **ContaminatedMixt** package, Section 3 outlines the ECM algorithm for maximum likelihood parameters estimation, and Section 4 illustrates some further computational/practical aspects. The relevance of the package is shown, via real and artificial data sets, in Section 5, and conclusions are finally given in Section 6.

2. Methodology

2.1. The general model

For a random vector $\boldsymbol{X}$, taking values in $\mathbb{R}^p$, a finite mixture of multivariate contaminated normal distributions (Punzo and McNicholas 2016) can be written as

$$p\left(\boldsymbol{x};\boldsymbol{\psi}\right)=\sum_{g=1}^{G}\pi_g\left[\alpha_g\phi\left(\boldsymbol{x};\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g\right)+\left(1-\alpha_g\right)\phi\left(\boldsymbol{x};\boldsymbol{\mu}_g,\eta_g\boldsymbol{\Sigma}_g\right)\right], \tag{1}$$

where, for the gth component, π_g is its mixing proportion, with $\pi_g > 0$ and $\sum_{g=1}^{G}\pi_g = 1$, $\alpha_g \in (0,1)$ is the proportion of good observations, and $\eta_g > 1$ denotes the degree of contamination. In (1), $\boldsymbol{\psi}$ contains all of the parameters of the mixture while $\phi\left(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$ represents the distribution of a p-variate normal random vector with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. As a special case, when $\alpha_g \to 1^-$ and $\eta_g \to 1^+$, for each $g = 1,\ldots,G$, we obtain classical mixtures of multivariate normal distributions.

2.2. Parsimonious variants of the general model

Because there are $p\left(p+1\right)/2$ free parameters for each component scale matrix $\boldsymbol{\Sigma}_g$, it is usually necessary to introduce parsimony in model (1) in order to avoid situations where the number of parameters is greater than, or however close to, the number of observations. Following

Family	Model	Volume	Shape	Orientation	$\boldsymbol{\Sigma}_g$	# of free parameters in $\boldsymbol{\Sigma}_g$
Spherical	EII	Equal	Spherical	–	$\lambda \boldsymbol{I}$	1
	VII	Variable	Spherical	–	$\lambda_g \boldsymbol{I}$	G
Diagonal	EEI	Equal	Equal	Axis-Align	$\lambda \boldsymbol{\Delta}$	p
	VEI	Variable	Equal	Axis-Align	$\lambda_g \boldsymbol{\Delta}$	$G + p - 1$
	EVI	Equal	Variable	Axis-Align	$\lambda \boldsymbol{\Delta}_g$	$1 + G\,(p-1)$
	VVI	Variable	Variable	Axis-Align	$\lambda_g \boldsymbol{\Delta}_g$	Gp
General	EEE	Equal	Equal	Equal	$\lambda \boldsymbol{\Gamma}\boldsymbol{\Delta}\boldsymbol{\Gamma}^\top$	$p\,(p+1)\,/2$
	VEE	Variable	Equal	Equal	$\lambda_g \boldsymbol{\Gamma}\boldsymbol{\Delta}\boldsymbol{\Gamma}^\top$	$G + p - 1 + p\,(p-1)\,/2$
	EVE	Equal	Variable	Equal	$\lambda \boldsymbol{\Gamma}\boldsymbol{\Delta}_g\boldsymbol{\Gamma}^\top$	$1 + G\,(p-1) + p\,(p-1)\,/2$
	EEV	Equal	Equal	Variable	$\lambda \boldsymbol{\Gamma}_g\boldsymbol{\Delta}\boldsymbol{\Gamma}_g^\top$	$p + Gp\,(p-1)\,/2$
	VVE	Variable	Variable	Equal	$\lambda_g \boldsymbol{\Gamma}\boldsymbol{\Delta}_g\boldsymbol{\Gamma}^\top$	$Gp + p\,(p-1)\,/2$
	VEV	Variable	Equal	Variable	$\lambda_g \boldsymbol{\Gamma}_g\boldsymbol{\Delta}\boldsymbol{\Gamma}_g^\top$	$G + p - 1 + Gp\,(p-1)\,/2$
	EVV	Equal	Variable	Variable	$\lambda \boldsymbol{\Gamma}_g\boldsymbol{\Delta}_g\boldsymbol{\Gamma}_g^\top$	$1 + G\,(p-1) + Gp\,(p-1)\,/2$
	VVV	Variable	Variable	Variable	$\lambda_g \boldsymbol{\Gamma}_g\boldsymbol{\Delta}_g\boldsymbol{\Gamma}_g^\top$	$Gp\,(p+1)\,/2$

Table 1: Nomenclature, covariance structure, and number of free parameters in $\boldsymbol{\Sigma}_g$ for each member of the family of parsimonious mixtures of multivariate contaminated normal distributions.

Celeux and Govaert (1995), Punzo and McNicholas (2016) consider the eigen-decomposition

$$\boldsymbol{\Sigma}_g = \lambda_g \boldsymbol{\Gamma}_g \boldsymbol{\Delta}_g \boldsymbol{\Gamma}_g^\top, \tag{2}$$

where $\lambda_g = |\boldsymbol{\Sigma}_g|^{1/p}$, $\boldsymbol{\Delta}_g$ is the scaled ($|\boldsymbol{\Delta}_g| = 1$) diagonal matrix of the eigenvalues of $\boldsymbol{\Sigma}_g$ sorted in decreasing order, and $\boldsymbol{\Gamma}_g$ is a $p \times p$ orthogonal matrix whose columns are the normalized eigenvectors of $\boldsymbol{\Sigma}_g$, ordered according to their eigenvalues. Each element in the right-hand side of (2) has a different geometric interpretation: λ_g determines the size (or volume) of the cluster, $\boldsymbol{\Delta}_g$ its shape, and $\boldsymbol{\Gamma}_g$ its orientation.

Following Banfield and Raftery (1993), Celeux and Govaert (1995), and Dang, Punzo, McNicholas, Ingrassia, and Browne (2017), among others, Punzo and McNicholas (2016) impose constraints on the three components of (2) resulting in a family of fourteen parsimonious mixtures of multivariate contaminated normal distributions (Table 1). Sufficient conditions for the identifiability of the models in this family are given in Punzo and McNicholas (2016).

2.3. Modeling framework: Model-based classification

Model-based classification, also known as semi-supervised classification (Chapelle, Schölkopf, and Zien 2010), is receiving renewed attention (see, e.g., Dean, Murphy, and Downey 2006, McNicholas 2010, Andrews *et al.* 2011, Browne and McNicholas 2012, and Subedi, Punzo, Ingrassia, and McNicholas 2013, 2015). However, despite being a more general framework, it remains the "poor cousin" of model-based clustering within the literature.

Consider the random sample $\{\boldsymbol{x}_i\}_{i=1}^n$ from model (1). Without loss of generality, suppose that the first m observations are known to belong to one of G groups; these are the so-called labeled observations. Let $\boldsymbol{z}_i$ be the G-dimensional component-label vector in which the gth

element is $z_{ig} = 1$ if $\boldsymbol{x}_i$ belongs to component g and $z_{ig} = 0$ otherwise, $g = 1, \ldots, G$. If the ith observation is labeled, denote with $\widetilde{\boldsymbol{z}}_i = (\widetilde{z}_{i1}, \ldots, \widetilde{z}_{iG})$ its component-membership indicator. In model-based classification, we use all n observations to estimate the parameters of the mixture; the fitted model is adopted to classify each of the $n - m$ unlabeled observations through the corresponding maximum *a posteriori* (MAP) probability. Note that

$$\text{MAP}\left(z_{ig}\right) = \begin{cases} 1 & \text{if } \max_h\{z_{ih}\} \text{ occurs in component } g, \\ 0 & \text{otherwise.} \end{cases}$$

Using this notation, the model-based classification likelihood can be written as

$$\mathcal{L}\left(\boldsymbol{\psi}\right) = \prod_{i=1}^{m}\prod_{g=1}^{G}\left\{\pi_g\left[\alpha_g\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g\right) + \left(1-\alpha_g\right)\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\eta_g\boldsymbol{\Sigma}_g\right)\right]\right\}^{\widetilde{z}_{ig}} \prod_{i=m+1}^{n} p\left(\boldsymbol{x}_i;\boldsymbol{\psi}\right).$$

We obtain the model-based clustering scenario as a special case when $m = 0$ (see, e.g., Punzo 2014).

