A Flexible EM-like Clustering Algorithm for Noisy Data

Violeta Roizman, Matthieu Jonckheere, Frédéric Pascal

▶ **To cite this version:**

Violeta Roizman, Matthieu Jonckheere, Frédéric Pascal. A Flexible EM-like Clustering Algorithm for Noisy Data. IEEE Transactions on Pattern Analysis and Machine Intelligence, In press, pp.1-14. 10.1109/TPAMI.2023.3337195 . hal-04366787

HAL Id: hal-04366787
https://hal.science/hal-04366787

Submitted on 2 Feb 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

arXiv:1907.01660v4 [stat.ML] 5 Oct 2020

A Flexible EM-like Clustering Algorithm for Noisy Data

Violeta Roizman[1,2], Matthieu Jonckheere[2,3], Frédéric Pascal[1]

[1]Laboratoire des Signaux et Systèmes (L2S), CentraleSupélec-CNRS-Université Paris-Sud, Université Paris-Saclay, 3, rue Joliot Curie, 91192, Gif-sur-Yvette, France
[2]Instituto de Cálculo, Universidad de Buenos Aires, Intendente Güiraldes 2160, Ciudad Universitaria-Pabellón II, Buenos Aires, Argentina
[3]IMAS-CONICET, Buenos Aires, Argentina

October 6, 2020

Abstract

Though very popular, it is well known that the EM for GMM algorithm suffers from non-Gaussian distribution shapes, outliers and high-dimensionality. In this paper, we design a new robust clustering algorithm that can efficiently deal with noise and outliers in diverse data sets. As an EM-like algorithm, it is based on both estimations of clusters centers and covariances. In addition, using a semi-parametric paradigm, the method estimates an unknown scale parameter per data-point. This allows the algorithm to accommodate for heavier tails distributions and outliers without significantly loosing efficiency in various classical scenarios. We first derive and analyze the proposed algorithm in the context of elliptical distributions, showing in particular important insensitivity properties to the underlying data distributions. We then study the convergence and accuracy of the algorithm by considering first synthetic data. Then, we show that the proposed algorithm outperforms other classical unsupervised methods of the literature such as k-means, the EM for Gaussian mixture models and its recent modifications or spectral clustering when applied to real data sets as MNIST, NORB and *20newsgroups*.

Keywords
clustering, robust estimation, mixture models, semi-parametric model, high-dimensional data.

1 Introduction

The clustering task consists in arranging a set of elements into groups with homogeneous properties/features that capture some important structure of the whole set. As other unsupervised learning tasks, clustering has become of great interest due to the considerable increase in the amount of unlabeled data in the recent years. As the characteristics of real-life data—in geometrical and statistical terms—are very diverse, an intensive research effort has been dedicated to define various clustering algorithms which adapt to some particular features and structural properties. We refer to Hennig [2015] and the clustering review by scikit-learn developers [2019], for discussions on the different methods and on how to choose one depending on the settings. Among the different types of clustering algorithms, the Expectation-Maximization (EM) procedure to estimate the parameters of an underlying Gaussian Mixture Model (GMM) [see for instance the review work by McLachlan, 1982] is a very popular method as its model-based

nature typically allows other algorithms to be outperformed when the data is low dimensional and the clusters have elliptical shapes. This model represents the distribution of the data as a random variable given by a mixture of Gaussian distributions. The corresponding clustering criterion is simple: all points drawn from a given normal distribution are considered to belong to the same cluster. The Expectation-Maximization algorithm (EM) [Dempster et al., 1977] is a general statistical method used to estimate the parameters of a probabilistic model, based on the maximization of the likelihood. It is an iterative algorithm with two main steps: the expectation part and the maximization part. In particular for the GMM case, closed-form expressions exist to obtain parameters estimations at the maximization step.

However, its performance decreases significantly in various scenarios of particular interest for machine learning applications:

- When the data distribution has heavier (or lighter) tails than the Gaussian one and/or in presence of outliers or noise as in Figure 1 [see for instance Fraley and Raftery, 2002]. This phenomenon can be simply explained by the non-robustness of the estimators that are computed by the algorithm: means and sample covariance matrices [Maronna, 1976].

- The presence of different scales in the data might complicate the global ordering of the observations around their closest centers (for instance through Mahalanobis distances). The usual normalization procedure for the estimation of covariance matrices might be too rigid to get satisfactory clustering results in the presence of significant variability intra and inter-clusters [García-Escudero et al., 2008].

- When the dimension increases (even in the Gaussian case), the estimation of the covariance matrix is crucially affected by the high-dimensionality as it has been shown by Bouveyron and Brunet-Saumard [2014]. Some solutions in that direction include regularization and parsimonious models that restrict the shape of the covariance matrix in order to decrease the number of parameters to be estimated [Celeux and Govaert, 1995].

In order to improve the performance of the GMM-EM clustering algorithm in the context of noisy and diverse data, two main strategies were contemplated. One consists in modifying the model to take into account the noise and the other one is to keep the original model and replace the estimators by others that are able to deal with outliers [McNicholas, 2016]. In that line of research, several variations of the Gaussian mixture model have been developed. In particular, some variations target the problem of mixtures of more general distributions, which allow to model a wider range of data, and possibly allowing for the presence of noise and outliers. Regarding the use of non-Gaussian distributions, Peel and McLachlan [2000] proposed an important model defined as a mixture of multivariate t-distributions. In this work, the authors suggested an algorithm (t-EM or EMMIX in the literature) to estimate the parameters of the mixture with known and unknown degrees of freedom by maximizing the likelihood and addressed the clustering task. More recently, Wei et al. [2017], Browne and McNicholas [2015], Lee and McLachlan [2014] considered hyperbolic and skew t-distributions.

Other robust clustering approaches worth mentioning are models which add an extra term to the usual Gaussian likelihood and algorithms with modifications inspired by usual robust techniques as robust point estimators, robust scales, weights for observations and trimming techniques. For instance, Banfield and Raftery [1993] considered the presence of a uniform noise as background while Coretto and Hennig [2017] proposed RIMLE, a pseudo-likelihood based

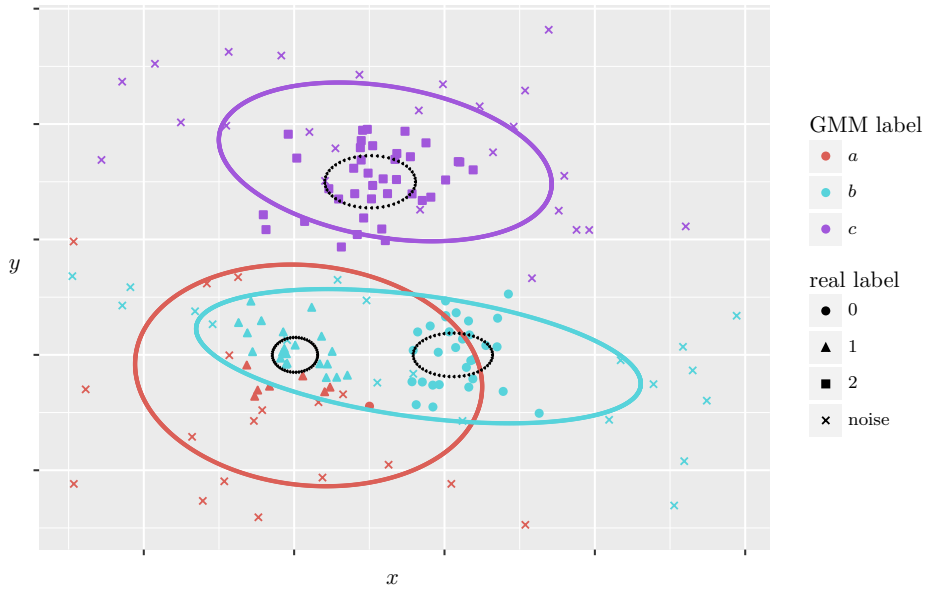

Figure 1: Clustering result of applying the classic EM for GMM in the presence of uniform noise. The shape of each observation represents the real label (the cross represents the noise). The color of each point represents the assigned label. The dashed ellipses represent the real clusters and the solid ellipses represent the contours of the estimated distributions.

algorithm that filters the low density areas. Yu et al. [2015] replaced the usual mean and sample covariance by the spatial median and the rank covariance matrix (RCM). Gonzalez et al. [2019] introduced a robust scale that is used to define a k-means-like algorithm that can deal with outliers. Furthermore, Gonzalez [2019] proposes a robust mixture of distributions estimation based on robust functionals. Moreover, in the work of Campbell [1984], Tadjudin and Landgrebe [2000] and Gebru et al. [2016] different weights for the observations were proposed where small weights correspond, as usual in the robust literature, to observations that are far from the cluster centers. Finally, trimming algorithms such as TCLUST [García-Escudero et al., 2008] leave out a proportion of data points that are far from all the means in order to better estimate the parameters in the M-step.

This article aims at defining an algorithm that can both outperform traditional ones under an assumption of diverse data and be adaptive to a large class of underlying distributions. Following the path of robust statistical approaches, we propose to complement it by using a semi-parametric setting, allowing us to reach an important flexibility for the data distributions. Our method is also inspired by the robust applications of the Elliptical Symmetric (ES) distributions [Boente et al., 2014, Ollila et al., 2012]. Of course, elliptical distributions have been widely used in many applications where Gaussian distributions were not able to approximate the underlying data distribution because of presence of heavy tails or outliers [Conte et al., 2002a, Gini et al., 2000]. This general family includes, among others, the class of compound-Gaussian distributions that contains Gaussian, $t-$ and $k-$ distributions [Gini and Farina, 2002, Conte and Longo, 1987, Conte et al., 2002b] as well as the class of Multivariate Generalized Gaussian Distributions [Pascal et al., 2013].

