Available online at www.sciencedirect.com

ScienceDirect

Computational Statistics & Data Analysis 00 (2016) 1–21

Journal Logo

The joint role of trimming and constraints in robust estimation for mixtures of Gaussian factor analyzers

Luis Angel García-Escudero[a], Alfonso Gordaliza[a], Francesca Greselin[b], Salvatore Ingrassia[c], Agustín Mayo-Iscar[a]

[a]*Department of Statistics and Operations Research and IMUVA, University of Valladolid (Spain)*
[b]*Department of Statistics and Quantitative Methods, Milano-Bicocca University (Italy)*
[c]*Department of Economics and Business, University of Catania (Italy)*

Abstract

Mixtures of Gaussian factors are powerful tools for modeling an unobserved heterogeneous population, offering - at the same time - dimension reduction and model-based clustering. The high prevalence of spurious solutions and the disturbing effects of outlying observations in maximum likelihood estimation may cause biased or misleading inferences. Restrictions for the component covariances are considered in order to avoid spurious solutions, and trimming is also adopted, to provide robustness against violations of the normality assumptions of the underlying latent factors. A detailed AECM algorithm for this new approach is presented. Simulation results and an application to the AIS dataset show the aim and effectiveness of the proposed methodology.

Keywords: Constrained estimation, Factor Analyzers Modeling, Mixture Models, Model-Based Clustering,, Robust estimation.

1. Introduction and motivation

Factor analysis is an effective method of summarizing the variability between a number of correlated features, through a much smaller number of unobservable, hence named *latent*, factors. It originated from the consideration that, in many phenomena, several observed variables could be explained by a few unobserved ones. Under this approach, each single variable (among the p observed ones) is assumed to be a linear combination of d underlying common factors with an accompanying error term to account for that part of the variability which is unique to it (not in common with other variables). Ideally, d should be substantially smaller than p, to achieve parsimony.

Clearly, the effectiveness of this method is limited by its global linearity, as happens for principal components analysis. Hence, Ghahramani and Hilton (1997), Tipping and Bishop (1999) and McLachlan and Peel (2000a) solidly widened the applicability of these approaches by combining local models of Gaussian factors in the form of finite mixtures. The idea is to employ latent variables to perform dimensional reduction in each component, thus providing a statistical method which concurrently performs clustering and, within each cluster, local dimensionality reduction.

In the literature, error and factors are routinely assumed to have a Gaussian distribution because of their mathematical and computational tractability: however, statistical methods which ignore departure from normality may cause

Email addresses: lagarcia@eio.uva.es (Luis Angel García-Escudero), alfonsog@eio.uva.es (Alfonso Gordaliza), francesca.greselin@unimib.it (Francesca Greselin), s.ingrassia@unict.it (
Salvatore Ingrassia), agustin@med.uva.es (Agustín Mayo-Iscar)

biased or misleading inferences. Moreover, it is well known that maximum likelihood estimation for mixtures often leads to ill-posed problems because of the unboundedness of the objective function to be maximized, which favors the appearance of non-interesting local maximizers and degenerate or *spurious* solutions.

The lack of robustness in mixture fitting arises whenever the sample contains a certain proportion of data that does not follow the underlying population model. Spurious solutions can even appear when ML estimation is applied to artificial data drawn from a given finite mixture model, i.e. without adding any kind of contamination. Hence, robust estimation is needed. Many contributions in this sense can be found in the literature: from the Mclust model with a noise component in Fraley and Raftery (1998), mixtures of t-distributions in McLachlan and Peel (2000), the trimmed likelihood mixture fitting method in Neykov et al. (2007), the trimmed ML estimation of contaminated mixtures in Gallegos and Ritter (2009), and the robust improper ML estimator introduced in Coretto and Hennig (2011), among many others. Some important applications in such fields as computer vision, pattern recognition, analysis of microarray gene expression data, or tomography suggest that more attention should be paid to robustness, because noise in the data sets may be frequent in all these fields of application.

Different types of constraints have been traditionally applied in Gaussian mixtures of factor analyzers, for instance, some authors propose taking a common (diagonal) error matrix (as for the Mixtures of Common Factor Analyzers, denoted by MCFA, in Baek et al., 2010) or imposing an isotropic error matrix (Bishop and Tipping, 1998). This strategy has proven to be effective in many cases, at the expenses of stronger distributional restrictions on the data. To avoid singularities and spurious solutions, under milder conditions, Greselin and Ingrassia (2015) recently proposed maximizing the likelihood by constraining the eigenvalues of the covariance matrices, following the previous work of Ingrassia (2004) and going back to Hathaway (1985). Furthermore, mixtures of t-analyzers have been considered (see McLachlan and Bean, 2005; Lin et al., 2014, and references therein) in an attempt to make the model less sensitive to outliers, but they, too, are not robust against very extreme outliers (Hennig, 2004).

The purpose of the present work is to introduce an estimating procedure for the mixture of Gaussian factor analyzers that can resist the effect of outliers and avoid spurious local maximizers. The proposed constraints can also be used to take into account prior information about the scatter parameters.

Trimming has been shown to be a simple, powerful, flexible and computationally feasible way to provide robustness in many different statistical frameworks. The basic idea behind trimming here is the removal of a small proportion α of observations whose values would be the most unlikely to occur if the fitted model were true. In this way, trimming avoids the problem of a small fraction of outlying observations exerting a harmful effect on the estimation. Incorporating constraints into the mixture fitting estimation method moves the mathematical problem to a well-posed setting and hence minimizes the risk of incurring spurious solutions. Moreover, a correct statement of the problem allows the desired statistical properties for the estimators to be obtained, such as the existence and consistency results, as in García-Escudero et al. (2008).

The rest of the paper has been organized as follows. In Section 2, the notation is introduced and the main ideas about Gaussian Mixtures of Factor Analyzers (hereafter denoted by MFA) are summarized. Then, in Section 3, the trimmed likelihood for MFA is presented, and fairly extensive notes are provided concerning the EM algorithm, with incorporated trimming and constrained estimation. In Section 4, the performance of the new procedure is discussed, on the grounds of some numerical results obtained from simulated and real data. In particular, the bias and MSE of robustly estimated model parameters for different cases of data contamination, are compared, using Monte Carlo experiments. The application to the Australian Institute of Sports dataset shows how classification and factor analysis can be developed using the new model. Section 5 contains concluding notes and provides ideas for further research.

2. Gaussian Mixtures of Factor Analyzers

The density of the p-dimensional random variable $\mathbf{X}$ of interest is modeled as a mixture of G multivariate normal densities in some unknown proportions $\pi_1, \dots \pi_G$, whenever each data point is taken to be a realization of the following density function:

$$f(\mathbf{x}; \boldsymbol{\theta}) = \sum_{g=1}^{G} \pi_g \phi_p(\mathbf{x}; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) \tag{1}$$

where $\phi_p(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the p-variate normal density function with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Here, the vector $\boldsymbol{\theta} = \boldsymbol{\theta}_{GM}(p, G)$ of unknown parameters consists of the $(G-1)$ mixing proportions π_g, the Gp elements of the component means $\boldsymbol{\mu}_g$, and the $\frac{1}{2}Gp(p+1)$ distinct elements of the component-covariance matrices $\boldsymbol{\Sigma}_g$. MFA postulates a finite mixture of linear sub-models for the distribution of the full observation vector $\mathbf{X}$, given the (unobservable) factors $\mathbf{U}$. That is, MFA provides local dimensionality reduction by assuming that the distribution of the observation $\mathbf{X}_i$ can be given as

$$\mathbf{X}_i = \boldsymbol{\mu}_g + \boldsymbol{\Lambda}_g \mathbf{U}_{ig} + \mathbf{e}_{ig} \quad \text{with probability} \quad \pi_g \; (g = 1, \ldots, G) \quad \text{for } i = 1, \ldots, n, \tag{2}$$

where $\boldsymbol{\Lambda}_g$ is a $p \times d$ matrix of *factor loadings*, the *factors* $\mathbf{U}_{1g}, \ldots, \mathbf{U}_{ng}$ are $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ distributed independently of the *errors* $\mathbf{e}_{ig}$. The latter are independently $\mathcal{N}(\mathbf{0}, \boldsymbol{\Psi}_g)$ distributed, and $\boldsymbol{\Psi}_g$ is a $p \times p$ diagonal matrix ($g = 1, \ldots, G$). The diagonality of $\boldsymbol{\Psi}_g$ is one of the key assumptions of factor analysis: the observed variables are independent given the factors. Note that the factor variables $\mathbf{U}_{ig}$ model correlations between the elements of $\mathbf{X}_i$, while the errors $\mathbf{e}_{ig}$ account for independent noise for $\mathbf{X}_i$. We suppose that $d < p$, which means that d unobservable factors are jointly explaining the p observable features of the statistical units. Under these assumptions, the mixture of factor analyzers model is given by (1), where the g-th component-covariance matrix $\boldsymbol{\Sigma}_g$ has the form

$$\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}_g \quad (g = 1, \ldots, G). \tag{3}$$

The parameter vector $\boldsymbol{\theta} = \boldsymbol{\theta}_{MFA}(p, d, G)$ now consists of the elements of the component means $\boldsymbol{\mu}_g$, the $\boldsymbol{\Lambda}_g$, and the $\boldsymbol{\Psi}_g$, along with the mixing proportions π_g ($g = 1, \ldots, G - 1$), on putting $\pi_G = 1 - \sum_{i=1}^{G-1} \pi_g$.

Note that, in the case of $d > 1$, there is an infinity of choices for $\boldsymbol{\Lambda}_g$, since model (2) is still satisfied if we replace $\boldsymbol{\Lambda}_g$ by $\boldsymbol{\Lambda}_g \mathbf{H}'$, where $\mathbf{H}$ is any orthogonal matrix of order d. As $d(d-1)/2$ constraints are needed for $\boldsymbol{\Lambda}_g$ to be uniquely defined, the number of free parameters for each component of the mixture is given by

$$pd + p - \frac{1}{2}d(d-1).$$

The following condition on p and d assures the desired parsimony:

$$[(p-d)^2 - (p+d)] > 0.$$

3. Robust Mixtures of Factor Analyzers

In this section, the *trimmed (Gaussian) mixtures of factor analyzers model* (trimmed MFA) is presented and a feasible algorithm for its implementation is provided.

3.1. Problem statement

Let $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a given data set in $\mathbb{R}^p$. With the theoretical underlying model described in Section 3 in mind, a mixture of Gaussian factor components can be robustly fitted to this dataset $\mathbf{x}$ by maximizing a *trimmed mixture log-likelihood* (see Neykov et al. 2007, Gallegos and Ritter 2009 and García-Escudero et al. 2014) defined as:

$$\mathcal{L}_{trim} = \sum_{i=1}^{n} \zeta(\mathbf{x}_i) \log \left[\sum_{g=1}^{G} \phi_p(\mathbf{x}_i; \mu_g, \boldsymbol{\Sigma}_g) \pi_g \right] \tag{4}$$

where $\zeta(\cdot)$ is a 0-1 trimming indicator function that tells us whether observation $\mathbf{x}_i$ is trimmed off: $\zeta(\mathbf{x}_i)$=0, or not: $\zeta(\mathbf{x}_i)$=1 and $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g^{'} + \boldsymbol{\Psi}_g$ as in (3). A fixed fraction α of observations can be unassigned by setting $\sum_{i=1}^{n} \zeta(\mathbf{x}_i) = [n(1-\alpha)]$ and, hence, the parameter α denotes the trimming level.

