Electronic Journal of Applied Statistical Analysis
EJASA, Electron. J. App. Stat. Anal.
http://siba-ese.unisalento.it/index.php/ejasa/index
e-ISSN: 2070-5948
DOI: 10.1285/i20705948v13n1p146

Outlier detection through mixtures with an improper component

By Novi Inverardi, Taufer

Published: 02 May 2020

Electronic Journal of Applied Statistical Analysis
Vol. 13, Issue 01, May 2020, 146-163
DOI: 10.1285/i20705948v13n1p146

Outlier detection through mixtures with an improper component

Pier Luigi Novi Inverardi*and Emanuele Taufer

University of Trento - Department of Economics and Management
via Inama, 5 - 38123 Trento (Italy)

Published: 02 May 2020

The paper investigates the use of a finite mixture model with an additional uniform density for outlier detection and robust estimation. The main contribution of this paper lies in the analysis of the properties of the improper component and the introduction of a modified EM algorithm which, beyond providing the maximum likelihood estimates of the mixture parameters, endogenously provides a numerical value for the density of the uniform distribution used for the improper component. The mixing proportion of outliers may be known or unknown. Applications to robust estimation and outlier detection will be discussed with particular attention to the normal mixture case.

keywords: Gaussian mixture, outlier detection, robust estimation, improper EM algorithm, improper component.

1 Introduction

Mixture models have been widely employed in many statistical procedures such as discriminant analysis or cluster analysis and primarily for checking the stability of classical inferential techniques under violation of the underlying assumptions. Mixture models also play a relevant role in the detection of outliers, an important task in data mining since it has applications in problems such as fraud detection, intrusion detection, analysis of microarray experiments for gene expression and data cleaning (Chandola et al., 2009).

*Corresponding author: pierluigi.noviinverardi@unitn.it

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

The goal of outlier detection is to find an unusual/atypical datum (outlier) in a given dataset. Many measures have been proposed to detect/classify outliers. This goal can also be addressed by mixture modelling (McLachlan and Peel, 2000), in which one or a few components are reserved for outliers or contaminants (Banfield and Raftery, 1993), (Longford and D'Urso, 2011).

Some recent and less recent contributions in outlier detection are those of Kutsuna and Yamamoto (2017) based on binary decision diagrams, Ernst and Haesbroeck (2017) which consider methods for spatial data, (Longford and D'Urso, 2011) and Longford (2013) which discuss a method based on a mixture with an improper component; Limas et al. (2004) which address multivariate non normal data and Yamanishi et al. (2004) which discuss an algorithm based on Gaussian mixtures and a kernel version of it.

This paper investigates the use of a finite mixture model for outlier detection and robust estimation of the parameters of a given distribution. More specifically, the use of an improper component of the mixture will be the main tool to detect outliers. The addition of a mixture component accounting for noise as a uniform distribution was discussed by Fraley and Raftery (1998) and Hennig (2004) in the context of cluster analysis while some other recent contributions (given above) introduce it in outlier detection. A common feature of the these papers is that the numerical value of the improper component is determined exogenously.

The main contribution of this paper lies in the analysis of the properties of the improper component and the introduction, as we term it, of an Improper EM-Algorithm (I-EM) which, beyond providing the maximum likelihood estimates of the mixture parameters, endogenously provides a numerical value for the density of the uniform distribution used for the improper component. The mixing proportion of outliers may be known or unknown. The posterior probabilities permit to identify the observations which are outliers and to use them to reduce their influence in the parameter estimation procedure. Applications to robust estimation and outlier detection will be discussed.

In the following section we present the rationale of the proposed procedure focusing on a simple mixture of two components f_1 and f_0 which may be univariate or multivariate densities. This simple structure allows us to analyze from a theoretical point of view some important features of the procedure. Section 3 is devoted to the generalisation of the I-EM algorithm which the proper component may itself be a mixture. We compare this algorithm to some of its competitors. The last section is devoted to the conclusions.

2 The proposed procedure

Mixture distributions may represent a natural environment for facing the problem of outliers detection. Consider a mixture pdf $f(y;\theta) = \pi\, f_1(y;\theta) + (1-\pi) f_0(y)$, where $\pi \in (0,1)$ and f_1 depends on some unknown parameter $\theta \in \Theta \subseteq \mathbb{R}^k$, $k \in I\!N$ (and may be a mixture itself) while the component f_0 can be regarded as an improper pdf that may generate/accommodate outliers or atypical data. The commonly used method of estimating the parameters of a mixture model is the EM-Algorithm, based on a sample $(z_1, y_1), \ldots, (z_n, y_n)$ where z_j are the values of the (unobserved) indicator variable Z_j,

with $P[Z_j = 1] = \pi$ and $P[Z_j = 0] = 1 - \pi$. Conditionally on $Z_j = \alpha$, Y_j has pdf $f_\alpha(y)$, $\alpha = 0, 1$. The complete data log-likelihood function (if the value of Z_j is known for each j) is given by

$$\ell_C(\pi, \theta) = \sum_{j=1}^{n} [z_j \log \pi + (1 - z_j) \log(1 - \pi)] + \sum_{j=1}^{n} z_j \log f_1(y_j; \theta) + \\ + \sum_{j=1}^{n} (1 - z_j) \log f_0(y_j). \quad (1)$$

The first sum in (1) is a Binomial log-likelihood which is maximized at $\pi = \frac{1}{n} \sum_{j=1}^{n} z_j$. The second sum indicates the log-likelihood function for θ. Typically, the third sum would also be a log-likelihood function for parameters that determine f_0, but for the moment f_0 is regarded as known. For instance, if f_1 is a normal pdf with unknown mean and variance, we set $\theta = (\mu, \sigma^2)$, and the second term in (1) reaches its maximum at

$$\mu = \frac{\sum_{j=1}^{n} y_j}{\sum_{j=1}^{n} z_j}, \quad \sigma^2 = \frac{\sum_{j=1}^{n} (y_j - \mu)^2}{\sum_{j=1}^{n} z_j}. \quad (2)$$

