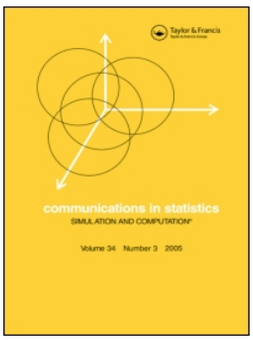

Taylor & Francis
Taylor & Francis Group

Communications in Statistics - Simulation and Computation

ISSN: 0361-0918 (Print) 1532-4141 (Online) Journal homepage: http://www.tandfonline.com/loi/lssp20

Robust fitting of mixtures of factor analyzers using the trimmed likelihood estimator

Li Yang, Sijia Xiang & Weixin Yao

To cite this article: Li Yang, Sijia Xiang & Weixin Yao (2017) Robust fitting of mixtures of factor analyzers using the trimmed likelihood estimator, Communications in Statistics - Simulation and Computation, 46:2, 1280-1291, DOI: 10.1080/03610918.2014.999088

To link to this article: https://doi.org/10.1080/03610918.2014.999088

Accepted author version posted online: 25 Mar 2015.
Published online: 09 Nov 2016.

COMMUNICATIONS IN STATISTICS—SIMULATION AND COMPUTATION®
2017, VOL. 46, NO. 2, 1280–1291
http://dx.doi.org/10.1080/03610918.2014.999088

Taylor & Francis
Taylor & Francis Group

Robust fitting of mixtures of factor analyzers using the trimmed likelihood estimator

Li Yang[a], Sijia Xiang[b], and Weixin Yao[c]

[a]Department of Statistics, Kansas State University, Manhattan, Kansas, USA; [b]School of Mathematics and Statistics, Zhejiang University of Finance and Economics, Hangzhou, China; [c]Department of Statistics, University of California, Riverside, California, USA

ABSTRACT

Mixtures of factor analyzers (MFAs) have been popularly used to cluster the high-dimensional data. However, the traditional estimation method is based on the normality assumptions of random terms and thus is sensitive to outliers. In this article, we introduce a robust estimation procedure of MFAs using the trimmed likelihood estimator. We use a simulation study and a real data application to demonstrate the robustness of the trimmed estimation procedure and compare it with the traditional normality-based maximum likelihood estimate.

ARTICLE HISTORY
Received 14 June 2014
Accepted 12 December 2014

KEYWORDS
EM algorithm; Factor analysis; Mixture models; Robustness; Trimmed likelihood estimator

MATHEMATICS SUBJECT CLASSIFICATION
Primary 62F35; Secondary 62H30

1. Introduction

Factor analysis (FA) is a statistical dimension-reduction technique for modeling the covariance structure of high-dimensional data using a small number of latent variables (Ghahramani and Hinton, 1997). It can be extended by allowing different local factor models in different regions of the input space. This results in a model that performs clustering and dimension reduction at the same time, and can be thought of as a reduced dimension mixture of Gaussians. Ghahramani and Hinton (1997) and Hinton et al. (1997) originally proposed mixtures of factor analyzers (MFAs) model. They used this model to visualize high-dimensional data in a lower-dimensional space to explore the grouping structure. Bishop (1998) and Tipping and Bishop (1997, 1999) considered the related model of mixtures of principal component analyzers for the same purpose. MFA model is in fact a nonlinear model, which can be considered as a combination of traditional FA model and the finite mixture models. Therefore, MFA model offers a way to overcome the linear limitation of the traditional FA model. In recent years, MFA model has received considerable interest. See, for example, Arminger et al. (1999), Dolan and Van der Maas (1998), Fokoué and Titterington (2003), and Yung (1997). McLachlan et al. (2003) discussed the application of MFAs to density estimation and the clustering of high-dimensional data.

MFA has been traditionally fitted using the maximum likelihood estimator (MLE) based on the normality assumptions of the random terms. Ghahramani and Hinton (1997) introduced an exact Expectation-Maximization (EM) algorithm to compute the MLE of MFA. However, it is well known that the normal-based MLE can be very sensitive to outliers. In fact, even a

CONTACT Sijia Xiang fxbxsj@live.com School of Mathematics and Statistics, Zhejiang University of Finance and Economics, Hangzhou, 310018 China.

single outlier can make an enormous impact on the MLE, which in mixture models means that at least one of the component parameter estimates might be arbitrarily large.

In this article, a robust fitting of MFAs is introduced based on the idea of trimmed likelihood estimator (TLE; Neykov et al., 2007). The TLE is designed to fit the majority of the data, whereas the remaining data will be considered as outliers and thus will not be used for parameter estimation. We use a simulation study and a real data application to demonstrate the robustness of the new estimation procedure and compare it with the traditional normality-based maximum likelihood estimate.

The rest of the article is organized as follows. In Section 2, we briefly introduce the EM algorithm for the FA and the MFA. Section 3 presents the robust fitting of the MFA using the TLE. Simulation results and a real data application are presented in Section 4. A discussion section ends the article.

2. Mixtures of factor analyzers

2.1. Factor analysis

Let $\mathbf{y}_1, \ldots, \mathbf{y}_n$ be a random sample of size n on a p-dimensional random vector. A typical FA model is given by:

$$\mathbf{y}_i = \boldsymbol{\mu} + \Lambda \mathbf{z}_i + \mathbf{e}_i, i = 1, \ldots, n, \tag{2.1}$$

where $\boldsymbol{\mu}$ is the mean of $\mathbf{y}_i$, $\mathbf{z}_i$ is a q-dimensional ($q < p$) vector of latent or unobservable variables called factors, and Λ ($p \times q$) is a factor loading matrix. The factors $\mathbf{z}_i$ are assumed to be i.i.d. $\mathcal{N}_q(\mathbf{0}, \boldsymbol{I}_q)$, independent of the errors $\mathbf{e}_i$, which are assumed to be i.i.d. $\mathcal{N}_p(\mathbf{0}, \Psi)$ with Ψ a diagonal matrix $\Psi = \text{diag}(\sigma_1^2, \ldots, \sigma_p^2)$. The marginal density of $\mathbf{y}_i$ is then $\mathcal{N}_p(\boldsymbol{\mu}, \Lambda\Lambda^T + \Psi)$. For the purpose of classifying and reducing data, the traditional FA is a useful tool for reducing a mass of information to an efficient description and grouping interdependent variables into descriptive categories. In statistics, it is a method used for explaining data, in particular, correlations between variables in multivariate observations.