Moreover, to avoid the unboundedness of $\mathcal{L}_{trim}$, *constrained maximization* of (4) is introduced. In more detail, with reference to the diagonal elements $\{\psi_{gk}\}_{k=1,\ldots,p}$ of the noise matrices $\boldsymbol{\Psi}_g$ for $g = 1, \ldots, G$, it is required that

$$\psi_{g_1 k} \leq c_{noise} \;\; \psi_{g_2 h} \qquad \text{for every } 1 \leq k \neq h \leq p \text{ and } 1 \leq g_1 \neq g_2 \leq G \tag{5}$$

The constant c_{noise} is finite and such that $c_{noise} \geq 1$, to avoid the $|\Sigma_g| \to 0$ case. This constraint can be seen as an adaptation to MFA of those introduced in Ingrassia and Rocci (2007), García-Escudero et al. (2008), and is similar to the mild restrictions implemented for MFA in Greselin and Ingrassia (2015). They all go back to the seminal paper of Hathaway (1985). We will look for the maximization of $\mathcal{L}_{trim}$ on $\mathbf{\Psi}_g$ under the given constraints: this setting leads to a well-defined maximization problem, and at the same time allows singularities to be discarded and the occurrence of spurious solutions to be reduced.

Our methodology also includes the possibility of controlling the relative variability of the norms of the p dimensional column vectors of the matrices $\mathbf{\Lambda}_g$ (for $g = 1, \ldots, G$). If $\{\eta_{k\,g}\}_{k=1,...,d}$ denotes the set of these norms, a second set of constraints applies on their values

$$\eta_{k\,g_1} \leq c_{load}\ \ \eta_{h\,g_2} \quad \text{for every} \quad 1 \leq k \neq h \leq d \quad \text{and} \quad 1 \leq g_1 \neq g_2 \leq G. \tag{6}$$

In fact, these types of constraints are not needed to avoid singularities in the target function, but they could be useful to achieve more sensible solutions.

Hereafter, Θ_c will denote the constrained parameter space for $\boldsymbol{\theta} = \{\pi_g, \mu_g, \Psi_g, \Lambda_g; g = 1, \ldots, G\}$ under the requirements (5) and (6).

3.2. *Algorithm*

The maximization of $\mathcal{L}_{trim}$ in (4) for $\theta \in \Theta_c$ is not an easy task, obviously. We will give a feasible algorithm obtained by combining the Alternating Expectation-Conditional Maximization algorithm (AECM) for MFA with that (with trimming and constraints) introduced in García-Escudero et al. (2014) (see, also, Fritz et al., 2013).

As usual in the EM framework, each observation $\mathbf{x}_i$ is associated with an unobserved state $\mathbf{z}_i = (z_{i1}, \ldots, z_{iG})'$ for $i = 1, \ldots, n$ where z_{ig} is one or zero, depending on whether $\mathbf{x}_i$ does or does not belong to the g-th component. The component label vectors $\mathbf{z}_1, \ldots, \mathbf{z}_n$ are taken to be the realized values of the random vectors $\mathbf{Z}_1, \ldots, \mathbf{Z}_n$, where, for independent feature data, it is appropriate to assume that they are (unconditionally) multinomially distributed. i.e. $\mathbf{Z}_1, \ldots, \mathbf{Z}_n \sim^{i.i.d.} Mult_G(1; \pi_1, ..., \pi_G)$. The AECM is an extension of the EM, suggested by the factor structure of the model, which uses different specifications of missing data at each stage. The idea is to partition the vector of parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ in such a way that $\mathcal{L}_{trim}$ is easy to be maximized for $\boldsymbol{\theta}_1$ given $\boldsymbol{\theta}_2$ and viceversa, replacing the M-step by a number of computationally simpler conditional maximization (CM) steps. In more detail, in the first cycle we set $\boldsymbol{\theta}_1 = \{\pi_g, \boldsymbol{\mu}_g; g = 1, \ldots, G\}$ and the missing data are the unobserved group labels $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)'$; while in the second cycle we set $\boldsymbol{\theta}_2 = \{\mathbf{\Lambda}_g, \mathbf{\Psi}_g; g = 1, \ldots, G\}$ and the missing data are the group labels $\mathbf{z}$ and the unobserved latent factors $\mathbf{u} = (\mathbf{u}_{11}, \ldots, \mathbf{u}_{nG})'$. Hence, the application of the AECM algorithm consists of two cycles, and there is one E-step and one CM-step, alternatively considering $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ in each cycle. A trimming step, to evaluate the trimming function, precedes each cycle. The trimming function has the role of discarding the $\alpha 100\%$ of observations with lowest contribution to the likelihood. Before describing the algorithm, we remark that the unobserved group labels $\mathbf{Z}$ are considered missing data in both cycles. Therefore, during the l-th iteration, $z_{ig}^{(l+1/2)}$ and $z_{ig}^{(l+1)}$ denote the conditional expectations at the first and second cycle, respectively.

The algorithm has to be run multiple times on the same dataset, with different starting values, to prevent the attainment of a local, rather than global, maximum log-likelihood. In each run it executes the following steps:

1 *Initialization:*
Each iteration begins by selecting initial values for $\theta^{(0)}$ where $\theta^{(0)} = (\pi_g^{(0)}, \mu_g^{(0)}, \mathbf{\Lambda}_g^{(0)}, \mathbf{\Psi}_g^{(0)}; g = 1, \ldots, G)$. Inspired from results obtained in a series of extensive test experiments about initialization strategies (see Maitra, 2009), and aiming to allow the algorithm to visit the entire parameter space, $p+1$ units are randomly selected (without replacement) for group g from the observed data $\mathbf{x}$. In this way, a subsample $\mathbf{x}^g$ is obtained that may be arranged in a $(p+1)\times p$ matrix, and its sample mean will be the initial $\mu_g^{(0)}$. Additionally, based on these $p+1$ observations, a new *ad hoc* approach for providing an initialization procedure for $\mathbf{\Psi}_g^{(0)}$ and $\mathbf{\Lambda}_g^{(0)}$ has been developed, to deal with the possible existence of gross outlying observations among the subsamples, which could inflate some of their eigenvalues. The rationale under the proposed procedure is, as usual, to fill in randomly the missing information in the complete model through random subsamples and, then, to estimate the other parameters. The missing information here are the factors $\mathbf{u}_{ig}$ for $i = 1, \ldots, n$ and $g = 1, \ldots, G$, which, under the assumptions for the model, are realizations from independently $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ distributed $\mathbf{U}_{ig}$ random variables. We may consider

model (2) in group g as a regression of $\mathbf{X}_i$ with intercept $\boldsymbol{\mu}_g$, regression coefficients given by $\boldsymbol{\Lambda}_g$, where the explanatory variables are the latent factors $\mathbf{U}_{ig}$, and with regression errors $\mathbf{e}_{ig}$. Hence, we draw $p+1$ random independent observations from the d-variate standard Gaussian to fill a $(p+1)\times d$ matrix $\mathbf{u}^g$. Then we set $\boldsymbol{\Lambda}_g^{(0)} = ((\mathbf{u}^g)'\mathbf{u}^g)^{-1}(\mathbf{u}^g)'\mathbf{x}_c^g$, where $\mathbf{x}_c^g$ is obtained by centering the columns of the $\mathbf{x}^g$ matrix. To provide a restricted random generation of $\boldsymbol{\Psi}_g$, the $(p+1)\times p$ matrix $\varepsilon_g = \mathbf{x}_c^g - \boldsymbol{\Lambda}_g^{(0)}\mathbf{u}^g$ is computed, and the diagonal elements of $\boldsymbol{\Psi}_g^{(0)}$ are set equal to the variances of the p columns of the ε_g matrix. After repeating this for $g=1,\ldots,G$, if the obtained matrices $\boldsymbol{\Lambda}_g^{(0)}$ and $\boldsymbol{\Psi}_g^{(0)}$ do not satisfy the required constraints (5) and (6), then the constrained maximizations described in step 2.4 must be applied. Finally, weights $\pi_1^{(0)},...,\pi_G^{(0)}$ in the interval $(0,1)$ and summing up to 1 are randomly chosen.

2 *Trimmed AECM steps:*
The following steps 2.1–2.6. are alternatively executed until a maximum number of iterations, *MaxIter*, is reached. The implementation of trimming is related to the "concentration" steps applied in high-breakdown robust methods (Rousseeuw and Van Driessen, 1999). Trimming is performed before each E-step, while constraints are enforced during the second cycle CM step.

2.1 *First cycle. Trimming:* Evaluate the n quantities

$$D(\mathbf{x}_i;\theta^{(l)}) = \sum_{g=1}^{G}\phi_p(\mathbf{x}_i;\mu_g,\boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g'+\boldsymbol{\Psi}_g)\pi_g \quad \text{for} \quad i=1,\ldots,n$$

and sort them to obtain their α quantile denoted by $D_{([n\alpha])}$. Notice that $D(\mathbf{x}_i;\theta^{(l)})$ is the contribution given from $\mathbf{x}_i$ to the overall likelihood. Now consider the set of indices $I\subset\{1,2,...,n\}$ defined as

$$I=\{i: D(\mathbf{x}_i;\theta^{(l)})\geq D_{([n\alpha])}\}.$$

Then set the trimming function as $\zeta(\mathbf{x}_i)=1$ for $i\in I$, and $\zeta(\mathbf{x}_i)=0$ otherwise. To update the parameters, only the observations with indices in I will be taken into account. In other words, the proportion α of observations with the smallest $D(\mathbf{x}_i;\theta^{(l)})$ values are tentatively discarded.

2.2 *First cycle. E-step:*
Here $\boldsymbol{\theta}_1=\{\pi_g,\boldsymbol{\mu}_g; g=1,\ldots,G\}$ and the missing data are the unobserved group labels $\mathbf{z}=(\mathbf{z}_1,\ldots,\mathbf{z}_n)'$. The E-step on the first cycle on the $(l+1)$-th iteration requires the calculation of

$$Q_1\left(\boldsymbol{\theta}_1;\boldsymbol{\theta}^{(l)}\right)=\mathbb{E}_{\boldsymbol{\theta}^{(l)}}\Big[\sum_{i=1}^{n}\zeta(\mathbf{x}_i)\sum_{g=1}^{G}Z_{ig}\left(\log\pi_g+\log\phi_p\left(\mathbf{x}_i;\boldsymbol{\mu}_g^{(l)},\boldsymbol{\Sigma}_g^{(l)}\right)\right)\Big|\,\mathbf{x}\Big],$$

which is the expected trimmed complete-data log-likelihood, given the data $\mathbf{x}$ and using the current estimate $\boldsymbol{\theta}^{(l)}$ for $\boldsymbol{\theta}$, where $\boldsymbol{\Sigma}_g^{(l)}=\boldsymbol{\Lambda}_g^{(l)}[\boldsymbol{\Lambda}_g^{(l)}]'+\boldsymbol{\Psi}_g^{(l)}$. In practice, it is necessary to calculate $\mathbb{E}_{\boldsymbol{\theta}^{(l)}}[Z_{ig}|\,\mathbf{x}]=z_{ig}^{(l+1/2)}$, where the latter are the "posterior probabilities" often considered in standard EM algorithms and which are evaluated as follows. Let us define

$$D_g(\mathbf{x};\theta^{(l)})=\phi_p\left(\mathbf{x};\boldsymbol{\mu}_g^{(l)},\boldsymbol{\Sigma}_g^{(l)}\right)\pi_g^{(l)}$$

then, set

$$z_{ig}^{(l+1/2)}=\frac{D_g(\mathbf{x}_i;\theta^{(l)})}{D(\mathbf{x}_i;\theta^{(l)})}.$$

2.3 *First cycle. CM-step:* This first CM step requires the maximization of $Q_1(\boldsymbol{\theta}_1;\boldsymbol{\theta}^{(l)})$ over $\boldsymbol{\theta}_1$, with $\boldsymbol{\theta}_2$ held fixed at $\boldsymbol{\theta}_2^{(l)}$. We get $\boldsymbol{\theta}_1^{(l+1)}$ by updating π_g and μ_g as follows

$$\pi_g^{(l+1)} = \frac{\sum_{i=1}^n z_{ig}^{(l+1/2)}\zeta(\mathbf{x}_i)}{[n(1-\alpha)]}$$

and

$$\mu_g^{(l+1)} = \frac{\sum_{i=1}^n z_{ig}^{(l+1/2)}\zeta(\mathbf{x}_i)\mathbf{x}_i}{n_g^{(l+1/2)}}$$

where $n_g^{(l+1/2)} = \sum_{i=1}^n z_{ig}^{(l+1/2)}\zeta(\mathbf{x}_i)$, for $g = 1,\ldots,G$.
According to notation in McLachlan and Peel (2000b), we set $\boldsymbol{\theta}^{(l+1/2)} = \left(\boldsymbol{\theta}_1^{(l+1)},\boldsymbol{\theta}_2^{(l)}\right)$.