For unobserved z_j, the E-Step of the EM-Algorithm replaces the z_j by their conditional expectation

$$\pi_{1j} = E[Z_j | y_j] = \frac{\pi f_1(y_j; \theta)}{\pi f_1(y_j; \theta) + (1 - \pi) f_0(y_j)}, \quad j = 1, 2, \ldots, n, \quad (3)$$

and the M-Step maximizes (1) with the z_j replaced by the π_{1j}. Assuming again that f_0 is known and that f_1 is a normal pdf with unknown mean and variance, the EM-Algorithm starting with some initial values $(\pi^{(0)}, \theta^{(0)})$, iterates until convergence between the following two steps:

E-**Step**:

$$\pi_{1j} = E[Z_j | y_j] = \frac{\pi f_1(y_j; \theta)}{\pi f_1(y_j; \theta) + (1 - \pi) f_0(y_j)}, \quad j = 1, 2, \ldots, n, \quad (4)$$

using current parameter values in the evaluation of f_1, and

M-**Step**:

$$\pi = \frac{1}{n} \sum_{j=1}^{n} \pi_{1j} \quad (5)$$

$$\mu = \frac{\sum_{j=1}^{n} \pi_{1j} y_j}{\sum_{j=1}^{n} \pi_{1j}}, \quad (6)$$

$$\sigma^2 = \frac{\sum_{j=1}^{n} \pi_{1j} (y_j - \mu)^2}{\sum_{j=1}^{n} \pi_{1j}}. \quad (7)$$

2.1 The Improper EM-Algorithm

The key idea in our approach is to choose for f_0 a density which does not depend on y, to accommodate arbitrarily large outliers and values that are unlikely under f_1. To achieve this, we will set

$$f_0(y_j) = c, \quad j = 1, \ldots, n, \tag{8}$$

where $c > 0$ is a constant. We will refer to f_0 as the *improper component* of the mixture distribution $f(y;\theta)$ and the choice of c represents one of the most crucial aspects of our approach.

Suppose at first that $\pi \in (0,1)$ is given. Note that if the mixing proportion π is assumed to be known, then the equation (5) disappears from the calculations while all other equations remain unchanged. Now continuing with the familiar normal density for f_1, the EM-Algorithm then uses equations (6) and (7) for the M-Step, while the E-Step is given by equation (3). An important question now arises: how should c be chosen? By the usual rule of conditioning,

$$E[Z_j] = \pi, \quad E[Z_j|y_j] = \pi_{1j}, \quad E[Z_j] = E[E(Z_j|y_j)] \tag{9}$$

which suggest to exploit equation (5) for choosing the value of c. Consequently equation (5) is brought back in again, but its purpose is now to give a hint for choosing appropriately the value of c rather than estimate π. Writing $f_{1j} = f(y_j;\theta)$ for sake of simplicity and using (3) in equation (5) we obtain,

$$\begin{aligned} 0 &= \sum_{j=1}^{n} \frac{\pi f_{1j}}{\pi f_{1j} + (1-\pi)\cdot c} - n\,\pi \\ &= \pi(1-\pi) \sum_{j=1}^{n} \frac{f_{1j} - c}{\pi(f_{1j} - c) + c} \\ &= \pi(1-\pi)\, h(c,\pi) \end{aligned} \tag{10}$$

or equivalently,

$$h(c,\pi) = \sum_{j=1}^{n} \frac{f_{1j} - c}{\pi(f_{1j} - c) + c} = 0. \tag{11}$$

It is straightforward to prove that $h(c,\pi)$ is a decreasing function of c with limits $\frac{n}{\pi}$ and $-\frac{n}{1-\pi}$ when $c = 0$ and $c \to +\infty$, respectively.

Therefore for each given $\pi \in (0,1)$ there is a unique solution $c > 0$ such that the equation (11) holds. Thus, for a given π, we will use as estimates of μ, σ^2, π_{0j} and π_{1j} the values obtained by iterating until convergence the following procedure:

I-EM1

Step 1: **Fix** the precision level ϵ and the value $\pi = \pi_{opt}$, the known proportion of non-contaminated observations

Step 1.1 (**Improper** E**-Step**): **Evaluate** f_{1j} at the current parameter values

Step 1.2: **Find** the value c_0 which solves the equation (11)

Step 1.3: Given π_{opt}, f_{1j} and $f_{0j} = c_0$ from equation (3) **compute** the posterior probabilities

$$\pi_{1j} = \frac{\pi_{opt}\, f_{1j}}{\pi_{opt}\, f_{1j} + (1 - \pi_{opt})\, c_0} \quad \text{and} \quad \pi_{0j} = 1 - \pi_{1j}$$

Step 1.4 (M-**Step**): **Update** μ and σ^2 according to the equations (6) and (7) using the current values of π_{1j} and π_{0j} and compute the $logLik$ for the $k = 1, 2, 3, \ldots$ current iteration

Step 1.5: If

$| logLik^{(k)} - logLik^{(k-1)} | > \epsilon$: **Iterate** again Step 1.1 - Step 1.5

$| logLik^{(k)} - logLik^{(k-1)} | \leq \epsilon$: **Stop** the estimation procedure: **goto** Step 2

Step 2: **Return** final parameter estimates $\hat{\mu}$ and $\hat{\sigma}^2$, $\hat{c}$ and posterior probabilities $\hat{\pi}_{1j}$ and $\hat{\pi}_{0j}$ for the data

The Step 1.2 of I-EM1 involves the solution of the equation $h(c, \pi) = 0$ with respect to c and this may be difficult to do. For fixed π, c_0 can be found by the Newton-Raphson algorithm, using the expression (11). The objective function is well behaved and the algorithm is unlikely to require more than a handful of iterations.

The purpose of the following two simple examples is to illustrate the role of the I-EM1 procedure in terms of robust estimation and suggest a possible rule for choosing the value of π when no information is available on it. Section 2.3 and Section 3.1 will give a definitely more convincing examples about how I-EM1 works as a robust estimator and/or outlier detector on more complex real and simulated datasets.