In this paper, we present

- A general mixture model, involving one scale parameter per data point, leading to an important flexibility in the resulting clustering algorithm. While not being parameters of interest for the clustering task, those parameters are estimated and their estimators are fully analyzed since they play an indirect role in the clustering algorithm;
- A clustering algorithm with the following characteristics: 1/ it follows the two steps expectation and maximization of EM algorithms, 2/ at the E-step, it provides estimated conditional probabilities, robust in the sense that the expected conditional log-likelihood leads to estimators independent of the distributions shapes 3/ at the M-step, it derives estimations of clusters centers and covariance matrices which turn out to be robust.

There are hence two types of estimations. On the one hand, as all EM-like algorithms, we perform an estimation of the parameters of interest: clusters proportions, means and covariance matrices. On the other hand, we use the estimation of scale (or nuisance) parameters, (which are not of direct interest) to improve the estimations of the parameters of interest as well as robustify the estimation of the probability for an observation to belong to a given cluster. More precisely, we show in this paper that, under mild assumptions, those probabilities estimates do not depend on the shape of data distributions, making the algorithm generic, simple and robust. It can be noticed that the scale/nuisance parameters could also be used for classification and outlier detection purposes by discriminating data and helping data assignment [Roizman et al., 2020] .

A key feature of the proposed algorithm is to be self-contained in the sense that no *extra-parameters* need to be tuned as it is the case for aforementioned approaches (*e.g.,* penalty parameters, rejection thresholds,and other distribution parameters such as shapes or the degrees of freedom).

In the sequel, we include practical and theoretical studies that provide evidence about the algorithm performance. In particular, we theoretically justify the efficiency of our algorithm using various arguments:

1. When the underlying model belongs to the class of elliptical distributions, with different means and dispersion matrix per cluster but with cluster-independent density generators (even different ones within clusters) then the estimation of membership probabilities does not depend on each specific density function. This is a consequence of the fact that those probabilities estimations do not depend on the scale factors of the covariance matrix but only on the scatter/dispersion matrices. Hence, the algorithm makes no mismatch error when the density generator is unknown, whenever this assumption is fulfilled. This is shown in Proposition 4
2. Even when the density function is different for every cluster, there are regimes, where the mismatch error can be controlled. We give an example using $t-$distributions with various degrees of freedom. See Proposition 5.
3. Finally, though the estimation of covariance matrix becomes clearly challenging in high-dimensional settings, estimations of the nuisance parameters get typically more accurate and faster when the dimension grows large, using a simple law of large numbers in the dimension. See Proposition 6.

From a practical perspective, the induced clustering performance is largely improved compared to k-means, the EM algorithm for GMM and HDBSCAN [Campello et al., 2015, McInnes and Healy, 2017, McInnes et al., 2017] when applied to real data sets such as MNIST variations [Lecun et al., 1998], NORB [LeCun and Bottou, 2004] and *20newsgroups* [Mitchell, 1997]. In agreement with the proposed results, previous works on classification of the MNIST dataset suggest the non-gaussianity of the clusters [Liao and Couillet, 2017]. Compared to spectral clustering and t-EM, TCLUST and RIMLE our algorithm performs similarly in classic cases and much better in others. Furthermore, the proposed algorithm is able to provide accurate estimations of location and dispersion parameters even in the presence of heavy tailed distributions or additive noise as proved in simulations where our algorithm beats the other compared models.

The rest of the paper is organized as follows. In Section 2, after introducing in details the models of interest, we present the clustering algorithm and discuss some of its important aspects, notably by proving convergence results on the parameters estimation. Section 3 is devoted to the experimental results, which allow us to show the improved performance of the proposed method for different synthetic and real data sets in comparison with other commonly used methods. Finally, conclusions and perspectives are stated in Section 4.

2 Model and Theoretical Justifications

In this section, we present a detailed description of the underlying theoretical model and the proposed clustering algorithm. Given $\{\mathbf{x}_i\}_{i=1}^n$ a set of n data points in $\mathbb{R}^m$, let us start by considering them as independent samples drawn from a mixture of distributions with the following probability density function (pdf):

$$f_i(\boldsymbol{x}_i) = \sum_{k=1}^{K} \pi_k f_{i,\theta_k}(\boldsymbol{x}_i) \quad \text{with} \quad \sum_{k=1}^{K} \pi_k = 1, \tag{1}$$

where π_k represents the proportion of the i^{th} distribution associated with some parameters θ_k in the mixture. The notation f_{i,θ_k} is used for simplicity and stands for a pdf f_{i,k,θ_k} that may depend in principle on cluster k and in general on some "cluster parameters" grouped in θ_k as well as on some extra nuisance parameter τ_{ik}. We remark that the subscript i is used in f_{i,θ_k} to stress that distributions can be different from one observation to another.

Remark 1 *Let us underline the level of generality of the model: the K clusters are only characterized by parameters θ_k while the shape of the distributions can change from one observation to another. We explain the relevance of such a general structure in the next paragraph, where we fix a set of distributions for the f_{i,θ_k}.*

In the sequel, we consider a very large class of distributions in order to generalize the classical Gaussian mixture model: the Elliptically Symmetric (ES) distributions. The pdf of an m-dimensional random vector $\mathbf{x}_i$ that is ES-distributed with mean $\boldsymbol{\mu}_k$ and covariance matrix $\tau_{ik}\boldsymbol{\Sigma}_k$ can be written as

$$f_{i,\theta_k}(\boldsymbol{x}_i) = A_{ik}|\boldsymbol{\Sigma}_k|^{-1/2}\tau_{ik}^{-m/2}\, g_{i,k}\left(\frac{(\boldsymbol{x}_i-\boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)}{\tau_{ik}}\right) \tag{2}$$

where A_{ik} is a normalization constant, $g_{i,k} : [0, \infty) \to [0, \infty)$ is any function (called the density generator) such that (2) defines a pdf. The matrix $\mathbf{\Sigma}_k$ reflects the structure of the covariance matrix of $\boldsymbol{x}_i$. Note that the covariance matrix is equal to $\mathbf{\Sigma}_k$ up to a scale factor if the distribution has a finite second-order moment [see for details Ollila et al., 2012]. This is denoted $ES(\boldsymbol{\mu}_k, \tau_{ik}\mathbf{\Sigma}_k, g_{i,k}(.))$. Note that the clustering parameter is $\theta_k = (\pi_k, \boldsymbol{\mu}_k, \mathbf{\Sigma}_k)$ while the nuisance parameter is τ_{ik}. For convenience, we denote by $\theta = (\theta_1, \ldots, \theta_K)$ the set of all clustering parameters.

At this stage, some comments have to be mentioned:

1. When $g_{i,k} = g, \ \forall i, k$, then we retrieve a classic paradigm for clustering modelling. All the points follow the same ES distribution but each class has a different mean and covariance matrix. However, note that this model is much more general than a Gaussian mixtures model as the class of ES distributions is much wider and includes in particular lighter and heavier tails than Gaussian ones. An important aspect of our results is that our algorithm is in that case **insensitive** to the function g, and hence allows to treat efficiently real data sets where g is not known.

2. When $g_{i,k} = g_i, \ \forall i, k$, we obtain a much more general model than the previous one, where, even if the distribution of the points do not depend on the classes except for their mean and covariances, the data within a class might follow e.g., a mixture of ES distributions. It has a practical importance since many data sets are compiled from different sources of data with different characteristics. In that case again, our algorithm is **still insensitive** to the functions g_i giving a lot of modelling flexibility and mismatch robustness.

3. When $g_{i,k} = g_k, \ \forall i, k$, then we consider one different ES distribution per class of data. For this non-standard settings, the clustering results **do depend on g_k which can be a practical obstacle to get sound results**. However, we show that our method can alleviate this dependence leading to good performance in some regimes.

Elliptical distributions have been used in many applications where one has to deal with the presence of heavy tails or outliers [Conte et al., 2002a, Gini et al., 2000]. This general family includes Gaussian, $t-$ and $k-$ distributions, among others [Gini and Farina, 2002, Conte and Longo, 1987, Conte et al., 2002b] . Such modelling admits a Stochastic Representation Theorem. A vector $\mathbf{x}_i \sim ES(\boldsymbol{\mu}_k, \tau_{ik}\mathbf{\Sigma}_k, g_{i,k}(.))$ if and only if it admits the following stochastic representation [Yao, 1973]

$$\mathbf{x}_i \stackrel{d}{=} \boldsymbol{\mu}_k + \sqrt{Q_{ik}}\sqrt{\tau_{ik}}\mathbf{A}_k\mathbf{u}_i, \tag{3}$$

where the non-negative real random variable Q_{ik}, called the modular variate, is independent of the random vector $\mathbf{u}_i$ that is uniformly distributed on the unit m-sphere and $\mathbf{A}_k\mathbf{A}_k^T$ is a factorization of $\mathbf{\Sigma}_k$ while τ_{ik} is a deterministic but unknown nuisance parameter.

Note that, in this work, one considers that $\mathbf{C}_{ik} = \tau_{ik}\,\mathbf{\Sigma}_k$ can also depend on the $i^{\underline{\text{th}}}$ observation, through the nuisance parameter. Now, for identifiability purposes, we assume that the distributions at hand have a second-order moment and that $\mathbf{C}_{ik}$ is the covariance matrix. This assumption implies the particular normalization on Q_{ik}, that is

$$E[Q_{ik}] = \text{rank}(\mathbf{C}_{ik})\,(= \text{rank}(\mathbf{\Sigma}_k)) = m, \text{ when } \mathbf{\Sigma}_k \text{ is full rank},$$

following for instance Ollila and Tyler [2012]. In the sequel, we hence call $\mathbf{C}_{ik}$ the covariance matrix and $\mathbf{\Sigma}_k$ the scatter matrix.