The FA model (2.1) can be fitted by maximizing the log-likelihood:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left\{ (2\pi)^{p/2} \left| \Lambda\Lambda^T + \Psi \right|^{-1/2} \exp \left[-\frac{1}{2} (\mathbf{y}_i - \boldsymbol{\mu})^T (\Lambda\Lambda^T + \Psi)^{-1} (\mathbf{y}_i - \boldsymbol{\mu}) \right] \right\},$$

with $\boldsymbol{\theta} = (\boldsymbol{\mu}^T, \Lambda^T, \Psi^T)^T$, which can be computed iteratively via the EM algorithm if $\mathbf{z}_i$ is considered the missing data.

E-step: Given the current estimator $\boldsymbol{\theta}^{(k)}$, calculate the following conditional expectation given the observed data $\mathbf{y}_i$:

$$\begin{aligned}
\mathbf{a}_i^{(k)} &= E\left(\mathbf{z}_i | \mathbf{y}_i, \boldsymbol{\theta}^{(k)}\right) = \Lambda^{(k)^T} \left(\Psi^{(k)} + \Lambda^{(k)} \Lambda^{(k)^T}\right)^{-1} \mathbf{y}_i, \\
\mathbf{b}_i^{(k)} &= E\left(\mathbf{z}_i \mathbf{z}_i^T | \mathbf{y}_i, \boldsymbol{\theta}^{(k)}\right) = \boldsymbol{I} - \Lambda^{(k)^T} \left(\Psi^{(k)} + \Lambda^{(k)} \Lambda^{(k)^T}\right)^{-1} \Lambda^{(k)} \\
&\quad + \left\{ \Lambda^{(k)^T} \left(\Psi^{(k)} + \Lambda^{(k)} \Lambda^{(k)^T}\right)^{-1} \mathbf{y}_i \right\} \left\{ \Lambda^{(k)^T} \left(\Psi^{(k)} + \Lambda^{(k)} \Lambda^{(k)^T}\right)^{-1} \mathbf{y}_i \right\}^T.
\end{aligned}$$

M-step: Calculate

$$\boldsymbol{\mu}^{(k+1)} = \sum_{i=1}^{n} \left(\mathbf{y}_i - \Lambda^{(k)} \mathbf{a}_i^{(k)} \right),$$

$$\Lambda^{(k+1)} = \left\{ \sum_{i=1}^{n} \mathbf{y}_i \mathbf{a}_i^{(k)T} \right\} \left\{ \sum_{i=1}^{n} \mathbf{b}_i^{(k)} \right\}^{-1},$$

$$\Psi^{(k+1)} = \frac{1}{n} \text{diag} \left\{ \sum_{i=1}^{n} \left(\mathbf{y}_i \mathbf{y}_i^T - \Lambda^{(k+1)} \mathbf{a}_i^{(k)} \mathbf{y}_i^T \right) \right\}.$$

2.2. Mixtures of factor analyzers

Although the FA model (2.1) provides a global linear model for the presentation of the data in a lower-dimensional subspace, its application is limited when the data are not homogenous. The MFA, which allows different local factor models in different regions of the input space, is a natural extension of the FA. Assume we have a mixture of m factor analyzers with mixing proportion π_j, $j = 1, \ldots, m$. The marginal density of $\mathbf{y}$ is given by:

$$f(\mathbf{y}; \boldsymbol{\theta}) = \sum_{j=1}^{m} \pi_j \mathcal{N}_p(\mathbf{y}; \boldsymbol{\mu}_j, \Lambda_j \Lambda_j^T + \Psi), \tag{2.2}$$

where $\boldsymbol{\theta} = (\boldsymbol{\pi}^T, \boldsymbol{\mu}^T, \Lambda^T, \Psi^T)^T$, $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_{m-1})^T$, $\boldsymbol{\mu} = (\boldsymbol{\mu}_1^T, \ldots, \boldsymbol{\mu}_m^T)^T$, $\Lambda = (\Lambda_1^T, \ldots, \Lambda_m^T)^T$. Here, $\boldsymbol{\mu}_j$ is the mean of the jth component, Λ_j is the factor loading matrix of the jth component, and Ψ is the diagonal matrix of the error terms. It will be useful in the estimation equations to have a definition of the MFA in terms of conditional densities. For the jth component, the conditional density function is:

$$f_j(\mathbf{y}|\mathbf{z}) = \mathcal{N}_p(\mathbf{y}; \boldsymbol{\mu}_j + \Lambda_j \mathbf{z}, \Psi).$$

Within each component of the mixture, we have the following joint density of $\mathbf{y}$ and $\mathbf{z}$:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{z} \end{bmatrix} \sim \mathcal{N}_{p+q} \left(\begin{bmatrix} \boldsymbol{\mu}_j \\ 0 \end{bmatrix}, \begin{bmatrix} \Lambda_j \Lambda_j^T + \Psi & \Lambda_j \\ \Lambda_j^T & \boldsymbol{I}_q \end{bmatrix} \right).$$

Similar to the FA, the MFA can be estimated by maximizing the following likelihood:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{j=1}^{m} \pi_j \left[(2\pi)^{p/2} |\Lambda_j \Lambda_j^T + \Psi|^{-1/2} \right.$$
$$\left. \exp \left\{ -\frac{1}{2} (\mathbf{y}_i - \boldsymbol{\mu}_j)^T (\Lambda_j \Lambda_j^T + \Psi)^{-1} (\mathbf{y}_i - \boldsymbol{\mu}_j) \right\} \right]. \tag{2.3}$$