3. Maximum likelihood estimation

3.1. An ECM algorithm

To fit the models in Table 1, Punzo and McNicholas (2016) illustrate the expectation-conditional maximization (ECM) algorithm of Meng and Rubin (1993). The ECM algorithm is a variant of the classical expectation-maximization (EM) algorithm (Dempster, Laird, and Rubin 1977), which is a natural approach for maximum likelihood estimation when data are incomplete. In our case, there are two sources of missing data: one arises from the fact that we do not know the component labels $\{\boldsymbol{z}_i\}_{i=m+1}^{n}$ and the other arises from the fact that we do not know whether an observation in group g is good or bad. To denote this second source of missing data, we use $\{\boldsymbol{v}_i\}_{i=1}^{n}$, with $\boldsymbol{v}_i = (v_{i1}, \ldots, v_{iG})$, where $v_{ig} = 1$ if observation i in group g is good and $v_{ig} = 0$ if observation i in group g is bad. By working on the complete-data likelihood

$$\begin{aligned}\mathcal{L}_c\left(\boldsymbol{\psi}\right) \;=\; & \prod_{i=1}^{m}\prod_{g=1}^{G}\left\{\pi_g\left[\alpha_g\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g\right)\right]^{v_{ig}}\left[\left(1-\alpha_g\right)\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\eta_g\boldsymbol{\Sigma}_g\right)\right]^{(1-v_{ig})}\right\}^{\widetilde{z}_{ig}} \\ & \times \prod_{i=m+1}^{n}\prod_{g=1}^{G}\left\{\pi_g\left[\alpha_g\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\boldsymbol{\Sigma}_g\right)\right]^{v_{ig}}\left[\left(1-\alpha_g\right)\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g,\eta_g\boldsymbol{\Sigma}_g\right)\right]^{(1-v_{ig})}\right\}^{z_{ig}} \quad (3)\end{aligned}$$

the ECM algorithm iterates between three steps – an E-step and two CM-steps – until convergence (which is evaluated via the Aitken acceleration criterion; see Aitken 1926 and Lindsay 1995). The only difference from the EM algorithm is that each M-step is replaced by two simpler CM-steps. They arise from the partition $\boldsymbol{\psi} = \{\boldsymbol{\psi}_1, \boldsymbol{\psi}_2\}$, where $\boldsymbol{\psi}_1 = \left\{\pi_g, \alpha_g, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g\right\}_{g=1}^{G}$ and $\boldsymbol{\psi}_2 = \{\eta_g\}_{g=1}^{G}$. In particular, for the most general model VVV, the $(r+1)$th iteration of the ECM algorithm can be summarized/simplified as follows (see Punzo and McNicholas 2016 for details on the model-based clustering paradigm):

E-step: The values of z_{ig} and v_{ig} in (3) are respectively replaced by

$$z_{ig}^{(r)} = \frac{\pi_g^{(r)}\left[\alpha_g^{(r)}\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g^{(r)},\boldsymbol{\Sigma}_g^{(r)}\right)+\left(1-\alpha_g^{(r)}\right)\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g^{(r)},\eta_g^{(r)}\boldsymbol{\Sigma}_g^{(r)}\right)\right]}{p\left(\boldsymbol{x}_i;\boldsymbol{\psi}^{(r)}\right)}$$

and

$$v_{ig}^{(r)} = \frac{\alpha_g^{(r)}\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g^{(r)},\boldsymbol{\Sigma}_g^{(r)}\right)}{\alpha_g^{(r)}\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g^{(r)},\boldsymbol{\Sigma}_g^{(r)}\right)+\left(1-\alpha_g^{(r)}\right)\phi\left(\boldsymbol{x}_i;\boldsymbol{\mu}_g^{(r)},\eta_g^{(r)}\boldsymbol{\Sigma}_g^{(r)}\right)}.$$

CM-step 1: With fixed $\boldsymbol{\psi}_2 = \boldsymbol{\psi}_2^{(r)}$, the parameters in $\boldsymbol{\psi}_1$ are updated as

$$\pi_g^{(r+1)} = \frac{n_g^{(r)}}{n},$$

$$\alpha_g^{(r+1)} = \frac{1}{n_g^{(r)}}\left(\sum_{i=1}^{m}\widetilde{z}_{ig}v_{ig}^{(r)} + \sum_{i=m+1}^{n} z_{ig}^{(r)}v_{ig}^{(r)}\right), \tag{4}$$

$$\boldsymbol{\mu}_g^{(r+1)} = \frac{1}{s_g^{(r)}}\left[\sum_{i=1}^{m}\widetilde{z}_{ig}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right)\boldsymbol{x}_i + \sum_{i=m+1}^{n} z_{ig}^{(r)}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right)\boldsymbol{x}_i\right], \tag{5}$$

and

$$\boldsymbol{\Sigma}_g^{(r+1)} = \frac{1}{n_g^{(r)}}\boldsymbol{W}_g^{(r)},$$

where

$$n_g^{(r)} = \sum_{i=1}^{m}\widetilde{z}_{ig} + \sum_{i=m+1}^{n} z_{ig}^{(r)},$$

$$s_g^{(r)} = \sum_{i=1}^{m}\widetilde{z}_{ig}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right) + \sum_{i=m+1}^{n} z_{ig}^{(r)}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right),$$

and

$$\begin{aligned}\boldsymbol{W}_g^{(r+1)} &= \sum_{i=1}^{m}\widetilde{z}_{ig}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right)\left(\boldsymbol{x}_i-\boldsymbol{\mu}_g^{(r+1)}\right)\left(\boldsymbol{x}_i-\boldsymbol{\mu}_g^{(r+1)}\right)^{\top} \\ &\quad + \sum_{i=m+1}^{n} z_{ig}^{(r)}\left(v_{ig}^{(r)}+\frac{1-v_{ig}^{(r)}}{\eta_g^{(r)}}\right)\left(\boldsymbol{x}_i-\boldsymbol{\mu}_g^{(r+1)}\right)\left(\boldsymbol{x}_i-\boldsymbol{\mu}_g^{(r+1)}\right)^{\top}.\end{aligned} \tag{6}$$

CM-step 2: With fixed $\boldsymbol{\psi}_1 = \boldsymbol{\psi}_1^{(r+1)}$, the parameters in $\boldsymbol{\psi}_2$ are updated as

$$\eta_g^{(r+1)} = \max\left\{1.001, \frac{b_g}{pa_g}\right\},$$

where

$$a_g = \sum_{i=1}^{m}\widetilde{z}_{ig}\left(1-v_{ig}^{(r)}\right) + \sum_{i=m+1}^{n} z_{ig}^{(r)}\left(1-v_{ig}^{(r)}\right)$$

and

$$b_g = \sum_{i=1}^{m} \widetilde{z}_{ig} \left(1 - v_{ig}^{(r)}\right) \delta\left(\boldsymbol{x}_i, \boldsymbol{\mu}_g^{(r+1)}; \boldsymbol{\Sigma}_g^{(r+1)}\right) + \sum_{i=m+1}^{n} z_{ig}^{(r)} \left(1 - v_{ig}^{(r)}\right) \delta\left(\boldsymbol{x}_i, \boldsymbol{\mu}_g^{(r+1)}; \boldsymbol{\Sigma}_g^{(r+1)}\right),$$

with $\delta\left(\boldsymbol{x}_i, \boldsymbol{\mu}_g^{(r+1)}; \boldsymbol{\Sigma}_g^{(r+1)}\right)$ denoting the squared Mahalanobis distance between $\boldsymbol{x}_i$ and $\boldsymbol{\mu}_g^{(r+1)}$ (with covariance matrix $\boldsymbol{\Sigma}_g^{(r+1)}$).

As it is well-documented in Punzo and McNicholas (2016), the weights

$$\left(v_{ig}^{(r)} + \frac{1 - v_{ig}^{(r)}}{\eta_g^{(r)}}\right)$$

in (5) and (6) reduce the impact of bad points in the estimation of the component means $\boldsymbol{\mu}_g$ and the component scale matrices $\boldsymbol{\Sigma}_g$, thereby providing robust estimates of these parameters. For a discussion on down-weighting for the multivariate contaminated normal distribution, see also Little (1988).