2.4 *Second cycle. Trimming:* Re-evaluate the n quantities $D(\mathbf{x}_i;\theta^{(l)})$ and, as done in step [2.1], update the trimming function $\zeta(\mathbf{x}_i)$.

2.5 *Second cycle. E- step:*
Here $\boldsymbol{\theta}_2 = \{(\boldsymbol{\Lambda}_g,\boldsymbol{\Psi}_g),\ g = 1,\ldots,G\}$ has to be considered, where the missing data are the unobserved group labels $\mathbf{Z}$ and the latent factors $\mathbf{U}$.
The E-step of the second cycle on the l-th iteration requires the calculation of the conditional expectation of the trimmed complete-data log-likelihood, given the observed data $\mathbf{x}$ and using the current estimate $\boldsymbol{\theta}^{(l+1/2)}$ for $\boldsymbol{\theta}$, i.e.

$$Q_2\left(\boldsymbol{\theta}_2;\boldsymbol{\theta}^{(l+1/2)}\right) = \mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}\Big[\sum_{i=1}^n \zeta(\mathbf{x}_i)\sum_{g=1}^G Z_{ig}\left(\log\pi_g^{(l+1)} + \log\phi_p(\mathbf{x}_i;\boldsymbol{\mu}_g^{(l+1)} - \boldsymbol{\Lambda}_g^{(l)}\mathbf{U}_{ig},\boldsymbol{\Psi}_g^{(l)}) + \log\phi_d(\mathbf{U}_{ig};0,\mathbf{I}_d)\right)\Big|\ \mathbf{x}\Big].$$

In addition to updating the posterior probabilities $\mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}[Z_{ig}|\mathbf{x}] = z_{ig}^{(l+1)}$ (and consequently $n_g^{(l+1)} = \sum_{i=1}^n z_{ig}^{(l+1)}\zeta(\mathbf{x}_i)$, for $g = 1,\ldots,G$, as previously done), this leads to an evaluation of the following conditional expectations: $\mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}[Z_{ig}\mathbf{U}_{ig}|\mathbf{x}]$ and $\mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}[Z_{ig}\mathbf{U}_{ig}\mathbf{U}_{ig}'|\mathbf{x}]$. Recalling that the conditional distribution of $\mathbf{U}_{ig}$, given $\mathbf{x}_i$, is

$$\mathbf{U}_{ig}|\mathbf{x}_i \sim \mathcal{N}\left(\boldsymbol{\gamma}_g(\mathbf{x}_i-\boldsymbol{\mu}_g),\mathbf{I}_q-\boldsymbol{\gamma}_g\boldsymbol{\Lambda}_g\right)$$

for $i = 1,\ldots,n$ and $g = 1,\ldots,G$ with

$$\boldsymbol{\gamma}_g = \boldsymbol{\Lambda}_g'(\boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g'+\boldsymbol{\Psi}_g)^{-1},$$

we obtain

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}[Z_{ig}\mathbf{U}_{ig}|\mathbf{x}_i] &= z_{ig}^{(l+1)}\boldsymbol{\gamma}_g^{(l)}\left(\mathbf{x}_i-\boldsymbol{\mu}_g^{(l+1)}\right)\\ \mathbb{E}_{\boldsymbol{\theta}^{(l+1/2)}}[Z_{ig}\mathbf{U}_{ig}\mathbf{U}_{ig}'|\mathbf{x}_i] &= z_{ig}^{(l+1)}\Big[\boldsymbol{\gamma}_g^{(l)}\left(\mathbf{x}_i-\boldsymbol{\mu}_g^{(l+1)}\right)\left(\mathbf{x}_i-\boldsymbol{\mu}_g^{(l+1)}\right)'\boldsymbol{\gamma}_g^{(l)\prime}+\mathbf{I}_q-\boldsymbol{\gamma}_g^{(l)}\boldsymbol{\Lambda}_g^{(l)}\Big]\end{aligned}$$

where we set

$$\boldsymbol{\gamma}_g^{(l)} = \boldsymbol{\Lambda}_g^{(l)\prime}\left(\boldsymbol{\Lambda}_g^{(l)}\boldsymbol{\Lambda}_g^{(l)\prime}+\boldsymbol{\Psi}_g^{(l)}\right)^{-1}.$$

2.6 *Second cycle. CM-step for constrained estimation of $\mathbf{\Lambda}_g$ and $\mathbf{\Psi}_g$:*

Here our aim is to maximize $Q_2\left(\boldsymbol{\theta}_2;\boldsymbol{\theta}^{(l)}\right)$ over $\boldsymbol{\theta}$, with $\boldsymbol{\theta}_1$ held fixed at $\boldsymbol{\theta}_1^{(l+1)}$. After some matrix algebra, this yields the updated ML-estimates

$$\begin{aligned}\mathbf{\Lambda}_g &= \mathbf{S}_g^{(l+1)}\boldsymbol{\gamma}_g^{(l)\prime}[\boldsymbol{\gamma}_g^{(l)}\mathbf{S}_g^{(l+1)}\boldsymbol{\gamma}_g^{(l)\prime}+\mathbf{I}_q-\boldsymbol{\gamma}_g^{(l)}\mathbf{\Lambda}_g^{(l)}]^{-1}\\ \mathbf{\Psi}_g &= \operatorname{diag}\left\{\mathbf{S}_g^{(l+1)}-\mathbf{\Lambda}_g^{(l+1)}\boldsymbol{\gamma}_g^{(l)}\mathbf{S}_g^{(l+1)}\right\}\end{aligned}$$

where $\mathbf{S}_g^{(l+1)}$ denotes the sample scatter matrix in group g, for $g=1,\ldots,G$

$$\mathbf{S}_g^{(l+1)} = (1/n_g^{(l+1)})\sum_{i=1}^{n} z_{ig}^{(l+1)}\zeta(\mathbf{x}_i)\left(\mathbf{x}_i-\boldsymbol{\mu}_g^{(l+1)}\right)\left(\mathbf{x}_i-\boldsymbol{\mu}_g^{(l+1)}\right)'.$$

During the iterations, due to the updates, it may happen that the $\mathbf{\Lambda}_g$ matrices do not belong to the constrained parameter space $\mathbf{\Theta}_c$. In the case where the additional constraints (6) have to be imposed, and the norms of the column vectors of the matrices $\mathbf{\Lambda}_g$ do not satisfy them, $\mathbf{\Lambda}_g^{(l+1)}\in\mathbf{\Theta}_c$ can be obtained as follows. After defining the diagonal matrix $\mathbf{E}_g=\operatorname{diag}(\eta_{g1},\eta_{g2},...,\eta_{gd})$, the truncated norms are then given as

$$[\eta_{gk}]_m=\min\left(c_{load}\cdot m,\ \max(\eta_{gk},m)\right),\ \text{for}\quad k=1,\ldots,d \text{ and } g=1,\ldots,G,$$

with m being some threshold value. The loading matrices are finally updated as $\mathbf{\Lambda}_g^{(l+1)}=\mathbf{\Lambda}_g\mathbf{E}_g^{-1}\mathbf{E}_g^{*}$ with

$$\mathbf{E}_g^{*}=\operatorname{diag}\left([\eta_{g1}]_{m_{\text{opt}}},[\eta_{g2}]_{m_{\text{opt}}},...,[\eta_{gd}]_{m_{\text{opt}}}\right)$$

and m_{opt} minimizing the real valued function

$$f_{load}(m)=\sum_{g=1}^{G}\pi_g^{(l+1)}\sum_{k=1}^{d}\left(\log\left([\eta_{gk}]_m\right)+\frac{\eta_{gk}}{[\eta_{gk}]_m}\right). \tag{7}$$

It may be mentioned here, in passing, that Proposition 3.2 in Fritz et al. (2013) shows that m_{opt} can be obtained by evaluating $2dG+1$ times the real valued function $f_{load}(m)$ in (7).
Given the $\mathbf{\Lambda}_g^{(l+1)}$, the matrices

$$\mathbf{\Psi}_g=\operatorname{diag}\left\{\mathbf{S}_g^{(l+1)}-\mathbf{\Lambda}_g^{(l+1)}\boldsymbol{\gamma}_g^{(l)}\mathbf{S}_g^{(l+1)}\right\}=\operatorname{diag}\left(\psi_{g1},...,\psi_{gp}\right)$$

can be obtained, and may not necessarily satisfy the required constraint (5). In this case, we set

$$[\psi_{gk}]_m=\min\left(c_{noise}\cdot m,\ \max(\psi_{gk},m)\right),\quad \text{for}\ \ k=1,\ldots,p;\ g=1,\ldots,G,$$

and fix the optimal threshold value m_{opt} by minimizing the following real valued function

$$f_{noise}(m)\mapsto\sum_{g=1}^{G}\pi_g^{(l+1)}\sum_{k=1}^{p}\left(\log\left([\psi_{gk}]_m\right)+\frac{\psi_{gk}}{[\psi_{gk}]_m}\right). \tag{8}$$

As before, in Fritz et al. (2013), it is shown that m_{opt} can be obtained in a straightforward way by evaluating $2pG+1$ times $f_{noise}(m)$ in (8). Thus, $\mathbf{\Psi}_g^{(l+1)}$ is finally updated as

$$\mathbf{\Psi}_g^{(l+1)}=\operatorname{diag}\left([\psi_{g1}]_{m_{\text{opt}}},...,[\psi_{gp}]_{m_{\text{opt}}}\right). \tag{9}$$

It is worth remarking that the given constrained estimation provides, at each step, the parameters $\mathbf{\Psi}_g$ and $\mathbf{\Lambda}_g$ that maximize the likelihood in the constrained parameter space Θ_c.

3 *Evaluate target function:* After applying the trimmed and constrained EM steps, and setting $\zeta(\mathbf{x}_i) = 0$ if $i \in I$ and $\zeta(\mathbf{x}_i) = 1$ if $i \notin I$, the associated value of the target function (4) is evaluated. If convergence has not been achieved before reaching the maximum number of iterations, *MaxIter*, the results are discarded.