However, there is no explicit solution for the above maximizer. Ghahramani and Hinton (1997) introduced an EM algorithm to maximize (2.3). More specifically, let ω_{ij} be an indicator variable indicating which component $\mathbf{y}_i$ comes from. That is,

$$\omega_{ij} = \begin{cases} 1, & \text{if } \mathbf{y}_i \text{ is from } j\text{th component,} \\ 0, & \text{otherwise.} \end{cases} \tag{2.4}$$

Then the complete log-likelihood for $\{(\mathbf{y}_i, \mathbf{z}_i, \omega_{ij}), i = 1, \ldots, n, j = 1, \ldots, m\}$ is

$$\ell_c(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \prod_{j=1}^{m} \pi_j^{\omega_{ij}} \left[(2\pi)^{p/2} |\Psi|^{-1/2} \exp\left\{ -\frac{1}{2} (\mathbf{y}_i - \boldsymbol{\mu}_j - \Lambda_j \mathbf{z}_i)^T \Psi^{-1} (\mathbf{y}_i - \boldsymbol{\mu}_j - \Lambda_j \mathbf{z}_i) \right\} \right]^{\omega_{ij}}.$$

The EM algorithm iterates between E-step, which computes the expected complete log-likelihood given current parameter estimates, and M-step, which maximizes the expected complete log-likelihood calculated in the E-step. We summarize the EM algorithm to maximize (2.3) as follows:

E-step: Given the current estimator $\boldsymbol{\theta}^{(k)}$, calculate the following conditional expectation given the observed data $\mathbf{y}$:

$$E(\omega_{ij}|\mathbf{y}_i, \boldsymbol{\theta}^{(k)}) = \frac{\pi_j^{(k)} \mathcal{N}_p(\mathbf{y}_i; \boldsymbol{\mu}_j^{(k)}, \Lambda_j^{(k)} \Lambda_j^{(k)^T} + \Psi^{(k)})}{\sum_{j=1}^{m} \pi_j^{(k)} \mathcal{N}_p(\mathbf{y}_i; \boldsymbol{\mu}_j^{(k)}, \Lambda_j^{(k)} \Lambda_j^{(k)^T} + \Psi^{(k)})} = p_{ij}^{(k)},$$

$$\mathbf{a}_{ij}^{(k)} = E\left(\mathbf{z}_i | \mathbf{y}_i, \omega_{ij} = 1, \boldsymbol{\theta}^{(k)}\right) = \Gamma_j^{(k)} \left(\mathbf{y}_i - \boldsymbol{\mu}_j^{(k)}\right),$$

$$\mathbf{b}_{ij}^{(k)} = E\left(\mathbf{z}_i \mathbf{z}_i^T | \mathbf{y}_i, \omega_{ij} = 1, \boldsymbol{\theta}^{(k)}\right) = I - \Gamma_j^{(k)} \Lambda_j^{(k)} + \Gamma_j^{(k)} \left(\mathbf{y}_i - \boldsymbol{\mu}_j^{(k)}\right) \left\{ \Gamma_j^{(k)} (\mathbf{y}_i - \boldsymbol{\mu}_j^{(k)}) \right\}^T,$$

where $\Gamma_j = \Lambda_j^T (\Psi + \Lambda_j \Lambda_j^T)^{-1}$.

M-step: Calculate

$$\pi_j^{(k+1)} = \frac{1}{n} \sum_{i=1}^{n} p_{ij}^{(k)},$$

$$\boldsymbol{\mu}_j^{(k+1)} = \left\{ \sum_{i=1}^{n} p_{ij}^{(k)} (\mathbf{y}_i - \Lambda_j^{(k)} \mathbf{a}_{ij}^{(k)}) \right\} \left\{ \sum_{i=1}^{n} p_{ij}^{(k)} \right\}^{-1},$$

$$\Lambda_j^{(k+1)} = \left\{ \sum_{i=1}^{n} p_{ij}^{(k)} (\mathbf{y}_i - \boldsymbol{\mu}_j^{(k+1)}) \mathbf{a}_{ij}^{(k)^T} \right\} \left\{ \sum_{i=1}^{n} p_{ij}^{(k)} \mathbf{b}_{ij}^{(k)} \right\}^{-1},$$

$$\Psi^{(k+1)} = \frac{1}{n} \text{diag} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij}^{(k)} \left(\mathbf{y}_i - \boldsymbol{\mu}_j^{(k+1)} - \Lambda_j^{(k+1)} \mathbf{a}_{ij}^{(k)}\right) \left(\mathbf{y}_i - \boldsymbol{\mu}_j^{(k+1)}\right)^T \right\}.$$

3. Robust fitting of mixtures of factor analyzers using the trimmed likelihood estimator

The MLE introduced in Section 2 is easy to implement, but very sensitive to outliers. Even a single outlier can make an enormous impact on the MLE, and make at least one of the component parameters to be arbitrarily large. To overcome this, Andrews et al. (2011), Baek and McLachlan (2011), and McLachlan et al. (2007) proposed mixtures of t-factor analyzers by assuming multivariate t-distributions for component errors and factor distributions. In this section, we apply the idea of TLE, proposed by Neykov et al. (2007), to fit the MFAs in a robust way. Compared to the proposed method based on TLE, the mixture of t-distributions has a very small breakdown point (BP) and is not robust when the outliers are extreme (Hennig, 2004; Yao et al., 2014).

Suppose a number k ($k \leq n$) of n observations are regular observations in the data, and the remaining $n-k$ observations may be gross or outliers. The basic idea of TLE is removing the $n-k$ observations that do not follow the model, and using only the k observations to fit the model. The combinatorial nature of the TLE can be expressed as:

$$\max_{I \in I_k} \max_{\boldsymbol{\theta}} \sum_{i \in I} \log f(\mathbf{y}_i; \boldsymbol{\theta}),$$

where I_k is the set of all k-subsets of $(1, \ldots, n)$ and $f(\mathbf{y}; \boldsymbol{\theta})$ is defined in (2.2). The fact that all possible $\binom{n}{k}$ partitions of the data have to be fitted by the MLE makes the estimation procedure computationally very expensive. To find an approximate TLE solution for large datasets, an algorithm called FAST-TLE was developed by Neykov and Müller (2003). The basic idea behind FAST-TLE algorithm contains two steps: a trial step followed by a refinement step.