The ECM algorithm for the other parsimonious models changes only with respect to the way the terms of the decomposition of $\boldsymbol{\Sigma}_g$ are obtained in the first CM-step. In particular, these updates are analogous to those given by Celeux and Govaert (1995) for their normal parsimonious clustering (GPC) models (corresponding to mixtures of multivariate normal distributions with eigen-decomposed covariance matrices). The only difference is that, on the $(r+1)$th iteration of the algorithm, $\boldsymbol{W}_g^{(r+1)}$ is used instead of the classical scattering matrix

$$\sum_{i=1}^{m} \widetilde{z}_{ig} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_g^{(r+1)}\right) \left(\boldsymbol{x}_i - \boldsymbol{\mu}_g^{(r+1)}\right)^\top + \sum_{i=m+1}^{n} z_{ig}^{(r)} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_g^{(r+1)}\right) \left(\boldsymbol{x}_i - \boldsymbol{\mu}_g^{(r+1)}\right)^\top.$$

4. Further aspects

4.1. Initialization

Many initialization strategies have been proposed for the EM algorithm applied to mixture models (see, e.g., Biernacki, Celeux, and Govaert 2003, Karlis and Xekalaki 2003, and Bagnato and Punzo 2013). The **ContaminatedMixt** package implements the following initializations, all based on providing the initial quantities $\boldsymbol{z}_i^{(0)}$, $\boldsymbol{v}_i^{(0)}$, and $\eta_g^{(0)} = 1.001$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$, to the first CM-step of the ECM algorithm.

`"random.post"`: Each $\boldsymbol{z}_i^{(0)}$ is substituted by a single observation randomly generated – via the `rmultinom()` function of the **stats** package – from a multinomial distribution with probabilities $(1/G, \ldots, 1/G)$. The values $v_{ig}^{(0)}$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$, are, by default, fixed to one, but they can be also provided by the user.

`"random.clas"`: The G values in $\boldsymbol{z}_i^{(0)}$ are randomly generated by a uniform distribution – via the `runif()` function of the **stats** package – and then normalized in order to sum to 1. The values $v_{ig}^{(0)}$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$, are, by default, fixed to one, but they can be also provided by the user.

`"kmeans"`: Hard values for $\boldsymbol{z}_i^{(0)}$, $i = 1, \ldots, n$, are provided by a preliminary run of the k-means algorithm, as implemented by the `kmeans()` function of the **stats** package.

`"mixt"`: For each parsimonious model, the n values $\boldsymbol{z}_i^{(0)}$ are substituted with the posterior probabilities arising from the fitting of the corresponding parsimonious mixture of multivariate normal distributions; the latter is estimated by the `gpcm()` function of the **mixture** package. The values $v_{ig}^{(0)}$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$, are fixed to one.

`"manual"`: The (soft or hard) values of $\boldsymbol{z}_i^{(0)}$, as well as the values of $\boldsymbol{v}_i^{(0)}$, are provided by the user.

4.2. Automatic detection of bad points

For a mixture of multivariate contaminated normal distributions, the classification of an observation $\boldsymbol{x}_i$ means:

Step 1. Determine its cluster membership.

Step 2. Establish if it is either a good or a bad observation in that cluster.

Let $\widehat{\boldsymbol{z}}_i$ and $\widehat{\boldsymbol{v}}_i$ denote, respectively, the expected values of $\boldsymbol{z}_i$ and $\boldsymbol{v}_i$ arising from the ECM algorithm, i.e., $\widehat{z}_{ig}$ is the value of $z_{ig}^{(r)}$ at convergence and $\widehat{v}_{ig}$ is the value of $v_{ig}^{(r)}$ at convergence. To determine the cluster membership of $\boldsymbol{x}_i$, we use the MAP classification, i.e., MAP $(\widehat{z}_{ig})$. We then consider $\widehat{v}_{ih}$, where h is selected such that MAP $(\widehat{z}_{ih}) = 1$, while $\boldsymbol{x}_i$ is considered good if $\widehat{v}_{ih} > 0.5$ and $\boldsymbol{x}_i$ is considered bad otherwise. The resulting information can be used to eliminate the bad points, if such an outcome is desired (Berkane and Bentler 1988). The remaining data may then be treated as effectively being distributed according to a mixture of multivariate normal distributions, and the clustering results can be reported as usual.

4.3. Constraints for detection of bad points

It may be required that in the gth cluster, $g = 1, \ldots, G$, the proportion of good data is at least equal to a pre-determined value α_g^*. In this case, it is easy to show that the update for α_g is

$$\alpha_g^{(r+1)} = \max\left\{\alpha_g^*, \frac{1}{n_g^{(r)}}\left(\sum_{i=1}^{m} \widetilde{z}_{ig} v_{ig}^{(r)} + \sum_{i=m+1}^{n} z_{ig}^{(r)} v_{ig}^{(r)}\right)\right\}.$$

Note that the **ContaminatedMixt** package also allows to fix α_g *a priori*. This is somewhat analogous to the trimmed clustering approach implemented by the **tclust** package (Fritz, García-Escudero, and Mayo-Iscar 2012), where one must specify the proportion of outliers (the so-called trimming proportion) in advance. However, pre-specifying the proportion of bad points *a priori* may not be realistic in many practical scenarios.

4.4. Model selection criteria

Thus far, the number of components G and the covariance structure (cf. Table 1) have been treated as *a priori* fixed. However, in most practical applications, they are unknown, so it is common practice to select them by evaluating a convenient (likelihood-based) model selection

Criterion	Definition	Reference
AIC	$2l\left(\widehat{\boldsymbol{\psi}}\right) - 2q$	Akaike (1973)
AIC_3	$2l\left(\widehat{\boldsymbol{\psi}}\right) - 3q$	Bozdogan (1994)
AICc	$\text{AIC} - 2\frac{q\,(q+1)}{n-q-1}$	Hurvich and Tsai (1989)
AICu	$\text{AICc} - n \ln \frac{n}{n-q-1}$	McQuarrie, Shumway, and Tsai (1997)
AWE	$2l\left(\widehat{\boldsymbol{\psi}}\right) - 2q\left(\frac{3}{2} + \ln n\right)$	Banfield and Raftery (1993)
BIC	$2l\left(\widehat{\boldsymbol{\psi}}\right) - q \ln n$	Schwarz (1978)
CAIC	$2l\left(\widehat{\boldsymbol{\psi}}\right) - q\,(1 + \ln n)$	Bozdogan (1987)
ICL	$\text{BIC} + 2\sum_{i=m+1}^{n}\sum_{g=1}^{G} \text{MAP}\left(\widehat{z}_{ig}\right) \ln \widehat{z}_{ig}$	Biernacki, Celeux, and Govaert (2000)

Table 2: Definition and key reference for the implemented model selection criteria.

criterion over a reasonable range of possible options (for the alternative use of likelihood-ratio tests to select either the parsimonious model or the number of components for a normal mixture, see Punzo, Browne, and McNicholas 2016). The **ContaminatedMixt** package supports the information criteria listed in Table 2, where $l\left(\widehat{\boldsymbol{\psi}}\right)$ is the observed-data log-likelihood and q is the number of free parameters.

Specifically for the model-based classification setting, the package provides three further criteria: (k-fold) cross-validation (CV), Bayesian entropy criterion (BEC; Bouchard and Celeux 2006) and AIC_{cond} (Vandewalle, Biernacki, Celeux, and Govaert 2013); details about the CV criterion can be found in Lebret *et al.* (2015).

5. Package description and illustrative example

In this section we provide a description of the main capabilities of the **ContaminatedMixt** package along with some illustrations.

5.1. Package description

The R package **ContaminatedMixt** is developed in an object-oriented design, using the standard S3 paradigm. For the sake of speed, most parts of the underlying code have been written using the C programming language. Parallel computation can be used on multicore PCs and computer clusters, when several models have to be fitted. Its main function, `CNmixt()`, fits the model(s) in Table 1 (if required, the function also fits the corresponding parsimonious mixtures of normal distributions) and returns a '`ContaminatedMixt`' class object; the arguments of this function, along with their description, are listed in Table 3. Four further functions are available in the package, and they are detailed in Table 4. Finally, the package contains several methods that allow for data extraction and visualization.