Finally, an ambiguity remains in the scatter matrix $\mathbf{\Sigma}_k$. Indeed, for any positive real number c, $(\tau_{ik}, \mathbf{\Sigma}_k)$ and $(\tau_{ik}/c, c\,\mathbf{\Sigma}_k)$ lead to the same covariance matrix $\mathbf{C}_{ik}$. In this work, we choose to fix the trace of $\mathbf{\Sigma}_k$ to m. Other normalizations could have been chosen instead as for instance imposing a unit-determinant for $\mathbf{\Sigma}_k$ without affecting the clustering results.

Ollila and Tyler [2012] showed in the complex case that, given random sample from $\mathbf{x}_i \sim CES(\mathbf{0}_m, \tau_{ik}\mathbf{\Sigma}_k, g_{i,k}(.))$, the estimation of τ_{ik} using Maximum Likelihood Estimation (MLE) is decoupled from the estimation of $\mathbf{\Sigma}_k$. Furthermore, the authors proved that the maximum likelihood estimator for $\mathbf{\Sigma}_k$ is the Tyler's estimator, regardless the functions g_i. This is a remarkable result, underlying the universal character of the Tyler estimator in this class of distributions. We will build on this distribution-free property of the Tyler's estimator which turns out to be central for our results.

2.1 The M-step: Parameter Estimation for the Mixture Model

Similarly to the EM for GMM, we extend the model with n discrete variables Z_i (with $i = 1 \dots n$), that are not observed (corresponding to the so-called latent variables), representing the cluster label of each observation $\mathbf{x}_i$. We compute the label for each observation and cluster in the E-step, while in the M-step we estimate the parameters of interest $\theta = (\theta_k)_{k=1}^K$.

Given a sample $\boldsymbol{x} = (\boldsymbol{x}_1^T, ..., \boldsymbol{x}_n^T)^T$, a set of parameters θ, and the latent variables $Z = (Z_1, ..., Z_n)^T$. The expected conditional log-likelihood of the model is

$$
\begin{aligned}
E_{Z|\boldsymbol{x},\theta^*}[l(Z, \boldsymbol{x}; \theta)] = & \sum_{i=1}^{n}\sum_{k=1}^{K} P_{i,\theta*}(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)\log(\pi_k f_{i,\theta_k}(\boldsymbol{x}_i)) \qquad (4)\\
= & \sum_{i=1}^{n}\sum_{k=1}^{K} p_{ik}\left[\log(\pi_k) + \log(A_{ik}) + \log\left(|\mathbf{C}_{ik}|^{-1/2} g_i((\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \mathbf{C}_{ik}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k))\right)\right],
\end{aligned}
$$

where, $p_{ik} = P_{i,\theta^*}(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)$ with $\sum_{k=1}^{K} p_{ik} = 1$ and $\mathbf{C}_{ik} = \tau_{ik}\mathbf{\Sigma}_k$.

We now include two propositions that summarize the derivation of the estimators for all the parameters of the model. As underlined previously, a key step using the ideas in Ollila and Tyler [2012], consists in factorizing the likelihood into two factors which further allows to describe fundamental properties of the estimators in the E and M steps. In Proposition 1, we derive the estimator for the τ parameters. Then, in Proposition 2, we derive the rest of the parameters of the model.

Proposition 1 *Suppose $\mathbf{x}_1, ..., \mathbf{x}_n$ an independent sample with $\mathbf{x}_i \sim ES(\boldsymbol{\mu}_k, \tau_{ik}\mathbf{\Sigma}_k, g_{i,k}(.))$ for some $k \in \{1, ..., K\}$. Suppose $\int t^{m/2} g_{i,k}(t)dt < \infty$, $\forall i, k$. Then, the derivation of the maximum likelihood estimation of the τ_{ik} parameters is decoupled from the one of the rest of the estimators. For fixed parameters $\mathbf{\Sigma}_k$ and $\boldsymbol{\mu}_k$, the τ_{ik}'s estimators are computed as*

$$
\widehat{\tau}_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)}{a_{ik}}, \ \forall 1 \le i \le n \ \ \textit{and} \ \ \forall 1 \le k \le K, \qquad (5)
$$

with $a_{ik} = \arg\sup_t \{t^{m/2} g_{i,k}(t)\}$.

Proof 1 *See Appendix A.1.*

We now describe the ML estimators for θ.

Proposition 2 *Given an independent random sample* $\mathbf{x}_1, ..., \mathbf{x}_n$*, the latent variables* Z_i*, and expected conditional log-likelihood of the model stated before, the maximization w.r.t.* θ_k *for* $k = 1, \ldots, K$*, leads to the following equations that the estimators have to fulfill. The closed equations*

$$\widehat{\pi}_k = \frac{1}{n} \sum_{i=1}^{n} p_{ik} \tag{6}$$

for the proportion of each distribution,

$$\widehat{\boldsymbol{\mu}}_k = \sum_{i=1}^{n} c_{ik}\, \mathbf{x}_i \quad \textit{with} \quad c_{ik} = \frac{\dfrac{p_{ik}}{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}}{\displaystyle\sum_{l=1}^{n} \frac{p_{lk}}{(\mathbf{x}_l - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_l - \widehat{\boldsymbol{\mu}}_k)}}, \tag{7}$$

for the mean of each distribution, and

$$\widehat{\boldsymbol{\Sigma}}_k = m \sum_{i=1}^{n} \frac{w_{ik} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T}{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)} \quad \textit{with} \quad w_{ik} = \frac{p_{ik}}{\displaystyle\sum_{l=1}^{n} p_{lk}}, \tag{8}$$

for the scatter matrices.

Proof 2 *See Appendix A.2.*

It follows from the derivation of Proposition 2 that there is a system of two fixed-point equations given by

$$\widehat{\boldsymbol{\mu}}_k = \frac{\displaystyle\sum_{i=1}^{n} \frac{p_{ik} \mathbf{x}_i}{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}}{\displaystyle\sum_{i=1}^{n} \frac{p_{ik}}{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}} \tag{9}$$

and

$$\widehat{\boldsymbol{\Sigma}}_k = m \sum_{i=1}^{n} \frac{w_{ik} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T}{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}, \tag{10}$$

that hold for the estimators of $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$, with w_{ik} defined in Eq. (8). In order to obtain these linked estimators, this system is iteratively solved as explained in Section 2.5.

We can now prove a fundamental property of the algorithm which is the monotonicity of the likelihood of the model. We later illustrate this property with simulations in Section 2.5. To establish more precise guarantees of convergence we would need a data-driven approach as it was developed for instance in [Wu et al., 2016]. We leave this analysis for future work.

Proposition 3 *Given the expected log-likelihood in* (4)*, the observed likelihood*

$$l(\boldsymbol{x};\theta) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k f_{i,\theta_k}(\boldsymbol{x}_i),$$

and assuming the convergence of the fixed-point equations system derived in Proposition 2, the steps defined by the estimator updates from Propositions 1 and 2 lead to a succession $\{\theta^t\}_{t=1}^N$ *with an increasing likelihood.*

Proof 3 *See Appendix A.3.*

It is important to notice that the derivation of estimators in our model results in usual robust estimators for the mean and covariance matrices. More specifically, both can be assimilated to M-estimators with a certain u function [Maronna, 1976]. Actually, both the expressions for the mean and the scatter matrix estimators are very close to the corresponding Tyler's M-estimator [see Tyler, 1987, Frontera-Pons et al., 2016, for more details]. Main differences arise from the mixture model that leads to different weights involved by the different distributions. However, in case of clusters with equal probability, i.e., $p_{ik} = 1/K$ for $k = 1, \ldots, K$ and $i = 1, \ldots, n$, one retrieves exactly the Tyler's M-estimator for the scatter matrix while the mean estimator differs only from the square-root at the denominator (see the explanation later on). Although our estimators are derived as usual MLE (but) for parametrized (thanks to the τ_{ik}) elliptical distributions, they are intrinsically robust. Indeed, as detailed in [Bilodeau and Brenner, 1999], Tyler's and Maronna's M-estimators can either be obtained through MLE approaches for particular models (e.g., Student-t M-estimators) or directly from other cost functions (e.g., Huber M-estimators) and all those estimators are by definition robust.

Thus, this approach can be seen as a generalization of Tyler's M-estimators to the mixture case. Indeed, one has for $\widehat{\boldsymbol{\mu}}_k$

$$\frac{1}{n}\sum_{i=1}^{n} u_1\left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)\right)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k) = \mathbf{0}, \text{ with } u_1(t) = \frac{p_{ik}}{t},$$

while $\widehat{\boldsymbol{\Sigma}}_k$ can be written as

$$\widehat{\boldsymbol{\Sigma}}_k = \frac{1}{n}\sum_{i=1}^{n} u_2\left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)\right)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T$$

with $u_2(t) = \dfrac{m\, w_{ik}}{t}$ where $w_{ik} = \dfrac{n\, p_{ik}}{\sum_{l=1}^{n} p_{lk}}$.

This similarity to classical Tyler's estimators explains the robust character of our proposal. Indeed, the difference lies in the weights terms p_{ik} and w_{ik} appearing in the $u_j(.)$ functions traditionally introduced in the robust statistics literature. These naturally implies that those $u_1(.)$ and $u_2(.)$ functions continue to respect Tyler's conditions (although Tyler [1987] used $u_1(t^{1/2})$ instead of $u_1(t)$, see Bilodeau and Brenner [1999] for more details).