(i) Trial step: Randomly select a subsample of size k^* from the data sample and then fit the model to that subsample to get a trial maximum likelihood estimate.
(ii) Refinement step: This step is based on the so-called concentration procedure.
   (a) Starting with the trial maximum likelihood estimate, find a combination with the k smallest negative log-likelihoods based on the current estimate.
   (b) Obtain an improved estimator by fitting the model to these k cases.
   (c) Repeat (a) and (b) until convergence.

   At the end of this step, the solution with the largest trimmed likelihood is stored. This value may not be guaranteed to be the global optimal but would be a close approximation to it.

The choice of trial size k^* and refinement subsample size k are related to the BP. The BP (i.e., the smallest fraction of contamination that can cause the estimator to take arbitrary large values) of TLE was studied by using d-fullness technique. Vandev and Neykov (1993) determined the value of d for the mixtures of normals to be $m(p+1)$. It was proved that if $\log f(y)$ is d-full, then the BP of TLE is not less than $\frac{1}{n}\min\{n-m+1, m-d+1\}$ (Neykov and Müller, 2003). The trial subsample size k^* should be greater than or equal to d for the existence of MLE. The choice of k can be any number within $[d, n]$. When $k = \lfloor (n+d+1)/2 \rfloor$, the BP of the TLE is maximized (Neykov and Müller, 2003). If the expected percentage of outliers α in the data is a known priori, a recommended choice of k is $\lfloor n(1-\alpha) \rfloor$, which can increase the efficiency of the TLE.

The process of TLE applied particularly to the MFAs can be performed as follows:

Input: A trial subset with sample size equals to k^* and initial parameters $\boldsymbol{\theta}^{(0)} = (\boldsymbol{\pi}^{(0)T}, \boldsymbol{\mu}^{(0)T}, \Lambda^{(0)T}, \Psi^{(0)T})^T$.

Output: A subset of size k, which has the k smallest negative log-likelihoods.

At the $(l+1)$th iteration:

E-step: Compute the expectation of component indicators ω_{ij}, latent variable $\mathbf{z}$, and $\mathbf{z}\mathbf{z}^T$ based on the current subsample of size k.

M-step: Maximize the complete log-likelihood of subsample of size k with respect to each unknown parameter and thus get a new parameter

$$\boldsymbol{\theta}^{(l+1)} = \left(\boldsymbol{\pi}^{(l+1)T}, \boldsymbol{\mu}^{(l+1)T}, \Lambda^{(l+1)T}, \Psi^{(l+1)T}\right)^T.$$

T-step: Define a new subsample of size k, which has the k smallest negative log-likelihoods with the new parameter $\boldsymbol{\theta}^{(l+1)}$.

Repeat **EMT** steps until convergence.

4. Simulation study and real data application

4.1. Simulation study

In this section, we use a simulation study to assess the performance of the MLE and the TLE to the MFAs. For TLE, 20 randomly generated initial values are used and TLE reports the single best estimate whose log-likelihood is the biggest. True value (T) is also used as initial value for MLE and TLE. For the 20 initial values, we first use the R code "hc" from the R package "mclust" to cluster the randomly generated subsets of the data and then use the R code "factanal" from the R package "stats" to do FA for each cluster. The trimming proportion α is set to be 5% and thus $k = \lfloor n(1-\alpha) \rfloor$ is used for TLE in all examples. We will discuss how to choose α data adaptively in Section 5.

A two-component MFA is considered in the simulation:

$$f(\mathbf{y}) = \sum_{j=1}^{2} \pi_j \mathcal{N}_p(\mathbf{y}; \boldsymbol{\mu}_j, \Lambda_j \Lambda_j^T + \Psi),$$

where the mixing proportions are $\pi_1 = 0.4$ and $\pi_2 = 0.6$. The means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are $p \times 1$ vectors with all the elements equal to 0 and 5, respectively, and the factor loading matrices Λ_1 and Λ_2 are $p \times 2$ matrices with all the elements equal to 0.5 and 1, respectively. That is,

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}_{p\times 1}, \boldsymbol{\mu}_2 = \begin{pmatrix} 5 \\ \vdots \\ 5 \end{pmatrix}_{p\times 1},$$

$$\Lambda_1 = \begin{pmatrix} 0.5 & 0.5 \\ \vdots & \vdots \\ 0.5 & 0.5 \end{pmatrix}_{p\times 2}, \Lambda_2 = \begin{pmatrix} 1 & 1 \\ \vdots & \vdots \\ 1 & 1 \end{pmatrix}_{p\times 2}.$$

We consider $p = 10, 20$, and 30. Sample sizes of $n = 200$ and $n = 400$ are conducted over 200 repetitions. To assess the robustness of the estimators, only $(1-\alpha_0) \times 100\%$ of the observations are generated from the above model with $\alpha_0 = 0, 0.01, 0.03$, and 0.05, and the remaining $\alpha_0 \times 100\%$ of the data is generated randomly from $U(20, 30)$. The simulation was done through R on a personal laptop with an i7-3610QM CPU and 8 GB of RAM. The computation time of the new algorithm (with 20 random initial values) is 45 seconds for $n = 200$ and 61 seconds for $n = 400$.

