Statistics and Computing (2020) 30:255–277
https://doi.org/10.1007/s11222-019-09881-1

Weighted likelihood mixture modeling and model-based clustering

Luca Greco[1] · **Claudio Agostinelli**[2]

Received: 8 October 2018 / Accepted: 4 June 2019 / Published online: 10 June 2019

Abstract
A weighted likelihood approach for robust fitting of a mixture of multivariate Gaussian components is developed in this work. Two approaches have been proposed that are driven by a suitable modification of the standard EM and CEM algorithms, respectively. In both techniques, the M-step is enhanced by the computation of weights aimed at downweighting outliers. The weights are based on Pearson residuals stemming from robust Mahalanobis-type distances. Formal rules for robust clustering and outlier detection can be also defined based on the fitted mixture model. The behavior of the proposed methodologies has been investigated by numerical studies and real data examples in terms of both fitting and classification accuracy and outlier detection.

Keywords Classification · EM · Mixture · Multivariate normal · Outlier detection · Pearson residuals · Robustness · Weighted likelihood

Mathematics Subject Classification 62F35 · 62G35 · 62H25 · 62H30

1 Introduction

Multivariate normal mixture models represent a very popular tool for both density estimation and clustering (McLachlan and Peel 41). The parameters of a mixture model are commonly estimated by maximum likelihood by resorting to the EM algorithm (Dempster et al. 19). Let $y = (y_1, y_2, \ldots, y_n)^\top$ be a random sample of size n. The mixture likelihood can be expressed as

$$L(y; \tau) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \phi_p(y_i; \mu_k, \Sigma_k) \,, \tag{1}$$

where $\tau = (\pi, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$, $\phi_p(\cdot; \cdot)$ is the p-dimensional multivariate normal density, $\pi = (\pi_1, \ldots, \pi_K)$ denotes the vector of prior membership probabilities and (μ_k, Σ_k) are the mean vector and variance-covariance matrix of the kth component, respectively. Rather than using the likelihood in (1), the EM algorithm works with the complete likelihood function

$$L^c(y; \tau) = \prod_{i=1}^{n} \prod_{k=1}^{K} \left[\pi_k \phi_p(y_i; \mu_k, \Sigma_k)\right]^{u_{ik}} \,, \tag{2}$$

where u_{ik} is an indicator of the ith unit belonging to the kth component. The EM algorithm iteratively alternates between two steps: expectation (E) and maximization (M). In the E-step, the posterior expectation of (2) is evaluated by setting u_{ik} equal to the posterior probability that y_i belongs to the kth component, i.e.

$$u_{ik} \propto \pi_k \phi_p(y_i; \mu_k, \Sigma_k) \,,$$

whereas at the M-step π, μ_k and Σ_k are estimated conditionally on u_{ik}.

An alternative strategy is given by the penalized classification EM (CEM) algorithm (Symon 46; Bryant 9; Celeux and Govaert 11): the substantial difference is that the E-step is followed by a C-step (where C stands for classification) in which u_{ik} is estimated as either 0 or 1, meaning that each unit is assigned to the most likely component, conditionally on the current parameters' values, i.e. $k_i = \text{argmax}_k u_{ik}$, $u_{ik_i} = 1$ and $u_{ik} = 0$ for $k \neq k_i$. The classification approach is aimed

✉ Luca Greco
luca.greco@unisannio.it

Claudio Agostinelli
claudio.agostinelli@unitn.it

[1] DEMM Department, University of Sannio, Benevento, Italy

[2] Department of Mathematics, University of Trento, Trento, Italy

at maximizing the corresponding classification likelihood (2) over both the mixture parameters and the individual components' labels. In the case $\pi_k = 1/K$, then the standard CEM algorithm is recovered. A detailed comparison of the EM and CEM algorithms can be found in [11].

When the sample data are prone to contamination and several unexpected outliers occur with respect to (w.r.t.) the assumed mixture model, maximum likelihood is likely to lead to unrealistic estimates and to fail in recovering the underlying clustering structure of the data (see Farcomeni and Greco 23, for a recent account). In the presence of noisy data that depart from the underlying mixture model, there is the need to replace maximum likelihood with a suitable robust procedure, leading to estimates and clustering rules that are not badly affected by contamination.

The need for robust tools in the estimation of mixture models has been first addressed in [10], who suggested to replace standard maximum likelihood with M-estimation. In a more general fashion, [24] proposed to resort to multivariate S-estimation of location and scatter in the M-step. Actually, the authors focused their attention on hidden Markov models, but their approach can be adapted from dynamic to static finite mixtures. According to such strategies, each data point is attached a weight lying in $[0, 1]$ (a strategy commonly addressed as soft trimming). An alternative approach to robust fitting and clustering is based on hard trimming procedures, i.e. a crispy weight $\{0, 1\}$ is attached to each observation: atypical observations are expected to be trimmed, and the model is fitted by using a subset of the original data. The `tclust` methodology (Garcia-Escudero et al. 29; Fritz et al. 28) is particularly appealing: model parameters are estimated by developing a penalized CEM algorithm augmented with an impartial trimming step. Very recent extensions have been discussed in [21], who proposed a reweighted trimming procedure (`rtclust`) and [20], in which trimming has been introduced in parsimonious model-based clustering (`mtclust`). A related proposal has been presented in [43] based on the so-called trimmed likelihood methodology. Furthermore, it is worth to mention that mixture model estimation and clustering can be also implemented by using the adaptive hard trimming strategy characterizing the Forward Search (Atkinson et al. 6).

There are also different proposals aimed at being robust that are not based on soft or hard trimming procedures. Some of them are characterized by the use of flexible components in the mixture. The idea is that of embedding the Gaussian mixture in a supermodel: [41] introduced a mixture of Student's t distributions, a mixture of skewed Student's t distributions has been proposed in [37] and [36], whereas [25;26] considered an additional component modeled as a Poisson process to handle noisy data (the method is available from package `mclust` (Fraley et al. 27) in `R` (R Core Team 44). A robust approach, named `otrimle`, has been proposed recently by [16;17], who considered the addition of an improper uniform mixture component to accommodate outliers.

We propose a robust version of both the EM and the penalized CEM algorithms to fit a mixture of multivariate Gaussian components based on soft trimming, in which weights are evaluated according to the weighted likelihood methodology (Markatou et al. 39). A first attempt in this direction has been pursued by [38]. Here, that approach has been developed further and made more general leading to a newly established technique, in which weights are based on the recent results stated in [5]. The methodology leads to a robust fit and is aimed at providing both cluster assignment of genuine data points and outlier detection rules. Data points flagged as anomalous are not meant to be classified into any of the clusters. Furthermore, a relevant aspect of our proposal is represented by the introduction of constraints, not considered in [38], aimed at avoiding local or spurious solutions (Fritz et al. 28).

Some necessary preliminaries on weighted likelihood estimation are given in Sect. 2. The weighted EM and penalized CEM algorithms are introduced in Sect. 3: some computational details are discussed concerning constraints, initialization issues, the tuning of the methods and classification and outlier detection rules are outlined. Section 4 states asymptotic results, whereas Sect. 5 is devoted to model selection. Numerical studies are presented in Sect. 6, and real data examples are discussed in Sect. 7.

2 Background

Let us assume a mixture model composed by K heterogeneous multivariate Gaussian components, where K is fixed in advance, with density function denoted by $m(y;\tau) = \sum_{j=1}^{K} \pi_j \phi_p(y_i;\mu_j,\Sigma_j)$. [38] suggested to work with the following weighted likelihood estimating equation (WLEE) in the M-step of the EM algorithm:

$$\sum_{i=1}^{n} w_i \sum_{j=1}^{k} u_{ij} \frac{\partial}{\partial \nu} \left[\log \pi_j + \log \phi_p(y_i;\mu_j,\Sigma_j)\right] = 0\,. \quad (3)$$

We notice that maximum likelihood equations are replaced by weighted equations. The weights are defined as

$$w_i = w(\delta(y_i)) = \frac{[A(\delta(y_i)) + 1]^+}{\delta(y_i) + 1}\,, \quad (4)$$

where $[\cdot]^+$ denotes the positive part, $\delta(y)$ is the Pearson residual function and $A(\delta)$ is the residual adjustment function (RAF, Basu and Lindsay 7). The Pearson residual gives a measure of the agreement between the assumed model $m(y;\tau)$ and the data that are summarized by a nonparametric

density estimate $\hat{m}_n(y) = n^{-1}\sum_{i=1}^{n} k(y; y_i, h)$, based on a kernel $k(y; t, h)$ indexed by a bandwidth h, that is

$$\delta(y) = \frac{\hat{m}_n(y)}{m(y;\tau)} - 1\,, \tag{5}$$

with $\delta \in [-1, \infty)$. In the construction of Pearson residuals, [38] suggested to use a smoothed model density in the continuous case, by using the same kernel involved in nonparametric density estimation (see Basu and Lindsay 7; Markatou et al. 39, for general results), i.e.

$$m^*(y;\tau) = \int k(y; t, h) m(t;\tau)\mathrm{d}t.$$

When the model is correctly specified, the Pearson residual function (5) evaluated at the true parameter value converges almost surely to zero, whereas, otherwise, for each value of the parameters, large Pearson residuals detect regions where the observation is unlikely to occur under the assumed model. The weight function (4) can be chosen to be unimodal so that it declines smoothly as the residual $\delta(y)$ departs from zero. Hence, those observations lying in such regions are attached a weight that decreases with increasing Pearson residual. Large Pearson residuals and small weights will correspond to data points that are likely to be outliers. The RAF plays the role to bound the effect of large residuals on the fitting procedure, as well as the Huber and Tukey bisquare function bound large distances in M-estimation and we assume is such that $|A(\delta)| < |\delta|$. Here, we consider the families of RAF based on the Power Divergence Measure

$$A_{pdm}(\delta) = \begin{cases} \nu\left((\delta+1)^{1/\nu} - 1\right) & \nu < \infty \\ \log(\delta+1) & \nu \to \infty \end{cases}$$

Special cases are maximum likelihood ($\nu = 1$, as the weights become all equal to one), Hellinger distance ($\nu = 2$), Kullback–Leibler divergence ($\nu \to \infty$) and Neyman's Chi-square ($\nu = -1$). Another example is given by the generalized Kullback–Leibler divergence (GKL) defined as

$$A_{gkl}(\delta) = \log(\nu\delta + 1)/\nu,\ 0 \le \nu \le 1.$$

Maximum likelihood is a special case when $\nu \to 0$ and Kullback–Leibler divergence is obtained for $\nu = 1$.

The shape of the kernel function has a very limited effect on weighted likelihood estimation. On the contrary, the smoothing parameter h directly affects the robustness/efficiency trade-off of the methodology in finite samples. Actually, large values of h lead to Pearson residuals all close to zero and weights all close to one and, hence, large efficiency, since the kernel density estimate is stochastically close to the postulated (smoothed) model. On the other hand, small values of h make the kernel density estimate more sensitive to the occurrence of outliers and the Pearson residuals become large for those data points that are in disagreement with the model. In other words, in finite samples more smoothing will lead to higher efficiency but larger bias under contamination.

2.1 Multivariate estimation

The computation of weights based on the Pearson residuals given in (5) becomes troublesome with growing dimensions since the data are more sparse and multivariate kernel density estimation may become unfeasible. In order to circumvent this *curse of dimensionality*, [5] proposed a novel technique which is based on the Mahalanobis distances

$$d = d(y;\mu,\Sigma) = [(y-\mu)^\top \Sigma^{-1}(y-\mu)]^{1/2}\,.$$

Then, Pearson residuals can be evaluated by comparing a univariate kernel density estimate based on squared distances and their underlying χ^2_p distribution at the assumed multivariate normal model, rather than working with multivariate data and multivariate kernel density estimates, that is

$$\delta(y) = \frac{\hat{m}_n\left(d^2\right)}{m_{\chi^2_p}\left(d^2\right)} - 1\,, \tag{6}$$

where

$$\hat{m}_n(t) = n^{-1}\sum_{i=1}^{n} k(t; d^2, h)$$

is an unbiased at the boundary univariate kernel density estimate over $(0, \infty)$ and $m_{\chi^2_p}(t)$ denotes the χ^2_p density function. It is worth noting that Pearson residuals can be evaluated w.r.t. the original χ^2_p density, so avoiding model smoothing (see also Kuchibhotla and Basu 34, 35). Assumptions and proofs concerning existence, convergence and asymptotic normality of the WLE of multivariate location and scatter have been also established (see the Supplementary material of Agostinelli and Greco 5).

3 Weighted likelihood mixture modeling

The technique for weighted likelihood mixture modeling proposed by [38] exhibits the same limitations that have been highlighted in [5] in the case of weighted likelihood estimation of multivariate location and scatter. The main drawbacks are driven by the employ of multivariate kernels.

The availability of consistent estimators of multivariate location and scatter based on the Pearson residuals (6) is the starting point to build a weighted likelihood methodology to

fit robustly the mixture model (5) that is also capable to handle situations in which the number of features is large enough. Therefore, by exploiting the approach developed in [5], we propose both a weighted EM algorithm and a weighted penalized CEM algorithm whose M-steps are characterized by a WLEE based on the Pearson residuals (6).

It is worth to notice that the method is expected to work in large dimensions, even if it is still over-parameterized in high-dimensional spaces. The technique is meant for and confined to the $n > p$ case and to dimensions that still allow evaluation of Mahalanobis distances. The development of weighted likelihood methodologies for model-based clustering in very large dimensions and in the $n < p$ situation is beyond the scope of the present work.

The weighted EM algorithm (WEM) is structured as follows:

1. *Initialization*

$$\tau^{(0)} = (\pi^{(0)}, \mu_1^{(0)}, \ldots, \mu_K^{(0)}, \Sigma_1^{(0)}, \ldots, \Sigma_1^{(K)}) .$$

Details on the sensitivity of the results to different initializations and the selection of the best solution will be given in Sect. 3.5.