The convergence of the fixed-point equations defining the M-estimators has been shown in Maronna [1976] but under a restrictive assumption on the u function, which is not fulfilled in our case. On the other hand, Kent proved in Kent et al. [1991] that for fixed mean, there is convergence of the fixed-point equation for the covariance estimator under a normalization constraint. Finally, he also showed that for some u function, the joint mean covariance estimations

boil down to a constrained covariance estimation. Unfortunately, this trick does not work in our case. Hence, the case of joint convergence of the fixed-point equations for the Tyler's estimators is still an open-problem in statistics even in the case of one distribution (no mixture).

We later perform analysis and simulations that confirm the robustness of the algorithm in practice. In particular, the setups in Section 3.1 include distributions with heavy tails, different distributions and noise.

2.2 The E-step: Computing the Conditional Probabilities

In contrast to the estimators derived in Proposition 2, (5) shows that the estimation of the τ_{ik} parameters are linked to the functions $g_{i,k}$ that characterizes the corresponding Elliptical Symmetric distribution. We now give a central result for our algorithm. The following proposition shows that the p_{ik}'s estimators do not depend on density generators when $g_{i,k} = g_i$.

Proposition 4 *Given an independent random sample $\mathbf{x}_i \sim ES(\boldsymbol{\mu}_k, \tau_{ik}\boldsymbol{\Sigma}_k, g_i(.))$ for some $k \in 1, ..., K$, the resulting estimated conditional probabilities $\widehat{p}_{ik} = \widehat{P}_{\theta_k}(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)$ have the following expression for all $i = 1, \ldots, n$ and $k = 1, \ldots, K$:*

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k) \right)^{-m/2} |\widehat{\boldsymbol{\Sigma}}_k|^{-1/2}}{\sum_{j=1}^{K} \widehat{\pi}_j \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)^T \widehat{\boldsymbol{\Sigma}}_j^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j) \right)^{-m/2} |\widehat{\boldsymbol{\Sigma}}_j|^{-1/2}}, \tag{11}$$

where $\widehat{\pi}_k$, $\widehat{\boldsymbol{\mu}}_k$ and $\widehat{\boldsymbol{\Sigma}}_k$ are given in Proposition 2.

Proof 4 *See Appendix A.4.*

Remark 2

- *Result of Proposition 4 is of utmost importance since it allows to derive the conditional probabilities required in the E-step independently of the distributions g_i's and of the τ_{ik}'s parameters. In other words, for any independent ES-distributed observation $\boldsymbol{x}_i$ with mean $\boldsymbol{\mu}_k$ and covariance matrix $\tau_{ik}\boldsymbol{\Sigma}_k$, a unique EM algorithm is derived that does not depend on the shapes of the various involved distributions. This is essential because the absence of precise knowledge on the specific data distribution is the most usual situation in a real life applications, while estimating it might degrade significantly the performance.*

- *Secondly, it evidences the fact that the particular normalization of the $\boldsymbol{\Sigma}$ estimator does not affect the probability computation in the E-step. In other words, the normalization of the scatter matrices are not relevant for the clustering results. On the other hand, the normalization of $\widehat{\boldsymbol{\Sigma}}$ does affect the scale of the τ_{ik} parameters. Thus, using them to classify points or reject outliers needs to be treated with care and is out of the scope of this paper.*

- *The particular case where the data points arise from a mixture of one ES distribution, $g_i = g, \forall 1 \leq i \leq n$, is contained in Proposition 4. We remark the particular example, included in this case, when all the distributions are Gaussian. If $\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_k, \tau_{ik}\boldsymbol{\Sigma}_k)$*

then the corresponding density generator is $g(t) = e^{-t/2}$. *The corresponding maximizer is* $\arg\sup_t\{t^{m/2}g(t)\} = m$, *consequently the estimator is, as derived in* (5), *as follows:*

$$\widehat{\tau}_{ik} = \frac{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}{m}. \tag{12}$$

- *The case where* $g_i = g_k$ *cannot be directly handled as a particular case of Proposition 4. Indeed, assuming each class is drawn by a common ES distribution* g_k *implies in general that extra-parameters, such as, for instance, the degree of freedom* ν_k *for* t*-distributions, the shape parameters for the* K*-distributions and for the generalized Gaussian distributions, depend on* k. *Those parameters have to be estimated in the M-step. We give an example in the next section in the particular case of mixture of* t*-distributions.*

2.3 Different Density Generator per Class

When the density generator depends on the class, our computations show that the p_{ik} do depend on the g_k, as opposed to the previous case.

When the density generators are known (which is quite unrealistic in practice), this assumption naturally increases the clustering performance since extra *a priori* information is added to the model. On the contrary, it implies a performance loss when the real data distribution is not the assumed one.

To illustrate the type of dependence reached in that case, we derive the E-step for the particular case of a mixture of multivariate $t-$distributions with different degrees of freedom ν_k. That is, the case where there are K different g_k functions, one for each cluster. The probability density function of each distribution is given by

$$f_{i,\theta_k}(\boldsymbol{x}_i) = \frac{\Gamma(\frac{\nu_k+m}{2})}{\Gamma(\frac{\nu_k}{2})|\boldsymbol{\Sigma}_k|^{1/2}}(\nu_k\pi\tau_{ik})^{-m/2}\left[1 + \frac{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}\nu_k}\right]^{-(\nu_k+m)/2}.$$

The next proposition states a quantitative approximation of the estimated conditional probabilities in terms of the "Gaussian value" (i.e. the value obtained for class independent g_i), when the ν_k parameters and m grow at the same rate.

Proposition 5 *Given an independent sample of a mixture of* K $t-$*distributions, with* $\mathbf{x}_i \sim t_{\nu_k}$, ν_k *being the degrees of freedom. If for each* k, $\frac{\nu_k}{m} \approx c_k$, *then*

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k \widehat{L}_{0ik}\sqrt{\frac{c_k}{1+c_k}}}{\sum_{j=1}^{K}\widehat{\pi}_j\widehat{L}_{0ij}\sqrt{\frac{c_j}{1+c_j}}} + O\left(\frac{1}{m}\right)$$

Proof 5 *See Appendix A.5.*

This scenario includes of course the case where all the ν_k are equal (See Remark 2). Additionally, it includes when the degrees of freedom are large, with fixed dimension m, the Gaussian case as detailed in the proof. Finally, the other intermediate situations where all the ν parameters do not differ much from the dimension m, are hence shown to be very close to the Gaussian computation. If neither of these conditions apply, an *ad hoc* estimation of the ν_k is of course possible and it has to be performed in the M-step.

2.4 High-Dimensional Regime and estimation of τ_{ik}

2.4.1 Gaussian data

Related to Proposition 5 that considers the case where m and ν_k grow at the same rate, we study in this section how the estimation of the nuisance parameters behaves when the dimension grows. Of course, it is well-known that the breakdown-point of the $\mathbf{\Sigma}$ estimator gets smaller when the dimension grows. Nevertheless, an underlying law of large numbers allows to show that the larger the dimension m, the better the τ estimation performance. For Gaussian data and under mild assumptions, if we take $\mathbf{x}_i$ drawn from the cluster k, we can show that the $\widehat{\tau}_{ik}$ estimator converges to the true value of τ_{ik} when m grows with n. This is more rigorously stated in the following proposition.

Proposition 6 *Suppose that*

$$\mathbf{x}_i = \boldsymbol{\mu}_k + \sqrt{\tau_{ik}}\mathbf{A}_k\mathbf{q}_i,$$

with a deterministic $\tau_{ik} \geq 0$, $\mathbf{A}_k^T\mathbf{A}_k = \mathbf{\Sigma}_k$, *rank*$(\mathbf{\Sigma}_k) = m$ *and* $\mathbf{q}_i \sim \mathcal{N}(0, \mathbf{I}_m)$. *Assume that there exists a sequence of random variables* $(t_i)_{i\in\mathbb{N}}$ *that converges in distribution such that, for* $\alpha \geq 0$, $n^\alpha \widehat{\boldsymbol{\mu}}_k^T\widehat{\boldsymbol{\mu}}_k \leq t_n$ *and that* $\widehat{\boldsymbol{\mu}}_k$ *converges in probability to* $\boldsymbol{\mu}_k$. *Then,* $(\widehat{\tau}_{ik} - \tau_{ik}) \sim \mathcal{N}(0, 2\tau_{ik}^2/m)$ *when* m *and* n *are large enough and fulfill the inequality* $n > m(2m-1)$.

Proof 6 *See Appendix A.6 .*

Remark 3

- *First, in the case where the mean parameter is known, recent Random Matrix Theory results [Couillet et al., 2014, Couillet et al., 2015, Zhang et al., 2016] are in agreement with this phenomenon and prove results for Maronna's and Tyler's M-estimators when* m *and* n *grow together at a fixed rate, i.e.,* $m/n \to \gamma \in [0, 1]$.
- *Secondly, Proposition 6 gives theoretical justification for obtaining better results in high-dimensional settings since in such cases* τ_{ik}*'s parameters will be more accurately estimated.*

2.4.2 Non-Gaussian data

In the case of more general elliptic distributions, one looses in general the convergence (in m) of $\widehat{\tau}_{ik}$ to a deterministic value. Still, $\widehat{\tau}_{ik}$ converges under mild assumptions towards a limit which can be used in principle to handle outliers detections via confidence intervals. As it is not the main topic of this paper, we just state a result for compound Gaussian distributions, illustrating the effect of the large dimension and the type of research that could be fostered in future work.