The performance of the estimates is measured by the misclassification probability (MCP), which is defined to be the proportion of observations that are misclassified:

$$\text{MCP} = 1 - \left\{ \sum_{i=1}^{n} \sum_{j=1}^{2} \omega_{ij} I_{p_{ij}>0.5} \right\} / n,$$

where ω_{ij}, defined in (2.4), indicates which component $\mathbf{y}_i$ comes from, and p_{ij} is the classification probability calculated by

$$p_{ij} = \frac{\hat{\pi}_j \mathcal{N}_p(\mathbf{y}_i; \hat{\boldsymbol{\mu}}_j, \hat{\Lambda}_j \hat{\Lambda}_j^T + \hat{\Psi})}{\sum_{j=1}^{2} \hat{\pi}_j \mathcal{N}_p(\mathbf{y}_i; \hat{\boldsymbol{\mu}}_j, \hat{\Lambda}_j \hat{\Lambda}_j^T + \hat{\Psi})}, i = 1, \ldots, n, j = 1, 2.$$

Table 1. Average (Std) of MCP, with $n = 200$.

Dimension	Method	$\alpha_0 = 0$	$\alpha_0 = 0.01$	$\alpha_0 = 0.03$	$\alpha_0 = 0.05$
$p = 10$	MLE	0.016 (0.012)	0.117 (0.032)	0.103 (0.031)	0.089 (0.029)
	TLE(T)	0.016 (0.011)	0.017 (0.010)	0.018 (0.012)	0.017 (0.012)
	TLE(I)	0.018 (0.012)	0.019 (0.011)	0.020 (0.013)	0.020 (0.014)
$p = 20$	MLE	0.018 (0.012)	0.089 (0.030)	0.097 (0.029)	0.140 (0.029)
	TLE(T)	0.018 (0.012)	0.019 (0.013)	0.020 (0.013)	0.067 (0.010)
	TLE(I)	0.020 (0.014)	0.022 (0.015)	0.022 (0.014)	0.070 (0.013)
$p = 30$	MLE	0.151 (0.354)	0.076 (0.025)	0.105 (0.031)	0.100 (0.032)
	TLE(T)	0.151 (0.353)	0.026 (0.014)	0.033 (0.018)	0.021 (0.012)
	TLE(I)	0.145 (0.347)	0.029 (0.021)	0.040 (0.036)	0.026 (0.029)

Note that for mixture models there are well-known label switching issues (Celeux, et al., 2000; Grün and Leisch, 2009; Jasra et al., 2005; Stephens, 2000; Yao, 2012a, 2012b; Yao and Lindsay, 2009). In our simulations, the labels are found by minimizing the MCP.

Tables 1 and 2 report the means and standard deviations of MCP for $n = 200$ and 400, respectively. Based on the above tables, both TLE(T) and TLE(I) have smaller MCP than MLE for all three p values and both $n = 200$ and $n = 400$. In Tables 3 and 4, we also report the means and standard deviations of the Euclidean distance between the estimates $\hat{\pi}_1$, $\hat{\boldsymbol{\mu}}_1$, and $\hat{\boldsymbol{\mu}}_2$ and their corresponding true values based on 200 repetitions. From the tables, we can see that the TLEs with both true initial values and random initial values have better performance than the MLE when there are outliers, especially for $\boldsymbol{\mu}_2$ and π_1. The TLEs with randomly generated initial values work almost the same as those with true initial values. In addition, the TLE still works well when the trimming proportion is larger than the proportion of outliers. Furthermore, when there are no outliers ($\alpha_0 = 0$), TLE has comparable performance to the traditional MLE.

4.2. Real data application

In this example, we consider applying both MLE and TLE of the MFA to the wine data, which is available at the Machine Learning Repository of the University of California. The dataset contains the results of chemical analysis of wines grown in the same region in Italy, but derived from three different cultivars. Therefore, a three-component mixture model is suitable to fit the data if we do not use the cultivars of the wines. The analysis determined the quantities of $p = 13$ constituents found in each of $n = 178$ wines. Both MLE and TLE of the MFA were fitted to this dataset. Similar to the simulation study, the trimming proportion is set to be 0.05 for TLE.

Table 2. Average (Std) of MCP, with $n = 400$.

Dimension	Method	$\alpha_0 = 0$	$\alpha_0 = 0.01$	$\alpha_0 = 0.03$	$\alpha_0 = 0.05$
$p = 10$	MLE	0.014 (0.006)	0.125 (0.024)	0.123 (0.020)	0.130 (0.019)
	TLE(T)	0.014 (0.006)	0.025 (0.007)	0.044 (0.006)	0.064 (0.006)
	TLE(I)	0.014 (0.006)	0.026 (0.008)	0.044 (0.006)	0.064 (0.006)
$p = 20$	MLE	0.014 (0.006)	0.110 (0.021)	0.123 (0.022)	0.131 (0.019)
	TLE(T)	0.014 (0.006)	0.025 (0.007)	0.044 (0.006)	0.065 (0.007)
	TLE(I)	0.014 (0.006)	0.025 (0.007)	0.045 (0.006)	0.065 (0.007)
$p = 30$	MLE	0.016 (0.008)	0.096 (0.021)	0.124 (0.020)	0.091 (0.022)
	TLE(T)	0.016 (0.009)	0.025 (0.006)	0.047 (0.008)	0.016 (0.007)
	TLE(I)	0.016 (0.009)	0.025 (0.007)	0.047 (0.008)	0.017 (0.008)

Table 3. Average (Std) of Euclidean distance, with $n = 200$.