2. *E-step* the standard E-step is left unchanged, with

$$u_{ik}^{(s)} = \frac{\pi_k^{(s-1)} \phi_p \left(y_i; \mu_k^{(s-1)}, \Sigma_k^{(s-1)} \right)}{\sum_{k=1}^{K} \pi_k^{(s-1)} \phi_p \left(y_i; \mu_k^{(s-1)}, \Sigma_k^{(s-1)} \right)}$$

3. *Weighted M-step* based on current parameter estimates,

(a) *Soft trimming* let us evaluate component-wise Mahalanobis-type distances

$$d_{ik}^{(s)} = d \left(y_i; \mu_k^{(s-1)}, \Sigma_k^{(s-1)} \right) .$$

Then, for each group, compute Pearson residuals and weights as

$$\delta_{ik}^{(s)} = \frac{\hat{m}_n \left(d_{ik}^{(s)^2} \right)}{m_{\chi_p^2} \left(d_{ik}^{(s)^2} \right)} - 1$$

and

$$w_{ik}^{(s)} = \frac{\left[A \left(\delta_{ik}^{(s)} \right) + 1 \right]^+}{\delta_{ik}^{(s)} + 1}$$

respectively.

(b) *Update membership probabilities and component-specific parameter estimates*

$$\pi_k^{(s+1)} = \frac{\sum_{i=1}^{n} u_{ik}^{(s)} w_{ik}^{(s)}}{\sum_{i=1}^{n} \sum_{k=1}^{K} u_{ik}^{(s)} w_{ik}^{(s)}}$$

$$\mu_k^{(s+1)} = \frac{\sum_{i=1}^{n} y_i w_{ik}^{(s)} u_{ik}^{(s)}}{\sum_{i=1}^{n} w_{ik}^{(s)} u_{ik}^{(s)}}$$

$$\Sigma_k^{(s+1)} = \frac{\sum_{i=1}^{n} \left(y_i - \mu_k^{(s+1)} \right) \left(y_i - \mu_k^{(s+1)} \right)^\top w_{ik}^{(s)} u_{ik}^{(s)}}{\sum_{i=1}^{n} w_{ik}^{(s)} u_{ik}^{(s)}}$$

(c) *Set* $\tau^{(s+1)} = \left(\pi^{(s+1)}, \mu_1^{(s+1)}, \ldots, \mu_K^{(s+1)}, \Sigma_1^{(s+1)}, \ldots, \Sigma_K^{(s+1)} \right)$.

It is worth noting that at the M-step it is proposed to solve the following WLEE

$$\sum_{i=1}^{n} \sum_{j=1}^{k} u_{ij} \frac{\partial}{\partial \tau} \left[\log \pi_j + \log \phi_p (y_i; \mu_j, \Sigma_j) \right] w_{ij} = 0, \quad (7)$$

that is characterized by the evaluation of K component-wise sets of weights, rather than one weight for each observation, as in equation (3).

The weighted penalized CEM algorithm (WCEM) is obtained by introducing a standard C-step between the E-step and the weighted M-step. The main feature of the WCEM algorithm is that one single weight is attached to each unit, based on its current assignment after the C-step, rather than component-wise weights. Then, the resulting WLEE shows the same structure as in (3) but with the difference that $u_{ij} = 1$ or $u_{ij} = 0$. The WCEM is described as follows:

1. *Initialization*

$$\tau^{(0)} = (\pi^{(0)}, \mu_1^{(0)}, \ldots, \mu_K^{(0)}, \Sigma_1^{(0)}, \ldots, \Sigma_K^{(0)}) .$$

2. *E-step*

$$u_{ik}^{(s)} = \frac{\pi_k^{(s-1)} \phi_p \left(y_i; \mu_k^{(s-1)}, \Sigma_k^{(s-1)} \right)}{\sum_{k=1}^{K} \pi_k^{(s-1)} \phi_p \left(y_i; \mu_k^{(s-1)}, \Sigma_k^{(s-1)} \right)}$$

3. *C-step* let $k_i^{(s)} = \text{argmax}_k u_{ik}^{(s)}$ identify the cluster assignment for the ith unit at the sth iteration. Then

$$\tilde{u}_{ik}^{(s)} = \begin{cases} 1 & \text{if } k = k_i, \\ 0 & \text{if } k \neq k_i. \end{cases}$$

4. *Weighted M-step* based on current parameter estimates $\tau^{(s)}$ and cluster assignments k_i,

(a) *Soft trimming* evaluate the Mahalanobis-type distances of each point w.r.t. the component it belongs in

$$d_{ik_i}^{(s)} = d\left(y_i; \mu_{k_i}^{(s-1)}, \Sigma_{k_i}^{(s-1)}\right).$$

Then, compute the corresponding Pearson residuals and weights as

$$\delta_{ik_i}^{(s)} = \frac{\hat{m}_n\left(d_{ik_i}^{(s)^2}\right)}{m_{\chi_p^2}\left(d_{ik_i}^{(s)^2}\right)} - 1$$

and

$$w_i^{(s)} = w_{ik_i}^{(s)} = \frac{\left[A\left(\delta_{ik_i}^{(s)}\right) + 1\right]^+}{\delta_{ik_i}^{(s)} + 1}$$

respectively, where

$$\begin{aligned}\hat{m}_n(d^2) &= \frac{1}{\sum_{i=1}^n \tilde{u}_{ik_i}} \sum_{i=1}^n k(d^2; d_{ik_i}^2, h)\,, \\ &= \frac{1}{\sum_{i=1}^n \tilde{u}_{ik}} \sum_{i=1}^n k(d^2; d_{ik}^2, h)\tilde{u}_{ik}\,.\end{aligned}$$

Hence, component-wise kernel density estimates only involve distances conditionally on cluster assignment.

(b) *Update membership probabilities and component-specific parameter estimates*

$$\begin{aligned}\pi_k^{(s+1)} &= \frac{\sum_{i=1}^n \tilde{u}_{ik}^{(s)} w_{ik_i}^{(s)}}{\sum_{i=1}^n w_{ik_i}^{(s)}}\,, \\ \mu_k^{(s+1)} &= \frac{\sum_{i=1}^n y_i w_{ik_i}^{(s)} \tilde{u}_{ik}^{(s)}}{\sum_{i=1}^n w_{ik_i}^{(s)} \tilde{u}_{ik}^{(s)}}\,, \\ \Sigma_k^{(s+1)} &= \frac{\sum_{i=1}^n \left(y_i - \mu_k^{(s+1)}\right)\left(y_i - \mu_k^{(s+1)}\right)^\top w_{ik_i}^{(s)} \tilde{u}_{ik}^{(s)}}{\sum_{i=1}^n w_{ik_i}^{(s)} \tilde{u}_{ik}^{(s)}}.\end{aligned}$$

(c) *Set* $\tau^{(s+1)} = \left(\pi^{(s+1)}, \mu_1^{(s+1)}, \ldots, \mu_K^{(s+1)}, \Sigma_1^{(s+1)}, \ldots, \Sigma_K^{(s+1)}\right)$

It is worth noting that both weighted algorithms return weighted estimates of covariance. The final output can be suitably modified in order to provide unbiased weighted estimates.

3.1 Eigen-ratio constraint

It is well known that maximization of the mixture likelihood (1) or the classification likelihood (2) is an ill-posed problem since the objective function may be unbounded (Day 18; Maronna and Jacovkis 40). Therefore, in order to avoid such problems, the optimization is performed under suitable constraints. In particular, we employed the eigen-ratio constraint defined as

$$\frac{\max_j \max_k \lambda_j(\Sigma_k)}{\min_j \min_k \lambda_j(\Sigma_k)} \leq c, \qquad \begin{matrix} j = 1, 2 \ldots, p, \\ k = 1, 2, \ldots, K \end{matrix} \tag{8}$$

where $\lambda_j(\Sigma_k)$ denoted the jth eigenvalue of the covariance matrix Σ_k and c is a fixed constant not smaller than one aimed at tuning the strength of the constraint. For $c = 1$ spherical clusters are imposed, while as c increases varying shaped clusters are allowed. The eigen-ratio constraint (8) can be satisfied at each iteration by adjusting the eigenvalues of each $\Sigma_k^{(s)}$. This is achieved by replacing them with a truncated version

$$\lambda_j^*(\Sigma_k) = \begin{cases} c & \text{if } \lambda_j(\Sigma_k) < c \\ \lambda_j(\Sigma_k) & \text{if } c \leq \lambda_j(\Sigma_k) \leq c\theta_c \\ c\theta_c & \text{if } \lambda_j(\Sigma_k) > c\theta_c \end{cases}$$

where θ_c is an unknown bound depending on c. The reader is pointed to [28;31] for a feasible solution to the problem of finding θ_c.

3.2 Classification and outlier detection

The WCEM automatically provides a classification of the sample units, since the value of $\tilde{u}_{ik}$ at convergence is either zero or one. With the WEM, by paralleling a common approach, a maximum a posteriori criterion can be used for cluster assignment, that is, a C-step is applied after the last E-step. Such criteria lead to classify all the observations, both genuine and contaminated data, meaning that also outliers are assigned to a cluster. Actually, we are not interested in classifying outliers and for purely clustering purposes outliers have to be discarded.

We distinguish two main approaches to outlier detection. According to the first, outlier detection should be based on the robust fitted model and performed separately by using formal rules. The key ingredients in multivariate outlier detection are the robust distances (Rousseeuw and Van Zomeren 45; Cerioli 12). The reader is pointed to [13] for a recent account on outlier detection. An observation is flagged as an outlier when its squared robust distance exceeds a fixed threshold, corresponding to the $(1 - \alpha)$-level quantile of the reference (asymptotic) distribution of the squared robust distances. A

common solution is represented by the use of the χ^2_p, and popular choices are $\alpha = 0.025$ and $\alpha = 0.01$. In the case of finite mixtures, the main idea is that the outlyingness of each data point should be measured conditionally on the final assignment. Hence, according to a proper testing strategy, an observation is declared as outlying when

$$d^2_{ik_i} > \chi^2_{p;1-\alpha}\ ,\ d^2_{ik_i} = (y_i - \hat{\mu}_{k_i})^\top \hat{\Sigma}_{k_i}(y_i - \hat{\mu}_{k_i})\ . \tag{9}$$

The second approach stems from hard trimming procedures, such as `tclust`, `rtclust` and `otrimle`. These techniques are not meant to provide simultaneous robust fit and outlier detection based on formal testing rules, but outliers are identified with those data points falling in the trimmed set or assigned to the improper density component, respectively. Therefore, by paralleling what happens with hard trimming, one could flag as outliers those data points whose weight, conditionally on the final cluster assignment, is below a fixed (small) threshold. Values as 0.10 or 0.20 seem reasonable choices. Furthermore, the empirical downweighting level represents a natural upper bound for the cutoff value that would give an indication of the largest tolerable swamping and of the minimum feasible masking for the given level of smoothing. This approach is motivated by the fact that the multivariate WLE shares important features with hard trimming procedures, even if it is based on soft trimming, as claimed in [4].

The process of outlier detection may result in type I and type II errors. In the former case, a genuine observation is wrongly flagged as outlier (swamping); in the latter case, a true outlier is not identified (masking). Swamped genuine observations are false positives, whereas masked outliers are false negatives. According to the first strategy, the larger α the more swamping and the less masking. In a similar fashion, the higher the threshold the more swamping and the less masking will characterize the second approach to outlier detection.

In the following, both approaches to outlier detection will be taken into account and critically compared.

3.3 The selection of *h*

The selection of h is a crucial task. According to authors' experience (see Agostinelli and Greco 5,4; Greco 32, for instance), but also as already suggested by [39], a safe selection of h can be achieved by monitoring the empirical downweighting level $(1 - \hat{\bar{\omega}})$ as h varies, with $\hat{\bar{\omega}} = n^{-1}\sum_{i=1}^n \hat{w}_i$, where the weights at convergence $\hat{w}_i = \hat{w}_{ik_i}$ are evaluated at the fitted parameter value and conditionally on the final cluster assignment, both for WEM and WCEM, along the lines outlined in Sect. 3.2. The monitoring of WLE analyses has been applied successfully in [4] to the case of robust estimation of multivariate location and scatter. The reader is pointed to [14] for an account on the benefits of monitoring. A good strategy in the tuning of the smoothing parameter would be to monitor several quantities of interest stemming from the fitted mixture model in addition to the empirical downweighting level. One could monitor the weighted log-likelihood at convergence, unit-specific robust distances conditionally on the final cluster assignment, unit-specific weights, a misclassification error if a training set with known labels is available. For instance, an abrupt change in the monitored empirical downweighting level or in the robust distances may indicate the transition from a robust to a non-robust fit and aid in the selection of a value of h that gives an appropriate compromise between efficiency and robustness. Values beyond this threshold would lead to at least one arbitrarily biased fitted component that can compromise the accuracy of clustering. It is worth to note that, the trimming level in `tclust` or the improper density constant in `otrimle` is selected in a monitoring fashion, as well.