The set of parameters yielding the highest value of the target function (among the multiple runs) and the associated trimmed indicator function $\zeta(\cdot)$ are returned as the final output of the algorithm. In the framework of model-based clustering, each unit is assigned to one group, based on the maximum a posteriori probability. Notice, in passing, that a high number of initializations is not needed, and nor a high value for *MaxIter*, as will be seen in Section 4.

In relation with the initialization strategy, the obtained initial values for the parameters in each population are based on small subsamples, aiming at ideally covering, in many trials, the full parameter space. Our proposal is based on the following idea: a small subsample has to be drawn for each group and then the information extracted from the subsample is completed with random data generated under the model assumptions. The expected consequence of this exploration of the parameter space is that spurious solutions, even singularities, can arise when running EM iterations, and the constraints on the scatters play the role of protecting against these undesired solutions. By considering many random initializations, we are confident that the best point, in terms of the likelihood, can be approached inside the restricted parameter space. The number of random initializations should increase with the number of groups G, the dimension p and in the case of very different group sizes.

It is worth remarking that the usual monotone convergence of the likelihood in the robust AECM algorithm holds true when incorporating trimming and constrained estimation for $\mathbf{\Psi}_g$. To prove this, notice firstly that, when performing the trimming step, the optimal observations have been retained, i.e. the ones with the highest contributions to the objective function. Secondly, the first cycle is the usual one in the AECM algorithm for MFA and, therefore, shares its optimality properties. Finally, it can be easily proved that, in the second EM cycle, the evaluation of the optimal $\left(\mathbf{\Lambda}_g, \mathbf{\Psi}_g\right)$ for $g = 1, \ldots, G$ corresponds to the usual way of obtaining firstly the optimal $\mathbf{\Lambda}_g$, which is not affected by the restrictions on $\mathbf{\Psi}_g$ and, then the optimal $\mathbf{\Psi}_g$ is obtained as

$$\arg\max \sum_{g=1}^{G} \frac{n_g}{2} \log(|\mathbf{\Psi}_g|) + \sum_{g=1}^{G} \frac{1}{2} \text{trace}\left[\mathbf{\Psi}_g^{-1}\left(\mathbf{S}_g^{(l+1)} - \mathbf{\Lambda}_g^{(l+1)} \boldsymbol{\gamma}_g^{(l)} \mathbf{S}_g^{(l+1)}\right)\right], \tag{10}$$

and this corresponds to (9).

On the other hand, when the algorithm also requires the constrained estimation of $\mathbf{\Lambda}_g$, the latter is based on a heuristic approach. We have empirical evidences about the monotonicity of this second EM cycle for the huge majority of the steps in which we applied it, producing low decreases in the objective function in extremely rare cases. In any case, after each entire AECM cycle, an increased likelihood was always observed.

4. Numerical studies

In this section, numerical studies will be presented, based on simulated and real data, to show the performance of the constrained and trimmed AECM algorithm with respect to unconstrained and/or untrimmed approaches.

4.1. Artificial data

We consider here the following mixture of G components of d-variate normal distributions. To perform each estimation, 40 different random initializations have been considered to start the algorithm at each run, as described in the previous section, and the best solution is retained. The needed routines have been written in R-code (R Team, 2013), and are available from the authors upon request.

Mixture: $G = 3$, $d = 6$, $q = 2$, $n = 150$.

The sample has been generated with weights $\boldsymbol{\pi} = (0.3, 0.4, 0.3)'$ according to the following parameters:

$$\boldsymbol{\mu}_1 = (0, 0, 0, 0, 0, 0)' \qquad \mathbf{\Psi}_1 = \text{diag}(0.1, 0.1, 0.1, 0.1, 0.1, 0.1)$$
$$\boldsymbol{\mu}_2 = (5, 5, 0, 0, 0, 0)' \qquad \mathbf{\Psi}_2 = \text{diag}(0.4, 0.4, 0.4, 0.4, 0.4, 0.4)$$
$$\boldsymbol{\mu}_3 = (10, 10, 0, 0, 0, 0)' \qquad \mathbf{\Psi}_3 = \text{diag}(0.2, 0.2, 0.2, 0.2, 0.2, 0.2)$$

$$\mathbf{\Lambda}_1 = \begin{pmatrix} 0.50 & 1.00 \\ 1.00 & 0.45 \\ 0.05 & -0.50 \\ -0.60 & 0.50 \\ 0.50 & 0.10 \\ 1.00 & -0.15 \end{pmatrix} \qquad \mathbf{\Lambda}_2 = \begin{pmatrix} 0.10 & 0.20 \\ 0.20 & 0.50 \\ 1.00 & -1.00 \\ -0.20 & 0.50 \\ 1.00 & 0.70 \\ 1.20 & -0.30 \end{pmatrix} \qquad \mathbf{\Lambda}_3 = \begin{pmatrix} 0.10 & 0.20 \\ 0.20 & 0.00 \\ 1.00 & 0.00 \\ -0.20 & 0.00 \\ 1.00 & 0.00 \\ 0.00 & -1.30 \end{pmatrix}.$$

Figure 1 shows a specimen of randomly generated data from the given mixture.

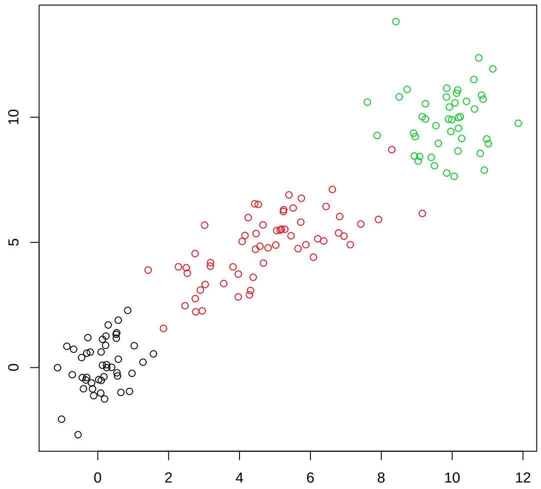

Figure 1. A specimen of 150 data points generated from the mixture (the first two coordinates are plotted, groups in black, red and green)

Our analysis begins by running the AECM algorithm on the generated sample, and considering the following six settings, namely:

S1. a ”virtually” unconstrained approach (i.e. $c_{\text{noise}} = c_{\text{load}} = 10^{10}$) without trimming ($\alpha = 0$),
S2. an adequate constraint on $\mathbf{\Psi}_g$, no constraint on $\mathbf{\Lambda}_g$ ($c_{\text{noise}} = 5, c_{\text{load}} = 10^{10}$) and no trimming ($\alpha = 0$),
S3. adequate constraints on $\mathbf{\Psi}_g$ and $\mathbf{\Lambda}_g$ ($c_{\text{noise}} = 5$, $c_{\text{load}} = 3$), and still no trimming ($\alpha = 0$),
S4. a ”virtually” unconstrained approach (i.e. $c_{\text{noise}} = c_{\text{load}} = 10^{10}$) with trimming ($\alpha = 0.06$),
S5. an adequate constraint on $\mathbf{\Psi}_g$, no constraint on $\mathbf{\Lambda}_g$ ($c_{\text{noise}} = 5, c_{\text{load}} = 10^{10}$), with trimming ($\alpha = 0.06$),
S6. adequate constraints on $\mathbf{\Psi}_g$ and $\mathbf{\Lambda}_g$ ($c_{\text{noise}} = 5$, $c_{\text{load}} = 3$), with trimming ($\alpha = 0.06$)

It is worth noticing that when setting $c_{noise} = 10^{10}$ singularities are surely discarded, and the estimation is allowed to move in a wide parameter space that contains the global maximum, among several local ones. In this situation, the estimation could incur spurious solutions. We expect the algorithm to improve its performance when giving the ”right” constraints. The adequate constraints can by evaluated by obtaining the maximum ratio among the eigenvalues of $\mathbf{\Psi}_g$ and among the singular values of $\mathbf{\Lambda}_g$. As the singular values of $\mathbf{\Lambda}_1$ are (3.069, 1.528), of $\mathbf{\Lambda}_2$ are (3.777, 1.873) and of $\mathbf{\Lambda}_3$ are (2.091, 1.729), hence $c_{load} \geq 2.471$; while the diagonal elements of $\mathbf{\Psi}_g$ are 0.1, 0.4, and 0.2, so $c_{noise} \geq 4$. When trimming is also applied to the artificially generated data, the effect of an unneeded elimination of the outermost points in the model estimation and subsequent classification can be seen. To measure the performance of the algorithm, the average misclassification error η is evaluated, over 1000 repetitions of the estimation procedure. The misclassification error is defined as the relative frequency of points of the sample erroneously labeled, taking into account that noise and pointwise contamination (when added) should be identified, as they virtually do not belong to the three groups. We see that the algorithm, applied without trimming, gives a superb classification with and without constraints. While adding trimming, the misclassification error, as expected, is pretty close to the trimming level, and all non-trimmed observations are perfectly classified (with the sole exception of 1 misclassified unit, that happened only once, when $c_{noise} = c_{load} = 10^{10}$, and occurred 4 times when $c_{noise} = 5$ and $c_{load} = 10^{10}$, over 1000 runs). The results are summarized in the first row of Table 1. Moreover, the other parameters, such as the means $\boldsymbol{\mu}_g$, and $\boldsymbol{\Psi}_g$, $\boldsymbol{\Lambda}_g$ for $g = 1, 2, 3$, are close to the values from which the data have been generated, as will be shown in Subsection 4.1.1.

Table 1. Misclassification error η (percent average values on 1000 repetitions of the estimation procedure) of the AECM algorithm with settings S1-S6, applied on the artificially generated data, and with contamination

	S1	S2	S3	S4	S5	S6
c_{noise}	10^{10}	5	5	10^{10}	5	5
c_{load}	10^{10}	10^{10}	3	10^{10}	10^{10}	3
α	0	0	0	0.06	0.06	0.06
D	0.003	0.006	0.001	6.001	6.003	6.000
D+N	29.606	29.654	48.324	3.131	3.128	0.004
D+PC	28.098	15.035	17.317	24.966	3.128	0.004
D+N+PC	44.690	45.089	47.044	12.289	5.887	0.003

In Table 1, four scenarios have been considered, i.e.:

D: the artificially generated data,

D+N: 10 points of uniform noise have been added around the data,

D+PC: 10 points of pointwise contamination have been added outside the range of the data,

D+N+PC: both the 10 points of uniform noise and the 10 points of pointwise contamination have been added to the data.

The algorithm has been applied to the different datasets in the six previous settings S1-S6 (i.e. with/without constraints and trimming), to obtain and compare the misclassification errors. In the case of D+N+PC, the right trimming level should obviously be raised at $\alpha = 0.12$. Results in the second row of Table 1 show that trimming is very effective to identify and discard noise in the data, and constraints contribute getting close to a perfect classification. The misclassification error (reported in the third row of Table 1) shows that, when concentrated outliers occur in the data, the constrained estimation is also needed to achieve a pretty good behavior of the algorithm. Noise and pointwise contamination could cause very messy estimations, as can be seen in the first three columns of the table, whenever the estimation only relies/does not rely on constraints.