Dimension	Method		$\alpha_0 = 0$	$\alpha_0 = 0.01$	$\alpha_0 = 0.03$	$\alpha_0 = 0.05$
$p = 10$	MLE	μ_1:	0.023 (0.026)	0.051 (0.032)	0.042 (0.038)	0.044 (0.033)
		μ_2:	0.025 (0.034)	1.359 (0.469)	2.979 (1.505)	6.368 (0.825)
		π_1:	0.001 (0.002)	0.021 (0.012)	0.021 (0.016)	0.030 (0.016)
	TLE(T)	μ_1:	0.024 (0.020)	0.023 (0.020)	0.021 (0.021)	0.025 (0.014)
		μ_2:	0.028 (0.030)	0.030 (0.035)	0.024 (0.028)	0.032 (0.035)
		π_1:	0.001 (0.002)	0.001 (0.002)	0.002 (0.003)	0.003 (0.004)
	TLE(I)	μ_1:	0.026 (0.022)	0.025 (0.021)	0.021 (0.022)	0.030 (0.030)
		μ_2:	0.030 (0.034)	0.033 (0.038)	0.031 (0.066)	0.036 (0.038)
		π_1:	0.001 (0.002)	0.001 (0.002)	0.002 (0.003)	0.003 (0.004)
$p = 20$	MLE	μ_1:	0.022 (0.015)	0.046 (0.091)	0.042 (0.024)	0.042 (0.027)
		μ_2:	0.027 (0.029)	0.849 (0.298)	2.792 (0.479)	5.449 (0.690)
		π_1:	0.001 (0.001)	0.013 (0.009)	0.020 (0.012)	0.028 (0.014)
	TLE(T)	μ_1:	0.023 (0.015)	0.026 (0.024)	0.023 (0.018)	0.025 (0.016)
		μ_2:	0.029 (0.030)	0.036 (0.046)	0.030 (0.030)	0.031 (0.036)
		π_1:	0.001 (0.002)	0.001 (0.002)	0.002 (0.003)	0.003 (0.003)
	TLE(I)	μ_1:	0.024 (0.016)	0.029 (0.025)	0.027 (0.025)	0.029 (0.023)
		μ_2:	0.039 (0.057)	0.047 (0.068)	0.037 (0.037)	0.038 (0.040)
		π_1:	0.001 (0.002)	0.002 (0.003)	0.002 (0.003)	0.003 (0.003)
$p = 30$	MLE	μ_1:	0.004 (0.010)	0.034 (0.022)	0.040 (0.024)	0.018 (0.032)
		μ_2:	0.005 (0.021)	0.528 (0.213)	2.248 (0.392)	1.551 (2.216)
		π_1:	0.001 (0.001)	0.008 (0.008)	0.019 (0.012)	0.010 (0.016)
	TLE(T)	μ_1:	0.004 (0.009)	0.024 (0.015)	0.024 (0.014)	0.010 (0.018)
		μ_2:	0.009 (0.043)	0.027 (0.033)	0.028 (0.031)	0.008 (0.020)
		π_1:	0.001 (0.001)	0.002 (0.002)	0.002 (0.003)	0.001 (0.003)
	TLE(I)	μ_1:	0.004 (0.010)	0.047 (0.201)	0.079 (0.465)	0.044 (0.401)
		μ_2:	0.012 (0.063)	0.037 (0.048)	0.039 (0.049)	0.013 (0.036)
		π_1:	0.001 (0.001)	0.002 (0.002)	0.003 (0.007)	0.001 (0.005)

Table 4. Average (Std) of Euclidean distance, with $n = 400$.

Dimension	Method		$\alpha_0 = 0$	$\alpha_0 = 0.01$	$\alpha_0 = 0.03$	$\alpha_0 = 0.05$
$p = 10$	MLE	μ_1:	0.010 (0.007)	0.031 (0.020)	0.023 (0.015)	0.020 (0.013)
		μ_2:	0.013 (0.018)	1.566 (0.289)	3.757 (0.364)	6.630 (0.595)
		π_1:	0.001 (0.001)	0.025 (0.011)	0.026 (0.010)	0.030 (0.011)
	TLE(T)	μ_1:	0.011 (0.008)	0.012 (0.009)	0.012 (0.009)	0.012 (0.008)
		μ_2:	0.015 (0.021)	0.016 (0.017)	0.013 (0.014)	0.012 (0.012)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.001)	0.002 (0.002)
	TLE(I)	μ_1:	0.011 (0.009)	0.012 (0.009)	0.013 (0.009)	0.012 (0.009)
		μ_2:	0.016 (0.022)	0.017 (0.019)	0.015 (0.016)	0.014 (0.014)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.001)	0.002 (0.002)
$p = 20$	MLE	μ_1:	0.011 (0.006)	0.025 (0.013)	0.021 (0.012)	0.020 (0.014)
		μ_2:	0.013 (0.013)	1.056 (0.235)	2.963 (0.324)	5.713 (0.511)
		π_1:	0.001 (0.001)	0.018 (0.008)	0.024 (0.010)	0.028 (0.010)
	TLE(T)	μ_1:	0.011 (0.007)	0.011 (0.006)	0.012 (0.008)	0.012 (0.008)
		μ_2:	0.014 (0.016)	0.016 (0.016)	0.013 (0.013)	0.013 (0.015)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.001)	0.002 (0.002)
	TLE(I)	μ_1:	0.012 (0.008)	0.011 (0.006)	0.012 (0.008)	0.013 (0.014)
		μ_2:	0.016 (0.020)	0.018 (0.017)	0.014 (0.014)	0.015 (0.016)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.001)	0.002 (0.002)
$p = 30$	MLE	μ_1:	0.011 (0.008)	0.021 (0.013)	0.022 (0.014)	0.016 (0.014)
		μ_2:	0.014 (0.015)	0.715 (0.171)	2.503 (0.316)	3.616 (2.238)
		π_1:	0.001 (0.001)	0.013 (0.008)	0.022 (0.010)	0.021 (0.016)
	TLE(T)	μ_1:	0.012 (0.009)	0.011 (0.007)	0.012 (0.007)	0.009 (0.008)
		μ_2:	0.018 (0.024)	0.014 (0.013)	0.017 (0.019)	0.009 (0.011)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.002)	0.001 (0.002)
	TLE(I)	μ_1:	0.013 (0.009)	0.012 (0.008)	0.012 (0.007)	0.009 (0.008)
		μ_2:	0.019 (0.023)	0.016 (0.015)	0.019 (0.019)	0.010 (0.013)
		π_1:	0.001 (0.001)	0.001 (0.001)	0.001 (0.002)	0.001 (0.002)