3.4 Synthetic data

Let us consider a three component mixture model with $\pi = (0.2, 0.3, 0.5)$, $\mu_1 = (-5, 0)^\top$, $\mu_2 = (0, -5)^\top$, $\mu_3 = (5, 0)^\top$ and

$$\Sigma_1 = \begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix},\quad \Sigma_2 = \begin{pmatrix} 2 & 1.25 \\ 1.25 & 2 \end{pmatrix},$$
$$\Sigma_3 = \begin{pmatrix} 3 & -1.75 \\ -1.75 & 3 \end{pmatrix}.$$

and a simulated sample of size $n = 1000$, with 40% of background noise. Outliers have been generated uniformly within an hypercube whose dimensions include the range of the data and are such that the distance to the closest component is larger than the 0.99-level quantile of a χ^2_2 distribution. WEM and WCEM have been run by setting the eigen-ratio restriction constant to $c = 15$. (The true value is 9.5.) The weights are based on the generalized Kullback–Leibler divergence and a folded normal kernel. Initialization has been provided by running `tclust` with a 50% level of trimming. The smoothing parameter h has been selected by monitoring the empirical downweighting level and unit-specific clustering-conditioned distances over a grid of h values (Agostinelli and Greco 4). Figure 1 displays the monitoring analyses of the empirical downweighting level, the robust distances and the misclassification error for the WEM. In all panels an abrupt change is detected, meaning that for h values on the right side of the vertical line the procedure is no more able to identify the outliers, hence being not robust w.r.t. the presence of contamination. Similar trajectories are observed for the WCEM and not reported here. In the monitoring of robust distances, a color map has been used that goes from light gray to dark gray in order to highlight those trajectories corresponding to observations that are flagged as outlying for

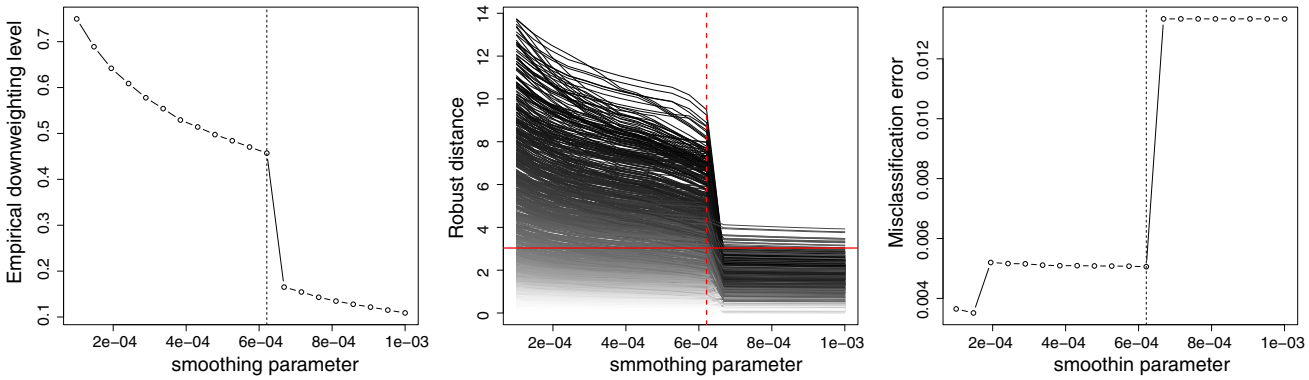

Fig. 1 Simulated data. Monitoring the empirical downweighting level (left), robust distances (middle), misclassification error (right) based on WEM. The vertical lines give the selected h. The horizontal line in the middle panel gives the $\chi^2_{2;0.99}$ quantile

Table 1 Simulated data

	ϵ	swamp.	mask.
WEM			
$\alpha = 0.010$	0.388	0.013	0.050
$\alpha = 0.025$	**0.408**	0.027	0.020
$\hat{w} < 0.1$	0.353	**0.003**	0.123
$\hat{w} < 0.2$	0.386	0.013	0.055
$\hat{w} < 1 - \hat{\tilde{w}}$	0.429	0.052	**0.005**
WCEM			
$\alpha = 0.010$	0.378	0.017	0.080
$\alpha = 0.025$	**0.408**	0.027	**0.020**
$\hat{w} < 0.1$	0.333	**0.003**	0.173
$\hat{w} < 0.2$	0.366	0.017	0.105
$\hat{w} < 1 - \hat{\tilde{w}}$	0.417	0.038	**0.015**

Outlier detection from different rules for WEM and WCEM. Boldface indicates the best performance

most of the monitoring. Figure 2 displays the result of applying both the WEM and WCEM algorithm to the sample at hand with an outlier detection rule based on the 0.99-level quantile of the χ^2_2 distribution and on a threshold for weights set at 0.2. Component-specific tolerance ellipses are based on the 0.95-level quantile of the χ^2_2 distribution. We notice that both methods succeed in recovering the underlying structure of the clean data despite the challenging contamination rate and that the outliers detection rules provide quite similar and satisfactory outcomes. The entries in Table 1 give the rate of detected outliers ϵ, swamping and masking stemming from the alternative strategies.

3.5 Sensitivity to initialization and root selection

In order to initialize WEM and WCEM, `tclust` based on a large rate of trimming is an appealing solution. Other candidate solutions can be used to initialize the algorithm. For instance, one approach has been discussed in [17] that is based on a combination of nearest neighbor denoising and agglomerative hierarchical clustering. Given the initial partition, starting values for component-specific parameters are obtained by the sample mean and covariance matrix of the points belonging to each cluster. Here, we also consider a safer strategy that is based on the evaluation of clusterwise robust estimates (for instance, by using the OGK estimator), since the initial denoising may still include dangerous outliers, especially with a large rate of contamination.

In order to check the extent to which results change by varying the initialization, we ran a numerical study based on 500 Monte Carlo trials according to the data configuration of the example in Sect. 3.4. In each trial, for a fixed h, the WEM starts iterating from `tclust` (with 50% trimming, `tclust50`), the initial values from [17] (`InitClust`) and its robust counterpart described above (`InitClustOGK`). The same numerical study has been also performed when the level of contamination is set to null, 10% and 20%. In the former scenario, with 40% of noisy points, `InitClust` leads to one inflated estimate of covariance and a smaller empirical downweighting level. On the contrary, the other two starting values give solutions with negligible differences in parameters estimates, depending on the chosen stopping rule and tolerance (here the stopping rule is based on the absolute value of the maximum difference between consecutive estimates of the centroids matrix and the tolerance is 10^{-4}) and the same final classification and detected outliers. In the latter cases, the algorithm was less dependent on the initial values in the sense that the all three alternatives led to practically indistinguishable fitted models.

When, for several initial values and a fixed value of the bandwidth h, the algorithm ends with different estimates, but still characterized by close empirical downweighting levels, by paralleling the classical approach, one could select the solution leading to the largest value of the weighted likelihood. The likelihood evaluated at the WLE, in general,

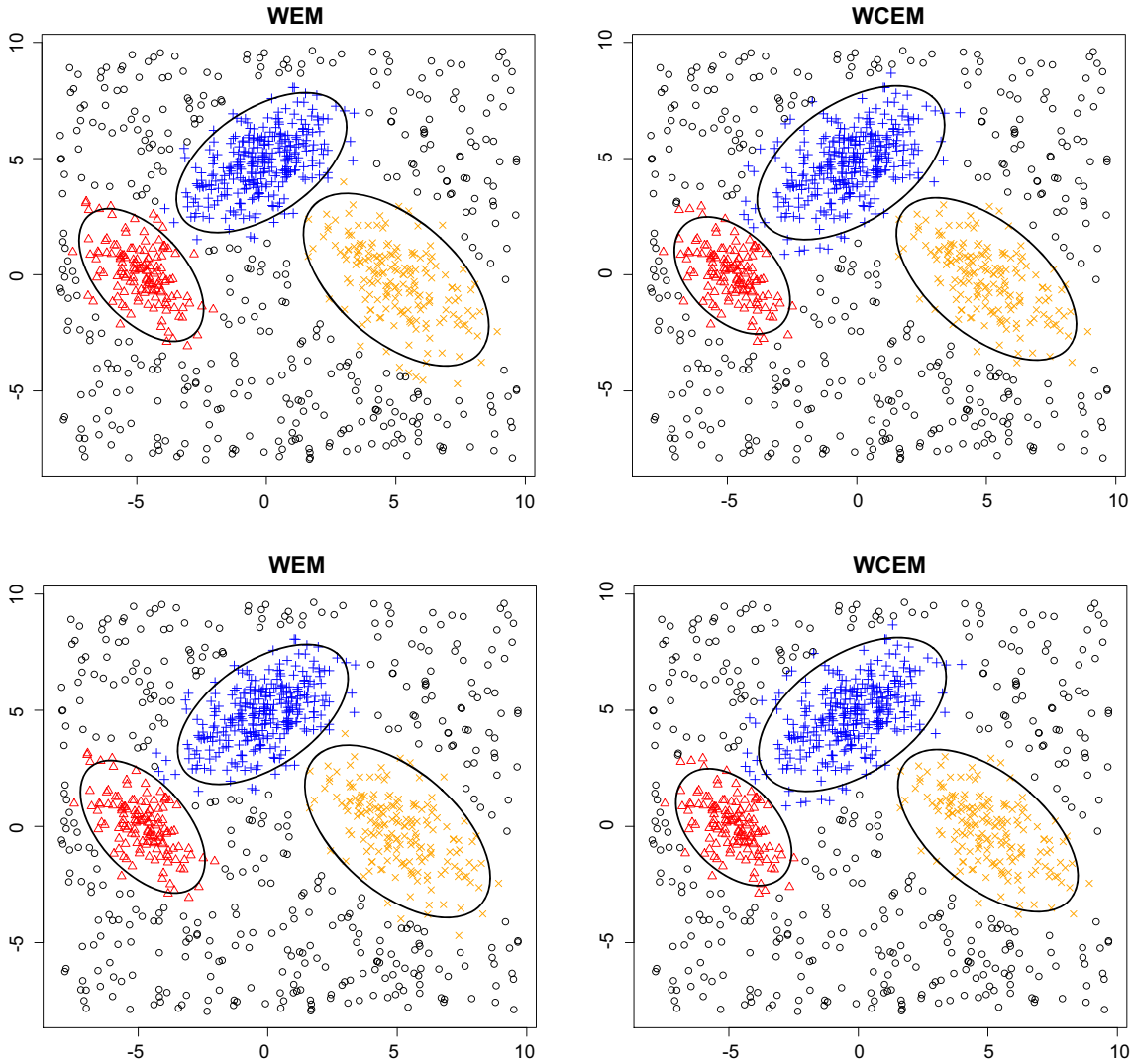

Fig. 2 Simulated data. Fitted components, cluster assignments and outlier detection by WEM (left) and WCEM (right). Top row: outlier detection based on $d^2_{k_i} < \chi^2_{p;0.99}$. Bottom row: outlier detection based on $w_{k_i} < 0.2$, 95% tolerance ellipses overimposed

could be misleading, as also discussed in Sect. 5. In the numerical studies, the solutions characterized by a smaller empirical downweighting level for $\epsilon = 40\%$ showed lower weighted likelihood values at convergence but larger likelihoods. At least in this example, `tclust50` provides the largest value of the weighted likelihood on average but the three initializations leads to the largest weighted likelihood value with almost equal frequencies, but for $\epsilon = 40\%$, where `tclust50` gives the selected root slightly more often than `InitClustOGK`. Figure 3 gives the distributions of the weighted likelihood values at convergence for the case $\epsilon = 20\%$. We do not observe significant differences. These findings give evidence supporting the convergence of the proposed algorithm. Similar results are also valid for the WCEM.

A simple and common strategy to check the stability of the results is to run the algorithm for a number (say, 20 to 50) of starting values. For instance, different initial solutions can be obtained by randomly perturbing the deterministic starting solution and/or the final one obtained from it (Farcomeni and Greco 24).

The empirical downweighting level provides one guidance to assess the reliability of the fitted model. Actually, if the sum of the estimated weights is approximately 1 then the WLE is close to the MLE, whereas if it is too small, then the corresponding WLE is a degenerate solution, indicating that it only represents a small subset of the data. In the case of excess of downweighting, the criterion based on the weighted likelihood can fail.

A strategy to root selection in the WLE framework has been introduced by [2] and extended to the multivariate framework in [5]. The main idea is that the probability of observing a very small value of the Pearson residual is

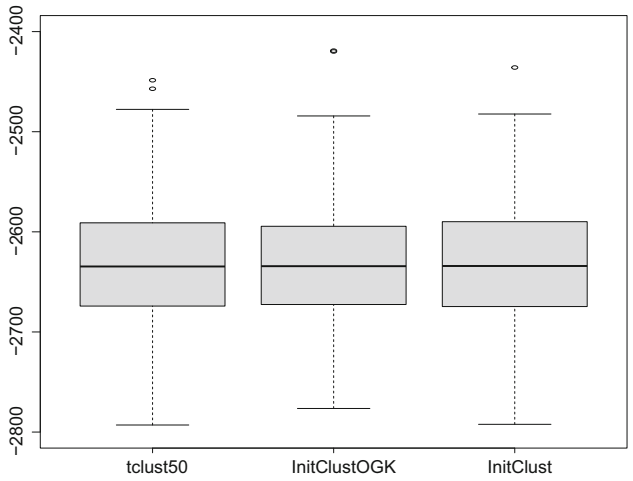

Fig. 3 Distribution of weighted likelihood values at convergence when the WEM algorithm is initialized by `tclust50`, `InitClustOGK` and `InitClust`, with $\epsilon = 20\%$ and the data configuration of example in Sect. 3.4

expected to be as small as the fitted model is close to the model underlying the majority of the data. Then, the selected root is that with the lowest fitted probability

$$\Pr_{\hat{\tau}}\left[\delta(\hat{d}^2; \hat{\tau}, \hat{M}_n) < -q\right], \tag{10}$$

with $q = 0.9, 0.95$. The probability in (10) is obtained by drawing a large number of instances (say $N = 10,000$) from the fitted model. In the example of Sect. 3.4, the criterion in (10) correctly discards the biased root with the lowest weighted likelihood value, whereas it is not able to discriminate between the other two set of estimates, since they are very close and essentially lead to the same results.

4 Properties

The WEM and WCEM algorithms have been obtained by introducing a different set of estimating equations that defines a WLEE as in (7) or (3), in place of the likelihood equations. In particular, in the M-step K separate WLE of multivariate location and scatter are obtained. The proposed algorithms are a special case of the algorithm first introduced by [22], where an EM-type algorithm has been established for very general estimating equations. Here, solving the WLEE, for each separate problem, corresponds to solve a complete data estimating equation of the form

$$\Psi(y; \tau) = (\Psi_\pi(y; \tau), \Psi_\mu(y; \tau), \Psi_\Sigma(y; \tau))^\top = 0 \tag{11}$$

Very general conditions for consistency and asymptotic normality of the solution to (11) are given in [22]. The main requirements are that

1. $\Psi(y; \tau)$ defines an unbiased estimating equation at the assumed model, i.e. $E_\tau[\Psi(Y; x, \tau)] = 0$;
2. $E_\tau[\Psi(Y; x, \tau)\Psi(Y; x, \tau)^\top]$ exists and is positive definite;
3. $E_\tau[\partial\Psi(Y; x, \tau)/\partial\tau]$ exists and is negative definite, $\forall\tau$.