Extractors for '`ContaminatedMixt`' class objects are illustrated in Table 5. When several models have been fitted, extractor functions consider the best model according to the information criterion in `criterion` (refer to Table 2), within the subset of estimated models

Arguments	Description
`X`	Matrix of dimension $n \times p$.
`G`	Vector containing the numbers of groups to be tried.
`contamination`	Optional Boolean indicating if the model(s) to be fitted have to be contaminated or not. If `NULL`, then both types of models are fitted.
`model`	Vector indicating the models (`"EII"`, `"VII"`, `"EEI"`, `"VEI"`, `"EVI"`, `"VVI"`, `"EEE"`, `"VEE"`, `"EVE"`, `"EEV"`, `"VVE"`, `"VEV"`, `"EVV"`, `"VVV"`) to be used. If `model = NULL` (default), then all 14 models are fitted.
`initialization`	Initialization strategy for the ECM algorithm. Possible values are `"random.post"`, `"random.clas"`, `"kmeans"`, `"mixt"`, and `"manual"` (see Section 4.1 for details). Default is `initialization = "mixt"`.
`alphafix`	Vector, of dimension G, with fixed *a priori* values for $\alpha_1, \ldots, \alpha_G$. If the length of `alphafix` is different from G, its first element is replicated G times. If `alphafix = NULL` (default), then $\alpha_1, \ldots, \alpha_G$ are estimated.
`alphamin`	Vector with values $\alpha_1^*, \ldots, \alpha_G^*$ (see Section 4.3). If the length of `alphamin` is different from G, its first element is replicated G times. If `alphamin = NULL`, then $\alpha_1, \ldots, \alpha_G$ are estimated without constraints, as in (4). Default value is `0.5`.
`seed`	Seed for the random number generator, when random initializations are used; if `NULL` (default), current seed is not changed.
`start.z`	$n \times G$ matrix with values $z_{ig}^{(0)}$, when `initialization = "manual"`.
`start.v`	$n \times G$ matrix with values $v_{ig}^{(0)}$. If `start.v = NULL` (default), then $v_{ig}^{(0)} = 1$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$.
`start`	When `initialization = "mixt"`, the initialization used for the `gpcm()` function of the **mixture** package (see Browne *et al.* 2018, for details).
`label`	Vector of n integers. It indicates the membership group of each observation. Use `0` when membership is not known. Use `NULL` when membership is unknown for all observations.
`AICcond`	When `TRUE`, AIC_{cond} and BEC are computed (see Section 4.4).
`iter.max`	Maximum number of iterations in the ECM algorithm. Default is `1000`.
`threshold`	Threshold for Aitken's acceleration procedure. Default is `1.0e-03`.
`eps`	Smallest value for the eigenvalues of $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_G$. It is used to prevent the estimation algorithms to be affected by local maxima or degeneracy of the likelihood (Hathaway 1986 and Ingrassia 2004). Default is `1e-100`.
`parallel`	When `TRUE`, the package **parallel** is used for parallel computation. The number of cores to use may be set with the global option `cl.cores`; default value is detected using `detectCores()`.
`k`	Number of (approximately) equal-sized subsamples used in the (k-fold) cross-validation.

Table 3: List of arguments for the function `CNmixt()`.

Functions	Description
`dCN()`	Density of observations based on the multivariate contaminated normal distribution.
`rCN()`	Generates random deviates from the multivariate contaminated normal distribution.
`CNmixtCV()`	List with the CV error rate estimated for each fitted model (see Section 4.4).
`CNpredict()`	Cluster prediction for observations based on a uncontaminated/contaminated normal mixture model whose parameters are specified by the user.

Table 4: Functions included in the **ContaminatedMixt** package in addition to `CNmixt()`.

Extractors	Description
`getBestModel()`	A '`ContaminatedMixt`' class object containing the best model only.
`getPosterior()`	Estimated posterior probabilities $\widehat{z}_{ig}$, $i = 1, \ldots, n$ and $g = 1, \ldots, G$.
`getSize()`	Estimated groups sizes (from the hard classification induced by the MAP operator).
`getCluster()`	Classification vector.
`getDetection()`	Matrix with two columns: the first gives the MAP group memberships whereas the second specifies if the observations are either good or bad (see Section 4.2).
`getPar()`	Estimated parameters (i.e., $\widehat{\psi}$).
`getIC()`	Values for the considered criteria in `criteria`.
`getCV()`	Values for the CV criterion.
`whichBest()`	Position of the model, in the '`ContaminatedMixt`' class object, for the criteria specified in `criteria`.
`whichBestCV()`	Position of the best model, in the '`ContaminatedMixt`' class object, according to the CV criterion.

Table 5: Extractors for '`ContaminatedMixt`' class objects.

having a number of components among those in `G`, a parsimonious model among those in `model`, and being contaminated or not as specified in `contamination`. Note that `getIC()` and `whichBest()` have an argument `criteria`, in substitution to `criterion`, which allows to select more than one criterion.

The package also includes some methods for '`ContaminatedMixt`' class objects; they are: `plot()` and `pairs()`, to display clustering/classification results in terms of scatter plots (in the cases $p = 2$ and $p \geq 2$, respectively), `summary()`, to visualize the estimated parameters and further inferential/clustering details, `print()`, to print at video the selected model(s) according to the information criteria in Table 2, `agree()` to evaluate the agreement of a given partition with respect to the partition arising from a fitted model, and `predict()` which provides the cluster prediction of observations based on a fitted uncontaminated/contaminated normal mixture model which is selected according to `getBestModel()`. As usual, further details can be found in the functions' help pages.

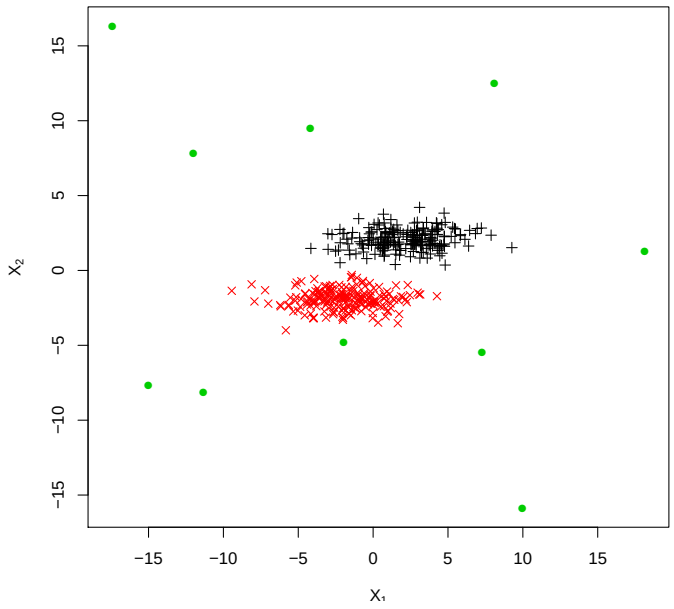

Figure 1: Scatterplot of the artificial data of Section 5.2. Bad points are represented by green bullets.

5.2. Artificial data

To illustrate the use of the package, we begin with an artificial data set from a mixture of $G = 2$ bivariate normal distributions, of equal size, with an EEI structure for the component covariance matrices. Ten bad points are also added from a uniform distribution over the range -20 to 20 on each variable. The data are generated by the following commands.

```
R> library("ContaminatedMixt")
R> library("mnormt")
R> p <- 2
R> set.seed(16)
R> n1 <- n2 <- 200
R> X1 <- rmnorm(n = n1, mean = rep(2, p), varcov = diag(c(5, 0.5)))
R> X2 <- rmnorm(n = n2, mean = rep(-2, p), varcov = diag(c(5, 0.5)))
R> bad <- matrix(runif(n = 20, min = -20, max = 20), nrow = 10, ncol = 2)
R> X <- rbind(X1, X2, bad)
```

The scatterplot of these data, in Figure 1, is obtained via the following commands.

```
R> group <- rep(c(1, 2, 3), times = c(n1, n2, 10))
R> plot(X, col = group, pch = c(3, 4, 16)[group], asp = 1,
+    xlab = expression(X[1]), ylab = expression(X[2]))
```

Model-based clustering

We start with a model-based clustering analysis by considering both contaminated and uncontaminated normal components, all the fourteen models in Table 1, and a number G of clusters

ranging from 1 to 3, resulting in 84 different models. The following command performs the fitting of the models and returns an object of class 'ContaminatedMixt'.

```
R> options(cl.cores = 4)
R> res1 <- CNmixt(X, G = 1:3, parallel = TRUE, seed = 2)

With G = 1, some models are equivalent, so only one model from each set of
equivalent models will be run.

 Using 4 cores


Best model according to AIC, AIC3, AICc, AICu is uncontaminated, with G = 3
group(s), and parsimonious structure VVI

Best model according to BIC, CAIC, AWE, ICL is contaminated, with G = 2
group(s), and parsimonious structure EEI
```