Example 1. 35 observations were generated from a $\mathcal{N}(0, 1)$ distribution, and 5 further data values equal to -15, -30 and 31, 40 and 6 were added as atypical data. The algorithm was run with $\pi = 35/40 = 0.8750$, using the sample mean and variance of all 40 data points as initial parameters values; the estimation procedure stops when the difference between two consecutive log likelihood functions is less than 10^{-6}. Running the Improper EM algorithm, seven iterations were required for reaching convergence of the estimation procedure, giving final values $\hat{\mu} = -0.0361$ and $\hat{\sigma}^2 = 0.9491$ as displayed in Table 1:

The final values practically coincide with the mean and variance of the 35 "good" observations (-0.0362 and 0.9484 respectively). At the solution posterior probabilities $\hat{\pi}_{1j}$ are equal to 1 for the good (or non contaminated) observations (except for observation 9 where the posterior probability was equal to 0.9996) or equal to 0 for the atypical observations (except for observation 40 where the posterior probability was equal to 0.0008). This is the reason for that the estimates given back by the Improper EM algorithm slightly differ from mean and variance of the 35 non contaminated observations.

Table 1: Protocol of the estimation procedure via I-EM algorithm with known non-contaminated proportion π

Iteration	c	Mean	Variance	logLik Difference
1.000	1.7073E-02	-2.0024E-01	1.346E+01	4.9854E+01
2.000	1.9655E-02	1.1479E-01	1.910E+00	1.1447E+01
3.000	2.8703E-03	-2.2254E-02	1.021E+00	5.0467E+00
4.000	3.6900E-05	-3.5937E-02	9.497E-01	2.4592E+00
5.000	1.7400E-05	-3.6077E-02	9.491E-01	2.2827E-02
6.000	1.7300E-05	-3.6078E-02	9.491E-01	1.7036E-04
7.000	1.7300E-05	-3.6078E-02	9.491E-01	1.9120E-06

The results of Example 1 are practically the same as removing the five atypical observations from the sample. This is not at all surprising because we have assumed the mixing proportion π to be known. But, in most practical situations, this will be not the case; so that, we will need some rule for choosing π. Example 2 is a natural continuation of Example 1 and provides some ideas in that sense.

Example 2. Using the same dataset as in Example 1, the Improper EM-algorithm was run for values of π between 0.5 and 0.99 (step: 0.01). Figure 1 shows $\hat{c}$, $\log \hat{c}$, $\hat{\mu}$, $\ln(\hat{\sigma}^2)$ as functions of π. Once π exceeds the true (correct) value $\frac{35}{40} = 0.875$, the parameter estimates change suddenly, but the most striking aspect of the graph is that at $\pi = \frac{35}{40}$ the value of $\hat{c}$ reaches practically 0. This suggest a heuristic rule: choose the smallest value of π for which $\hat{c}$ is "very small" (minimum). Table 2 illustrates the Improper EM algorithm protocol: as before, the estimation procedure stops when the difference between two consecutive log likelihood functions is less than 10^{-6} and thirteen iterations were needed for reaching convergence getting the final values $\hat{\mu} = -0.0348$, $\hat{\sigma}^2 = 0.9503$ and $\hat{\pi} = 0.880$.

Some theoretical analysis supporting this rule is given later, but just for comparison Figure 2 shows the analogous graph using a sample of size $n = 100$ from standard normal distribution without contamination. In Figure 2, π varies from 0.5 to 0.99 (step: 0.01); note that in that interval $c(\pi)$ decreases regularly and never reaches the value 0.

2.2 More on the behaviour of c and the choice of π

To better understand the behaviour of c as a function of π it is useful to study the case there is no unknown parameter in f_1. Of course this case has no practical importance, but it gives some theoretical insights.

Table 2: Protocol of the estimation procedure via I-EM algorithm with unknown non-contaminated proportion π

Iteration	c	Mean	Variance	logLik Difference
1.000	1.635E-02	-2.020E-01	1.400E+01	1.044E+01
2.000	1.635E-02	1.143E-01	1.974E+00	1.144E+01
3.000	1.097E-03	3.045E-03	1.175E+00	5.310E+00
4.000	4.000E-06	-1.358E-02	1.079E+00	5.264E-01
5.000	2.000E-06	-2.428E-02	1.014E+00	6.634E-01
6.000	2.000E-06	-3.193E-02	9.679E-01	1.470E+00
7.000	2.000E-06	-3.431E-02	9.534E-01	2.337E-01
8.000	2.000E-06	-3.474E-02	9.508E-01	4.946E-02
9.000	2.000E-06	-3.481E-02	9.504E-01	7.934E-03
10.000	2.000E-06	-3.482E-02	9.503E-01	1.209E-03
11.000	2.000E-06	-3.482E-02	9.503E-01	1.828E-04
12.000	2.000E-06	-3.482E-02	9.503E-01	2.761E-05
13.000	2.000E-06	-3.482E-02	9.503E-01	4.169E-06

Let $f_{1j} = f_1(y_j),\ j = 1, 2, \ldots, n$ be known positive constants. Since for any given $\pi \in (0, 1)$, equation (11) has a unique solution, it implicitly defines a function

$$c = c(\pi), \quad 0 < \pi < 1. \tag{12}$$

Lemma 1 *For given constants $f_{11}, f_{12}, \ldots, f_{1n}$, the function $c = c(\pi)$ is monotonically decreasing from $c(0) = \frac{1}{n}\sum_{j=1}^{n} f_{1j}$ to $c(1) = n\left(\sum_{j=1}^{n} f_{1j}^{-1}\right)^{-1}$. If all f_{1j} are identical, then $c(\pi)$ is constant.*

Proof: The values $c(0)$ and $c(1)$ follow directly by solving the equations $h(c, 0) = 0$ and $h(c, 1) = 0$. Equation (11) defines a contour of constant value 0 of $h(c, \pi)$ considered as a function of two variables c and π. Since

$$\frac{\partial}{\partial c} h(c, \pi) < 0, \quad \forall \pi \in (0, 1)$$

and

$$\frac{\partial}{\partial \pi} h(c, \pi) \leq 0, \quad \forall c > 0$$

monotonicity follows.