This conditions are satisfied by the proposed WLEE that are characterized by weighted score functions stemming from (2). The reader is pointed to the Supplementary material in [5] for detailed assumptions and proofs.

5 Model selection

In model-based clustering, formal approaches to choose the number of components are based on the value of the log-likelihood function at convergence. Criteria such as the BIC or the AIC are commonly used to select K when running the classical EM algorithm. In a robust setting, in `tclust` the number of clusters is chosen by monitoring the changes in the trimmed classification likelihood over different values of K and contamination levels. A formal approach has not been investigated yet in the case of the `otrimle`, even if the authors conjecture that a monitoring approach or the development of information criteria can be pursued as well.

Here, when the robust fit is achieved by the WEM algorithm, we suggest to resort to a weighted counterparts of the classical AIC or BIC criteria. Then, the proposed strategy is based on minimizing

$$Q^w(K) = -2\ell^w(y; \hat{\tau}) + m(K) \tag{12}$$

where $\ell^w(y; \hat{\tau}) = \sum_{i=1}^n \hat{w}_{ik_i}\ell(y_i; \tau)$ and $m(K)$ is a penalty term reflecting model complexity. The rationale behind the use of a weighted criterion is that we want to implement a model selection device leading to results close to those one would obtain by using the standard criteria on the genuine part of the data only. Actually, if one uses the standard criteria based on the log-likelihood function evaluated at the WLE, outliers still contribute to its value and, even if these individual contributions are the smallest, the overall behavior of the corresponding BIC and AIC may be badly affected. Let us consider the set of simulated data of Sect. 3.4. The two panels in Figure 4 display, respectively, the behavior of the weighted BIC and the classical BIC evaluated at the WLE for different choices of K over a grid of values for the smoothing parameter h. The unpleasant behavior of the BIC is evident from the inspection of the right panel of Figure 4. On the contrary, the weighted BIC, shown in the left panel of Figure 4, allows selection of the correct clustering complexity. We notice that a similar trajectory is observed for both $K = 2$ and $K = 3$. The abrupt change is detected at the same value of h but the

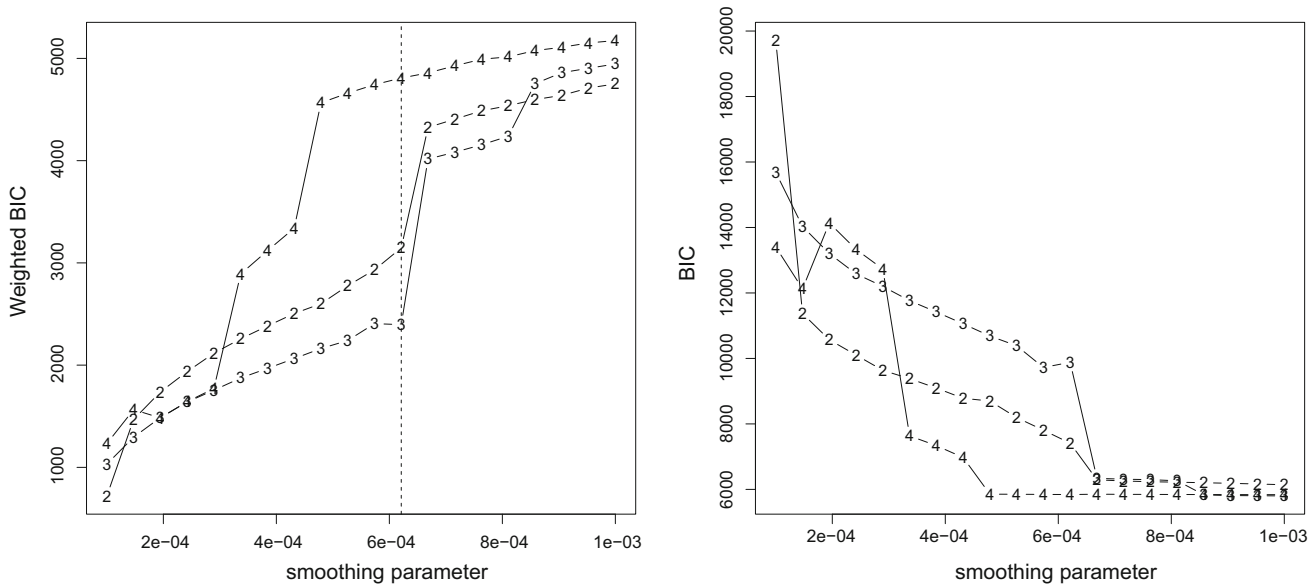

Fig. 4 Example 1. Monitoring the weighted BIC (left) and the classical BIC evaluated at the WLE (right). The vertical line in the left panel gives the selected h

choice $K = 3$ is preferred since it leads to a smaller weighted BIC before the robust fit turns into a non-robust one.

It is well known that the BIC approximate the twice log-Bayes factor for model comparison. One could extend the same relationship to the weighted BIC (12) and the weighted Bayes factor defined in [3]. Furthermore, for what concerns the WCEM algorithm, one could mimic the approach used in `tclust` and monitor the weighted conditional likelihood at convergence for varying K and h. Then, the number of clusters should be set equal to the minimum K for which there is no substantial improvement in the objective function when adding one group more.

It can be proved that the robust criterion in (12) is asymptotically equivalent to its classical counterparts at the assumed model, i.e. when the data are not prone to contamination. The proof is based on some regularity conditions about the kernel and the model that are required to assess the asymptotic behavior of the WLE (Agostinelli 1; Agostinelli and Greco 3,5). In the case of finite mixture models, it is assumed further that an ideal clustering setting holds under the postulated mixture model, that is, data are assumed to be well clustered. The following result holds.

Proposition *Let $\mathcal{Y}_j$ be the set of points belonging to the jth component, whose cardinality is n_j. The full data is defined as $\cup_{j=1}^{K}\mathcal{Y}_j$ with $\sum_{j=1}^{K} n_j = n$ and $\lim_{n_j\to\infty}\frac{n_j}{n} = 0$. Assume that (i) the model is correctly specified, (ii) the WLE $\hat{\tau}$ is a consistent estimator of τ, (iii) $\sup_{y\in\mathcal{Y}_j}|w(\delta(y)) - 1| \xrightarrow{p} 0$. Then, $|Q^w(k) - Q(k)| \xrightarrow{p} 0$.*

Proof Let $\tilde{\tau}$ denote the maximum likelihood estimate.

$$\begin{aligned}
\frac{1}{2}|Q^w(k) - Q(k)| &= \left|\sum_i w_i \ell(y_i;\hat{\tau}) - \sum_i \ell(y_i;\tilde{\tau})\right| \\
&\le \left\{\left|\sum_i (w_i - 1)\ell(y_i;\hat{\tau})\right|\right\} \\
&\quad + \left\{\left|\sum_i \left[\ell(y_i;\hat{\tau}) - \ell(y_i;\tilde{\tau})\right]\right|\right\} \\
&\le \sum_i |(w_i - 1)\ell(y_i;\hat{\tau})| \\
&\le \sup_y |w_i - 1| \sum_i \ell(y_i;\hat{\tau}) \\
&\xrightarrow{p} 0 \text{as } n_j \to \infty
\end{aligned}$$

□

6 Numerical studies

We investigate the finite sample behavior of the proposed WEM and WCEM algorithms. Both algorithms are still based on non-optimized `R` code. Nevertheless, the results that follow are satisfactory and computational time always lays in a feasible range. We set $n = 1000$, $K = 3$ and simulate data according to the *M5 scheme* as introduced in [29]. Clusters have been generated by p-variate Gaussian distributions with parameters

Table 2 Average measures of fitting accuracy for WEM, WCEM, EM, CEM and `otrimle` with $p = 2, 5, 10, 25$, $\epsilon = 0$, $\beta = 10, 8, 6$

p	$\beta = 10$			$\beta = 8$			$\beta = 6$		
	μ	Σ	π	μ	Σ	π	μ	Σ	π
WEM									
2	0.600	0.186	0.022	0.641	0.203	0.023	0.8550	0.248	0.034
5	0.651	0.802	0.025	0.767	0.822	0.035	0.867	0.886	0.032
10	0.664	1.158	0.022	0.703	1.183	0.024	0.921	1.254	0.036
25	0.778	1.807	0.027	1.191	1.881	0.044	3.039	2.075	0.131
WCEM									
2	0.614	0.193	0.030	0.687	0.208	0.029	1.273	0.261	0.056
5	0.653	0.795	0.033	0.767	0.822	0.035	1.422	0.904	0.064
10	0.681	1.156	0.028	0.829	1.194	0.031	1.715	1.305	0.074
25	0.867	1.816	0.030	2.242	1.941	0.074	6.543	2.336	0.165
EM									
2	0.551	0.154	0.022	0.597	0.169	0.023	0.765	0.184	0.029
5	0.622	0.764	0.025	0.651	0.713	0.025	0.816	0.832	0.029
10	0.659	1.113	0.022	0.697	1.137	0.023	0.933	1.214	0.033
25	0.773	1.772	0.025	1.126	1.833	0.038	3.387	2.045	0.140
CEM									
2	0.603	0.160	0.022	0.803	0.182	0.030	1.760	0.274	0.056
5	0.650	0.770	0.028	0.834	0.790	0.033	1.698	0.874	0.073
10	0.687	1.122	0.022	0.875	1.158	0.030	1.915	1.270	0.081
25	0.894	1.784	0.028	2.548	1.923	0.084	7.226	2.376	0.141
`otrimle`									
2	0.594	0.170	0.053	0.622	0.187	0.047	0.773	0.204	0.050
5	0.636	0.770	0.028	0.666	0.787	0.031	0.834	0.851	0.039
10	0.660	1.117	0.023	0.699	1.141	0.024	0.914	1.215	0.033
25	0.772	1.774	0.025	1.185	1.838	0.040	3.343	2.037	0.141

$\mu_1 = (-\beta, -\beta, 0, \ldots, 0)$,
$\mu_2 = (0, \beta, 0, \ldots, 0)$,
$\mu_3 = (\beta, 0, 0, \ldots, 0)$

and

$$\Sigma_1 = \begin{pmatrix} 15 & -10 & 0_{p-2} \\ -10 & 15 & 0_{p-2} \\ 0_{p-2}^\top & 0_{p-2}^\top & \mathrm{I}_{p-2} \end{pmatrix},$$

$$\Sigma_2 = \mathrm{I}_p,\ \Sigma_3 = \begin{pmatrix} 45 & 0 & 0_{p-2} \\ 0 & 30 & 0_{p-2} \\ 0_{p-2}^\top & 0_{p-2}^\top & \mathrm{I}_{p-2} \end{pmatrix},$$

where 0_d is a null row vector of dimension d and I_d is the $d \times d$ identity matrix. Dimensions $p = 2, 5, 10, 25$ have been taken into account. The parameter β regulates the degree of overlapping among clusters: smaller values yield severe overlapping whereas larger values give a better separation. Here, we set $\beta = 6, 8, 10$. Theoretical cluster weights are fixed as $\pi = (0.2, 0.4, 0.4)$. Outliers have been generated uniformly within an hypercube whose dimensions include the range of the data and are such that the distance to the closest component is larger than the 0.99-level quantile of a χ^2_p distribution. When $p = 25$, outliers only occur in the first ten dimensions. This setting is more challenging and allows to assess the quality of the proposed model-based clustering techniques in larger-dimensional problems. The rate of contamination has been set to $\epsilon = 0.10, 0.20$. The case $\epsilon = 0$ has been used to evaluate the efficiency of the proposed techniques when applied to clean data. The numerical studies are based on 500 Monte Carlo trials. The weighted likelihood algorithms are both based on a folded normal kernel and a GKL RAF (with $\tau = 0.9$), whereas we set $c = 50$ as eigen-ratio constraint. The smoothing parameter h has been selected in such a way that the empirical downweighting level lies in the range $(0.2, 0.35)$ under contamination, whereas it is about 10% when no outliers occur. The algorithm is assumed to reach convergence when $\max |\hat{\mu}^{(s+1)} - \hat{\mu}^{(s)}| < tol$, with a tolerance tol set to 10^{-4}, where $\hat{\mu}^{(s)}$ is the matrix of centroids estimates at the sth iteration and the differences are elementwise.

Table 3 Swamping rate for WEM, WCEM and `otrimle` with $p = 2, 5, 10, 25$, $\epsilon = 0$, $\beta = 10, 8, 6$

p	$\beta = 10$	$\beta = 8$	$\beta = 6$
WEM $\alpha = 0.01$			
2	0.024	0.021	0.017
5	0.018	0.017	0.016
10	0.018	0.017	0.017
25	0.020	0.020	0.020
WCEM $\alpha = 0.01$			
2	0.024	0.020	0.020
5	0.017	0.016	0.017
10	0.017	0.017	0.017
25	0.019	0.019	0.018
WEM $\hat{w} < 0.1$			
2	0.007	0.005	0.005
5	0.004	0.004	0.004
10	0.004	0.004	0.003
25	0.005	0.005	0.005
WCEM $\hat{w} < 0.1$			
2	0.007	0.005	0.005
5	0.004	0.003	0.004
10	0.003	0.003	0.003
25	0.004	0.005	0.005
WEM $\hat{w} < 0.2$			
2	0.016	0.012	0.010
5	0.010	0.009	0.008
10	0.009	0.008	0.008
25	0.010	0.010	0.010
WCEM $\hat{w} < 0.2$			
2	0.016	0.011	0.011
5	0.008	0.008	0.008
10	0.007	0.007	0.008
25	0.008	0.009	0.009
`otrimle`			
2	0.038	0.028	0.031
5	0.014	0.007	0.005
10	0.009	0.002	0.002
25	0.001	0.001	0.001

Fitting accuracy has been evaluated according to the following measures:

1. $||\hat{\mu} - \mu||$, where $\hat{\mu}$ and μ are $3 \times p$ matrices with $\hat{\mu}_j$ and μ_j in each row, respectively, for $j = 1, 2, 3$;
2. $\text{ave}_j \log \text{cond}\left(\hat{\Sigma}_j \Sigma_j^{-1}\right)$, where $\text{cond}(A)$ denotes the condition number of the matrix A;
3. $||\hat{\pi} - \pi||$.