In conclusion, to be protected against all types of data corruption, constrained estimation *and* trimming are needed. With the joint effect of these tools, the statistical problem is well posed, spurious solutions can be avoided and the estimator resists the influence of all classes of contaminating observations. With the application of only one of the previous instruments, these benefits are lost.

The algorithm for estimating a robust MFA have been written in R language and is available from the authors upon request, and an R package is currently under development. In terms of computational resources, the algorithm is not so heavily modified from the usual one, and the time for executing the robust AECM, if compared to the one without trimming and constraints, remains of the same order of magnitude: it needs about twice the computing time. To have an idea, 10 robust estimations (based on 40 random initializations and 60 iterations) on the artificial data required 3.96 seconds of system time, while the same experiment with the classical AECM required 1.911 seconds.

4.1.1. Properties of the estimators for the mixture parameters

Now, a second analysis on the same artificial data is performed and the main interest here is in assessing the effect of trimming and constraints on the properties of the model estimators. Namely, we estimate their bias and mean square error, when the data is affected by noise and/or pointwise contamination. The same four scenarios of the previous subsection are again considered, i.e.: the artificially generated data (D), the data with the added 10 points of uniform noise (D+N), the data with the added 10 points of pointwise contamination (D+PC), and finally the data with both uniform noise and the pointwise contamination (D+N+PC).

We apply the algorithm for estimating a trimmed MFA model in all the four scenarios, exploring the six settings on c_{noise}, c_{load} and α that have been shown in Table 1. The benchmark of all simulations is given by the results obtained on artificial data drawn from a given MFA without outliers. In each experiment, a sample of size $n = 150$ has been drawn 1000 times from the mixture described at the beginning of this Section, and the model parameters for the trimmed MFA have been estimated using the algorithm presented in the previous Section 3.2, by setting $c_{noise} = c_{load} = 10^{10}$ (a virtually unconstrained solution) or $c_{noise} = 5, c_{load} = 3$ for a constrained one, and $\alpha = 0$ for no trimming, while $\alpha = 0.06$ or $\alpha = 0.12$ when adopting adequate trimming.

Notice that the considered estimators in each component are vectors (apart from π_g which are scalar quantities, for $g = 1, \ldots, G$). We are interested in providing synthetic measures of their properties, such as bias and mean square error (MSE). As usual, let $\hat{T}$ be an estimator for the scalar parameter t, then the bias of $\hat{T}$ is given by $bias(\hat{T}) = \mathbb{E}(\hat{T}) - t$, i.e. it is the signed absolute deviation of the expected value $\mathbb{E}(\hat{T})$ from t. Therefore, we would have 6 biases for each component of the mean μ_g, 6 for diag(Ψ_g) and 12 for Λ_g. On the other hand, MSE is defined as a scalar quantity, namely $\mathbb{E}(|\hat{T} - t|^2) = \text{trace}(Var(\hat{T})) + \text{bias}(\hat{T})^2$, also for vector estimators. Hence, a synthesis of each parameter's biases is adopted by considering the mean of their absolute values on each component. Below the bias, in Tables 2 and 3, the MSE is provided in parenthesis.

The results on bias and mean square error for the case of estimating the trimmed MFA with trimming but without constraints or viceversa, show the harmful effects of distorted inference. The only exception comes from D+N, where trimming is pretty able to cope with the contamination. On the other hand, when reasonable constraints $c_{noise} = 5$, $c_{load} = 3$ and a right trimming level are applied to deal with the added outliers, the results come back being very close to the benchmark, shown in the first column of Table 2. Therefore, it has been shown that robust inference reduces bias and mean square error, in both cases of sparse and concentrated outliers.

Table 2. Bias and MSE (in parentheses) of $\hat{\pi}_i$, and bias as the sum of absolute deviations, followed by MSE (in parentheses) of the parameter estimators $\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Psi}}_i, \hat{\boldsymbol{\Lambda}}_i$, for $i = 1, 2, 3$ for the artificial data "D", and for the artificial data plus noise "D+N"; labels from S1 to S6 denote the estimation settings

	D						D+N					
	S1	S2	S3	S4	S5	S6	S1	S2	S3	S4	S5	S6
π_1	0 (0)	0 (0)	0 (0)	0.0154 (2e-04)	0.0166 (3e-04)	0.0167 (3e-04)	-0.037 (0.0017)	-0.1636 (0.0283)	-0.2206 (0.0502)	1e-04 (0)	0 (0)	0 (0)
π_2	0 (0)	0 (0)	0 (0)	-0.0227 (5e-04)	-0.0229 (5e-04)	-0.0238 (6e-04)	-0.2835 (0.0807)	0.065 (0.0058)	0.4077 (0.1677)	-1e-04 (0)	0 (0)	0 (0)
π_3	0 (0)	0 (0)	0 (0)	0.0073 (1e-04)	0.0064 (0)	0.0071 (1e-04)	0.3205 (0.1031)	0.0985 (0.0113)	-0.1871 (0.0365)	0 (0)	0 (0)	0 (0)
$\boldsymbol{\mu}_1$	0.001 (0.113)	0.001 (0.117)	0.003 (0.112)	0.003 (0.124)	0.004 (0.114)	0.006 (0.12)	0.215 (9.211)	2.289 (99.165)	5.468 (122.462)	0.002 (0.118)	0.003 (0.116)	0.002 (0.11)
$\boldsymbol{\mu}_2$	0.003 (0.132)	0.003 (0.133)	0.002 (0.131)	0.005 (0.173)	0.002 (0.179)	0.003 (0.185)	4.89 (74.119)	2.191 (69.868)	0.694 (34.888)	0.002 (0.143)	0.004 (0.126)	0.006 (0.139)
$\boldsymbol{\mu}_3$	0.006 (0.112)	0.003 (0.11)	0.001 (0.111)	0.003 (0.122)	0.004 (0.125)	0.003 (0.123)	1.492 (87.415)	1.026 (91.361)	10.338 (471.833)	0.004 (0.109)	0.004 (0.108)	0.002 (0.115)
$\boldsymbol{\Psi}_1$	0.007 (0.009)	0.002 (0.003)	0.002 (0.003)	0.009 (0.009)	0.005 (0.004)	0.005 (0.004)	0.311 (58.795)	1.132 (11.2)	0.795 (6.278)	0.007 (0.008)	0.003 (0.003)	0.002 (0.003)
$\boldsymbol{\Psi}_2$	0.029 (0.142)	0.045 (0.05)	0.045 (0.053)	0.074 (0.169)	0.1 (0.104)	0.1 (0.106)	9.194 (1837.338)	0.445 (2.547)	0.355 (2.248)	0.033 (0.136)	0.047 (0.053)	0.047 (0.055)
$\boldsymbol{\Psi}_3$	0.168 (1.018)	0.04 (0.053)	0.04 (0.055)	0.174 (0.952)	0.048 (0.048)	0.047 (0.048)	0.422 (2.613)	0.316 (1.375)	0.928 (7.718)	0.166 (0.989)	0.041 (0.055)	0.041 (0.054)
$\boldsymbol{\Lambda}_1$	0.526 (9.031)	0.529 (9.127)	0.533 (9.158)	0.523 (9.064)	0.526 (9.07)	0.551 (9.547)	0.558 (37.297)	0.537 (296.817)	0.525 (52.772)	0.523 (9.064)	0.527 (9.11)	0.517 (8.954)
$\boldsymbol{\Lambda}_2$	0.585 (11.495)	0.576 (11.369)	0.56 (11.011)	0.573 (11.326)	0.566 (10.335)	0.575 (10.634)	0.668 (585.002)	0.633 (205.661)	0.553 (49.041)	0.573 (11.326)	0.574 (11.319)	0.58 (11.569)
$\boldsymbol{\Lambda}_3$	0.333 (6.786)	0.332 (7.448)	0.341 (7.444)	0.34 (6.888)	0.342 (7.284)	0.337 (7.224)	0.346 (26.408)	0.363 (22.941)	0.346 (119.003)	0.34 (6.888)	0.341 (7.408)	0.337 (7.503)

Table 3. Bias and MSE (in parentheses) of $\hat{\pi}_i$, and bias as the sum of absolute deviations, followed by MSE (in parentheses) of the parameter estimators $\hat{\boldsymbol{\mu}}_i, \hat{\boldsymbol{\Psi}}_i, \hat{\boldsymbol{\Lambda}}_i$, for $i = 1, 2, 3$ for the artificial data plus pointwise contamination "D+PC", and for the artificial data plus noise and pointwise contamination "D+N+PC"; labels from S1 to S6 denote the estimation settings

	D+PC						D+N+PC					
	S1	S2	S3	S4	S5	S6	S1	S2	S3	S4	S5	S6
π_1	-0.0304 (0.001)	-0.0276 (0.001)	-0.0374 (0.0018)	-0.0664 (0.0051)	0 (0)	0 (0)	-0.218 (0.0482)	-0.2116 (0.0449)	-0.2163 (0.0468)	-0.0534 (0.003)	1e-04 (0)	0 (0)
π_2	-0.3076 (0.0947)	-0.0868 (0.0077)	-0.1376 (0.0193)	-0.1742 (0.031)	0 (0)	0 (0)	0.4025 (0.1627)	0.4418 (0.1953)	0.4822 (0.2325)	0.0316 (0.0011)	-1e-04 (0)	0 (0)
π_3	0.338 (0.1143)	0.1144 (0.0133)	0.175 (0.031)	0.2406 (0.0586)	0 (0)	0 (0)	-0.1845 (0.0347)	-0.2302 (0.0531)	-0.2659 (0.0707)	0.0218 (6e-04)	0 (0)	0 (0)
$\boldsymbol{\mu}_1$	0.346 (16.538)	0.931 (30.237)	1.365 (47.55)	1.918 (75.165)	0.002 (0.116)	0.003 (0.112)	21.143 (1416.987)	18.651 (985.875)	20.535 (881.201)	5.233 (735.9)	0.006 (0.114)	0.001 (0.117)
$\boldsymbol{\mu}_2$	8.59 (150.803)	2.682 (102.543)	4.489 (152.814)	6.005 (147.795)	0.005 (0.129)	0.004 (0.135)	1.146 (110.599)	0.534 (49.885)	0.001 (0.044)	2.208 (344.79)	0.007 (0.13)	0.003 (0.132)
$\boldsymbol{\mu}_3$	0.912 (3.31)	0.818 (7.14)	0.61 (5.662)	0.854 (5.55)	0.003 (0.111)	0.002 (0.113)	6.59 (383.624)	9.178 (640.891)	11.134 (381.857)	0.422 (24.125)	0.002 (0.11)	0.004 (0.115)
$\boldsymbol{\Psi}_1$	0.012 (0.011)	0.052 (0.072)	0.07 (0.108)	0.032 (0.027)	0.003 (0.003)	0.003 (0.003)	0.069 (17.255)	0.509 (3.089)	0.636 (4.316)	0.028 (0.021)	0.003 (0.004)	0.002 (0.003)
$\boldsymbol{\Psi}_2$	0.363 (0.969)	0.098 (0.22)	0.132 (0.285)	0.302 (0.747)	0.045 (0.052)	0.046 (0.052)	0.9 (101.563)	0.239 (0.942)	0.246 (0.788)	0.054 (0.306)	0.046 (0.051)	0.044 (0.05)
$\boldsymbol{\Psi}_3$	0.692 (3.528)	0.12 (0.332)	0.186 (0.499)	0.241 (0.747)	0.041 (0.054)	0.04 (0.054)	8.865 (2101.106)	1.006 (8.218)	1.176 (10.605)	0.124 (0.772)	0.042 (0.053)	0.042 (0.054)
$\boldsymbol{\Lambda}_1$	0.539 (9.581)	0.549 (26.694)	0.544 (18.72)	0.566 (57.769)	0.529 (9.111)	0.511 (8.76)	0.433 (470.528)	0.5 (594.275)	0.613 (257.523)	0.47 (555.065)	0.53 (9.126)	0.526 (9.072)
$\boldsymbol{\Lambda}_2$	0.559 (10.083)	0.562 (42.216)	0.565 (27.648)	0.596 (102.541)	0.57 (11.248)	0.572 (11.36)	0.568 (151.272)	0.598 (96.445)	0.596 (38.347)	0.599 (182.497)	0.594 (11.736)	0.57 (11.175)
$\boldsymbol{\Lambda}_3$	0.334 (11.31)	0.369 (34.879)	0.344 (15.699)	0.348 (18.004)	0.331 (7.337)	0.342 (7.488)	0.335 (470.156)	0.363 (376.344)	0.356 (193.749)	0.342 (28.224)	0.337 (7.468)	0.349 (7.587)