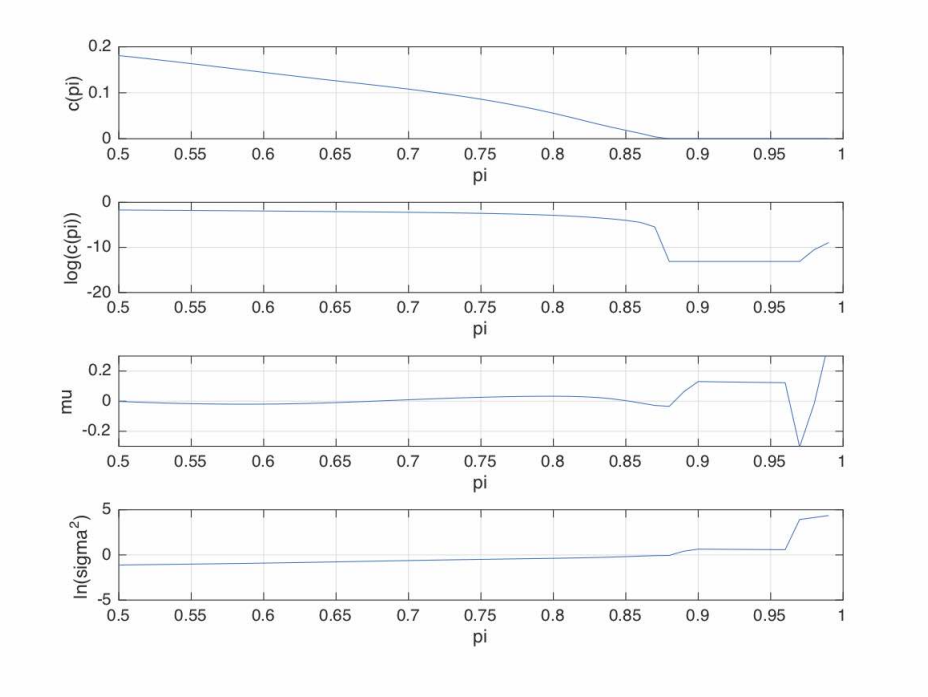

Figure 1: Behaviour of $\hat{c}$, $\log(\hat{c})$, $\hat{\mu}$ and $\log(\hat{\sigma}^2)$ as functions of π for Example 1 contaminated data.

Lemma 1 suggests that if the parameter θ determining f_1 is reasonably well estimated by the data at a given value π_0 of π, we can expect $\hat{c}(\pi)$ to decrease monotonically in a neighbourhood of π_0. On the other hand, if a small change in π affects the parameter values drastically, then the values of the f_{1j} will change, and the function $c(\pi)$ might then increase or stay roughly constant. This gives some support to the heuristic rule formulated at the end of Example 1.

More support for the heuristic rule can be given by considering the case where some of the observations are outside the support of the *pdf* f_1, as given in the following theorem.

Lemma 2 *Suppose the support of f_1 does not depend on the unknown parameter θ, and $n_2 < n$ of the observations are outside of the support, i.e., $f_{1j} = 0$ for some n_2 observations, and $n_1 = n - n_2$ observations are inside. Then,*

$$\hat{c}\left(\frac{n_1}{n}\right) = 0. \tag{13}$$

Proof: Suppose for simplicity that $f_{1j} > 0$ for $j = 1, 2, \ldots, n_1$ and $f_{1j} = 0$ for $j = n_1 + 1, \ldots, n$. Then

$$h(c, \pi) = \sum_{j=1}^{n_1} \frac{f_{1j} - c}{\pi\left(f_{1j} - c\right) + c} - \frac{n_2}{1 - \pi}, \tag{14}$$

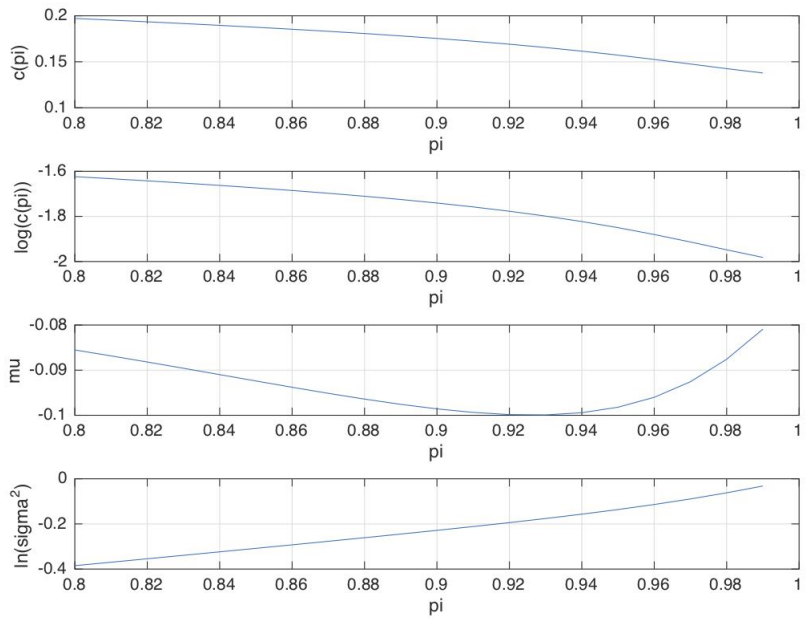

Figure 2: Behaviour of $\hat{c}$, $\log(\hat{c})$, $\hat{\mu}$ and $\log(\hat{\sigma}^2)$ as functions of π for non-contaminated standard normal data.

and $h(0; n_1/n) = 0$.

By Lemma 2, we can vary π only in the range from 0 to n_1/n, because for $\pi > n_1/n$ there exists no positive constant c that solves the equation (11). This is perfectly reasonable because at this point we have exhausted" all information given by the n_1 observations within the support of f_1. For further illustration, suppose again that θ is known, and that $f_{1j} = \alpha > 0$ for $j = 1, 2, \ldots, n_1$, while $f_{1j} = 0$ for $j = n_1 + 1, \ldots, n$. Then the equation $h(c; \pi) = 0$ has an explicit solution

$$c(\pi) = \alpha \, \frac{\frac{n_1}{n} - \pi}{1 - \pi}, \tag{15}$$

showing that values of π larger then n_1/n must be excluded.