Table 4 Average measures of classification accuracy for WEM, WCEM, EM, CEM and `otrimle` with $p = 2, 5, 10, 25$, $\epsilon = 0$, $\beta = 10, 8, 6$

p	$\beta = 10$		$\beta = 8$		$\beta = 6$	
	Rand	MCE	Rand	MCE	Rand	MCE
WEM						
2	0.98	0.01	0.93	0.02	0.83	0.06
5	0.98	0.01	0.93	0.02	0.83	0.06
10	0.97	0.01	0.92	0.03	0.82	0.07
25	0.97	0.01	0.88	0.04	0.67	0.14
WCEM						
2	0.98	0.01	0.93	0.02	0.83	0.06
5	0.98	0.01	0.93	0.02	0.83	0.06
10	0.97	0.01	0.93	0.03	0.80	0.07
25	0.96	0.01	0.82	0.07	0.57	0.21
EM						
2	0.97	0.01	0.93	0.02	0.83	0.06
5	0.97	0.01	0.93	0.02	0.83	0.06
10	0.97	0.01	0.92	0.03	0.82	0.07
25	0.97	0.01	0.89	0.04	0.66	0.15
CEM						
2	0.97	0.01	0.93	0.02	0.82	0.07
5	0.97	0.01	0.93	0.02	0.82	0.07
10	0.97	0.01	0.92	0.03	0.79	0.08
25	0.96	0.01	0.81	0.07	0.55	0.22
`otrimle`						
2	0.97	0.01	0.93	0.02	0.84	0.06
5	0.97	0.01	0.93	0.02	0.83	0.06
10	0.97	0.01	0.92	0.03	0.82	0.07
25	0.96	0.01	0.88	0.04	0.67	0.15

For what concerns the task of outlier detection, several strategies have been compared: we considered a detection rule based on the 0.99-level quantile of the χ^2_p distribution, according to (9), but also based on the fitted weights, with thresholds set at 0.1, 0.2 and $1 - \hat{\tilde{\omega}}$. For each decision rule, for the contaminated scenario, we report (a) the rate of detected outliers ϵ; (b) the swamping rate; (c) the masking rate. The first is a measure of the fitted contamination level, whereas the others give insights on the level and power of the outlier detection procedure. The comparisons across the different methods should be considered for close values of ϵ. Actually, the more outliers are detected the more likely a genuine observation can be misclassified, whereas, on the contrary, the more true outliers are correctly flagged. For $\epsilon = 0$, swamping only is taken into account.

Classification accuracy has been measured by (i) the adjusted Rand index and (ii) the misclassification error rate (MCE), both evaluated over true negatives for the robust techniques. The results are based on the testing decision rule (9).

Table 5 Average measures of fitting accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10, 25$, $\epsilon = 0.10, 0.20$, $\beta = 10$ (well-separated clusters)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	μ	Σ	π	μ	Σ	π
WEM						
2	0.651	0.219	0.029	0.745	0.249	0.026
5	0.679	0.886	0.025	0.694	0.894	0.026
10	0.685	1.237	0.023	0.741	1.278	0.027
25	0.852	1.885	0.025	0.914	1.987	0.031
WCEM						
2	0.629	0.213	0.035	0.761	0.245	0.059
5	0.731	0.944	0.045	0.719	1.128	0.045
10	0.732	1.301	0.035	0.780	1.642	0.040
25	0.944	1.890	0.027	0.993	1.991	0.037
`tclust`						
2	0.619	0.219	0.027	0.774	0.261	0.029
5	0.631	0.796	0.025	0.685	0.834	0.026
10	0.682	1.153	0.023	0.718	1.218	0.028
25	0.888	1.861	0.025	0.869	1.993	0.028
`otrimle`						
2	0.638	0.214	0.110	0.744	0.239	0.177
5	0.629	0.803	0.068	0.664	0.845	0.124
10	0.668	1.163	0.067	0.704	1.233	0.124
25	0.860	1.889	0.065	0.914	1.967	0.125

Table 6 Average measures of fitting accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10, 25$, $\epsilon = 0.10, 0.20$, $\beta = 8$ (moderate overlapping)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	μ	Σ	π	μ	Σ	π
WEM						
2	0.684	0.254	0.029	0.821	0.280	0.030
5	0.727	0.883	0.027	0.742	0.900	0.028
10	0.750	1.233	0.028	0.801	1.308	0.030
25	1.435	1.955	0.047	1.725	2.067	0.056
WCEM						
2	0.685	0.230	0.041	1.087	0.285	0.081
5	0.806	1.170	0.045	0.826	1.128	0.039
10	0.858	1.616	0.032	0.928	1.774	0.038
25	2.504	2.024	0.075	2.144	2.219	0.076
`tclust`						
2	0.802	0.264	0.034	0.884	0.336	0.039
5	0.829	0.818	0.033	0.876	0.871	0.033
10	0.860	1.187	0.033	0.909	1.258	0.036
25	2.266	1.988	0.067	1.969	2.068	0.067
`otrimle`						
2	0.652	0.238	0.111	0.731	0.276	0.183
5	0.684	0.812	0.068	0.719	0.872	0.126
10	0.731	1.192	0.068	0.782	1.261	0.126
25	1.659	1.962	0.083	1.992	2.044	0.142

In order to avoid problems due to label switching issues, cluster labels have been sorted according to the first entry of the fitted location vectors.

Under the assumed model, WEM and WCEM have been initialized by `tclust` with 20% of trimming and their behavior have been compared with the EM and CEM algorithms and the `otrimle`, for the same eigen-ratio constraint and the same initial values. In the presence of contamination, we do not report the results concerning the non-robust EM and CEM but only those regarding the WEM, WCEM, `otrimle` and *oracle* `tclust`, i.e. with trimming level equal to the actual contamination level (`tclust10` and `tclust20`, respectively). Under this scenario, starting values have been driven by `tclust` with 50% of trimming.

It is worth to stress, here, that the comparison in terms of outlier detection reliability between weighted likelihood estimation, `tclust` and `otrimle` can be considered fair only by looking at the rate of weights below the fixed threshold for the former methodology and trimmed observation or those assigned to the improper density group for the latter techniques, since formal testing rules have not been considered neither for `tclust` or `otrimle`.

First, let us consider the behavior of WEM and WCEM at the assumed model. The entries in Table 2 give the considered average measures of fitting accuracy; Table 3 gives the level of swamping according to the different strategies for WEM and WCEM that are based on the χ^2_p distribution and the inspection of weights; Table 4 reports classification accuracy. The overall behavior of WEM and WCEM is appreciable: we observe a tolerable efficiency loss, a negligible swamping effect and a reliable classification accuracy, indeed, as compared with the non-robust procedures. Furthermore, the results are quite similar to those stemming from `otrimle` and quite often the inspection of the weights from WEM and WCEM leads to a smaller number of false positives, on average.

The performance of WEM and WCEM under contamination is explored next. The fitting accuracy provided by the proposed weighted likelihood based strategies is illustrated in Tables 5, 6, 7. In all considered scenarios, the behavior of WEM and WCEM is satisfactory and they both compare well with the oracle `tclust` and `otrimle`. In particular, the good performance of WEM and WCEM has to be remarked in the challenging situation of severe overlapping. Furthermore, for all data configurations, we notice the ability of WEM to combine accurate estimates of component-specific parameters with those of the cluster weights. The entries in Tables 8, 9, 10 show the behavior of the testing procedure

Table 7 Average measures of fitting accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 6$ (severe overlapping)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	μ	Σ	π	μ	Σ	π
WEM						
2	1.043	0.313	0.038	1.131	0.370	0.056
5	0.837	0.946	0.032	0.899	0.977	0.035
10	0.990	1.294	0.040	1.115	1.375	0.046
25	3.203	2.153	0.135	3.608	2.229	0.156
WCEM						
2	1.112	0.344	0.058	1.386	0.362	0.081
5	1.121	1.356	0.072	1.465	1.236	0.054
10	1.909	1.756	0.086	1.889	1.951	0.076
25	7.826	2.541	0.198	5.752	2.279	0.244
`tclust`						
2	1.728	0.372	0.082	3.431	0.607	0.133
5	1.477	0.897	0.069	1.685	0.957	0.072
10	1.761	1.301	0.077	1.898	1.348	0.083
25	6.048	2.221	0.206	5.973	2.267	0.203
`otrimle`						
2	0.910	0.293	0.122	0.885	0.295	0.183
5	0.781	0.876	0.072	0.861	0.923	0.127
10	0.991	1.259	0.077	1.086	1.320	0.132
25	3.708	2.131	0.164	3.922	2.199	0.203

based on the χ^2_p distribution and the inspection of weights for all considered scenarios. The empirical level of contamination is always larger than the nominal one, but it is acceptable and stable as p and β change. Masking is always negligible, hence highlighting the appreciable power of the testing procedure. We remark that one could also consider multiple testing adjustments in outlier detection as outlined in [13]. To conclude the analysis, Tables 11, 12, 13 give the considered measures of classification accuracy as β varies. The results are quite stable across the four methods and all dimensions. As well as before, WEM and WCEM lead to a satisfactory classification, even in the challenging case of severe overlapping. The results obtained for $p = 25$ deserve some special remarks. Actually, fitting and classification accuracies deteriorate, particularly in the presence of moderate-to-severe overlapping. Nevertheless, the proposed WEM and WCEM still behave in a fashion not dissimilar from the other well-established techniques.

6.1 Computational burden

The numerical studies are enriched by evaluating the computational demand of the proposed methodology for increasing sample size and dimension. Time needed for convergence

Table 8 Outlier detection for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 10$ (well-separated clusters)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
WEM $\alpha = 0.01$						
2	0.118	0.021	0.004	0.227	0.034	0.002
5	0.120	0.022	0.000	0.217	0.021	0.000
10	0.119	0.021	0.000	0.218	0.022	0.000
25	0.119	0.021	0.000	0.223	0.028	0.000
WCEM $\alpha = 0.01$						
2	0.118	0.021	0.005	0.223	0.032	0.015
5	0.118	0.020	0.000	0.217	0.017	0.000
10	0.115	0.017	0.000	0.213	0.016	0.000
25	0.114	0.016	0.000	0.215	0.019	0.000
WEM $w < 0.1$						
2	0.096	0.005	0.086	0.211	0.017	0.012
5	0.108	0.009	0.000	0.205	0.006	0.000
10	0.106	0.007	0.000	0.207	0.009	0.000
25	0.109	0.010	0.000	0.209	0.011	0.000
WCEM $\hat{w} < 0.1$						
2	0.092	0.005	0.125	0.201	0.019	0.069
5	0.107	0.008	0.000	0.205	0.006	0.001
10	0.105	0.006	0.000	0.206	0.007	0.000
25	0.105	0.006	0.000	0.205	0.007	0.000
WEM $\hat{w} < 0.2$						
2	0.110	0.013	0.016	0.231	0.039	0.001
5	0.118	0.020	0.000	0.213	0.016	0.000
10	0.116	0.018	0.000	0.216	0.020	0.000
25	0.115	0.017	0.000	0.215	0.219	0.000
WCEM $\hat{w} < 0.2$						
2	0.108	0.013	0.038	0.227	0.039	0.022
5	0.116	0.018	0.000	0.211	0.014	0.000
10	0.114	0.016	0.001	0.213	0.016	0.000
25	0.110	0.011	0.000	0.210	0.012	0.000
WEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.012	0.018	0.263	0.078	0.000
5	0.126	0.031	0.000	0.224	0.031	0.000
10	0.122	0.024	0.000	0.238	0.048	0.000
25	0.117	0.019	0.000	0.223	0.029	0.000
WCEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.013	0.016	0.263	0.080	0.007
5	0.125	0.028	0.000	0.224	0.030	0.000
10	0.120	0.025	0.000	0.232	0.040	0.000
25	0.111	0.012	0.000	0.215	0.019	0.000
`tclust`						
2	0.100	0.005	0.043	0.200	0.008	0.031
5	0.100	0.000	0.000	0.200	0.000	0.000
10	0.100	0.000	0.000	0.200	0.000	0.000
25	0.100	0.000	0.000	0.200	0.000	0.000

Table 8 continued

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
`otrimle`						
2	0.135	0.040	0.019	0.254	0.069	0.005
5	0.106	0.003	0.007	0.203	0.003	0.004
10	0.102	0.002	0.001	0.202	0.003	0.000
25	0.101	0.001	0.005	0.200	0.001	0.003

with non-optimized `R` code on a 3.4 GHz Intel Core i5 processor is given in Table 14. We report the sample times needed for convergence of the algorithm on a single dataset with $K = 3$, $\epsilon = 20\%$, $b = 10$, $c = 50$. The smoothing parameter has been chosen in order to achieve an empirical downweighting level about equal to 0.25. Initialization has been included in time evaluation. It can be seen that computing time increases both with the sample size and the dimension but always at a reasonable slow rate. We underline that the speed of convergence also depends on the choice of h: values of the smoothing parameter leading to an excess of downweighting can make it slow down.

7 Real data examples

7.1 Swiss bank note data

Let us consider the well-known Swiss banknote dataset concerning $p = 6$ measurements of $n = 200$ old Swiss 1000-franc banknotes, half of which are counterfeit. The weighted likelihood strategy is based on a gamma kernel and a symmetric Chi-square RAF. Our first task is to choose the number of clusters. To this end, we look at the weighted BIC (12) on a fixed grid of h values for $K = 1, 2, 3, 4$ and a restriction factor $c = 12$. The inspection of Figure 5 clearly suggests a two-group structure for all considered values of the smoothing parameter h. The empirical downweighting level is fairly stable for a wide range of h values. We decided to set $h = 0.05$ leading to an empirical downweighting level equal to 0.10. The WEM algorithm based on the testing rule (9) with $\alpha = 0.01$ leads to identify 21 outliers that include 15 forged and 6 genuine bills.