Because several models have to be fitted, parallel computation is convenient; it is set with the argument `parallel = TRUE`. The number of CPU cores used is printed at video and it is followed, after a few seconds, by a description of the best model according to each of the 8 criteria in Table 2. Here, we can note that some of the considered criteria, namely BIC, CAIC, AWE, and ICL agree in suggesting a contaminated model with $G = 2$ clusters and the true but unknown parsimonious structure EEI. To find out more about the model selected by the BIC, which is the most commonly used criterion in this context, we run the following command.

```
R> summary(res1)

----------------------------------
Best fitted model according to BIC
----------------------------------
 log.likelihood   n par     BIC
        -1699.2 410  11 -3464.7

Clustering table:
  1   2
205 205

Prior:  = 0.50032,  = 0.49968
Model: Contaminated EEI (diagonal, equal volume and shape) with 2 components

Variables
 Means:
    group 1 group 2
X.1  2.0953 -1.9185
X.2  2.0732 -1.9338
```

```
 Variance-covariance matrices:
  Component 1
       X.1    X.2
X.1 5.0545 0.0000
X.2 0.0000 0.4356
  Component 2
       X.1    X.2
X.1 5.0545 0.0000
X.2 0.0000 0.4356

 Alpha
[1] 0.97135 0.97326

 Eta
[1] 113.11 103.84
```

As we can note from the estimates $\widehat{\eta}_1 = 113.11$ and $\widehat{\eta}_2 = 103.84$, there is a large enough degree of contamination in the two clusters, which together contribute to capture the added bad points (see also the estimates $\widehat{\alpha}_1 = 0.97135$ and $\widehat{\alpha}_2 = 0.97326$). In order to evaluate the agreement of the obtained clustering with respect to the true one, we can adopt the `agree()` function, included in the package, in the following way.

```
R> agree(res1, givgroup = group)

        groups
givgroup   1   2 bad points
       1 200   0          0
       2   0 200          0
       3   0   0         10
```

The obtained classification is totally in agreement with the true one. A plot of the clustering results for the best BIC model is displayed with the following command (Figure 2).

```
R> plot(res1, contours = TRUE, asp = 1, xlab = expression(X[1]),
+    ylab = expression(X[2]))
```

Isodensities are also displayed (`contours = TRUE`). Clustering results in Figure 2 look analogous to those in Figure 1.

Model-based classification

On the same data, we can also suppose to know the cluster membership of some of the available observations and evaluate the classification of the remaining ones. Via the following commands we first randomly select twenty good observations to be considered as labeled, and then we fit the EEI model (`model = "EEI"`) with $G = 2$ clusters, in both its uncontaminated and contaminated version, assuming the groups membership of these observations as known in advance.

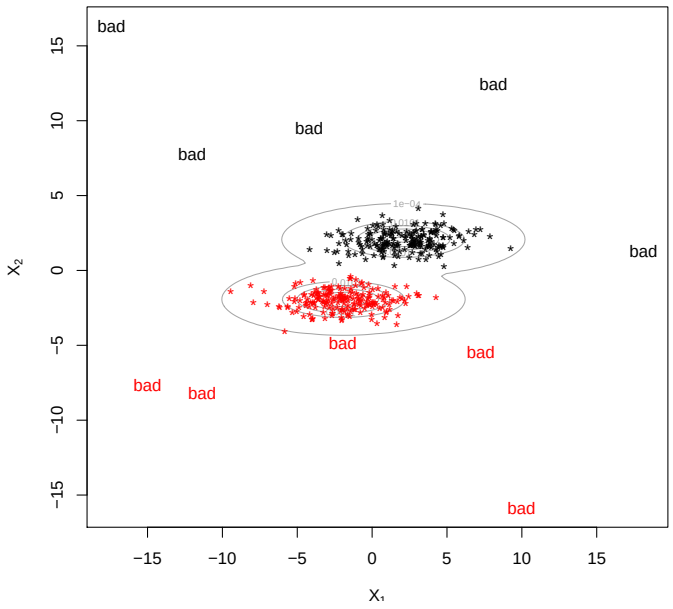

Figure 2: Clustering results from the model selected by the BIC on the artificial data of Section 5.2. Isodensities of the model are superimposed on the plot.

```
R> indlab <- sample(1:400, 20)
R> lab <- numeric(nrow(X))
R> lab[indlab] <- group[indlab]
R> res2 <- CNmixt(X, G = 2, model = "EEI", label = lab, AICcond = TRUE,
+    parallel = TRUE, seed = 2)

 Using 4 cores

Best model according to AIC, BIC, AIC3, AICc, AICu, CAIC, AWE, ICL, BEC,
AICcond is contaminated, with G = 2 group(s), and parsimonious structure EEI
```

The position of the labeled observations is contained in the object `indlab`, while their group membership is given in the object `lab`. Being a model-based classification analysis, we can add the AIC_{cond} (see Section 4.4) among the criteria considered to select the best model; this is done via the argument `AICcond = TRUE`, which implicitly activates the BEC too. From the results printed at video, we can note how all the considered criteria are in agreement in suggesting the uncontaminated version of the fitted model. To find out more about the selected model, we run the following command.

```
R> summary(res2, criterion = "AICcond")

--------------------------------------
Best fitted model according to AICcond
--------------------------------------
 log.likelihood   n par   AICcond
       -1699.3 410  11 -0.049517
```

```
Clustering table:
  1   2
205 205

Prior:  = 0.50025,  = 0.49975
Model: Contaminated EEI (diagonal, equal volume and shape) with 2 components

Variables
 Means:
    group 1 group 2
X.1  2.0955 -1.9185
X.2  2.0734 -1.9336

 Variance-covariance matrices:
  Component 1
       X.1     X.2
X.1 5.0548 0.00000
X.2 0.0000 0.43557
  Component 2
       X.1     X.2
X.1 5.0548 0.00000
X.2 0.0000 0.43557

 Alpha
[1] 0.97148 0.97325

 Eta
[1] 113.39 104.05
```

The agreement between the obtained classification and the true classification of the unlabeled observations only can be evaluated via the following command.

```
R> agree(res2, givgroup = group)

        groups
givgroup   1   2 bad points
       1 190   0          0
       2   0 190          0
       3   0   0         10
```

Naturally, the comparison is automatically focused on the unlabeled observations only. As we can see, the classification results are optimal in this case too.

5.3. The `wine` dataset

This second tutorial uses the `wine` data set included in the **ContaminatedMixt** package and available at the UCI Machine Learning Repository http://archive.ics.uci.edu/ml/