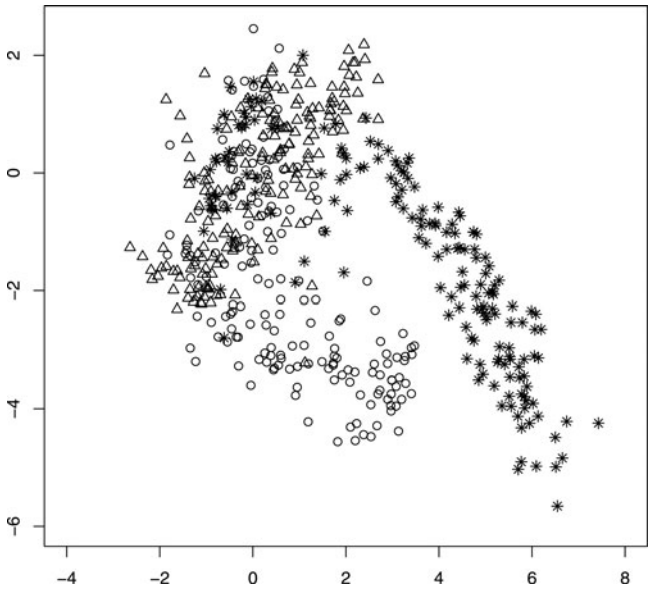

Figure 1. Wine data: plot of the estimated posterior means of the $q = 2$ factors (△, ○, and ∗ denote true component membership).

Based on McLachlan and Peel (2000), the misclassification rate is smallest for $q = 2$ and 3. In our analysis, $q = 2$ is used as our reduced dimension. Figure 1 shows the estimated posterior means of the $q = 2$ factors following a three-component MFA of the wine data, which is actually the $\mathbf{a}_{ij}$ calculated from E-step. These posterior means have been plotted with their true group labels corresponding to the three different cultivars displayed. From Fig. 1, we can see that MFAs have been useful here in exploring the grouping structure of the data in a much reduced dimension.

To assess the robustness of the two estimation methods, we also consider the contaminated data by adding 1% and 3% outliers from $U(9, 11)$. Table 5 displays the estimated means $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$, and $\boldsymbol{\mu}_3$ via MLE and TLE when the proportion of outliers are $\alpha_0 = 0$, 0.01, and 0.03, and Table 6 displays the estimated component proportions π_1 and π_2. The true parameter values are calculated by using true classification labels based on the cultivars of the wines. From both tables, we see that when there are no outliers ($\alpha_0 = 0$), both MLE and TLE can provide comparatively good estimators. When the data are contaminated, however, TLE performs much better than MLE. As the proportion of outliers gets higher, MLE departs further away from the original MLE, while TLE does not change much when the outliers are added to the data.

5. Discussion

MFAs have been popularly used to do dimension reduction and model-based clustering for high-dimensional data. In this article, we investigate a robust estimation procedure of the MFAs based on the TLE proposed by Neykov et al. (2007). The simulation study and real data analysis demonstrated the effectiveness of the TLE-based robust estimation procedure.

It is well know that the scale estimate by TLE is biased for univariate data. A scale factor is usually needed to make the scale estimate an unbiased consistent estimator. Based on our limited empirical experience, the TLE-based covariance estimates for MFAs are also biased.

Table 5. Wine data: estimated means with $\alpha_0 = 0$, 0.01, and 0.03.

		$\alpha_0 = 0$		$\alpha_0 = 0.01$		$\alpha_0 = 0.03$	
	True	MLE	TLE	MLE	TLE	MLE	TLE
μ_1	13.74	13.66	13.74	13.44	13.74	12.34	13.73
	2.01	1.99	2.01	1.61	2.02	0.21	1.99
	2.46	2.47	2.46	2.09	2.46	0.79	2.43
	17.04	17.49	17.05	16.42	17.18	15.77	17.01
	106.34	107.87	106.30	105.67	106.04	105.95	105.34
	2.84	2.85	2.84	2.50	2.84	1.29	2.84
	2.98	3.00	2.98	2.69	2.98	2.11	2.96
	0.29	0.29	0.29	− 0.03	0.29	− 1.25	0.28
	1.90	1.92	1.90	1.53	1.90	0.66	1.87
	5.53	5.44	5.52	5.29	5.53	7.09	5.50
	1.06	1.07	1.06	0.71	1.06	− 0.40	1.06
	3.16	3.16	3.16	2.78	3.14	1.53	3.14
	1115.71	1097.23	1114.12	1144.08	1115.45	1284.31	1115.80
μ_2	12.28	12.28	12.30	12.34	12.32	12.92	12.30
	1.93	1.95	1.96	1.98	1.95	1.97	1.97
	2.24	2.22	2.25	2.26	2.24	2.33	2.24
	20.24	19.96	20.26	20.21	20.09	18.88	20.08
	94.55	91.86	90.09	94.98	90.07	99.06	91.30
	2.26	2.23	2.23	2.30	2.24	2.51	2.24
	2.08	2.04	2.06	2.14	2.05	2.48	2.07
	0.36	0.37	0.38	0.37	0.37	0.33	0.38
	1.63	1.60	1.55	1.64	1.53	1.75	1.59
	3.09	3.05	3.07	3.17	3.07	4.11	3.06
	1.06	1.05	1.06	1.05	1.05	1.06	1.05
	2.79	2.77	2.79	2.82	2.78	2.95	2.78
	519.51	502.67	496.14	534.54	496.23	777.10	498.36
μ_3	13.15	13.12	13.13	13.12	13.12	13.11	13.12
	3.33	3.31	3.37	3.30	3.30	3.27	3.29
	2.44	2.44	2.43	2.44	2.44	2.43	2.44
	21.42	21.42	21.34	21.42	21.41	21.33	21.41
	99.31	100.03	99.35	100.03	100.04	100.02	100.05
	1.68	1.68	1.65	1.68	1.67	1.68	1.67
	0.78	0.79	0.77	0.79	0.79	0.80	0.79
	0.45	0.44	0.45	0.44	0.44	0.44	0.44
	1.15	1.16	1.12	1.16	1.16	1.15	1.16
	7.40	7.29	7.27	7.28	7.27	7.25	7.25
	0.68	0.69	0.69	0.69	0.69	0.69	0.69
	1.68	1.70	1.68	1.70	1.70	1.69	1.70
	629.90	630.27	629.56	630.53	631.24	627.43	632.32

However, it requires more theoretical studies whether a scale or vector factor could make the TLE-based covariance estimator unbiased and consistent.