On the contrary, there are 19 data points whose weight is lower than $1 - \hat{\tilde{w}}$ that include 14 forged and 5 genuine bills. The cluster assignments stemming from the latter approach are displayed in Figure 6. It is worth to note that the outlying forged bills coincide with the group that has been recognized to follow a different forgery pattern and characterized by a peculiar length of the diagonal (see García-Escudero et al. 30; Dotto et al. 21, and references therein). On the other hand, the

Table 9 Outlier detection for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 8$ (moderate overlapping)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
WEM $\alpha = 0.01$						
2	0.120	0.023	0.001	0.218	0.025	0.012
5	0.126	0.029	0.000	0.216	0.021	0.000
10	0.125	0.028	0.000	0.217	0.021	0.000
25	0.119	0.021	0.000	0.222	0.028	0.000
WCEM $\alpha = 0.01$						
2	0.117	0.024	0.048	0.216	0.029	0.038
5	0.127	0.031	0.000	0.214	0.017	0.000
10	0.123	0.026	0.000	0.213	0.016	0.000
25	0.115	0.017	0.000	0.220	0.025	0.000
WEM $\hat{w} < 0.1$						
2	0.096	0.005	0.086	0.202	0.012	0.038
5	0.108	0.009	0.000	0.205	0.006	0.000
10	0.106	0.007	0.000	0.207	0.009	0.000
25	0.105	0.005	0.000	0.209	0.011	0.000
WCEM $\hat{w} < 0.1$						
2	0.092	0.005	0.125	0.198	0.019	0.085
5	0.107	0.008	0.000	0.205	0.006	0.001
10	0.105	0.006	0.000	0.205	0.016	0.000
25	0.114	0.015	0.000	0.208	0.010	0.000
WEM $\hat{w} < 0.2$						
2	0.110	0.013	0.016	0.221	0.028	0.009
5	0.118	0.020	0.000	0.212	0.014	0.000
10	0.116	0.018	0.000	0.215	0.019	0.000
25	0.120	0.020	0.000	0.215	0.019	0.000
WCEM $\hat{w} < 0.2$						
2	0.108	0.013	0.038	0.225	0.041	0.042
5	0.116	0.018	0.000	0.211	0.013	0.000
10	0.114	0.016	0.001	0.212	0.016	0.000
25	0.120	0.020	0.000	0.215	0.019	0.000
WEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.012	0.018	0.250	0.063	0.002
5	0.126	0.031	0.000	0.223	0.029	0.000
10	0.122	0.024	0.000	0.236	0.046	0.000
25	0.122	0.026	0.000	0.223	0.029	0.000
WCEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.013	0.016	0.272	0.097	0.025
5	0.125	0.028	0.000	0.222	0.028	0.000
10	0.120	0.025	0.000	0.231	0.039	0.000
25	0.122	0.026	0.000	0.227	0.034	0.000
`tclust`						
2	0.100	0.007	0.064	0.200	0.009	0.036
5	0.100	0.000	0.001	0.200	0.000	0.001
10	0.100	0.000	0.000	0.200	0.000	0.000
25	0.100	0.000	0.000	0.200	0.000	0.000

Table 9 continued

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
`otrimle`						
2	0.141	0.047	0.005	0.256	0.069	0.000
5	0.101	0.001	0.007	0.206	0.008	0.004
10	0.102	0.002	0.001	0.203	0.004	0.000
25	0.100	0.001	0.004	0.201	0.002	0.002

outlying genuine bills all exhibit some extreme measures. For the same value of the eigen-ratio constraint, the `otrimle` assigns 19 bills to the improper component density, leading to the same classification of the WEM, whereas `rtclust` includes in the trimming set one counterfeit bill more.

A visual comparison between the three results is possible from Figure 7, whose panels show a scatterplot of the fourth against the sixth variable with the classification resulting from WEM (with both outlier detection rules), `rtclust` and `otrimle`, respectively. The WCEM has been tuned to achieve the same empirical downweighting level and leads to the same results.

7.2 2018 world happiness report data

In this section the weighted likelihood methodology is applied to a dataset from the 2018 World Happiness Report by the United Nations Sustainable Development Solutions Network (Helliwell et al. 33) (hereafter denoted by WHR18). The data give measures about six key variables used to explain the variation of subjective well-being across countries: per capita Gross Domestic Product (on log scale), Social Support, i.e. the national average of the binary responses to the Gallup World Poll (GWP) question *If you were in trouble, do you have relatives or friends you can count on to help you whenever you need them, or not?*, Health Life Expectancy at birth, Freedom to make life choices, i.e. the national average of binary responses to the GWP question *Are you satisfied or not with your freedom to choose what you do with your life?*, Generosity, measured by the residual of regressing the national average of GWP responses to the question *Have you donated money to a charity in the past month?* on GDP per capita, perception of Corruption, i.e. the national average of binary responses to the GWP questions *Is corruption widespread throughout the government or not?* and *Is corruption widespread within businesses or not?*. The dataset is made of 142 rows, after the removal of some countries characterized by missing values. The objective is to obtain groups of countries with a similar behavior, to identify possible countries with anomalous and unexpected traits and to

Table 10 Outlier detection for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 6$ (severe overlapping)

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
WEM $\alpha = 0.01$						
2	0.120	0.023	0.001	0.206	0.020	0.052
5	0.125	0.028	0.000	0.214	0.018	0.000
10	0.123	0.026	0.000	0.216	0.020	0.000
25	0.123	0.026	0.000	0.222	0.027	0.000
WCEM $\alpha = 0.01$						
2	0.145	0.050	0.001	0.253	0.045	0.006
5	0.128	0.031	0.000	0.214	0.018	0.000
10	0.124	0.027	0.000	0.213	0.017	0.000
25	0.119	0.021	0.000	0.219	0.024	0.000
WEM $\hat{w} < 0.1$						
2	0.096	0.005	0.086	0.209	0.020	0.003
5	0.108	0.009	0.000	0.205	0.006	0.000
10	0.106	0.007	0.000	0.207	0.008	0.000
25	0.111	0.010	0.000	0.210	0.009	0.000
WCEM $\hat{w} < 0.1$						
2	0.092	0.005	0.125	0.237	0.046	0.001
5	0.107	0.008	0.000	0.206	0.006	0.001
10	0.105	0.006	0.000	0.206	0.007	0.000
25	0.110	0.010	0.000	0.211	0.009	0.000
WEM $\hat{w} < 0.2$						
2	0.110	0.013	0.016	0.231	0.039	0.001
5	0.118	0.020	0.000	0.211	0.013	0.000
10	0.116	0.018	0.000	0.215	0.019	0.000
25	0.118	0.021	0.000	0.216	0.020	0.000
WCEM $\hat{w} < 0.2$						
2	0.108	0.013	0.038	0.227	0.039	0.022
5	0.116	0.018	0.000	0.211	0.013	0.001
10	0.114	0.016	0.001	0.213	0.016	0.000
25	0.118	0.019	0.000	0.218	0.018	0.000
WEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.012	0.018	0.209	0.020	0.003
5	0.126	0.031	0.000	0.222	0.027	0.000
10	0.122	0.024	0.000	0.235	0.044	0.000
25	0.127	0.032	0.000	0.238	0.048	0.000
WCEM $\hat{w} < 1 - \hat{\tilde{w}}$						
2	0.109	0.013	0.016	0.237	0.046	0.001
5	0.125	0.028	0.000	0.223	0.028	0.000
10	0.120	0.025	0.000	0.232	0.041	0.000
25	0.125	0.027	0.000	0.235	0.042	0.000
`tclust`						
2	0.100	0.007	0.061	0.200	0.024	0.095
5	0.100	0.000	0.002	0.200	0.000	0.001
10	0.100	0.000	0.000	0.200	0.000	0.000
25	0.100	0.000	0.000	0.200	0.000	0.000

Table 10 continued

p	$\epsilon = 0.10$			$\epsilon = 0.20$		
	ϵ	swamp.	mask.	ϵ	swamp.	mask.
`otrimle`						
2	0.145	0.050	0.004	0.251	0.064	0.003
5	0.102	0.003	0.006	0.202	0.003	0.006
10	0.102	0.003	0.001	0.203	0.004	0.000
25	0.101	0.001	0.002	0.201	0.001	0.001

Table 11 Average measures of classification accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 10$ (well-separated clusters)

p	$\epsilon = 0.10$		$\epsilon = 0.20$	
	Rand	MCE	Rand	MCE
WEM				
2	0.98	0.01	0.98	0.01
5	0.97	0.01	0.97	0.01
10	0.97	0.01	0.97	0.01
25	0.96	0.01	0.96	0.01
WCEM				
2	0.98	0.01	0.9	0.01
5	0.97	0.01	0.97	0.01
10	0.97	0.01	0.97	0.01
25	0.95	0.02	0.95	0.02
`tclust`				
2	0.97	0.01	0.97	0.01
5	0.97	0.01	0.97	0.01
10	0.97	0.01	0.97	0.01
25	0.95	0.02	0.95	0.02
`otrimle`				
2	0.98	0.01	0.98	0.01
5	0.97	0.01	0.97	0.01
10	0.97	0.01	0.97	0.01
25	0.96	0.01	0.96	0.01

highlight those features that are the main source of separation among clusters.

In this example, a GKL RAF has been chosen, the unbiased at the boundary kernel density estimate has been obtain by first evaluating a kernel density estimate on the log-transformed squared distance over the whole real line and then back-transforming the fitted density to $(0, \infty)$ (Agostinelli and Greco 5), and we set $c = 50$.

In order to select the number of clusters, we monitored the weighted BIC stemming from WEM and the classification log-likelihood at convergence from WCEM for different values of K and h. The corresponding monitoring plots are given in Figure 8, respectively. Based on the WEM algorithm, $K = 3$ is to be preferred, even if the gap with the

Table 12 Average measures of classification accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$, $\epsilon = 0.10, 0.20$, $\beta = 8$ (moderate overlapping)

p	$\epsilon = 0.10$		$\epsilon = 0.20$	
	Rand	MCE	Rand	MCE
WEM				
2	0.93	0.02	0.92	0.03
5	0.92	0.03	0.93	0.03
10	0.92	0.03	0.92	0.03
25	0.87	0.05	0.86	0.05
WCEM				
2	0.93	0.02	0.91	0.03
5	0.92	0.03	0.93	0.03
10	0.92	0.03	0.92	0.03
25	0.81	0.07	0.81	0.07
`tclust`				
2	0.93	0.02	0.92	0.03
5	0.92	0.03	0.93	0.03
10	0.92	0.03	0.92	0.03
25	0.80	0.08	0.80	0.08
`otrimle`				
2	0.93	0.02	0.93	0.02
5	0.92	0.03	0.93	0.03
10	0.92	0.03	0.92	0.03
25	0.86	0.05	0.84	0.06

case $K = 4$ is very small for all considered values of the smoothing parameter. On the other hand, the inspection of the weighted classification log-likelihood driven by the WCEM suggests $K = 4$. Therefore, we have applied our WEM and WCEM algorithms both based on $K = 3$ and $K = 4$. As with $K = 4$ two groups are not very separated, we preferred $K = 3$ and reported only those results for reasons of space. Moreover, the results stemming from WEM and WCEM were very similar both in terms of fitted parameters, cluster assignments and detected outliers. Then, in the following we only give the results driven by WEM. The empirical down-weighting level was found not to depend in a remarkable fashion on the number of groups. In particular, for $K = 3$, in the monitoring process of $1 - \hat{\tilde{w}}$ we did not observe any abrupt change but a smooth decline until a stabilization of the level of contamination occurred. Then, we decided to use a h value leading to $(1 - \hat{\tilde{w}}) \approx 0.10$. Figure 9 displays the distance plot stemming from WEM. According to (9) for a level $\alpha = 0.01$, 12 outliers are detected. A closer inspection of the plot unveils that some of such points are close to the cut-off value. Therefore, they are not considered as outliers but correctly assigned to the corresponding cluster. Furthermore, we notice that all the points leading to the largest distances are attached a very small weight (< 0.01). The weight cor-

Table 13 Average measures of classification accuracy for WEM, WCEM, `tclust` and `otrimle` with $p = 2, 5, 10$x, $\epsilon = 0.10, 0.20$, $\beta = 6$ (severe overlapping)

p	$\epsilon = 0.10$		$\epsilon = 0.20$	
	Rand	MCE	Rand	MCE
WEM				
2	0.82	0.07	0.79	0.07
5	0.82	0.06	0.82	0.06
10	0.82	0.07	0.81	0.07
25	0.66	0.15	0.64	0.16
WCEM				
2	0.82	0.06	0.79	0.08
5	0.81	0.07	0.81	0.07
10	0.79	0.08	0.79	0.08
25	0.53	0.20	0.52	0.19
`tclust`				
2	0.82	0.07	0.76	0.07
5	0.82	0.07	0.82	0.07
10	0.80	0.08	0.80	0.08
25	0.49	0.23	0.49	0.22
`otrimle`				
2	0.84	0.06	0.83	0.06
5	0.83	0.06	0.83	0.06
10	0.82	0.07	0.81	0.07
25	0.65	0.16	0.64	0.16

Table 14 Time (in s) for convergence of WEM on simulated data sets of increasing size $n = 1000, 2000, 5000, 10,000$ and dimensions $p = 2, 5, 10, 25$, with $\epsilon = 0.20$, $\beta = 10$, $c = 50$

Size	Dimension			
	2	5	10	25
1000	1.58	4.38	7.44	19.94
2000	8.08	11.50	15.24	30.06
5000	24.98	32.54	45.67	61.72
10,000	88.50	90.86	93.76	116.68

responding to Myanmar is about 0.40. The other countries near the threshold line all show weights about equal to 0.80.