The distributions of the estimators for the model parameters can be represented by box plots, and some of them are shown in Figure 2, namely with reference to $\hat{\pi}_1$ (upper panel), $\hat{\mu}_1[1,1]$ (second panel), $\hat{\Psi}_1[1,1]$ (third panel) and $\hat{\Lambda}_1[1,1]$ (bottom panel). In a direct comparison, the small efficiency reduction of the estimator when applying trimming and constraints on the true data (cases D / S2-S6) can be seen, the effect of using only trimming when uniform noise has been added to data (case D+N / S4) is apparent; finally, the joint usage of trimming and constraints on $\mathbf{\Psi}_g$ is shown to be effective to protect against all types of contamination.

4.2. Real data: the AIS data set

As an illustration, we apply the proposed technique to the Australian Institute of Sports (AIS) data, which is a famous benchmark dataset in the multivariate literature, originally reported by Cook and Weisberg (1994) and subsequently analyzed by Azzalini and Dalla Valle (1996), among many other authors. The dataset consists of $p = 11$ physical and hematological measurements on 202 athletes (100 females and 102 males) in different sports, and is available within the R package *sn* (Azzalini, A., 2011). The observed variables are: red cell count (RCC), white cell count (WCC), Hematocrit (Hc), Hemoglobin (Hg), plasma ferritin concentration (Fe), body mass index, weight/height^2 (BMI), sum of skin folds (SSF), body fat percentage (Bfat), lean body mass (LBM), height, cm (Ht), weight, kg (Wt), apart from gender and kind of Sport. A partial scatterplot of the AIS dataset is given in Figure 3.

Our purpose is to provide a model for the entire dataset, and since the group labels (athlete's gender) are provided in advance, the aim is to classify athletes by this feature.

Let us begin our analysis by fitting a mixture of multivariate Gaussian distributions, using the *Mclust* package in R. The routine *mclustBIC*, after fitting a set of normal mixture models, considering from 1 to 9 components in the mixture and different patterns for the covariance matrices, selects the best EEV model (ellipsoidal scatters, with equal volume and shape, different orientation of the component scatters) with $G = 2$ components, providing the highest BIC value, i.e. $BIC = -10251.6$. Now, using this model to classify AIS data, 18 misclassified units are obtained, i.e., a misclassification error equal to $18/202 = 9.4\%$. The classification results are shown in Figure 4 (left panel).

To improve the classification, we may exploit the conjecture that a strong correlation exists between the hematological and physical measurements. Therefore, a mixture of factor analyzers may be estimated, assuming the existence of some underlying unknown factors (like nutritional status, hematological composition, overweight status indices, and so on) which jointly explain the observed measurements. Through the factors, the aim is to find a perspective on data which disentangles the overlapping components. To avoid variables having a greater impact in the model (which is not affine equivariant) due to different scales, before performing the estimation, the variables have been divided by their interquartile range. We begin by adopting the *pGmm* package from R, that fits mixtures of factor analyzers with patterned covariances. Parsimonious Gaussian mixtures are obtained by constraining the loadings Λ_g and the errors Ψ_g to be equal or not among the components. We employed the routine *pGmmEM*, considering from 1 to 9 components, and number of underlying factors d ranging from 1 to 6, with 30 different random initializations, to provide the best iteration (in terms of BIC) for each case. The best model is a CUU mixture model with $d = 4$ factors and $G = 3$ components, with $BIC = -3127.424$. CUU means "Constrained" loading matrices $\Lambda_g = \Lambda$ and "Unconstrained" error matrices $\Psi_g = \omega_g \Delta_g$, where Δ_g are normalized diagonal matrices and ω_g is a real value varying across components. Using this model to classify athletes, we got 109 misclassified units and we discarded it.

As a second attempt using *pGmm*, a UUU model has been estimated by setting $G = 2$ components, and $d = 6$. The acronym UUU means that the estimation of loadings Λ_g and errors Ψ_g is unconstrained. Based on 30 random starts, the best UUU model has $BIC = -3330.306$, and the consequent classification of the AIS dataset produces 72 misclassified units (misclassification error=35.6%, see the right panel in Figure 4).

Finally, we want to show the performance of our trimmed and constrained estimation for MFA on the AIS data. All the results are generated by the procedure described in Section 3.2, based on 30 random initializations and returning the best obtained solution of the parameters, in terms of the highest value of the final likelihood. We see that the best solution, with only 3 misclassified points, has been obtained by combining trimming ($\alpha = 0.05$) and the constrained estimation of Ψ_g ($c_{noise} = 45$) and $\mathbf{\Lambda}_g$ ($c_{load} = 10$), with $d = 6$.

Notice that the choice of $G = 2$ and $d = 6$ could be motivated by estimating all models within a range of values for G and d, and choosing the pair of values providing the best BIC. A trimmed version of the $BIC = 2\mathcal{L}_{trim}(x;\hat{\theta}) - k \log n^*$ should be considered, where we denote by k the number of free parameters in the model, and by n^* the number of non trimmed observations (i.e. $n^* = [n(1-\alpha)]$). Results are shown in Table 5. In practice, we stopped our investigation at

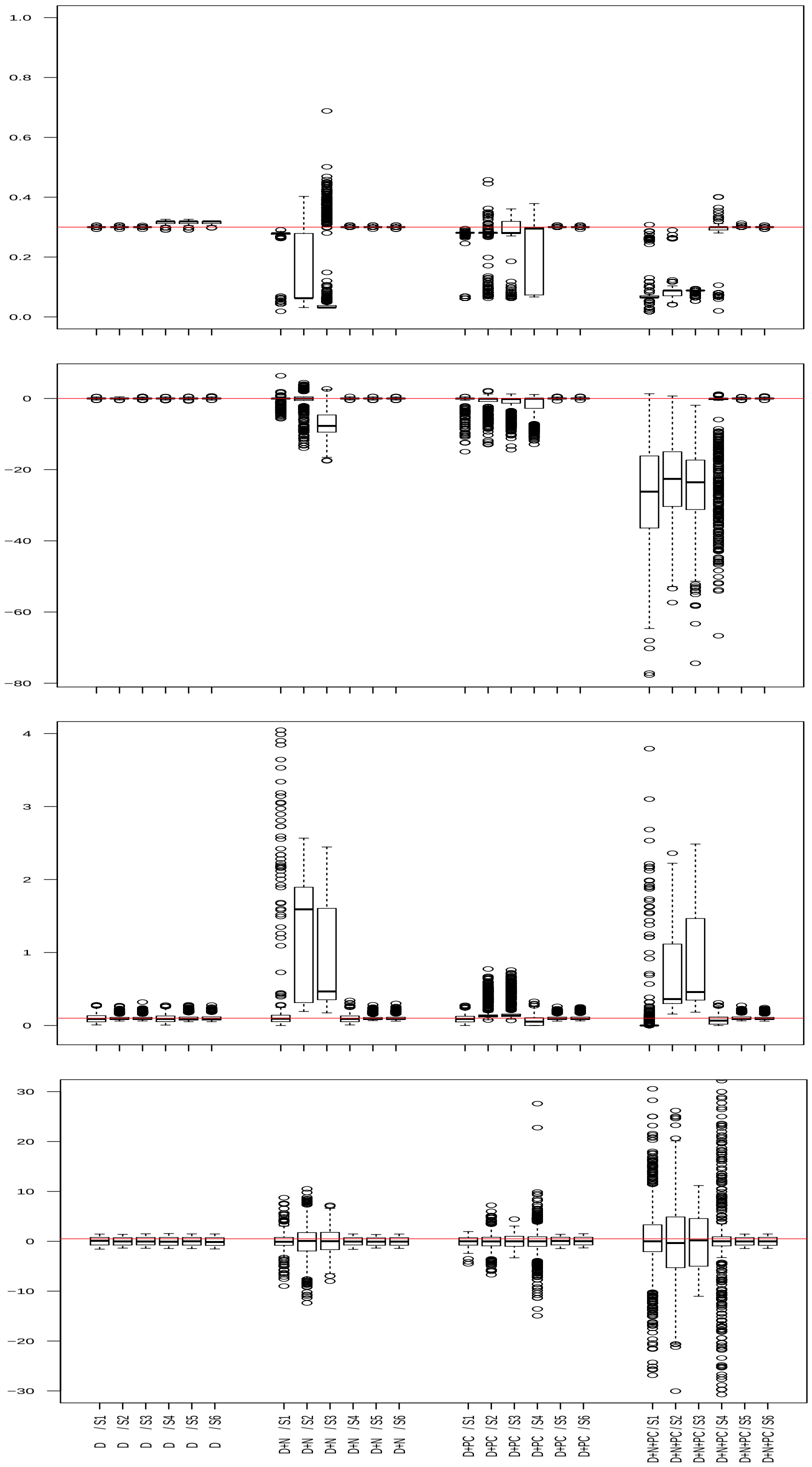

Figure 2. Boxplots of the simulated distributions of $\hat{\pi}_1$, estimator for $\pi_1 = 0.3$ (upper panel); $\hat{\mu}_1[1]$, estimator for $\mu_1[1] = 0$ (panel in 2nd row from above); $\hat{\Psi}_1[1, 1]$, estimator for $\Psi_1[1, 1] = 0.1$ (3rd panel from above); and $\hat{\Lambda}_1[1, 1]$, estimator for $\Lambda_1[1, 1] = 0.5$ (lower panel). As usual, "D stays for *data*, "+N" stays for *added noise*, "+PC" stays for *added pointwise contamination*, while S1-S6 denotes the estimation settings.