In practical applications with unknown θ one would probably exclude observations outside the support of f_1 a priori, but nevertheless Lemma 2 gives a valuable insight: if some n_2 observations are distinct outliers, and π approaches the value n_1/n from below, then f_{1j} will be extremely small for the outliers, and by a continuity argument we can the expect the function $\hat{c}(\pi)$ to take values very close to zero for $\pi \geq n_1/n$. On the other hand, for an uncontaminated distribution we expect that $\hat{c}(\pi)$ be monotonically decreasing in π. Thus, any increase in $\hat{c}(\pi)$ indicates the observations that are very unlikely under f_1 are becoming influential for the parameter estimates.

Turning to the general setup where f_1 depends on the parameter θ, the above lemma suggests that if the parameter θ determining f_1 is reasonably well estimated by the data at a given value π_{opt} of π, we can expect $c(\pi)$ to decrease monotonically when we are approaching π_{opt}. In other words, if a small change in π affects the parameter values drastically, then the values of the f_{1j} will change and the function $c(\pi)$ might then increase or stay roughly constant. Keeping in mind that π is the proportion of "good" data which follow the component f_1 of the mixture and that the function (or component) $c(\pi)$ was introduced with the aim to accommodate arbitrarily large outliers and values that are unlikely under f_1, this provides an empirical rule for the choice of π.

Finally, the estimation procedure may be set in the following way:

I-EM2

Step 1: **Fix** the precision level ϵ, the starting value of $\pi \in (0, 1)$ and the increment $d\pi$ of π

Step 1.1 **Run** the I-EM1 algorithm

Step 1.2 **Save** π, c_0, $h(c_0, \pi)$, final parameter estimates $\hat{\mu}$ and $\hat{\sigma}^2$, posterior probabilities $\hat{\pi}_{1j}$ and $\hat{\pi}_{0j}$ for the observations; then **goto** Step 2

Step 2: **Until** $\pi < 1$, **put** $\pi = \pi + d\pi$ and **goto** Step 1.1

Step 3 : **Return** as estimate of the unknown value of π the value π_{opt} corresponding to the minimum of c_0 values and the corresponding values of c_0, $\hat{\mu}$, $\hat{\sigma}^2$, $\hat{\pi}_{1j}$ and $\hat{\pi}_{0j}$ as estimates of the improper density component, parameters and posterior probabilities, respectively

2.3 Adulteration in wine production

Monetti et al. (1996) studied the chemical composition of $n = 344$ commercial samples of concentrated grape must (CGM) in wine production. In sugar adulteration controls, chemical values of a suspect sample are compared with those of a reference one. To improve classification the authors suggest using a multivariate approach and show that data can be modelled with a two multivariate normal components mixture.

Their dataset contains data collected on four variables (D/H_I, *myo-* and *scyllo-*inositol, D/H_{II}) suitable for discovering adulterations with added sugar from plants other than grapes: two polyalcohols (*myo*-inositol and *scyllo*-inositol) and two D/H ratios (D/H_I and D/H_{II}) in the methyl and methylene position of ethanol which depend on photosynthetic and physiological factors. The polyalcohol variables *myo*-inositol and *scyllo*-inositol have been logarithmically transformed. See plots of the data in Figure 3.

Here we are interested in separation of the group of adulterated grape samples from the other and in parameter estimation of the non-adulterated component of the CGM

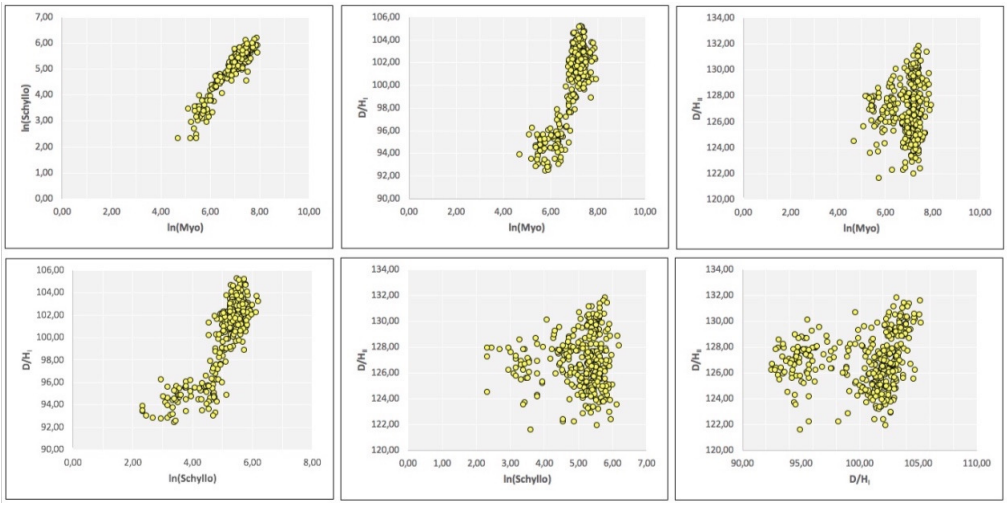

Figure 3: Plots of the Wine data along the $(\ln(Myo) - \ln(Schyllo)))$, $(\ln(Myo)$-$D/H_I)$, $(\ln(Myo)$-$D/H_{II})$, $(\ln(Schyllo)$-$D/H_I)$, $(\ln(Schyllo)$-$D/H_{II})$ and $(D/H_I$-$D/H_{II})$ directions.

samples without assuming knowledge of reference values. In practice the second component associated with the adulterated samples will be modelled through the density f_0.

To this end the *I-EM2* algorithm was run using the sample mean and covariance matrix of all 344 data points as initial values given by

$$\begin{pmatrix} 100.3831 & 6.9556 & 5.0907 & 126.8483 \end{pmatrix}$$

and

$$\begin{pmatrix} 11.2105 & 1.6893 & 2.1252 & 0.9270 \\ 1.6893 & 0.3625 & 0.4223 & 0.0853 \\ 2.1252 & 0.4223 & 0.5629 & 0.0532 \\ 0.9270 & 0.0853 & 0.0532 & 4.0071 \end{pmatrix}$$

The *I-EM2* algorithm returned $\hat{\pi} = 0.76$ for the proportion of non-adulterated musts in the original sample, very close to the number of non-adulterated samples equal to 261 as discussed at length in Monetti et al. (1996). In other words, 24% of the original 344 (83) concentrated grape musts have been recognized as adulterated by sugar addition.