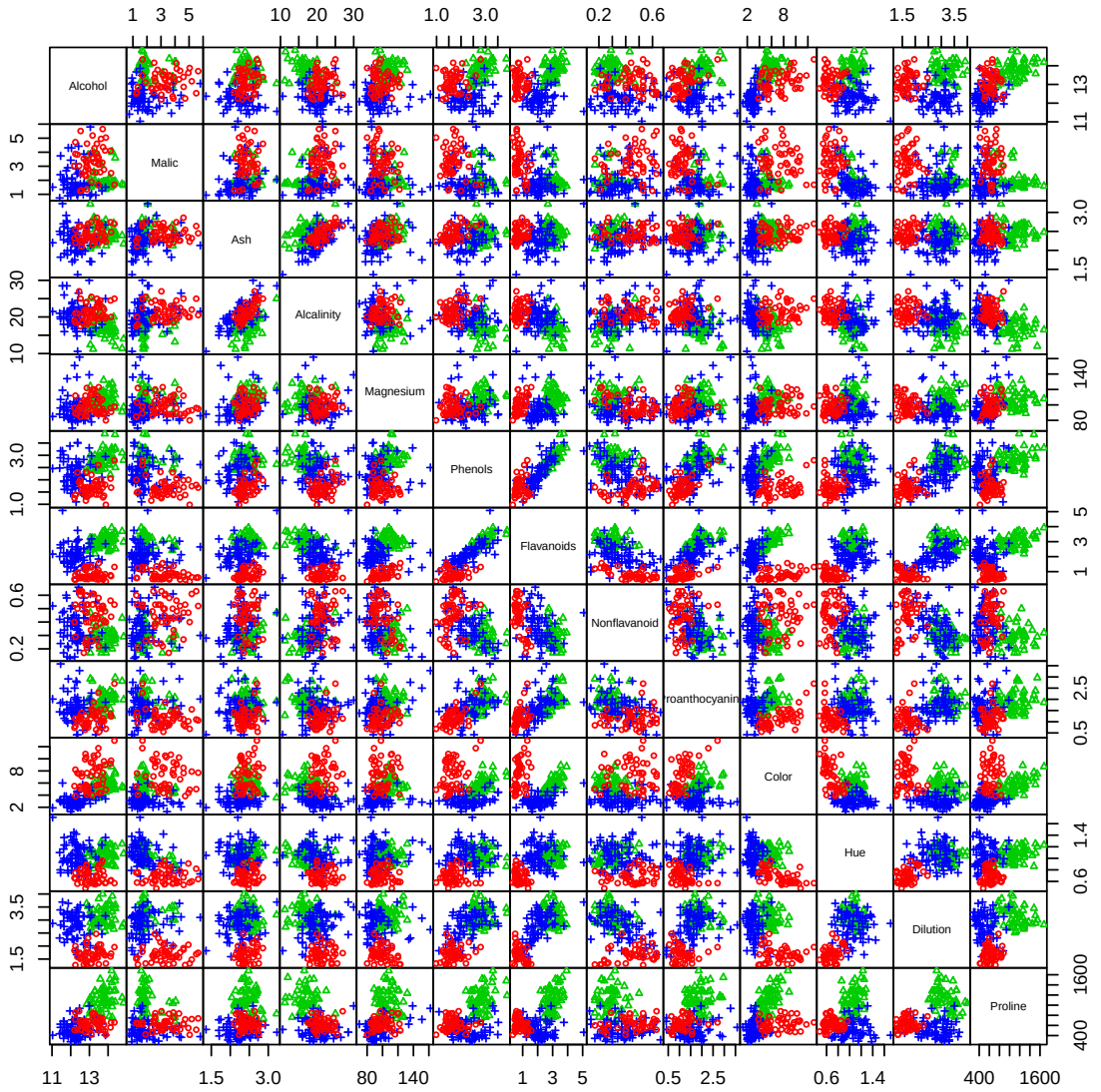

Figure 3: Wine data: Scatterplot matrix with clustering induced by the three cultivars.

datasets/Wine. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars (Barbera, Barolo, and Grignolino). The analysis determined the quantities of $p = 13$ constituents (continuous variables) found in each of the three types of wine. Data are loaded with:

```
R> data("wine", package = "ContaminatedMixt")
```

This command loads a data frame with the first column being a factor indicating the type of wine and the others containing the measurements about the 13 constituents. The plot of these data, displayed in Figure 3, is obtained by:

```
R> group <- wine[, 1]
R> pairs(wine[, -1], cex = 0.6, pch = c(1, 2, 3)[group],
+    col = c(2, 3, 4)[group], gap = 0, cex.labels = 0.6)
```

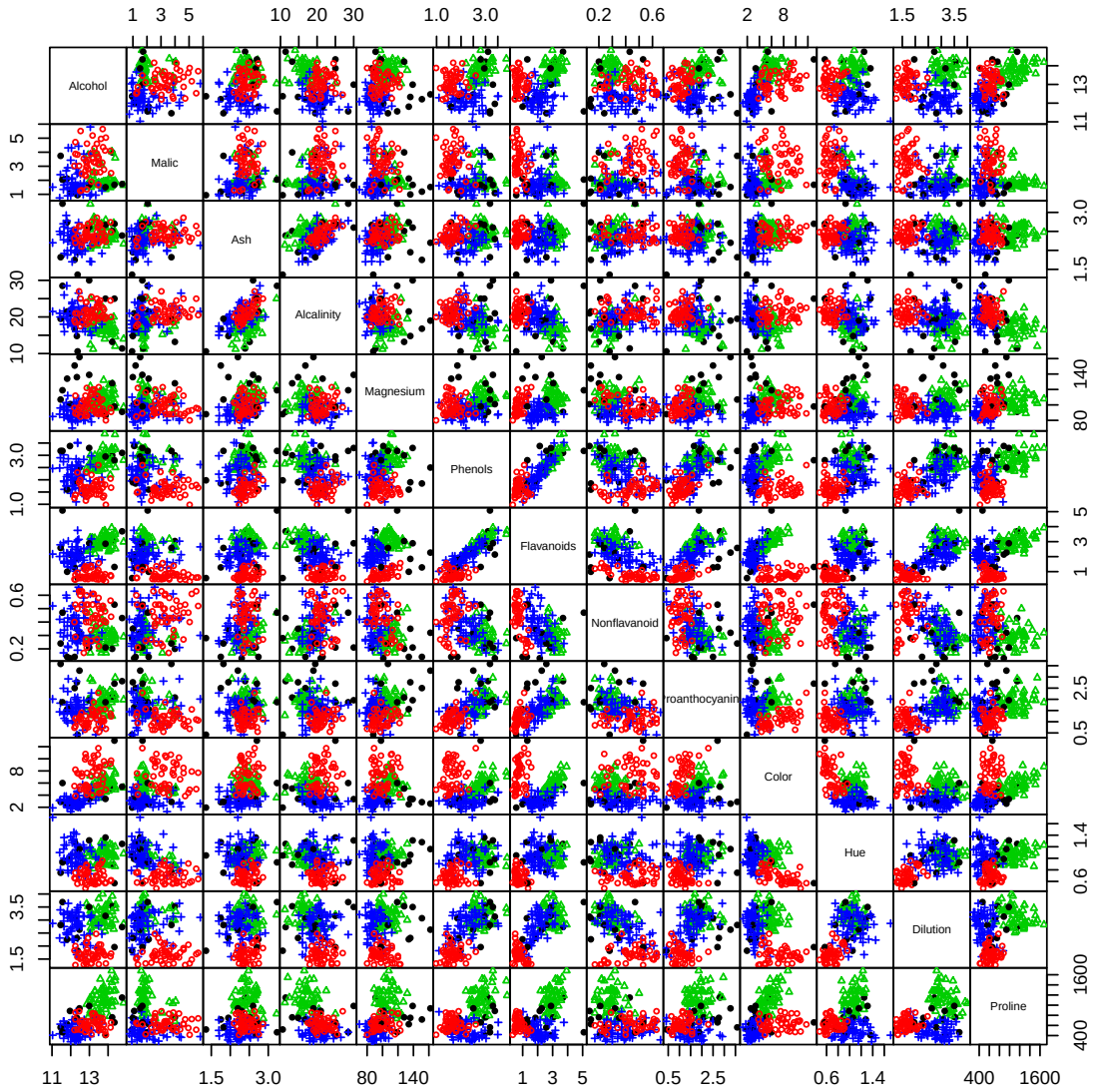

Figure 4: Wine data: Classification results from the model selected by BIC and ICL. Bad points are represented by black bullets.

The following command fits all the fourteen models, in their contaminated and uncontaminated version (`contamination = NULL`), for $G \in \{1, 2, 3, 4\}$, for a total of 112 models.

```
R> options(cl.cores = 12)
R> res3 <- CNmixt(wine[, -1], contamination = NULL, G = 1:4,
+    initialization = "random.post", seed = 138, parallel = TRUE)

With G = 1, some models are equivalent, so only one model from each set of
equivalent models will be run.

 Using 12 cores

Best model according to AIC is contaminated, with G = 3 group(s), and
```

```
parsimonious structure EVV

Best model according to BIC, AIC3, CAIC, ICL is contaminated, with G = 3
group(s), and parsimonious structure VVE

Best model according to AWE is contaminated, with G = 3 group(s), and
parsimonious structure EEI

Best model according to AICc, AICu is contaminated, with G = 3 group(s), and
parsimonious structure EVI
```

In this case, a random initialization of the posterior probabilities is used (`initialization = "random.post"`) with a pre-specified seed of random generation (`seed = 138`). The best model, for the most commonly used criteria BIC and ICL, is the VVE contaminated model with $G = 3$ clusters. The classification performance of this model can be seen via the following command.

```
R> agree(res3, givgroup = group)
```

```
           groups
givgroup     1  2  3 bad points
  Barbera   47  0  0          1
  Barolo     0 57  0          2
  Grignolino 0  0 61         10
```

As we can note, there are no misclassified wines; however, 13 wines are recognized as bad, 10 of which arise from the Grignolino cultivar. The graphical representation of the classification from the selected model can be obtained via the following command (see Figure 4).

```
R> pairs(res3, cex = 0.6, gap = 0, cex.labels = 0.6)
```

6. Conclusions

In this paper, we have introduced **ContaminatedMixt**, a package for the R software environment, specifically conceived for fitting and disseminating parsimonious mixtures of multivariate contaminated normal distributions. Although these models have been originally proposed for clustering applications (Punzo and McNicholas 2016), their use has been here extended to model-based classification, where information about the group membership of some of the observations is available. The package is also meant to be a user-friendly tool for an automatic detection of mild outliers (also referred to as bad points herein). Computation can take advantage of parallelization on multicore PCs and computer clusters, when a comparison among different models is needed. This is handy when, as it is often the case in practical applications, the number of clusters and/or the covariance structure of the model is not *a priori* known. We believe our package may be a practical tool supporting academics and practitioners who are involved in robust cluster/classification analysis applications.