The cluster profiles and raw measurements for the detected outlying countries are reported in Table 15. The three clusters are well separated in terms of all of the items considered, even if small differences are seen in terms of perception of Corruption between clusters 1 and 2. Cluster 3 includes all the countries characterized by the highest level of subjective well-being. The differences with the other two clusters concerning GDP, HLE and Corruption are outstanding. On the opposite, in cluster 1 we find all the countries with the hardest economic and social conditions. For what concerns the explanation of the outliers, we notice that the subgroup

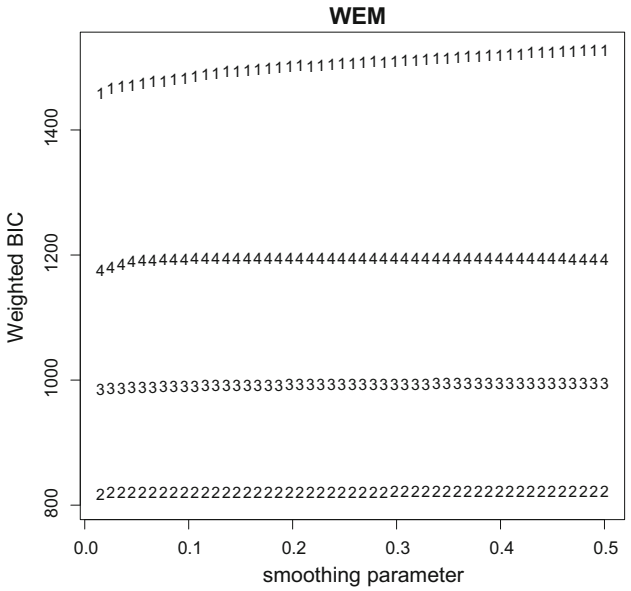

Fig. 5 Swiss banknote data. Monitoring the weighted BIC for WEM, $K = 1, 2, 3, 4$

of African countries composed by Central African Republic, Chad, Ivory Coast, Lesotho, Nigeria and Sierra Leone might belong to group 1 but is characterized by the six lowest HLE values. For what concerns Myanmar, it exhibits extremely large Freedom and Generosity indexes and a surprising small value for Corruption. It should belong to cluster 1 but it is closer to cluster 3 according to the last three measurements, indeed. For instance, it may be supposed that such measurements are not completely reliable because of problems with the questionnaires and the sampling or they may revel a surprising positive attitude despite the difficult economic and life conditions.

A spatial map of cluster assignments is given in Figure 10 that confirms the goodness and coherence of the results and the ability of the considered six features to find reasonable clusters. Cluster 1 is mainly localized in Africa, cluster 2 is composed by developing countries, whereas cluster 3 includes the world leading countries, among which there are USA, Canada, Australia and the countries in the European Union.

7.3 Anuran calls

This example concerns the problem of recognition and classification of anuran (frogs and tods) families through their calls. The classification task is a very complex problem due to the large anuran diversity (Colonna et al. 15). The data gives $p = 22$ normalized Mel-frequency cepstral coefficients (MFCCs), measures that are commonly used as features in sound processing and speech recognition, for each of $n = 7195$ syllables, extracted after the segmentation of 60

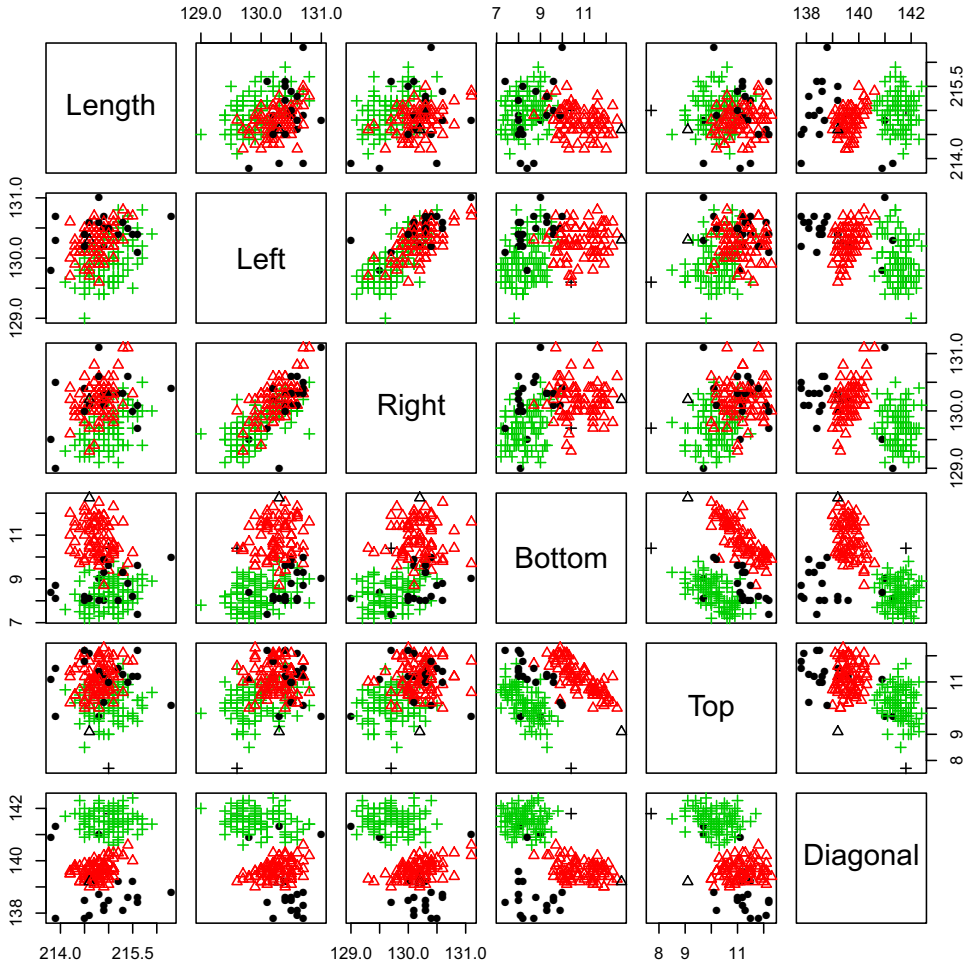

Fig. 6 Swiss banknote data. Cluster assignments by WEM. Observation whose weight is lower than $1 - \bar{w}$ are considered outliers. Genuine bills are denoted by a green +, forged bills by a red △. Outliers are denoted by a black filled circle. (Color figure online)

audio records belonging to $k = 4$ different families. The data are publicly available at the `UCI Machine Learning Repository`. Despite the challenging nature of the problem, in particular for model-based clustering techniques, we want to assess the reliability of the proposed methodology on such high-dimensional example and its behavior as an unsupervised learning device. To this end, we decided to split half the data in a training and test set.

The entries in Table 16 give the adjusted Rand index evaluated on both sets, averaged over 100 replicated splits, in order to honestly estimate the accuracy of classification. We compare the results from `mclust`, `tclust10`, `tclust20`, WEM and WCEM. WEM and WCEM are characterized by a GKL RAF and a gamma kernel. On the training set, the adjusted Rand index is evaluated only on those observations not flagged as outliers. By looking at the results, one can state that the robust methods are feasible also in this challenging settings and lead to improved classification accuracy w.r.t. the non-robust `mclust`, indeed.

On the other hand, the robust techniques only succeed to a limited extent in recovering the actual classification on the test set, because of the problem complexity, which comes from the large dimensionality and sample size but also from the severe overlapping of the groups corresponding to the anuran families. Nevertheless, the behavior of the weighted likelihood methodologies is satisfactory when compared with the other competing techniques. The behavior of `tclust10` on the training set is a consequence of the smaller number of detected outliers.

8 Conclusions

We have proposed a robust technique for fitting a finite mixture of multivariate Gaussian components based on recent developments in weighted likelihood estimation. Actually, the proposed methodology is meant to provide a step further with respect to the original proposal in [38]. The method is based on the idea of using a univariate kernel density estimate based on robust distances rather than a multivariate one based on the data in order to compute weights. Furthermore, the proposed technique is characterized by the introduction of an eigen constraint aimed at avoiding problems connected with an unbounded likelihood or spurious solutions.

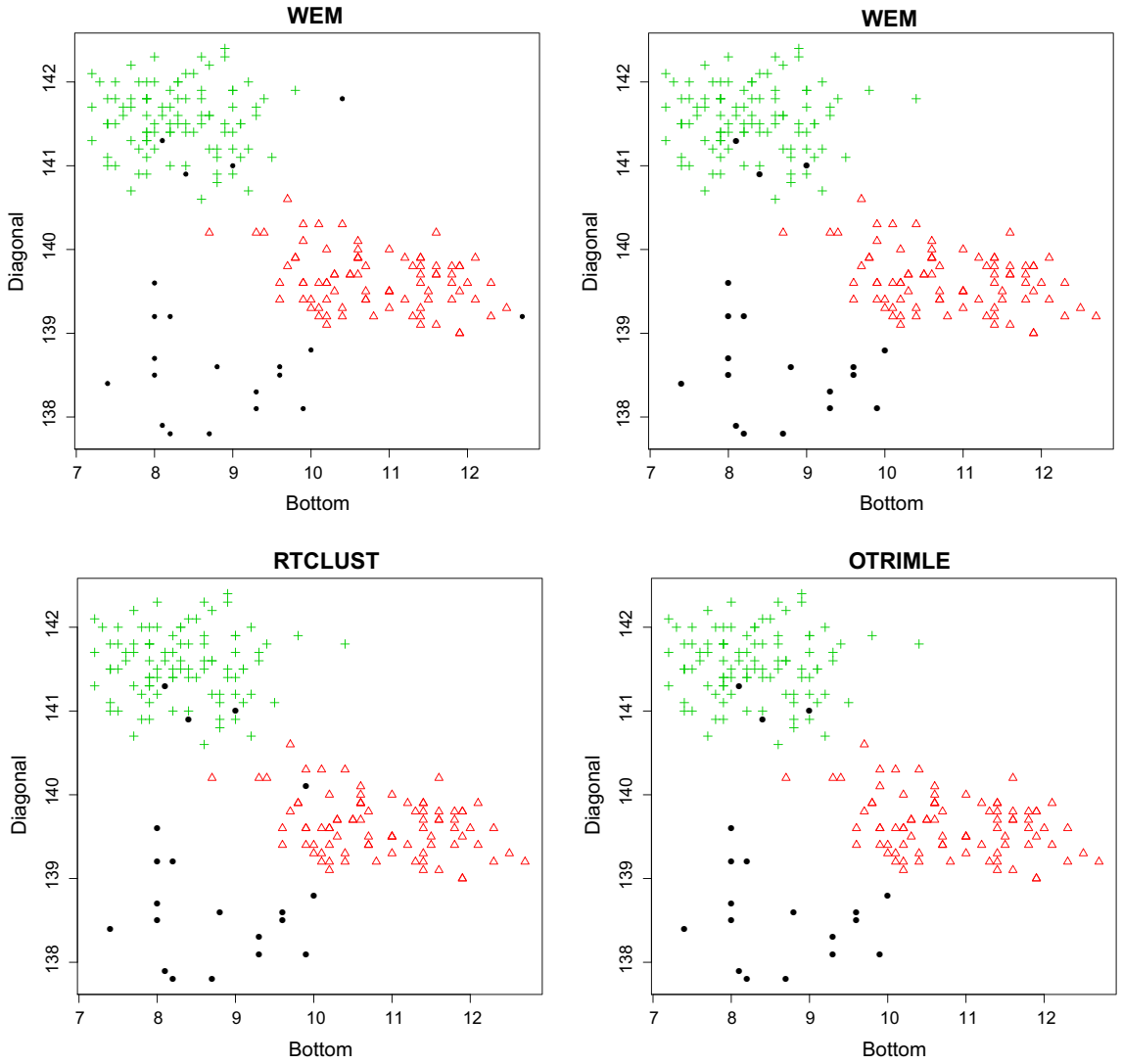

Fig. 7 Swiss banknote data. Fourth against the sixth variable with cluster assignments by WEM ($\alpha = 0.01$), WEM ($w_{k_i} < 1 - \bar{w}$), `rtclust` and `otrimle` in clockwise fashion. Genuine bills are denoted by a green +, forged bills by a red △. Outliers are denoted by a black filled circle. (Color figure online)

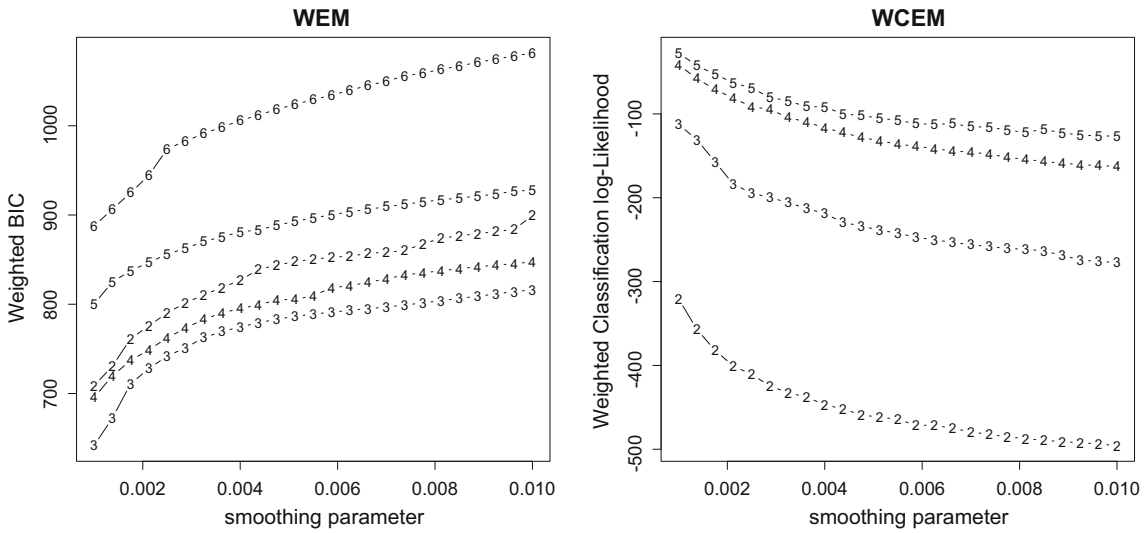

Fig. 8 WHR18 data. Monitoring the weighted BIC for WEM (left) and the weighted classification log-likelihood for WCEM, $K = 2, 3, 4, 5, 6$

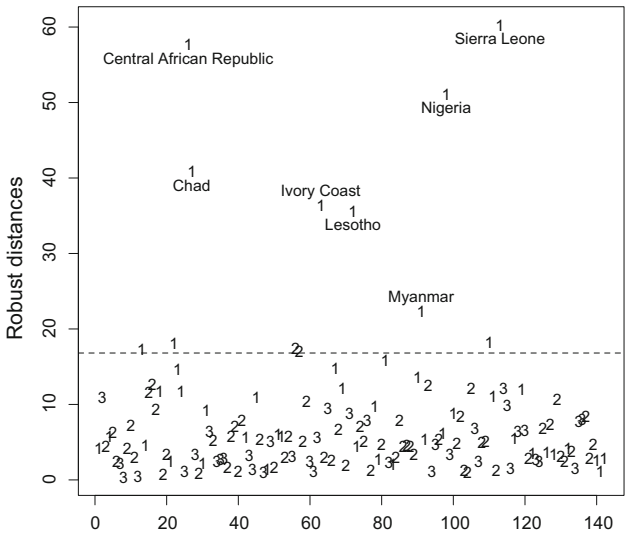

Fig. 9 WHR18 data. Distance plot for WEM with $K = 3$. The horizontal line gives the $\chi^2_{6;0.99}$ quantile

Based on the robustly fitted mixture model, a model-based clustering strategy can be built in a standard fashion by looking at the value of posterior membership probabilities. At the same time, formal rules for outlier detection can be derived, as well. Then, one could assign units to clusters provided that the corresponding outlyingness test is not significant that means that detected outliers have to be discarded and not assigned to any group. The numerical studies and the real data examples showed the satisfactory reliability of the proposed methodology.