Proposition 7 *Suppose that*

$$\mathbf{x}_i = \boldsymbol{\mu}_k + \sqrt{\eta_i\, \tau_{ik}}\mathbf{A}_k\mathbf{q}_i,$$

with a deterministic $\tau_{ik} \geq 0$, $\mathbf{A}_k^T\mathbf{A} = \boldsymbol{\Sigma}_k$, $rank(\boldsymbol{\Sigma}_k) = m$, $\mathbf{q}_i \sim \mathcal{N}(0, \mathbf{I}_m)$, *and* η_i *a positive random variable independent of* $\mathbf{q}_i$. *Assume further the consistence of* $\widehat{\boldsymbol{\mu}}_k, \widehat{\boldsymbol{\Sigma}}_k$ *and that*

$$\arg\sup_t\{t^{m/2}g_i(t)\}/m \to 1 \text{ when } \ m \to \infty.$$

Then

$$\widehat{\tau}_{ik} \xrightarrow[m\to\infty]{prob.} \tau_{ik}\,\eta_i.$$

Proof 7 *The Proof follows the same lines as the proof of Proposition A.6 and is omitted.*

Remark 4 *Note that as usual in ML estimation, one can find counter examples where* $\arg\sup_t\{t^{m/2}g_i(t)\}$ *is not equivalent to* m *when* m *grows large. This is however a quite pathological situation and the assumption of the proposition is fulfilled for most practical cases.*

2.5 Implementation Details and Numerical Considerations

The general structure of the proposed algorithm is the same as the one of the classical EM for GMM. The differences between both algorithms lie in the $\hat{p}_{ij}$ expression and the recursive update equations for the parameter estimations. We design slightly different variations of the M-step and study the convergence, precision and speed. We do this considering that the estimators for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are weighted versions of the classic estimators. More precisely, based on equations (9) and (10), we propose four alternatives thinking in accelerating the convergence speed. These versions depend on two different aspects. One aspect consists in using the just computed estimation of the mean or the estimator from the previous iteration of the loop. The other facet is proposed to emphasize the weights of the data points in the computation of the estimators based on the Tyler's estimator. In Section 2, we mention that the location and scatter estimators are close to Tyler's up to the square root of the Mahalanobis distance when the location is unknown. We propose to modify the weights by adding this square root in order to mimic Tyler's estimator. The different versions are defined as follows:

1. Version 1: the parameter $\boldsymbol{\mu}$ used to compute the estimator $\boldsymbol{\Sigma}$ is the one obtained in the **same iteration** of the fixed-point loop.

2. Version 2: the $\boldsymbol{\mu}$-parameter is the one obtained in the **previous iteration**.

3. Version 3: we propose an **accelerated method** where the quadratic forms $(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T\widehat{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)$ in the denominators of the fixed-point $\boldsymbol{\mu}$ equations are replaced by their square root, corresponding to the original Tyler's M-estimators.

4. Version 4: we implement the same **acceleration procedure** on top of the algorithm of Version 2.

For concreteness, in **Algorithm 1** we describe the complete algorithm in Versions 1 (left) and 4 (right). In the particular case described in Section 2.3, where all g_{ik} functions are known, the p_{ik} should be computed with the Bayes expression as in (21).

Algorithm 1: Scheme of the F-EM algorithm. The Version 1 of the M-step is on the left and the Version 4 is on the right. The differences are highlighted in red in the Version 4.

Input : Data $\{\boldsymbol{x}_i\}_{i=1}^n$, K the number of clusters
Output: Clustering labels $\mathcal{Z} = \{z_i\}_{i=1}^n$

Set initial random values $\theta^{(0)}$;
$l \leftarrow 1$;
while *not convergence* **do**
  E-step: Compute $p_{ik}^{(l-1)} = P_{i,\theta^{(l-1)}}(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)$ for each $1 \leq k \leq K$

$$p_{ik}^{(l)} = \frac{\pi_k^{(l-1)} \left((\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(l-1)})^T (\boldsymbol{\Sigma}_k^{(l-1)})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(l-1)}) \right)^{-m/2} |\boldsymbol{\Sigma}_k^{(l-1)}|^{-1/2}}{\sum_{j=1}^{K} \pi_j^{(l-1)} \left((\boldsymbol{x}_i - \boldsymbol{\mu}_j^{(l-1)})^T (\boldsymbol{\Sigma}_j^{(l-1)})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_j^{(l-1)}) \right)^{-m/2} |\boldsymbol{\Sigma}_j^{(l-1)}|^{-1/2}}$$

  M-step:
    For each $1 \leq k \leq K$:
      Update $\pi_k^{(l)} = \frac{1}{n} \sum_{i=1}^n p_{ik}^{(l)}$ and compute $w_{ik}^{(l)} = \frac{p_{ik}^{(l)}}{\sum_{l=1}^n p_{lk}^{(l)}}$;
      Set $\boldsymbol{\mu}_k^{'} = \boldsymbol{\mu}_k^{(l-1)}$ and $\boldsymbol{\Sigma}_k^{'} = \boldsymbol{\Sigma}_k^{(l-1)}$;
      while *not convergence* **do**

Version 1	Version 4
$\boldsymbol{\mu}_k^{''} = \frac{\sum_{i=1}^n \frac{p_{ik}^{(l)} \boldsymbol{x}_i}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})}}{\sum_{i=1}^n \frac{p_{ik}^{(l)}}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})}}$	$\boldsymbol{\mu}_k^{''} = \frac{\sum_{i=1}^n \frac{p_{ik}^{(l)} \boldsymbol{x}_i}{\left[(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'}) \right]^{1/2}}}{\sum_{i=1}^n \frac{p_{ik}^{(l)}}{\left[(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'}) \right]^{1/2}}}$
$\boldsymbol{\Sigma}_k^{''} = m \sum_{i=1}^n \frac{w_{ik}^{(l)} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{'})}$	$\boldsymbol{\Sigma}_k^{''} = m \sum_{i=1}^n \frac{w_{ik}^{(l)} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{''})(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{''})^T}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{''})^T (\boldsymbol{\Sigma}_k^{'})^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k^{''})}$

      end
      Update $\boldsymbol{\mu}_k^{(l)} = \boldsymbol{\mu}_k^{''}$ and $\boldsymbol{\Sigma}_k^{(l)} = \boldsymbol{\Sigma}_k^{''}$ and $\tau_{ik}^{(l)}$:
  $l \leftarrow l + 1$;
end
Set z_i as the index k that has the maximum p_{ik} value;

The plots in Figure 2 show the convergence of the fixed-point equations for the estimation of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ for the different versions of the algorithm in two different setups with two distributions each. We separately study the two parameters in order to see if the variations differently affect each convergence. The first one is a simple case with two well-separated Gaussian distributions in dimension $m = 10$ (means equal to $\mathbf{0}_m$ and $2^*\mathbf{1}_m$, and covariance matrices are the identity $\mathbf{I}_m$ and a diagonal matrix with elements 0.25, 3.5, 0.25, 0.75, 1.5, 0.5, 1, 0.25, 1, 1). The second one is a mixture of two t-distributions with heavy tails and the same parameter ($\nu_1 = \nu_2 = 3$). As one can see in both cases, the convergence is reached for all versions of the algorithm after approximately twenty iterations of the fixed-point loop. We see in Figure 2 that the speed of convergence is improved for Versions 3 and 4, as expected. On the other hand, we study in Figure 3 the evolution of the log-likelihood in these two different scenarios. In the case of the multivariate t-distributions, we computed the likelihood with the true degrees of freedom ($\nu_k = 3$). This Figure shows an increasing likelihood in all cases and a faster convergence of the Version 1 of the model because the correct values for the mean/scatter estimators are reached faster even though its computation takes a bit longer than for Versions 3 and 4. The estimation accuracy of Versions 1 and 2 are by construction (ML-based) better than for Versions 3 and 4. Based on these figures and previous studies about fixed-point fast convergence [see e.g., Pascal et al., 2008], Version 1 is kept since it follows the original proposal, and although slightly slower than Version 2 for the fixed-point loop, it is faster for the convergence of the algorithm. Furthermore, we fixed the number of iterations to 20 in all the experiments. Notice that increasing this number does not result in a significant increase in terms of clustering performance.

Let us now discuss initialization and thresholds used in the proposed algorithm. The mean parameters are initialized as the means resulting of the k-means algorithm. In the case where k-means outputs clusters of only one point, we run k-means again leaving out the isolated points. Due to singularity problems, we take the initial scatter matrix as the identity matrix. We set the initial value of all τ parameters to one. For the convergence flag, we consider 10^{-6} for the threshold of the l_2-norm difference of consecutive estimators, and the maximum number of iterations of the fixed-point loop length is set to 20 based on previous discussion. Using the initialization described above, we obtain the same final clustering results for each run. In the low-dimensional case, we truncate the τ value in order to avoid numerical issues induced by points that are very close to the mean. That is, if τ is smaller than 10^{-12} we change its value to the selected threshold. The implementation in Python of the algorithm is available at the repository github.com/violetr/fem.

Furthermore, it is important to remark that, in our approach, the constraint on the trace of $\widehat{\boldsymbol{\Sigma}}$ ($\text{tr}(\widehat{\boldsymbol{\Sigma}}) = m$) **does not** act as a regularization procedure, as it is usually the case in EM-like algorithms [García-Escudero et al., 2008, Coretto and Hennig, 2017]. As mentioned in Remark 2, the trace constraint does not affect the clustering results.

Finally, regarding the complexity of the algorithm, it happens to be the same as the one of the classical EM algorithm for mixture of Gaussian distributions. The E-step has the same complexity of the usual algorithm. For the M-step, even though a nested loop is included to solve the fixed-point equations, the complexity is not increased since the number of iterations is constant and the main cost of each iteration corresponds to the scatter matrix inversion as in EM for GMM.