References

Aitken AC (1926). "A Series Formula for the Roots of Algebraic and Transcendental Equations." *Proceedings of the Royal Society of Edinburgh*, **45**(1), 14–22. doi:10.1017/s0370164600024871.

Aitkin M, Wilson GT (1980). "Mixture Models, Outliers, and the EM Algorithm." *Technometrics*, **22**(3), 325–331. doi:10.1080/00401706.1980.10486163.

Akaike H (1973). "Information Theory and an Extension of Maximum Likelihood Principle." In BN Petrov, F Csaki (eds.), *Second International Symposium on Information Theory*, pp. 267–281. Akademiai Kiado, Budapest.

Andrews JL, McNicholas PD (2012). "Model-Based Clustering, Classification, and Discriminant Analysis via Mixtures of Multivariate t-Distributions." *Statistics and Computing*, **22**(5), 1021–1029. doi:10.1007/s11222-011-9272-x.

Andrews JL, McNicholas PD, Subedi S (2011). "Model-Based Classification via Mixtures of Multivariate t-Distributions." *Computational Statistics & Data Analysis*, **55**(1), 520–529. doi:10.1016/j.csda.2010.05.019.

Andrews JL, Wickins JR, Boers NM, McNicholas PD (2018). "**teigen**: An R Package for Model-Based Clustering and Classification via the Multivariate t Distribution." *Journal of Statistical Software*, **83**(7), 1–32. doi:10.18637/jss.v083.i07.

Bagnato L, Punzo A (2013). "Finite Mixtures of Unimodal Beta and Gamma Densities and the k-Bumps Algorithm." *Computational Statistics*, **28**(4), 1571–1597. doi:10.1007/s00180-012-0367-4.

Bagnato L, Punzo A, Zoia MG (2017). "The Multivariate Leptokurtic-Normal Distribution and Its Application in Model-Based Clustering." *The Canadian Journal of Statistics*, **45**(1), 95–119. doi:10.1002/cjs.11308.

Banfield JD, Raftery AE (1993). "Model-Based Gaussian and Non-Gaussian Clustering." *Biometrics*, **49**(3), 803–821. doi:10.2307/2532201.

Barnett V, Lewis T (1994). *Outliers in Statistical Data*. Wiley Series in Probability & Statistics. John Wiley & Sons.

Becker C, Gather U (1999). "The Masking Breakdown Point of Multivariate Outlier Identification Rules." *Journal of the American Statistical Association*, **94**(447), 947–955. doi:10.1080/01621459.1999.10474199.

Berkane M, Bentler PM (1988). "Estimation of Contamination Parameters and Identification of Outliers in Multivariate Data." *Sociological Methods & Research*, **17**(1), 55–64. doi:10.1177/0049124188017001003.

Biecek P, Szczurek E, Vingron M, Tiuryn J (2012). "The R Package **bgmm**: Mixture Modeling with Uncertain Knowledge." *Journal of Statistical Software*, **47**(3), 1–31. doi:10.18637/jss.v047.i03.

Biernacki C, Celeux G, Govaert G (2000). "Assessing a Mixture Model for Clustering with the Integrated Completed Likelihood." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(7), 719–725. doi:10.1109/34.865189.

Biernacki C, Celeux G, Govaert G (2003). "Choosing Starting Values for the EM Algorithm for Getting the Highest Likelihood in Multivariate Gaussian Mixture Models." *Computational Statistics & Data Analysis*, **41**(3–4), 561–575. doi:10.1016/s0167-9473(02)00163-9.

Bock HH (2002). "Clustering Methods: From Classical Models to New Approaches." *Statistics in Transition*, **5**(5), 725–758.

Böhning D (2000). *Computer-Assisted Analysis of Mixtures and Applications: Meta-Analysis, Disease Mapping and Others*, volume 81 of *Monographs on Statistics and Applied Probability*. Chapman & Hall/CRC, London.

Bouchard G, Celeux G (2006). "Selection of Generative Models in Classification." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(4), 544–554. doi:10.1109/tpami.2006.82.

Bozdogan H (1987). "Model Selection and Akaike's Information Criterion (AIC): The General Theory and Its Analytical Extensions." *Psycometrika*, **52**(3), 345–370. doi:10.1007/bf02294361.

Bozdogan H (1994). "Theory & Methodology of Time Series Analysis." In *Proceedings of the First US/Japan Conference on the Frontiers of Statistical Modeling: An Informational Approach*, volume 1. Kluwer Academic Publishers, Dordrecht.

Browne RP, ElSherbiny A, McNicholas PD (2018). ***mixture**: Mixture Models for Clustering and Classification*. R package version 1.5, URL https://CRAN.R-project.org/package=mixture.

Browne RP, McNicholas PD (2012). "Model-Based Clustering, Classification, and Discriminant Analysis of Data with Mixed Type." *Journal of Statistical Planning and Inference*, **142**(11), 2976–2984. doi:10.1016/j.jspi.2012.05.001.

Celeux G, Govaert G (1995). "Gaussian Parsimonious Clustering Models." *Pattern Recognition*, **28**(5), 781–793. doi:10.1016/0031-3203(94)00125-6.

Chapelle O, Schölkopf B, Zien A (2010). *Semi-Supervised Learning*. Adaptive Computation and Machine Learning. MIT Press. doi:10.7551/mitpress/9780262033589.001.0001.

Dang UJ, Browne RP, McNicholas PD (2015). "Mixtures of Multivariate Power Exponential Distributions." *Biometrics*, **71**(4), 1081–1089. doi:10.1111/biom.12351.

Dang UJ, Punzo A, McNicholas PD, Ingrassia S, Browne RP (2017). "Multivariate Response and Parsimony for Gaussian Cluster-Weighted Models." *Journal of Classification*, **34**(1), 4–34. doi:10.1007/s00357-017-9221-2.

Davies L, Gather U (1993). "The Identification of Multiple Outliers." *Journal of the American Statistical Association*, **88**(423), 782–792. doi:10.2307/2290763.

Dean N, Murphy TB, Downey G (2006). "Using Unlabelled Data to Update Classification Rules with Applications in Food Authenticity Studies." *Journal of the Royal Statistical Society C*, **55**(1), 1–14. doi:10.1111/j.1467-9876.2005.00526.x.

Dempster AP, Laird NM, Rubin DB (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm." *Journal of the Royal Statistical Society B*, **39**(1), 1–38.

Fraley C, Raftery AE (1998). "How Many Clusters? Which Clustering Method? Answers via Model-Based Cluster Analysis." *Computer Journal*, **41**(8), 578–588. doi:10.1093/comjnl/41.8.578.

Fraley C, Raftery AE (2007). "Model-Based Methods of Classification: Using the **mclust** Software in Chemometrics." *Journal of Statistical Software*, **18**(6), 1–13. doi:10.18637/jss.v018.i06.

Fraley C, Raftery AE, Scrucca L, Murphy TB, Fop M (2017). ***mclust**: Normal Mixture Modelling for Model-Based Clustering, Classification, and Density Estimation*. R package version 5.4, URL https://CRAN.R-project.org/package=mclust.

Fritz H, García-Escudero LA, Mayo-Iscar A (2012). "**tclust**: An R Package for a Trimming Approach to Cluster Analysis." *Journal of Statistical Software*, **47**(12), 1–26. doi:10.18637/jss.v047.i12.

Gallegos MT, Ritter G (2009). "Trimmed ML Estimation of Contaminated Mixtures." *Sankhyā: The Indian Journal of Statistics A*, **71**(2), 164–220.

Gómez-Villegas MA, Gómez-Sánchez-Manzano E, Maín P, Navarro H (2011). "The Effect of Non-Normality in the Power Exponential Distributions." In L Pardo, N Balakrishnan, MA Gil (eds.), *Modern Mathematical Tools and Techniques in Capturing Complexity*, Understanding Complex Systems, pp. 119–129. Springer-Verlag, Berlin.