There is still room for further work, along a path shared with `tclust`, `rtclust`, `mtclust` and `otrimle`. Actually the proposed method works for a given smoothing parameter h and a fixed number of clusters K. In addition, outlier detection depends upon a fixed threshold. At the moment, the selection of h stemming from the monitoring of several quantities, such as the empirical downweighting level, the unit-specific robust distances or even the fitted parameters, provides an acceptable adaptive solution. Such a procedure is not different from the implementation of a

Table 15 WHR18 data: cluster profiles and raw measurements for the detected outlying countries

	LogGDP	HLE	Social sup.	Freedom	Generosity	Corruption	Size
Cluster 1	7.88	53.68	0.69	0.69	0.19	0.78	38
Cluster 2	9.35	64.46	0.83	0.74	0.26	0.80	60
Cluster 3	10.48	71.63	0.90	0.83	0.44	0.61	37
Cen.Afr.Rep.	6.47	44.31	0.31	0.63	0.17	0.88	
Chad	7.55	45.66	0.68	0.53	0.17	0.84	
Ivory Coast	8.13	46.52	0.66	0.77	0.16	0.76	
Lesotho	7.91	46.48	0.80	0.73	0.10	0.74	
Myanmar	8.59	57.51	0.79	0.86	0.90	0.62	
Nigeria	8.61	45.50	0.78	0.76	0.28	0.89	
Sierra Leone	7.22	43.99	0.64	0.67	0.24	0.85	

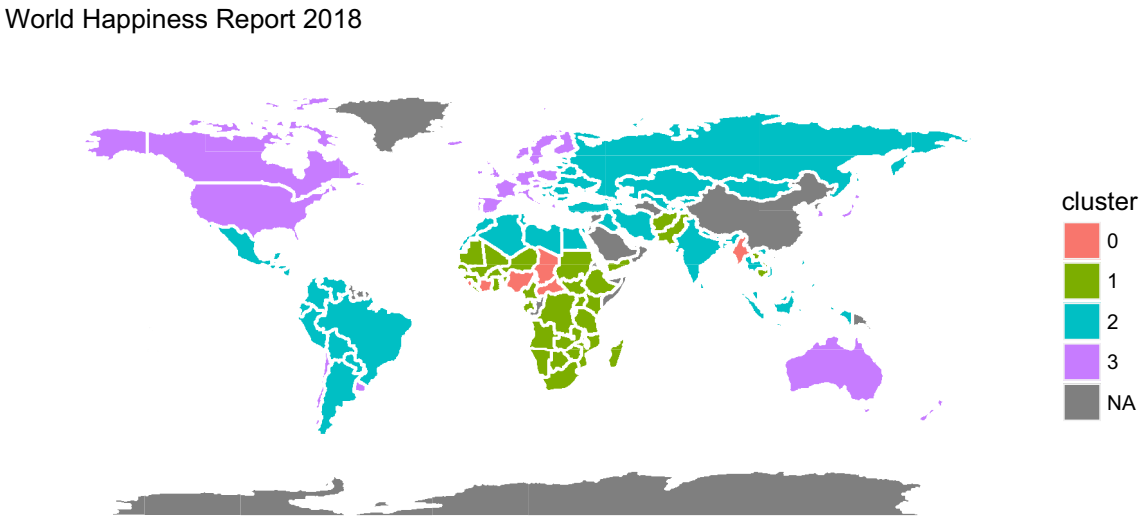

Fig. 10 WHR18 data. Spatial classification from WEM with $K = 3$

Table 16 Anuran calls

	mclust	tclust10	tclust20	WEM	WCEM
Training	0.25	0.58	0.65	0.68	0.61
	(0.06)	(0.03)	(0.02)	(0.02)	(0.03)
Test	0.25	0.53	0.52	0.53	0.53
	(0.07)	(0.04)	(0.02)	(0.02)	(0.02)

Adjusted Rand index for `mclust`, `tclust10`, `tclust20`, WEM and WCEM, averaged over 100 trials; standard errors are also given in parenthesis

sequence of refinement steps of an initial robust partition stemming from a sequence of decreasing values of h. The selection of K remains a difficult problem to deal with too, despite the satisfactory behavior of the proposed criteria, i.e. the weighted BIC and the weighted classification log-likelihood. Outlier detection is a novel aspect in the framework of robust mixture modeling and model-based clustering. In the specific context, the outlyingness of each unit is tested conditionally on the final cluster assignment. The number of outliers clearly depends on the chosen level α or the selected threshold for the final weights. A fair choice of the level of the test is still an open problem in outlier detection. However, the suggested testing strategies work satisfactory, at least in those considered scenarios, and provide a good compromise between swamping and masking that could be improved further by using multiplicity adjustments (Cerioli and Farcomeni 13). The extent to which the proposed methodology allows to deal with very large-dimensional problems remains limited, as well as for the other robust model-based clustering techniques we also considered in this paper. Nevertheless, the weighted likelihood methodology looks promising if one is willing to develop robust procedures specifically suited for high-dimensional problems. For instance, multivariate weighted likelihood estimation could be considered in model-based subspace clustering methods and in particular in the framework of mixtures of factor analyzers (McLachlan et al. 42). The reader is pointed to [8] for a recent account on high-dimensional clustering.

Acknowledgements The authors are grateful to the coordinating editor and two anonymous referees for their valuable suggestions.

References

Agostinelli, C.: Robust model selection in regression via weighted likelihood methodology. Stat. Probab. Lett. **56**(3), 289–300 (2002)

Agostinelli, C.: Notes on pearson residuals and weighted likelihood estimating equations. Stat. Probab. Lett. **76**(17), 1930–1934 (2006)

Agostinelli, C., Greco, L.: A weighted strategy to handle likelihood uncertainty in Bayesian inference. Comput. Stat. **28**(1), 319–339 (2013)

Agostinelli, C., Greco, L.: Discussion on "The power of monitoring: how to make the most of a contaminated sample". Stat. Methods Appl. (2017). https://doi.org/10.1007/s10260-017-0416-9

Agostinelli, C., Greco, L.: Weighted likelihood estimation of multivariate location and scatter. Test (2018). https://doi.org/10.1007/s11749-018-0596-0

Atkinson, A., Riani, M., Cerioli, A.: Exploring Multivariate Data with the Forward Search. Springer, Berlin (2013)

Basu, A., Lindsay, B.: Minimum disparity estimation for continuous models: efficiency, distributions and robustness. Ann. Inst. Stat. Math. **46**(4), 683–705 (1994)

Bouveyron, C., Brunet-Saumard, C.: Model-based clustering of high-dimensional data: a review. Comput. Stat. Data Anal. **71**, 52–78 (2014)

Bryant, P.: Large-sample results for optimization-based clustering methods. J. Classif. **8**(1), 31–44 (1991)

Campbell, N.: Mixture models and atypical values. Math. Geol. **16**(5), 465–477 (1984)

Celeux, G., Govaert, G.: Comparison of the mixture and the classification maximum likelihood in cluster analysis. J. Stat. Comput. Simul. **47**(3–4), 127–146 (1993)

Cerioli, A.: Multivariate outlier detection with high-breakdown estimators. J. Am. Stat. Assoc. **105**(489), 147–156 (2010)

Cerioli, A., Farcomeni, A.: Error rates for multivariate outlier detection. Comput. Stat. Data Anal. **55**(1), 544–553 (2011)

Cerioli, A., Riani, M., Atkinson, A., Corbellini, A.: The power of monitoring: how to make the most of a contaminated sample. Stat. Methods Appl. (2017). https://doi.org/10.1007/s10260-017-0409-8

Colonna, J.G., Gama, J., Nakamura, E.: Recognizing Family, Genus, and Species of Anuran Using a Hierarchical Classification Approach. Lecture Notes in Computer Science, pp. 198–212. Springer, Berlin (2016)

Coretto, P., Hennig, C.: Robust improper maximum likelihood: tuning, computation, and a comparison with other methods for robust gaussian clustering. J. Am. Stat. Assoc. **111**(516), 1648–1659 (2016)

Coretto, P., Hennig, C.: Consistency, breakdown robustness, and algorithms for robust improper maximum likelihood clustering. J. Mach. Learn. Res. **18**(1), 5199–5237 (2017)

Day, N.: Estimating the components of a mixture of normal distributions. Biometrika **56**(3), 463–474 (1969)

Dempster, A., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B Methodol. **39**, 1–38 (1977)

Dotto, F., Farcomeni, A.: Robust inference for parsimonious model-based clustering. J. Stat. Comput. Simul. **89**(3), 414–442 (2019)

Dotto, F., Farcomeni, A., Garcia-Escudero, L.A., Mayo-Iscar, A.: A reweighting approach to robust clustering. Stat. Comput. **28**(2), 477–493 (2016)

Elashoff, M., Ryan, L.: An em algorithm for estimating equations. J. Comput. Graph. Stat. **13**(1), 48–65 (2004)

Farcomeni, A., Greco, L.: Robust Methods for Data Reduction. CRC Press, Boca Raton (2015a)
Farcomeni, A., Greco, L.: S-estimation of hidden Markov models. Comput. Stat. **30**(1), 57–80 (2015b)
Fraley, C., Raftery, A.: How many clusters? Which clustering method? Answers via model-based cluster analysis. Comput. J. **41**(8), 578–588 (1998)
Fraley, C., Raftery, A.: Model-based clustering, discriminant analysis, and density estimation. J. Am. Stat. Assoc. **97**(458), 611–631 (2002)
Fraley, C., Raftery, A., Murphy, T., Scrucca, L.: mclust version 4 for r: normal mixture modeling for model-based clustering, classification, and density estimation. Technical Report 597, University of Washington, Seattle (2012)
Fritz, H., Garcia-Escudero, L., Mayo-Iscar, A.: A fast algorithm for robust constrained clustering. Comput. Stat. Data Anal. **61**, 124–136 (2013)
Garcia-Escudero, L., Gordaliza, A., Matran, C., Mayo-Iscar, A.: A general trimming approach to robust cluster analysis. Ann. Stat. **36**, 1324–1345 (2008)
García-Escudero, L.A., Gordaliza, A., Matrán, C., Mayo-Iscar, A.: Exploring the number of groups in robust model-based clustering. Stat. Comput. **21**(4), 585–599 (2011)
Garcia-Escudero, L., Gordaliza, A., Matran, C., Mayo-Iscar, A.: Avoiding spurious local maximizers in mixture modeling. Stat. Comput. **25**(3), 619–633 (2015)
Greco, L.: Weighted likelihood based inference for $p(x<y)$. Commun. Stat. Simul. Comput. **46**(10), 7777–7789 (2017)
Helliwell, J., Layard, R., Sachs, J.: World Happiness Report 2018 (2018)
Kuchibhotla, A., Basu, A.: A general set up for minimum disparity estimation. Stat. Probab. Lett. **96**, 68–74 (2015)
Kuchibhotla, A., Basu, A.: A minimum distance weighted likelihood method of estimation. Technical report, Interdisciplinary Statistical Research Unit (ISRU), Indian Statistical Institute, Kolkata, India (2018). https://faculty.wharton.upenn.edu/wp-content/uploads/2018/02/attemptv4p1.pdf. Accessed 17 Jan 2018
Lee, S., McLachlan, G.: Finite mixtures of multivariate skew t-distributions: some recent and new results. Stat. Comput. **24**(2), 181–202 (2014)
Lin, T.: Robust mixture modeling using multivariate skew t distributions. Stat. Comput. **20**(3), 343–356 (2010)
Markatou, M.: Mixture models, robustness, and the weighted likelihood methodology. Biometrics **56**(2), 483–486 (2000)
Markatou, M., Basu, A., Lindsay, B.G.: Weighted likelihood equations with bootstrap root search. J. Am. Stat. Assoc. **93**(442), 740–750 (1998)
Maronna, R., Jacovkis, P.: Multivariate clustering procedures with variable metrics. Biometrics **30**(3), 499–505 (1974)
McLachlan, G., Peel, D.: Finite Mixture Models. Wiley, New York (2004)
McLachlan, G.J., Peel, D., Bean, R.: Modelling high-dimensional data by mixtures of factor analyzers. Comput. Stat. Data Anal. **41**(3–4), 379–388 (2003)
Neykov, N., Filzmoser, P., Dimova, R., Neytchev, P.: Robust fitting of mixtures using the trimmed likelihood estimator. Comput. Stat. Data Anal. **52**(1), 299–308 (2007)
R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2019). https://www.R-project.org/
Rousseeuw, P., Van Zomeren, B.: Unmasking multivariate outliers and leverage points. J. Am. Stat. Assoc. **85**(411), 633–639 (1990)
Symon, M.: Clustering criterion and multi-variate normal mixture. Biometrics **77**, 35–43 (1977)