Once π has been determined, 25 iterations have been required to the known proportion *I-EM1* to reach convergence, giving back the final estimates for mean vector and covariance matrix of the set of four variables (D/H_I, *myo*-inositol, *scyllo*-inositol and D/H_{II}) for non-adulterated samples

$$\hat{\mu}' = \begin{pmatrix} 102.0385 & 7.2400 & 5.4354 & 126.8276 \end{pmatrix}$$

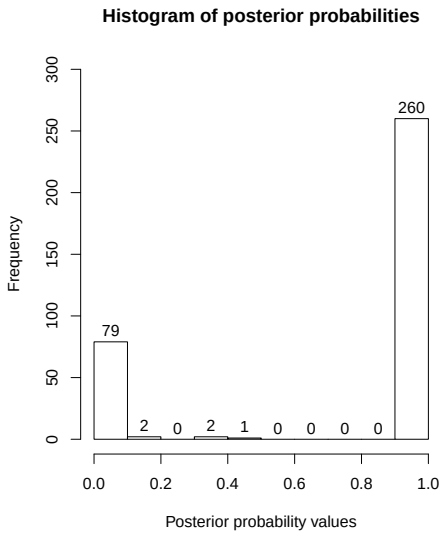

Figure 4: Posterior probabilities histogram (Wine data - Monetti et al. (1996))

and

$$\hat{\Sigma} = \begin{pmatrix} 2.2549 & 0.0930 & 0.2037 & 1.1761 \\ 0.0930 & 0.0594 & 0.0472 & 0.0725 \\ 0.2037 & 0.0472 & 0.0774 & 0.0197 \\ 1.1761 & 0.0725 & 0.0197 & 4.4815 \end{pmatrix}$$

Finally the inspection of the posterior probabilities associated to the original units permits to identify the adulterated (or outlier) samples. Figure 4 refers to the histogram of posterior probabilities and we may adopt a threshold of 0.4 or 0.5 to split the original 344 wine samples in adulterated ($\pi_{1j} \leq 0.5$) and non-adulterared ($\pi_{1j} > 0.5$). Estimates parameters of non-adulterated samples strongly agree with those of Monetti et al. (1996).

3 Generalizing the algorithm

Depending on the form of f_1, which might be a mixture itself, it might be not always possible to determine the value π_{opt} as discussed in Section 2.2 since, as simulations show, a numerical search for π_{opt} may lead to a trivial unit value.

Notwithstanding, it will be shown that the I-EM algorithm can still perform very well in outlier detection when the posterior probabilities associated with f_0 are used to rank potential outlying observations

Under the assumption that f_1 is Gaussian or a finite Gaussian mixture, we propose here a generalization of the *I-EM2* algorithm which does not rely on a fixed value of $\pi = \pi_{opt}$, rather, it provides an automatic update of the value π. As we see, there will be no formal

proof the final value of π obtained by the algorithm will be a good estimate of the true proportion of outliers in the data, however, the posterior probabilities associated with f_0 do provide an effective tool for evaluating each observation. The proposed algorithm runs as follows:

I-EM3

Step 1: in M- **Step** of the I-EM2 algorithm include updating of π according to equation (5)

Step 2: **Return** final parameter estimates $\hat{\mu}$ and $\hat{\sigma}^2$, $\hat{\pi}$, $\hat{c}$ and posterior probabilities $\hat{\pi}_{1j}$ and $\hat{\pi}_{0j}$ for the data

Extensive simulations show that the performance of the algorithm in outlier detection is rather insensitive to the initial choice of π and better performances are observed when π is in the range $[0.6, 0.9]$.

If f_1 is a mixture of K normal (multivariate) distributions, set $\pi_k = \pi/K$, $k = 1, \ldots, K$ and use the standard updating equations of the EM algorithm. Also, an initial estimate of f_1 can be obtained by separate analysis, using, e.g. the *mclust* function of the *mclust* package (Scrucca et al., 2017) with no pre-determined number of groups.

In the next Section synthetic and real data will be used to evaluate the I-EM algorithm as an automatic outlier detection method and show the relevance of the empirical rules given above.

3.1 Application to outlier detection

Comparisons will be done with established outlier detection methods: the one-class support vector machine method (SVM); the local outlier factor (LOF) and the Isolation Forest (IF). For these methods, packages for their implementation in R (R Core Team, 2018) are available. The following functions and packages were used: *ksvm* from the *kernlab* package (Karatzoglou et al., 2004), the *lofactor* function from the package *DMwR* (Torgo, 2010) and the function *IsolationTrees* in the *IsolationForest* package (Liu et al., 2008)

3.1.1 Gaussian mixture data

As a first example we replicate an experiment carried on in Yamanishi et al. (2004) where the data and outliers are provided by mixtures of tri-variate normal distributions. We generate random numbers from a distribution with density

$$f(y) = \pi f_1(y) + (1 - \pi) f_0(y) \tag{16}$$

where both densities f_1 and f_0 are a Gaussian mixture, namely

$$f_1(y) = \pi_{11} f(y; \mu_1, \Lambda_1) + (1 - \pi_{11}) f(y; \mu_2, \Lambda_2) \tag{17}$$

and the outlier part is

$$f_0(y) = \pi_{01} f(y; \mu_3, \Lambda_3) + \pi_{02} f(y; \mu_4, \Lambda_4) + \pi_{03} f(y; \mu_5, \Lambda_5) \tag{18}$$

with $\pi_{01} + \pi_{02} + \pi_{03} = 1$. The parameter values for the experiments are set to $\pi = 0.97$, $\pi_{11} = 1/2$, $\pi_{01} = \pi_{02} = \pi_{03} = 1/3$. For $f_1(y)$ set $\mu_1 = (0, 0, 0)'$, $\mu_2 = (0, 7, 0)'$, $\Lambda_1 = diag(1.2, 1, 1)$ and $\Lambda_2 = diag(1, 1.2, 1)$. For $f_0(y)$ set $\mu_3 = (3.5, 0, 0)'$, $\mu_4 = (3.5, 2, 0)'$, $\mu_5 = (9, -3, 0)'$ $\Lambda_3 = \Lambda_4 = \Lambda_5 = diag(0.2, 0.2, 0.2)$. We refer to the outliers generated around μ_3, μ_4, μ_5 respectively as groups 1, 2 and 3.