Hathaway RJ (1986). "A Constrained EM Algorithm for Univariate Normal Mixtures." *Journal of Statistical Computation and Simulation*, **23**(3), 211–230. doi:10.1080/00949658608810872.

Hennig C (2002). "Fixed Point Clusters for Linear Regression: Computation and Comparison." *Journal of Classification*, **19**(2), 249–276. doi:10.1007/s00357-001-0045-7.

Hurvich CM, Tsai CL (1989). "Regression and Time Series Model Selection in Small Samples." *Biometrika*, **76**(2), 297–307. doi:10.1093/biomet/76.2.297.

Ingrassia S (2004). "A Likelihood-Based Constrained Algorithm for Multivariate Normal Mixture Models." *Statistical Methods and Applications*, **13**(2), 151–166. doi:10.1007/s10260-004-0092-4.

Karlis D, Xekalaki E (2003). "Choosing Initial Values for the EM Algorithm for Finite Mixtures." *Computational Statistics & Data Analysis*, **41**(3–4), 577–590. doi:10.1016/s0167-9473(02)00177-9.

Lange KL, Little RJA, Taylor JMG (1989). "Robust Statistical Modeling Using the t Distribution." *Journal of the American Statistical Association*, **84**(408), 881–896. doi:10.1080/01621459.1989.10478852.

Lebret R, Iovleff S, Langrognet F, Biernacki C, Celeux G, Govaert G (2015). "**Rmixmod**: The R Package of the Model-Based Unsupervised, Supervised, and Semi-Supervised Classification Mixmod Library." *Journal of Statistical Software*, **67**(6), 1–29. doi:10.18637/jss.v067.i06.

Leisch F, Grün B (2018). *CRAN Task View: Cluster Analysis & Finite Mixture Models*. Version 2018-03-09, URL https://CRAN.R-project.org/view=Cluster.

Lindsay BG (1995). *Mixture Models: Theory, Geometry and Applications*, volume 5. NSF-CBMS Regional Conference Series in Probability and Statistics, Institute of Mathematical Statistics, Hayward, California.

Little RJA (1988). "Robust Estimation of the Mean and Covariance Matrix from Data with Missing Values." *Journal of the Royal Statistical Society C*, **37**(1), 23–38. doi:10.2307/2347491.

Maruotti A, Punzo A (2017). "Model-Based Time-Varying Clustering of Multivariate Longitudinal Data with Covariates and Outliers." *Computational Statistics & Data Analysis*, **113**, 475–496. doi:10.1016/j.csda.2016.05.024.

McLachlan GJ, Basford KE (1988). *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York.

McLachlan GJ, Peel D (1998). "Robust Cluster Analysis via Mixtures of Multivariate t-Distributions." In A Amin, D Dori, P Pudil, H Freeman (eds.), *Advances in Pattern Recognition*, volume 1451 of *Lecture Notes in Computer Science*, pp. 658–666. Springer-Verlag, Berlin.

McNicholas PD (2010). "Model-Based Classification Using Latent Gaussian Mixture Models." *Journal of Statistical Planning and Inference*, **140**(5), 1175–1181. doi:10.1016/j.jspi.2009.11.006.

McNicholas PD (2016). *Mixture Model-Based Classification*. Chapman & Hall/CRC, Boca Raton.

McNicholas PD, ElSherbiny A, McDaid AF, Murphy TB (2018). ***pgmm**: Parsimonious Gaussian Mixture Models*. R package version 1.2.2, URL https://CRAN.R-project.org/package=pgmm.

McQuarrie A, Shumway R, Tsai CL (1997). "The Model Selection Criterion AICu." *Statistics & Probability Letters*, **34**(3), 285–292. doi:10.1016/s0167-7152(96)00192-7.

Meng XL, Rubin DB (1993). "Maximum Likelihood Estimation via the ECM Algorithm: A General Framework." *Biometrika*, **80**(2), 267–278. doi:10.1093/biomet/80.2.267.

Peel D, McLachlan GJ (2000). "Robust Mixture Modelling Using the t Distribution." *Statistics and Computing*, **10**(4), 339–348. doi:10.1023/a:1008981510081.

Punzo A (2014). "Flexible Mixture Modeling with the Polynomial Gaussian Cluster-Weighted Model." *Statistical Modelling*, **14**(3), 257–291. doi:10.1177/1471082x13503455.

Punzo A, Blostein M, McNicholas PD (2017). "High-Dimensional Clustering with the Contaminated Gaussian Distribution." arXiv:1408.2128v2 [stat.ME], URL https://arxiv.org/abs/1408.2128v2.

Punzo A, Browne RP, McNicholas PD (2016). "Hypothesis Testing for Mixture Model Selection." *Journal of Statistical Computation and Simulation*, **86**(14), 2797–2818. doi:10.1080/00949655.2015.1131282.

Punzo A, Maruotti A (2016). "Clustering Multivariate Longitudinal Observations: The Contaminated Gaussian Hidden Markov Model." *Journal of Computational and Graphical Statistics*, **25**(4), 1097–1116. doi:10.1080/10618600.2015.1089776.

Punzo A, Mazza A, McNicholas PD (2018). ***ContaminatedMixt**: Model-Based Clustering and Classification with the Multivariate Contaminated Normal Distribution*. R package version 1.3.3, URL https://CRAN.R-project.org/package=ContaminatedMixt.

Punzo A, McNicholas PD (2016). "Parsimonious Mixtures of Multivariate Contaminated Normal Distributions." *Biometrical Journal*, **58**(6), 1506–1537. doi:10.1002/bimj.201500144.

Punzo A, McNicholas PD (2017). "Robust Clustering in Regression Analysis via the Contaminated Gaussian Cluster-Weighted Model." *Journal of Classification*, **34**(2), 249–293. doi:10.1007/s00357-017-9234-x.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Ritter G (2015). *Robust Cluster Analysis and Variable Selection*. Chapman & Hall/CRC, Boca Raton.

Schwarz G (1978). "Estimating the Dimension of a Model." *The Annals of Statistics*, **6**(2), 461–464. doi:10.1214/aos/1176344136.

Subedi S, Punzo A, Ingrassia S, McNicholas PD (2013). "Clustering and Classification via Cluster-Weighted Factor Analyzers." *Advances in Data Analysis and Classification*, **7**(1), 5–40. doi:10.1007/s11634-013-0124-8.

Subedi S, Punzo A, Ingrassia S, McNicholas PD (2015). "Cluster-Weighted t-Factor Analyzers for Robust Model-Based Clustering and Dimension Reduction." *Statistical Methods & Applications*, **24**(4), 623–649. doi:10.1007/s10260-015-0298-7.

Tukey JW (1960). "A Survey of Sampling from Contaminated Distributions." In I Olkin (ed.), *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, Stanford Studies in Mathematics and Statistics, chapter 39, pp. 448–485. Stanford University Press, California.

Vandewalle V, Biernacki C, Celeux G, Govaert G (2013). "A Predictive Deviance Criterion for Selecting a Generative Model in Semi-Supervised Classification." *Computational Statistics & Data Analysis*, **64**, 220–236. doi:10.1016/j.csda.2013.02.010.

Zhang J, Liang F (2010). "Robust Clustering Using Exponential Power Mixtures." *Biometrics*, **66**(4), 1078–1086. doi:10.1111/j.1541-0420.2010.01389.x.

Affiliation:

Antonio Punzo
Department of Economics and Business
University of Catania
Corso Italia, 55, 95129 Catania, Italy
Telephone: +39/095/7537640
E-mail: antonio.punzo@unict.it
URL: http://www.economia.unict.it/punzo

Angelo Mazza
Department of Economics and Business
University of Catania
Corso Italia, 55, 95129 Catania, Italy
Telephone: +39/095/7537736
E-mail: a.mazza@unict.it
URL: http://docenti.unict.it/a.mazza

Paul D. McNicholas
Department of Mathematics & Statistics
McMaster University
Hamilton, Ontario, L8S 4L8, Canada
Telephone: +1-905-525-9140, ext. 23419
E-mail: mcnicholas@math.mcmaster.ca
URL: http://ms.mcmaster.ca/~paul/

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
July 2018, Volume 85, Issue 10 *Submitted:* 2015-12-21
doi:10.18637/jss.v085.i10 *Accepted:* 2017-05-30