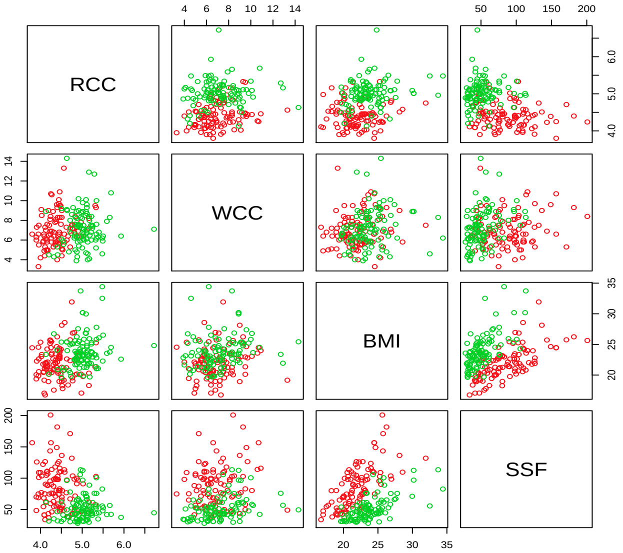

Figure 3. Scatterplot of some pairs of the AIS variables (female data in red, male in green)

Table 4. Trimmed and constrained MFA estimation on the AIS data set (best results over 30 random initializations). Misclassification error η (in percentage) under different settings

c_{noise}	10^{10}	45	10^{10}	45	10^{10}	45	10^{10}	45
c_{load}	10^{10}	10^{10}	10	10	10^{10}	10^{10}	10	10
α	0	0	0	0	0.05	0.05	0.05	0.05
η	0.1040	0.0891	0.1040	0.0891	0.0347	0.0149	0.0347	0.0149

$d = 6$ because in a factor analyzer, to reach parsimony, we should have $(p - d)^2 \geq p + d$ (as discussed in detail at the end of Section 2). We must thank an anonymous Referee for raising a question on this issue, that also confirmed our previous assessment of $d = 6$, obtained by performing a factor analysis on the observations coming from the group of male athletes, and employing a scree-plot to test the hypothesis that 6 factors are sufficient (chi square statistic equal to 97.81 on 4 degrees of freedom, and p-value= $2.88 \cdot 10^{-20}$).

Moreover, we recall here that Bekker (1997) showed that $d < \phi(p)$ is a necessary and sufficient condition for global identifiability of the diagonal matrix $\mathbf{\Psi}$, where

$$\phi(p) := \frac{2p + 1 - \sqrt{8p + 1}}{2}$$

is the so-called Ledermann bound (Ledermann, 1937). In the present case we have that $d = 6 < \phi(p) = 6.78$, hence no identifiability issues arise.

The constraints, and in first place the constraint c_{noise} on $\mathbf{\Psi}_g$, play an important role (compare results in columns 2-4-6 and 8 to the ones displayed in the odd columns), but trimming is needed to reach the best result. This is motivated by the fact that the data, in both groups, are not following an 11-dimensional multivariate Gaussian, as it can be easily checked by performing a Mardia test. Two results of the fitted models and the subsequent classifications are displayed

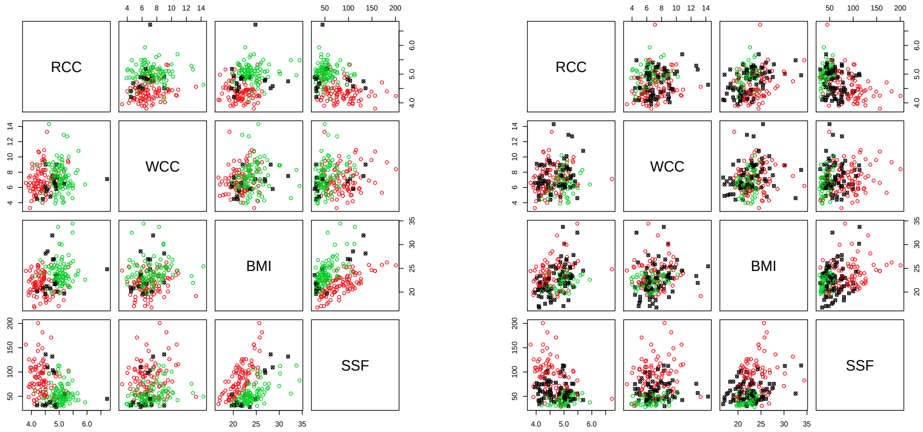

Figure 4. The classification of AIS data obtained through the best model from *Mclust* (left panel), and through the best UUU model from *pGmm* (right panel) with $G = 2$ and $d = 6$ (female data in red, male in green, misclassified units as black circlecrosses).

Groups G	1	**2**	3	4	5
Factors d					
1	3484.32	2833.18	2744.96	2702.89	2780.27
2	2436.87	2311.24	2283.16	2331.43	2433.88
3	1938.75	1885.03	1967.93	2068.36	2261.20
4	1490.72	1513.43	1662.43	1818.78	2002.08
5	1409.03	1390.86	1559.12	1774.64	1971.56
6	1142.15	**1099.89**	1371.98	1772.44	1976.96

Table 5. *trimmed* BIC for different choices of the number of factors d and the number of groups G, on AIS data.

in Figure 5, by selecting the 2 variables in the scatterplot that enable us to point out the trimmed and misclassified units. We have chosen to represent the best solution (left panel), with only 3 misclassified points, colored in black, and with 10 trimmed points, denoted by “X”. In the right panel, to make a comparison, we report classification results obtained by the non-robustly fitted model, whose details are shown in the first column of Table 4. In this second case, we were doing an almost unconstrained estimation of $\mathbf{\Psi}_g$ and $\mathbf{\Lambda}_g$ and we were not applying trimming, obtaining 21 misclassified observations.

In the robustly estimated model, the misclassified observations are in rows 70, 121 and 153 in the AIS dataset. Two misclassified units are from male athletes, one is a female athlete. The discriminant function of the mixture components for the observation in position 153 are close ($D_1(\mathbf{x};\theta) = 0.0115$ and $D_2(\mathbf{x};\theta) = 0.0855$), while for the other two observations they are neatly different.

Finally, we recall that trimmed observations have been discarded to provide robustness to the parameter estimation. After estimating the model, it thus also makes sense to classify these observations. The trimmed observations are in rows 11, 75, 93, 99, 133, 160, 163, 166, 178, 181 and, if we assign them by the Bayes rule to the component g having greater value of $D_g(\mathbf{x};\theta) = \phi_p(\mathbf{x};\boldsymbol{\mu}_g, \mathbf{\Lambda}_g\mathbf{\Lambda}_g' + \mathbf{\Psi}_g)\pi_g$, we classify the first four in the female group of athletes, and the second group of six in the male group. This means that all the trimmed observations have been assigned to their true group. Table 6 shows the details of the classification, and the rightmost panel in Figure 5 plots the final result of the robust model fitting.

As a last analysis on the AIS dataset, we are interested in factor interpretation. The rotated factor loading matrices have been obtained by employing a Gradient Projection algorithm, available through the R package *GPArotation* (Bernaards and Jennrich, 2005; Browne, 2001). We opted for an oblimin transformation, which yielded results shown

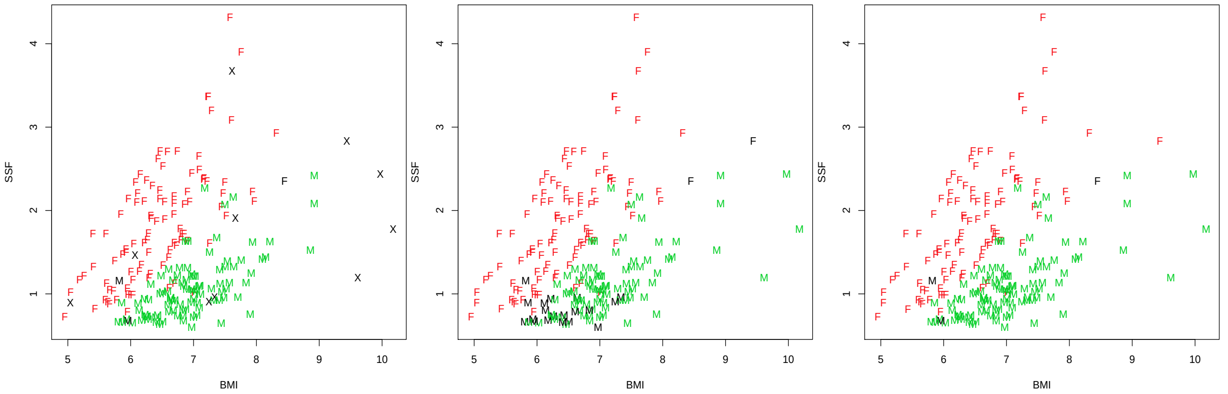

Figure 5. Classification of AIS data with fitted trimmed and constrained MFA (left panel), compared to non-robust MFA (central panel), i.e. the model in the first column of Table 4. Misclassified data are colored in black, trimmed data are denoted by “X”. In the right panel, AIS data after classifying also the trimmed observations. The three misclassified points (in black) represent only 1.5% of the data

Table 6. Trimmed units in the AIS dataset and their final classification

unit	11	75	93	99	133	160	163	166	178	181
$D_1(\mathbf{x};\theta)$	3.4e-16	9.8e-08	2.8e-06	4.8e-11	2.7e-93	6.4e-64	9.1e-89	1.3e-14	1.1e-20	1.2e-08
$D_2(\mathbf{x};\theta)$	9.6e-19	1.1e-12	4.3e-11	6.5e-87	6.2e-14	1.9e-06	7.1e-08	2.0e-12	9.5e-13	2.5e-05
Sex	F	F	F	F	M	M	M	M	M	M

in Table 7. We observe that the two groups highlight the same factors, while in a slightly different order of importance. The first factor for the group of observations for female athletes, may be labelled as a *hematological factor*, with a very high loading on Hc, followed by RCC and Hg. The second factor, loading heavily on Ht, and in a lesser extent on Wt and LBM, may be denoted as a *general nutritional status*. The third and fourth factors are related only to Fe and BMI, respectively. The fifth factor can be viewed as an *overweight assessment index*, since SSF and $Bfat$ load highly on it. The sixth factor is related only to WCC. Noticing that WCC is not joined to the *hematological factor*, we observe that the specific role of lymphocytes, cells of the immune system that are involved in defending the body against both infectious disease and foreign invaders, seems to be pointed out. Analogous comments may be done on the factor loadings for the group of male athletes.

We would like to add a final remark on this data analysis. When approaching the AIS dataset, we run a Mardia test on both groups and measured asymmetry and kurtosis, finding that both are significantly different from the Gaussian case. Hence one may argue that a mixture of two skew distributions, as in Lin et al. (2014a), is more suited for this dataset. Unfortunately, the obtained misclassification error, comparing different skew components, ranges from 4.5% to 5.9%. We want to show here that trimming is a convenient and competitive tool to be adopted, when one tail (skewness) or both tails (kurtosis) are contaminated in the data. As a general principle, trimming enables robust estimation of the location and scatter, and may offer an effective alternative to the adoption of more parameterized skew models. Our results show that we estimated the core of the data through a Gaussian density, obtaining such a good classification. We therefore argue that the flexibility obtained by the robust approach may provide a pretty good fit, even in the presence of some asymmetry in data tails.