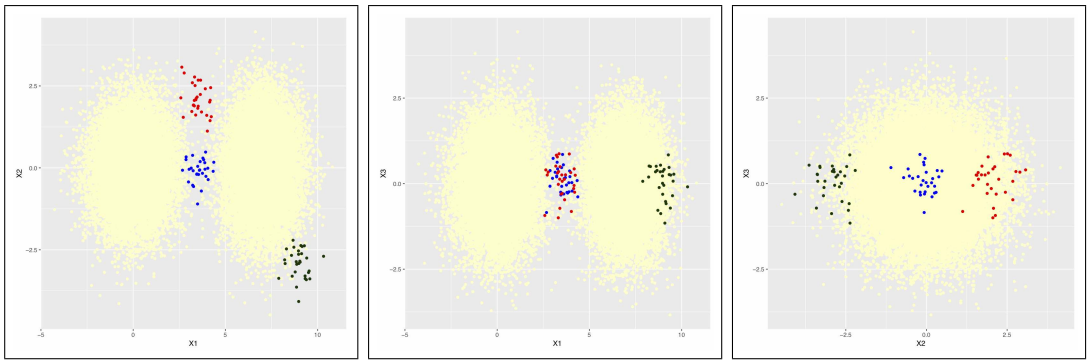

Figure 5: Plots of the data generated from the mixture (16) along the $x_1 - x_2$, $x_1 - x_3$ and $x_2 - x_3$ directions. In darker colours the outlier groups. Blue: group 1, red: group 2, dark green: group 3.

Figure 5 shows 30000 simulated data from the mixture (16) from different points of view. On the direction x_1, x_2 outliers from group 3 are far from the bulk distribution with respect to the Euclidean distance, hence are expected to be easily detected by while those in group 2, are between the means of the two Gaussian components of $f_1(y)$ and are expected to be difficult to detect.

The simulation is run as follows: a data set of $n = 30000$ was generated according to the parameters set above and an initial estimate of $f_1(y)$ was obtained by running the *mclust* function of the *mclust* package (Scrucca et al., 2017) with no pre-determined number of groups; then the *I-EM3* algorithm was run by setting an initial value $\pi = 0.8$ and the estimated π_{0i}, $i = 1, \ldots, 30000$ obtained by the algorithm where used to evaluate the probability of a single observation to be an outlier.

This experiment was iterated $m = 50$ times. For the function *ksvm* we set the following options: *kernel="rbfdot", kpar="automatic" and* $\nu = 0.1$; in the function *lofactor* the number of neighbours was set to $k = 10$ and $k = 50$ and the default setting was used for the function *IsolationTrees*.

In order to compare the performances, the area under the ROC curve (AUC) was computed separately for each group of outliers and for all outliers grouped together. Recall that the ROC curve is created by plotting the true positive rate (TPR) of outliers classification against the false positive rate (FPR) at various threshold settings; the

AUC is widely used in comparing the performance of classification techniques, see, e.g. Bradley (1997). In our computations for the ROC analysis the package *ROCR* (Sing et al., 2005) was used.

Table 3 reports the mean AUC and the mean timing for the experiment. We observe that the *I-EM3* always outperform the other algorithms. A careful inspection of the results in Yamanishi et al. (2004) shows that the *I-EM3* also outperforms, especially for groups 1 and 2, either the Gaussian mixture-based and kernel-based SmartSifter approach of Yamanishi et al. (2004) and the method of Burge and Shawe-Taylor (1997). As far as the setting of the value π for the *I-EM3* algorithm, we tried different starting values obtaining always similar results, indicating that this parameter may not be crucial in the performance of the algorithm.

It is worth noting that the estimated parameters for the two main (non-outlier groups) agree strongly with the parameter values obtained by applying the *mclust* function of the *mclust* package with no pre-determined number of groups.

Table 3: Gaussian mixture data: average AUC and timing over the 50 experiments. Data from mixture (16) with $n = 30000$ randomly generated data.

	G1	G2	G3	All	Timing
I-EM	0.965	0.991	0.984	0.981	0.515
SVM	0.909	0.818	0.979	0.903	23.643
IF	0.821	0.930	0.974	0.909	0.214
LOF10	0.645	0.603	0.678	0.642	68.177
LOF50	0.645	0.603	0.678	0.918	71.503

3.1.2 Swiss Banknotes

The Swiss Banknotes dataset (Flury and Riedwyl, 1988) is a well-known benchmark dataset and contains data on 100 genuine and 100 forged banknotes; for classification purposes there are 6 different measurements on each banknote (length, left, right bottom, top, diagonal). Since the variable diagonal perfectly separates the two groups, it is excluded from the computations; classification of outliers is then done using only the first five variables.

We run an experiment as follows: 50 new data sets are created; each data set contains the 100 genuine banknotes and 5 randomly selected forged banknotes. On each dataset the *I-EM3* and the other competing algorithms are run in order to detect the forged notes.

In order to run the *I-EM3* algorithm, since in this case the data does not show the presence of any mixture, the initial estimate of $f_1(y)$ is obtained by simply estimating

of the mean vector and covariance matrix of the observations. The initial value for π is again set to 0.8.

Table 4 contains the mean AUC and the mean timing of the algorithms tested. In this case LOF has the best performance, however the *I-EM3* is quite close to the highest precision.

For one of the datasets generated, Figure 6 reports the data points with size proportional to the probability of being an outlier produced by the *I-EM3* algorithm along the directions *Length-Left* and *Right-Bottom*. For this example all forged banknotes were assigned a probability of being an outlier greater than 0.98; as far as the genuine banknotes are concerned, 4% of them had an assigned probability of being an outlier greater than 0.90; 11% was assigned a probability greater than 0.5.