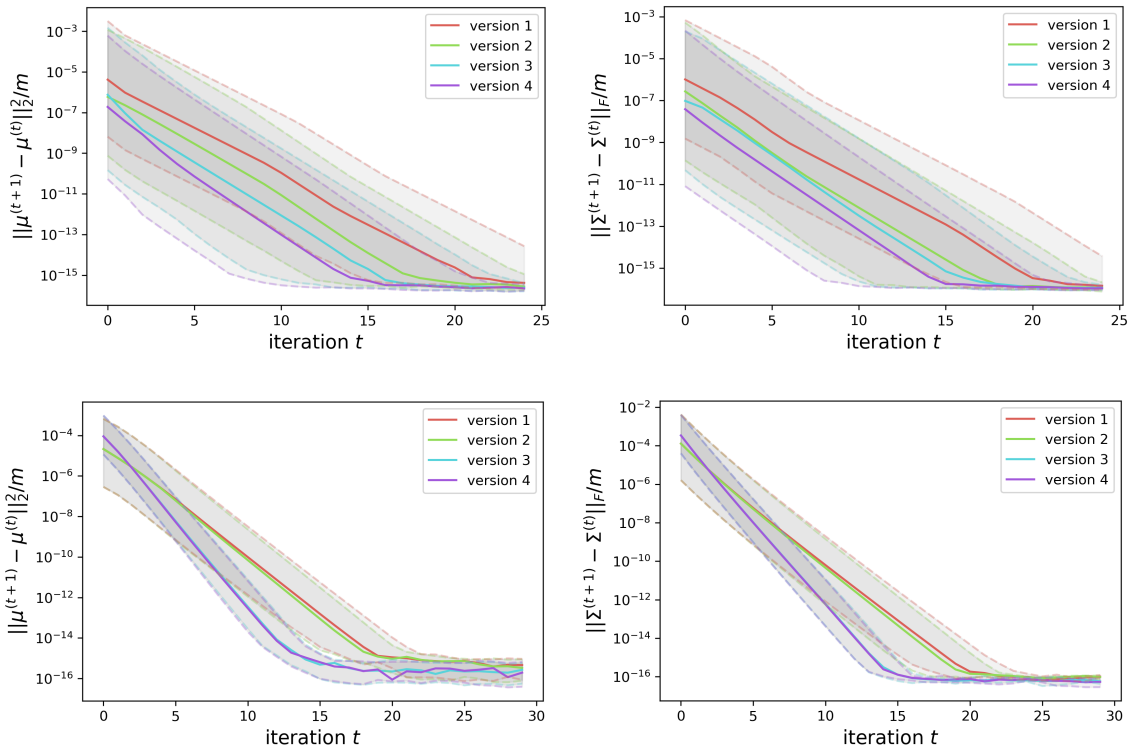

Figure 2: Convergence speed of the fixed-point equations for the estimation of $\boldsymbol{\mu}$ (left) and $\boldsymbol{\Sigma}$ (right). The results in Gaussian case are plotted on the top and on the bottom and the ones for a mixture of t-distributions are shown on the bottom. Each line represents the median of the values obtained on each iteration of the fixed-point iteration for all the iterations of the F-EM algorithms and for all clusters. The gray areas represent the quartile range of each iteration of each version.

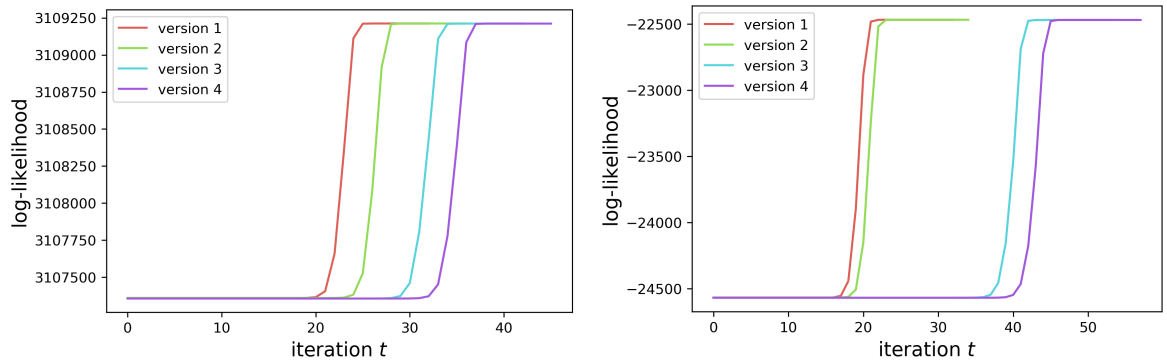

Figure 3: Evolution of the log-likelihood of the models for the different versions. On the left, the results in the Gaussian case and on the right the results for a mixture of t-distributions.

3 Experimental Results

In this section, we present experiments obtained with both synthetic and real data. We study the convergence of the fixed point equations and the estimation error in the case of synthetic data (for which we know the true parameter values). We compare our results to the ones of the classical EM for GMM, EM for multivariate t-distributions, TCLUST [García-Escudero et al., 2008] and RIMLE [Coretto and Hennig, 2017]. Additionally, for the real data, we compare the clustering results with the ground truth labels for k-means, HDBSCAN and spectral clustering [Ng et al., 2001]. The comparison between the former three and our algorithm is straightforward because they all have in common only one main parameter (the number of clusters) that we fix and suppose known in our experiments. Regarding the implementations, we use Scikit-learn [Pedregosa et al., 2011] for k-means and the Gaussian Mixture and the R package EMMIXskew [Wang et al., 2009] for the mixture of $t-$distributions. Concerning TCLUST and the RIMLE algorithms, we set the number of clusters and use the default values for the rest of the parameters. When possible, we avoided the artificial constraint on the TCLUST algorithm solution caused by the eigenvalue constraint threshold. We used the OTRIMLE version of RIMLE that selects the main parameter of the model with a data-driven approach [Coretto and Hennig, 2019]. For both of them, we use the R implementation provided by the authors. In the case of spectral clustering, we run the Scikit-learn implementation where it is necessary to tune an extra parameter in order to build the neighborhood graph. We set the number of neighbors in the graph equal to the number that maximizes the silhouette score [Rousseeuw, 1987]. A fair comparison with HDBSCAN is even more difficult to set because the parameters to tune are completely different and less intuitive than those of the other algorithms. Once again, we select the best silhouette score pair of parameters by sweeping a grid of selected values.

We then quantify the differences of performance by using the usual metrics for the clustering task known as the adjusted mutual information (AMI) index and the adjusted rand (AR) index [Vinh et al., 2010]. For real datasets, one also provides the rate of correct classification when matching each clustering label with a classification label. In the case of real datasets, we also report the clustering classification rate as done in Weber and Robinson [2016]. In some cases, we visualize the 2D embedding of the data obtained by the UMAP algorithm [McInnes et al., 2018] colored with the resulting labels of the different clustering algorithms in order to better understand the nature of the data and the clustering results. This dimensional reduction algorithm has the same objective as t-SNE [van der Maaten and Hinton, 2008] but its implementation in Python is much faster.

3.1 Synthetic Data

In order to compare the clustering performance of the different algorithms, data are simulated according to different distributions, different values for the τ_{ik}'s and different parameters. The different setups are reported in Tables 1 and 2. The setups 1 and 2 are mixtures of multivariate t-distributions. Setup 3 is a mixture of $k-$distributions, $t-$distributions and Gaussian distributions. On the other hand, in Setup 4 we add uniform noise background to three Gaussian distributions. This noise accounts for 10% of the data. Finally, Setup 5 includes clusters that are a mixture of two distributions. In other words, all points from a given cluster are generated with the same parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ but we used different distributions. In this situation, we mixed generalized Gaussian distributions (noted $\mathcal{GN}$), $t-$distributions and Gaussian distributions (noted $\mathcal{N}$). In Table 2, diag and diag† are diagonal matrices with trace m and diag*

has trace 12. Consequently, Setup 1 tests the performance of the algorithm in the presence of covariance matrices with different traces.

Setup	m	n	**distribution 1**	**distribution 2**	**distribution 3**
1	8	1000	t, $dof = 3$	t, $dof = 3$	t, $dof = 3$
2	8	1000	t, $dof = 10$	t, $dof = 10$	t, $dof = 10$
3	40	1300	K, $dof = 3$	t, $dof = 6$	$\mathcal{N}$
4	8	1200	$\mathcal{N}$	$\mathcal{N}$	$\mathcal{N}$
5	6	1200	$0.7\mathcal{N} + 0.3\mathcal{GN}, s = 0.1$	$0.6\mathcal{N} + 0.4t, dof = 2.3$	$\mathcal{N}$

Table 1: Dimension and shape of the distributions in the five different setups. The distribution of each of the three clusters in each setup is specified. The distribution can be multivariate Gaussian ($\mathcal{N}$), Generalized Gaussian ($\mathcal{GN}$), $t-$distribution or $k-$distribution. In the case of the latter three distributions the extra parameters (dof or s) are indicated.