In our examples, we have fixed the trimming proportion to be 0.05 for TLE. It works well whenever the true proportions of outliers are no more than 5%. However, it requires more research to find a data adaptive optimal or conservative trimming proportion for TLE in

Table 6. Wine data: estimated component proportions with $\alpha_0 = 0$, 0.01, and 0.03.

		$\alpha_0 = 0$		$\alpha_0 = 0.01$		$\alpha_0 = 0.03$	
	True	MLE	TLE	MLE	TLE	MLE	TLE
π_1	0.3315	0.3516	0.3516	0.3049	0.3386	0.0331	0.3201
π_2	0.3989	0.3726	0.3726	0.4190	0.3697	0.6853	0.3869

practice. Neykov et al. (2007) recommended a graphical tool to choose the trimming proportion in their examples. However, based on our limited empirical experience, such graphical tool was not very successful in choosing the trimming proportion for MFAs. There have been many methods proposed for choosing the trimming proportion for TLE in the nonmixture context. For example, Jurećková et al. (1994) studied the problem of choosing the trimming proportion for a trimmed L-estimator of location, and recommended the L-estimators with smooth weight functions. For the trimmed mean in the location modeling and for the trimmed least-squares estimator in the linear regression model, Dodge and Jurećková (1997) proposed a partially adaptive estimator of the trimming proportion based on a rank-based decision procedure. Clark and Schubert (2010) studied an adaptive TLE of regression, whose algorithm tends to expose the outliers automatically and provide the estimators with the outliers removed. It will be interesting to know whether we can extend the foregoing methods to adaptively choose the trimming proportion for TLE in the mixture context.

References

Andrews, J. L., McNicholas, P. D., Subedi, S. (2011). Model-based classification via mixtures of multivariate *t*-distribution. *Computational Statistics & Data Analysis* 55(1):520–529.

Arminger, G., Stein, P., Wittenberg, J. (1999). Mixtures of conditional mean and covariance structure models. *Psychometrika* 65:475–494.

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

Bishop, C. M. (1998). Latent variable models. In: Jordan, M. I., ed. *Learning in Graphical Models*. Dordrecht: Kluwer Academic, pp. 371–403.

Clark, B. R., Schubert, D. (2010). Adaptive trimmed likelihood estimation in regression. *Probability and Statistics* 30:203–219.

Celeux, G., Hurn, M., Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association* 95:957–970.

Dodge, Y., Jurećková, J. (1997). Adaptive choice of proportion in trimmed least-squares estimation. *Statistics & Probability Letters* 33(2):167–176.

Dolan, C., Van der Maas, H. (1998). Fitting multivariate normal finite mixtures subject to structural equation modelling. *Psychometrika* 63:227–253.

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

Ghahramani, Z., Hinton, G. E. (1997). *The EM Algorithm for Mixture of Factor Analyzers*. Technical Report, CRG-TR-96. Toronto, ON: University of Toronto.

Grün, B., Leisch, F. (2009). Dealing with label switching in mixture models under genuine multimodality. *Journal of Multivariate Analysis* 100:851–861.

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

Hinton, G. E., Dayan, P., Revow, M. (1997). Modeling the manifolds of images of handwritten digits. *IEEE Transaction on Neural Networks* 8:65–73.

Jasra, A., Holmes, C. C., Stephens, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science* 20:50–67.

Jurećková, J., Koenker, R., Welsh, A. H. (1994). Adaptive choice of trimming proportions. *Annals of the Institute of Statistical Mathematics* 46(4):737–755.

McLachlan, G. J., Bean, R. W., Ben-Tovim Jones, L. (2007). Extension of the mixture of factor analyzers model to incorporate the multivariate *t*-distribution. *Computational Statistics & Data Analysis* 51:5327–5338.

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

McLachlan, G. J., Peel, D., Bean, R. W. (2003). Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis* 41:379–388.

Neykov, N., Filzmoser, P., Dimova, R., Neytchev, P. (2007). Robust fitting of mixtures using the trimmed likelihood estimator. *Computational Statistics & Data Analysis* 52:299–308.

Neykov, N. M., Müller, C. H. (2003). Breakdown point and computation of trimmed likelihood estimators in generalized linear models. In: Dutter, R., Filzmoser, P., Gather, U., Rousseeuw, P., eds. *Developments in Robust Statistics*. Berlin: Springer, pp. 277–286.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B* 62:795–809.

Tipping, M. E., Bishop, C. M. (1997). *Mixtures of Probabilistic Principal Component Analysers*. Technical Report, No. NCRG/97/003. Birmingham: Neural Computing Research Group, Aston University.

Tipping, M. E., Bishop, C. M. (1999). Mixtures of probabilistic principal component analysers. *Neural Computation* 11:443–482.

Vandev, D. L., Neykov, N. M. (1993). Robust maximum likelihood in the Gaussian case. In: Morgenthaler, S., Ronchetti, E., Stahel, W. A., eds. *New Directions in Data Analysis and Robustness*. Basel: Birkhauser, pp. 257–264.

Yao, W. (2012a). Model based labeling for mixture models. *Statistics and Computing* 22:337–347.

Yao, W. (2012b). Bayesian mixture labeling and clustering. *Communications in Statistics - Theory and Methods* 41:403–421.

Yao, W., Lindsay, B. G. (2009). Bayesian mixture labeling by highest posterior density. *Journal of American Statistical Association* 104:758–767.

Yao, W., Wei, Y., Yu, C. (2014). Robust mixture regression using T-distribution. *Computational Statistics and Data Analysis* 71:116–127.

Yung, Y. F. (1997). Finite mixtures in confirmatory factor analysis models. *Psychometrika* 62:297–330.