5. Concluding remarks

We propose a robust estimation for the mixture of Gaussian factor model by adopting trimming and constrained estimation. To resist pointwise contamination and sparse outliers that could arise in data collection, we incorporate

Table 7. Factor loadings in the AIS data set

rotated $\boldsymbol{\Lambda}_1$ (female athletes)

RCC	$\boxed{0.697}$	-0.006	-0.009	-0.055	0.001	-0.035
WCC	0.000	0.009	0.000	-0.015	0.012	$\boxed{-0.941}$
Hc	$\boxed{0.794}$	-0.015	0.040	-0.004	0.010	0.026
Hg	$\boxed{0.682}$	0.021	-0.025	0.047	-0.002	0.007
Fe	0.002	-0.005	$\boxed{-0.510}$	0.003	-0.004	0.000
BMI	0.029	-0.008	0.023	$\boxed{0.644}$	-0.316	-0.057
SSF	-0.040	-0.012	-0.037	0.033	$\boxed{-0.889}$	-0.017
$Bfat$	0.014	-0.024	0.013	-0.007	$\boxed{-0.826}$	0.008
LBM	0.022	$\boxed{-0.419}$	0.020	0.295	0.054	-0.025
Ht	0.020	$\boxed{-0.924}$	0.023	-0.128	-0.076	-0.002
Wt	0.029	$\boxed{-0.468}$	0.023	0.330	-0.235	-0.031

rotated $\boldsymbol{\Lambda}_2$ (male athletes)

RCC	-0.033	$\boxed{0.663}$	0.077	-0.015	-0.025	-0.053
WCC	0.003	-0.004	0.024	-0.003	$\boxed{1.024}$	0.013
Hc	0.048	$\boxed{0.622}$	-0.008	0.005	0.036	-0.048
Hg	-0.002	$\boxed{0.604}$	-0.051	0.009	0.001	0.079
Fe	0.010	-0.006	0.027	$\boxed{1.103}$	-0.004	0.008
BMI	-0.371	0.109	$\boxed{-0.656}$	0.074	0.070	-0.261
SSF	$\boxed{-0.616}$	0.002	0.009	-0.001	0.015	-0.026
$Bfat$	$\boxed{-0.610}$	-0.009	0.003	0.007	-0.000	0.040
LBM	0.036	0.071	-0.344	0.037	0.053	$\boxed{-0.885}$
Ht	0.036	0.005	0.170	-0.022	0.009	$\boxed{-1.157}$
Wt	-0.222	0.071	-0.357	0.042	0.056	$\boxed{-0.884}$

a trimming procedure in the iterations of the EM algorithm. The key idea is that a small portion of observations, which are highly unlikely to occur under the current fitted model assumption, are discarded from contributing to the parameter estimates. Furthermore, to reduce spurious solutions and avoid singularities of the likelihood, a constrained ML estimation for the component covariances has been implemented. Results from the Monte Carlo experiments show that the bias and MSE of the estimators, in several cases of contaminated data, are comparable to results obtained on data without noise. Finally, the analysis on a real dataset illustrates that robust estimation leads to better classification and provides direct interpretation of the factor loadings.

Further investigations are needed to tune the choice of the parameters, such as the portion of trimming data and the values of the constraints. Though interesting, this issue is beyond the scope of the present paper. Surely, some data-dependent diagnostic based on trimmed BIC notions (Neykov et al., 2007) may provide a way to select the number of groups and underlying factors, as has been shown. With reference to the choice of α, other tools can be adapted to the present case, such as silhouette plots to assess the strength of cluster assignments and the classification trimmed likelihood curves (García-Escudero et al., 2011). These curves provide helpful exploratory tools by monitoring the estimation results when moving α in [0,1] and $G = 1, 2, \ldots$. On the other hand, considering an α higher than needed does not necessarily spoil trimmed-based clustering results. We could have erroneously discarded some non-outlying data points but the "main" clustering structure may still be detected, as has been documented by the simulation results. Clearly, when thinking of tuning parameters, one has also to take into account that they are interrelated. For instance, a high trimming level α could lead to smaller G values, since components with fewer observations may be trimmed off.

Moreover, larger values of the constraints could lead to higher values of G, since new components with few, almost collinear observations may arise. With respect to the choice of the specific values for the constraints, our experience tells us that a moderate interval of values for c_{load} and c_{noise} produces almost the same estimation and exactly the same classification. Also, from an interpretative point of view, this corresponds to the fact that, generally, the user has some intuition (or rough knowledge) about the order of magnitude of these constraints, and this partial information can be incorporated into the estimation. The encouraging results obtained here suggest that a deeper discussion of these implementation details could be developed as a future work.

As a final remark, following the suggestions for further investigation we received from an unknown referee, the proposed method can also be extended to accommodate missing values, as in Wang (2015); and much faster convergence to the EM-based algorithm could be improved along the lines of Zhao and Yu (2008).

Acknowledgements

The authors would like to thank two anonymous referees for many interesting and helpful comments that led to a clearer presentation of the material. This research is partially supported by the Spanish Ministerio de Ciencia e Innovación, grant MTM2014-56235-C2-1-P, by Consejería de Educación de la Junta de Castilla y León, grant VA212U13, by grant FAR 2015 from the University of Milano-Bicocca and by a grant FIR 2014 from the University of Catania.

References

Azzalini A., R package sn: The skew-normal and skew-t distributions (version 0.4-17), *URL http://azzalini. stat. unipd. it/SN*, (2011).

Azzalini A., Dalla Valle A., The multivariate skew-normal distribution, *Biometrika*, 83 (4), 715–726, (1996).

Baek J., McLachlan G., Flack L., Mixtures of Factor Analyzers with Common Factor Loadings: Applications to the Clustering and Visualization of High-Dimensional Data, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (7), 1298 –1309, (2010).

Baek J., McLachlan G., Mixtures of common t-factor analyzers for clustering high-dimensional microarray data, *Bioinformatics*, 27 (9), 1269–1276, (2011).

Bernaards C. A., Jennrich R. I. , Gradient projection algorithms and software for arbitrary rotation criteria in factor analysis, *Educational and Psychological Measurement*, 65 (5), 676–696, (2005).

Bekker P. A., ten Berge J. M. F. , Generic global identification in factor analysis, *Linear Algebra and its Applications*, 264, 255–263, (1997).

Bickel D.R., Robust cluster analysis of microarray gene expression data with the number of clusters determined biologically, *Bioinformatics*, 19 (7), 818–824, (2003).

Bishop C. M., Tipping M. E., A Hierarchical Latent Variable Model for Data Visualization, *IEEE Transactions on Pattern analysis and Machine Intelligence* , 20, 281–293, (1998).

Browne M. W., An overview of analytic rotation in exploratory factor analysis, *Multivariate Behavioral Research*, 36 (1), 111–150, (2001).

Campbell J., Fraley C., Murtagh F., Raftery A., Linear flaw detection in woven textiles using model-based clustering, *Pattern Recognition Letters*, 18 (14), 1539 – 1548, (1997).

Cook R. D., Weisberg S., *An introduction to regression graphics*, vol. 405, John Wiley & Sons, 1994.

Coretto P., Hennig C., Maximum likelihood estimation of heterogeneous mixtures of Gaussian and uniform distributions, *Journal of Statistical Planning and Inference*, 141 (1) 462–473, (2011) .

Fokoué E., Titterington D., Mixtures of factor analysers. Bayesian estimation and inference by stochastic simulation, *Machine Learning*, 50 (1-2), 73–94, (2003).

Fraley C., Raftery A. E., How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis, *Computer Journal*, 41 (8), 578–588, (1998).

Fritz H., García-Escudero L., Mayo-Iscar A., A fast algorithm for robust constrained clustering, *Computational Statistics & Data Analysis*, (61), 124–136, (2013).

Gallegos M., Ritter G., Trimmed ML estimation of contaminated mixtures, *Sankhya (Ser. A)*, (71), 164–220, (2009).

García-Escudero L. A., Gordaliza A., Matrán C., Mayo-Iscar A., A General Trimming Approach to Robust Cluster Analysis, *The Annals of Statistics*, 36 (3) 1324–1345 (2008).

García-Escudero L. A., Gordaliza A., Matrán C., Mayo-Iscar A., Exploring the Number of Groups in Robust Model-Based Clustering, *Statistics and Computing*, 21 (4), 585–599, (2011).

García-Escudero L., Gordaliza A., Mayo-Iscar A., A constrained robust proposal for mixture modeling avoiding spurious solutions, *Advances in Data Analysis and Classification*, 8 (1), 27–43, (2014).

Ghahramani Z., Hilton G., The EM algorithm for mixture of factor analyzers, *Techical Report CRG-TR-96-1*, (1997).

Greselin F., Ingrassia S., Maximum likelihood estimation in constrained parameter spaces for mixtures of factor analyzers, *Statistics and Computing*, 25, 215–226, (2015).

Hathaway R., A constrained formulation of maximum-likelihood estimation for normal mixture distributions, *The Annals of Statistics*, 13 (2), 795–800, (1985).

Hennig C., Breakdown points for maximum likelihood-estimators of location-scale mixtures, *Annals of Statistics*, 32, 1313–1340, (2004).

Ingrassia S., A likelihood-based constrained algorithm for multivariate normal mixture models, *Statistical Methods & Applications*, 13, 151–166, (2004).

Ingrassia S., Rocci R., Constrained monotone EM algorithms for finite mixture of multivariate Gaussians, *Computational Statistics & Data Analysis*, 51, 5339–5351, (2007).

Ledermann, W., On the rank of the reduced correlational matrix in multiple-factor analysis, *Psychometrika*, 2 (2), 85-93, (1937).

Lin T.-I., Ho H. J., Lee C. R., Flexible mixture modelling using the multivariate skew-t-normal distribution, *Statistics and Computing*, 24 (4), 531-546, (2014).

Lin T.-I., McNicholas P. D., Ho H. J., Capturing patterns via parsimonious t mixture models, *Statistics & Probability Letters*, 88, 80–87, (2014).

Maitra R., Clustering Massive Datasets With Application in Software Metrics and Tomography, *Technometrics*, 43 (3), 336–346, (2001).

Maitra R., Initializing partition-optimization algorithms, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 6 (1) 144–157, (2009).

McLachlan G. J., Bean R. W., Maximum likelihood estimation of mixtures of *t* factor analyzers, *Technical Report, University of Queensland*, (2005).

McLachlan G., Peel D., Robust mixture modelling using the *t* distribution, *Statistics and Computing*, 10 (4) 335–344, (2000).

McLachlan G., Peel D., Mixtures of factor analyzers, in: *Proceedings of the Seventeenth International Conference on Machine Learning*, P. Langley (Ed.)., San Francisco: Morgan Kaufmann, 599–606, (2000a).

McLachlan G. J., Peel D., *Finite Mixture Models*, John Wiley & Sons, New York, (2000b).

McNicholas P., Murphy T., Parsimonious Gaussian mixture models, *Statistics and Computing*, 18 (3), 285–296, (2008).

Neykov N., Filzmoser P., Dimova R., Neytchev P., Robust Fitting of Mixtures Using the Trimmed Likelihood Estimator, *Computational Statistics & Data Analysis*, 52 (1), 299–308, (2007).

R. D. C. Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, URL `http://www.R-project.org`, (2013).

Rousseeuw P. J., Van Driessen K., A Fast Algorithm for the Minimum Covariance Determinant Estimator, *Technometrics*, 41, 212–223, (1999).

Steane M. A., McNicholas P. D., R. Y. Yada, Model-based classification via mixtures of multivariate t-factor analyzers, *Communications in Statistics-Simulation and Computation*, 41 (4), 510–523, (2012).

Stewart C. V., Robust parameter estimation in computer vision, *SIAM review*, 41 (3), 513–537, (1999).

Tipping M. E., Bishop C. M., Mixtures of probabilistic principal component analyzers, *Neural computation*, 11 (2), 443–482, (1999).

Wang, W. L., Mixtures of common t-factor analyzers for modeling high-dimensional data with missing values, *Computational Statistics & Data Analysis*, 83, 223–235, (2015).

Zhao J. H., Yu P. L. Fast ML estimation for the mixture of factor analyzers via an ECM algorithm, *Neural Networks, IEEE Transactions on*, 19 (11), 1956–1961, (2008).