Setup	$\boldsymbol{\mu}_1$	$\boldsymbol{\mu}_2$	$\boldsymbol{\mu}_3$	$\boldsymbol{\Sigma}_1$	$\boldsymbol{\Sigma}_2$	$\boldsymbol{\Sigma}_3$
1	$\mathcal{U}_{(0,1)}$	$6 * 1_m$	$1.5 * 1_m + 3e_1$	diag	diag*	$\mathbf{I}_m/m * 4$
2	$\mathcal{U}_{(0,1)}$	$5 * 1_m$	$1.5 * 1_m + \mathcal{N}(0, \varepsilon)$	diag	diag†	$\mathbf{I}_m$
3	$2 * 1_m$	$6 * 1_m$	$7 * 1_m$	$toep(\rho = 0.2)$	$\mathbf{I}_m$	$toep(\rho = 0.5)$
4	$5 * 1_m$	$7 * 1_m$	$9 * 1_m$	$toep(\rho = 0.2)$	$\mathbf{I}_m$	$toep(\rho = 0.5)$
5	$\mathcal{U}_{(0,0.2)}$	$2 * 1_m$	$4 * 1_m + 2e_1$	$toep(\rho = 0.4)$	$\mathbf{I}_m$	$toep(\rho = 0.7)$

Table 2: Parameters of the distributions in the different setups. The means are either deterministic vectors or stochastic random uniform or Gaussian vectors. The options for the scatter matrix are diagonal with different eigenvalues, the identity matrix or Toeplitz matrix where we specify the constant.

We repeat each experiment $nrep = 200$ times and collect the mean and standard deviation of estimation errors. For the matrices, we compute the Frobenius norm of the difference between the real scatter matrix parameter and its estimation, divided by the matrix size. In order to make a fair comparison of the estimation performance, we take into account the estimations of $\boldsymbol{\Sigma}$'s up to a constant. In other words, we normalize all the $\boldsymbol{\Sigma}$ estimators to have the correct trace. The reported estimation error is computed as follows:

$$\sqrt{\sum_{l=1}^{m}\sum_{o=1}^{m}\left((\boldsymbol{\Sigma}_k)_{lo} - \left(\frac{\widehat{\boldsymbol{\Sigma}}_k \mathrm{tr}(\boldsymbol{\Sigma}_k)}{\mathrm{tr}(\widehat{\boldsymbol{\Sigma}}_k)}\right)_{lo}\right)^2 / m^2}.$$

When estimating $\boldsymbol{\mu}$, the l_2 norm of the error is computed. The π_k vectors, corresponding to the distribution proportions, are randomly chosen from a set of possibilities that avoid trivial and giant clusters. These cases are avoided due to the ill posed clustering problem that it implies.

In these experiments, we include all the considered algorithms that estimate parameters. Thus, we leave out of the comparison k-means, spectral clustering and HDBSCAN. Table 3 shows the estimation error when estimating the main parameters of the model for all the setups. Furthermore, we report the clustering metrics in Table 4. Complementarily, figures 4 and 5 visually summarize with boxplots the distribution of these measures. In most cases, the EM for

GMM (GMM-EM) method has poor results and a high variance.

In setups 1 and 2, the distributions are multivariate $t-$Student and the difference between them is only in the degrees of freedom. In these setups, the proposed algorithm referred to as flexible EM algorithm (F-EM) and t-EM error values are smaller than GMM-EM values. This increase in the predictive performance can be simply explained by the robustness of the estimators in the case of heavy-tailed distributions or in the presence of outliers. It is interesting to confirm that, as in Setup 2, the considered distributions have larger degrees of freedom (tails are lighter), GMM-EM performs much better than in Setup 1. However, while TCLUST and RIMLE perform similarly in the Setup 1, RIMLE has a very big variance and worse estimation in the Setup 2. This phenomenon is due to the overestimation of point as noise/outliers. On the other hand, F-EM and t-EM perform very similarly in both settings, with a slight improvement of F-EM in the $\mathbf{\Sigma}$ estimation. As shown in both tables, even in the $t-$distributed case where the t-EM algorithm is completely adapted, our robust algorithm performs similarly in average. We remark that F-EM performs very good as expected, even if in Setup 1 the traces are very different. Then, for Setups 3, in the case of mixture of three different distributions (k-distribution, t-distribution and Gaussian distribution), the F-EM algorithm outperforms the other algorithms in the majority of runs. As Figure 4 shows, there are only very few runs where F-EM had bad performance. Thus, it is important to notice that the model assumptions used to derive the F-EM algorithm, *i.e.,* unknown τ_{ik}'s and different distributions for each observation, is very general and it allows to successfully handle the case of mixtures of different distributions without additive computational cost, which appears to be an important contribution of this work.

Furthermore, Figure 5 shows the performance in Setup 4 and Setup 5. In Setup 4, in which uniform background noise in the cube $[0, 14]^m$ is included, the best performances are the ones from TCLUST and RIMLE which appears reasonably since their design matches the data generation process. After them, F-EM has a very good performance taking into account that we do not reject outliers and as a consequence, that those are intrinsically misclassified. When we exclude the noise for the metric computation, the classification performance is equally good for these three algorithms, althougth the TCLUST algorithm is computed with the true proportion of outliers. Besides, the parameter estimation is equally good for F-EM, RIMLE and TCLUST. The performance analysis in this Setup (for which RIMLE and TCLUST are designed to provide the best performance) highlights the flexibility and the robustness of the proposed algorithm. Finally, Setup 5 displays a very good behaviour of F-EM and RIMLE compared to the rest of the algorithms. The performance of EM-GMM is really bad there because it cannot deal with outliers coming from heavy tails. The combination of two distributions for one cluster is difficult to fit for $t-$EM and TCLUST. The model is too general for $t-$EM and TCLUST probably suffers from a noise rate that is not sufficient to avoid the heavy tails.

To conclude, the proposed algorithm shows by design very stable performance among a wide range of cases. Indeed, when the data perfectly follows a specific model such as *e.g.,* a mixture of t-distributions, the best algorithm will be the ML-based one (in this case the t-EM algorithm). However, the F-EM algorithm does perform almost as good as the t-EM. But in various other scenarios (data drawn from different models, outliers in the data), the F-EM will clearly outperform traditional model-based algorithms that are not adaptive.

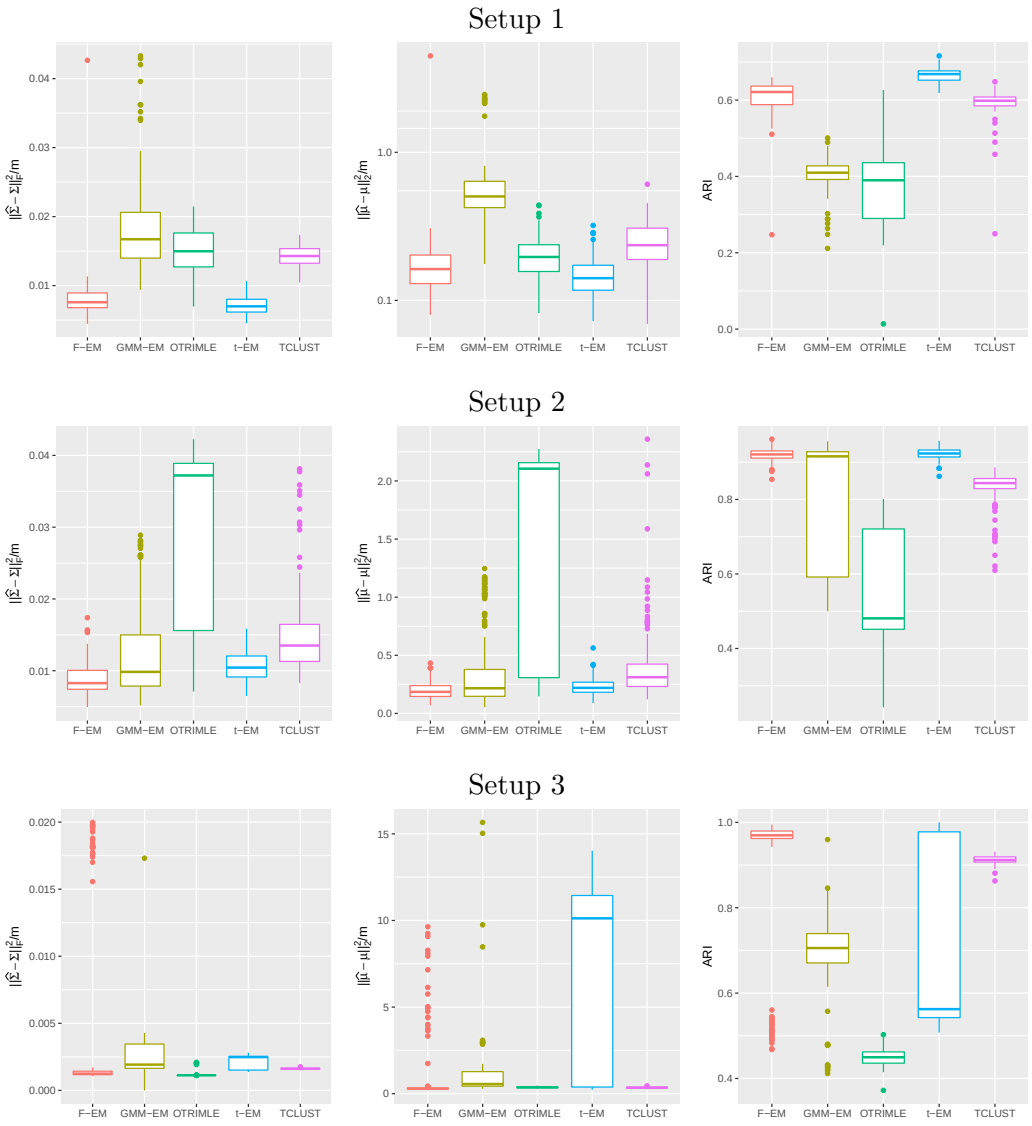

Figure 4: Boxplots representing the performance of the algorithms in the estimation and classification for the different setups. Each row represents one setup. From the left to the right, the Figure summarizes the estimation error of the scatter matrix up to a constant, the estimation error of the mean and the AR index of the classification when comparing to the ground truth.