Table 4: Swiss banknotes data: average AUC and timing over the 50 experiments for outlier detection. Genuine banknotes:100; forged banknotes: 5.

	I-EM	SVM	IF	LOF10	LOF50
AUC	0.978	0.848	0.923	0.982	0.989
Timing	0.027	0.023	0.082	0.030	0.043

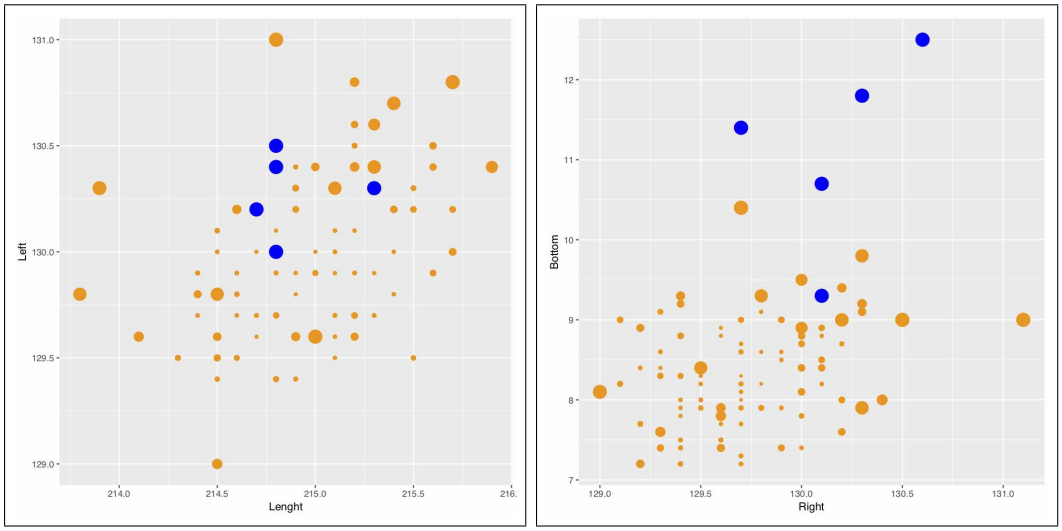

Figure 6: Plot of the datapoints with size proportional to the probability of being an outlier for one dataset of genuine and forged banknotes along the directions *Length-Left* and *Right-Bottom*. Blue (darker) points: forged; orange (lighter) points: genuine.

4 Conclusions

We have discussed and approach to robust estimation and outlier detection with finite mixtures and an improper component with special reference to Gaussian mixtures. An EM algorithm including an additional step taking care of the improper component has been discussed from practical and theoretical point of view. The approach has shown an excellent performance in many examples. A further interesting development will be to consider general mixtures or kernel based mixtures as, for example in Yamanishi et al. (2004).

Acknowledgement

The authors wish to thank two anonymous referees for their meticulous reading and constructive suggestions that surely improved the presentation of this paper.

References

Aitkin, M. and Tunnicliffe-Wilson, G. (1980). Mixture Models, Outliers, and the EM Algorithm. *Technometrics*, 22(3):325-331.

Banfield, J.D. and Raftery, A.E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49:803-821.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145-1159.

Burge, P. and Shawe-Taylor, J. (1997). Detecting cellular fraud using adaptive prototypes. In *Proc. of AI Approaches to Fraud Detection and Risk Management*, pages 9-13.

Chandola, V., Banerjee, A. and Kumar, V. (2009). Anomaly detection: A Survey. *ACM Computing Surveys*, 41(3), article 15:1-58.

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977). Maximum likelihood for incomplete data via EM algorithm. *Journal of the Royal Statistical Society*, Series B, 39:1-38.

Ernst, M., & Haesbroeck, G. (2017). Comparison of local outlier detection techniques in spatial multivariate data. *Data mining and knowledge discovery*, 31(2):371-399.

Flury, B., Riedwyl, H. (1988). *Multivariate Statistics. A Practical Approach.* Chapman and Hall.

Fraley, C. and Raftery, A.E. (1998). How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal*, 41:578-588.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J. and Stahel, W. A. (1986). *Robust Statistics. The Approach Based on Influence Functions.* John Wiley and Sons.

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

Huber, P. J. (1981) *Robust Statistics.* John Wiley and Sons.

Karatzoglou, A., Smola, A., Hornik, K., Zeileis, A. (2004). kernlab-an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1-20.

Kutsuna, T., Yamamoto, A. (2017). Outlier detection using binary decision diagrams. *Data mining and knowledge discovery*, 31(2):548-572.

Limas, M. C., Meré, J. B. O., de Pisón Ascacibar, F. J. M., González, E. P. V. (2004). Outlier detection and data cleaning in multivariate non-normal samples: the PAELLA algorithm. *Data Mining and Knowledge Discovery*, 9(2):171-187.

Longford, N.T., D'Urso, P. (2011) Mixture models with an improper component. *Journal of Applied Statistics*, 38:2511-2521.

Longford, N. T. (2013). Searching for contaminants. *Journal of Applied Statistics*, 40(9):2041-2055.

Liu, F.T., Ting, K.M., Zhou, Z.H. (2008). Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413-422.

McLachlan, G.J and Peel, D. (2000). *Finite Mixture Models*. John Wiley and Sons.

Monetti, A., Versini, G., Dalpiaz, G. and Raniero, F. (1996). Sugar Adulterations Control in Concentrated Rectified Grape Musts by Finite Mixture Distribution Analysis of the *myo*- and *scyllo*-Inositol Content and D/H Methyl Ratio of Fermentative Ethanol. *Journal of Agricultural and Food Chemistry*, 44:2194-2201.

R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Scrucca L., Fop M., Murphy T. B. and Raftery A. E. (2017). mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):205-233.

Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T. (2005). ROCR: visualizing classifier performance in R. *Bioinformatics*, 21(20):3940–3941.

Torgo, L. (2010). *Data Mining with R, learning with case studies*. Chapman and Hall/CRC.

Yamanishi, K., Takeuchi, J. I., Williams, G., Milne, P. (2004). On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. *Data Mining and Knowledge Discover*y, 8(3):275-300.