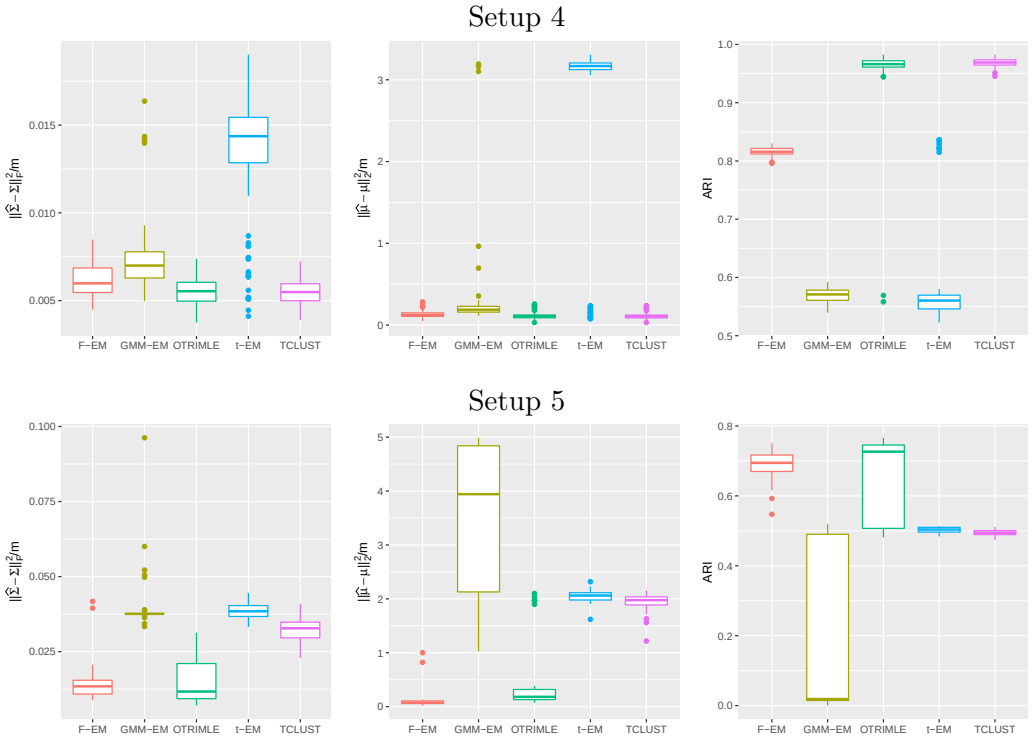

Figure 5: Boxplots representing the performance of the algorithms in the estimation and classification for the different setups. Each row represents one setup. From the left to the right, the Figure summarizes the estimation error of the scatter matrix up to a constant, the estimation error of the mean and the AR index of the classification when comparing to the ground truth.

3.2 Real Data

The proposed F-EM algorithm has been tested on three different real data sets: MNIST [Lecun et al., 1998], small NORB [LeCun and Bottou, 2004] and *20newsgroup* [Mitchell, 1997]. The MNIST hand-written digits (Figure 6) data set has become a standard benchmark for classification/clustering methods. We apply F-EM to discover groups in balanced subsets of similar pairs of digits (3-8 and 1-7) and the set of digits (3-8-6). We additionally contaminate the later subset with a small proportion of noise by randomly adding some of the remaining different digits.

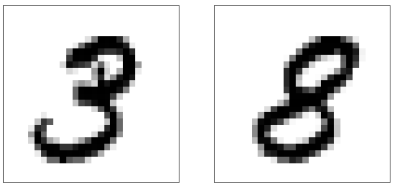

Figure 6: Two samples of the pair 3-8 from the hand-written MNIST data set.

As in many application examples in the literature, we first applied PCA to work with some meaningful features instead of the original data [van der Maaten and Hinton, 2008]. We make a trade-off between explained variance and curse of dimensionality effects. The dimension of the reduced data is shown in Table 5 under the column m. Because of the stochastic character of the algorithms, we run each of them multiple times ($nrep = 50$) and we report the median value of the metrics. The metrics for the F-EM algorithm are almost always the same and this explains why we do not report the variance.

As can be seen in Tables 6, 7 and 8, one obtains, in most cases, better values for all the metrics than those produced by the other partitioning techniques. This can be explained by the increment in flexibility and the smaller impact of outliers in the estimation process. More precisely, the F-EM algorithm does not provide the best results in these scenarios:

- MNIST 7-1 scenario for AMI and AR indices, where the t-EM performs the best,
- MNIST 3-8-6 and its noisy variation for the three criteria where the spectral clustering and TCLUST respectively perform the better,

The loss in performance of the F-EM algorithm is, in most cases, around or less than 1% highlighting the robustness of the approach: **"better or strongly better than existing methods in most cases and comparable in other cases"**. Moreover, those scenarios always correspond to the simpler scenarios, without noise and with well-separated clusters or completely designed to be managed by the best algorithm (MNIST 3-6-8 plus noise for the TCLUST).

We collected the clustering results from the HDBSCAN algorithms fed with a grid of values for its two main parameters. All the computed metrics comparing the results with the ground truth were poor, close to 0. We show the best clustering result of the 3-8 MNIST subset in Figure 7, where a high amount of data points is classified as noise by the algorithm. If the metric is computed only in the non-noise labeled data points then the clustering is almost perfect. This behavior might be explained by the dimension of the data, that seems to be too high for

HDBSCAN to deal with.

Additionally, we have tested dimensional reduction techniques UMAP and t-SNE prior to the clustering task. All metrics were improved after carefully tuning the parameters. In this scenario, the proposed method performs similarly to the classical GMM-EM because these embedding methods tend to attract outliers and noise to clusters. However, these non-linear visualization approaches are not recommended to extract features before clustering because fictitious effects might appear depending on the parameters choice.

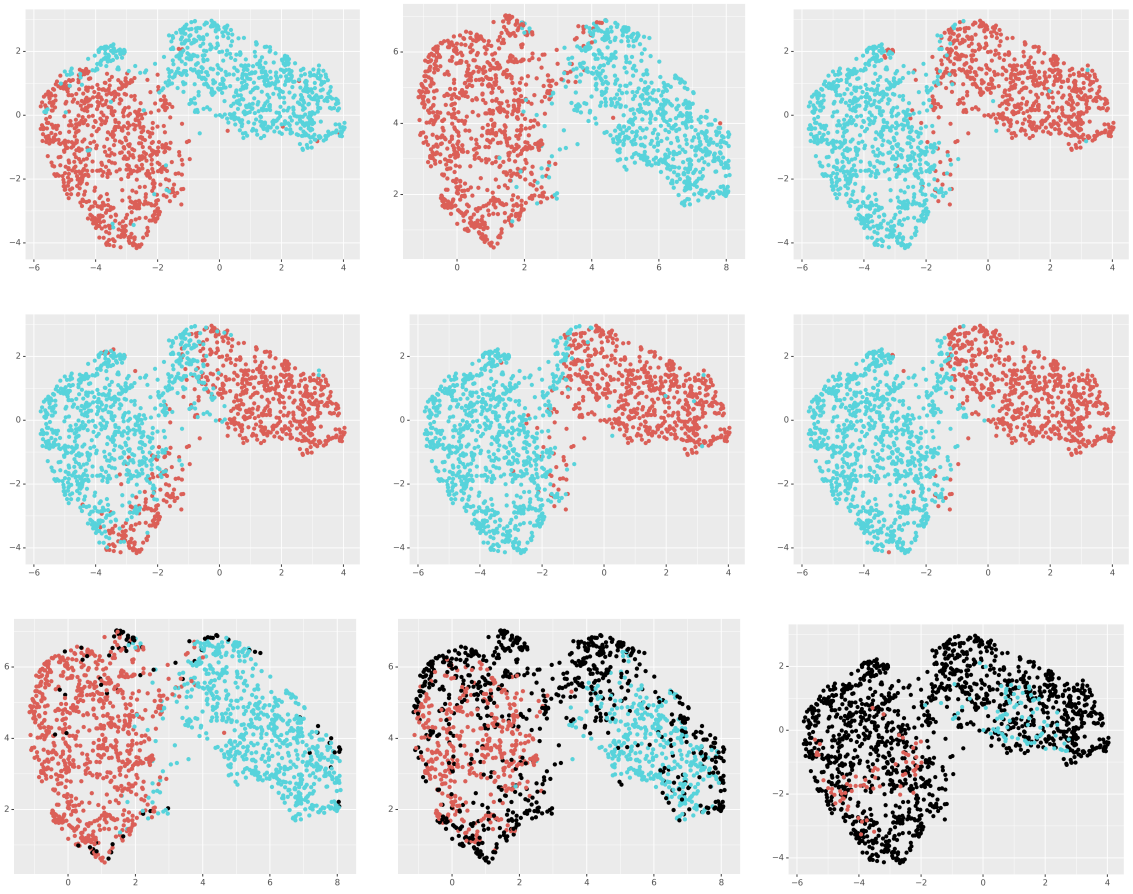

Figure 7: UMAP embedding of the 3-8 pair MNIST subset colored with labels. On the first row, from left to right, the real ground truth labels, the F-EM clustering labels and the t-EM clustering labels. On the second row, from left to right, the k-means clustering labels, the GMM-EM labels and the spectral clustering labels. On the bottom from the left to the right, the TCLUST labels, the RIMLE labels and the HDBSCAN labels. Points colored with black are labelled as noise.

For the NORB dataset (some representatives are shown in Figure 8), k-means, GMM-EM, spectral clustering and UMAP+HDBSCAN do not perform in a satisfactory way since they end-up capturing the luminosity as the main classification aspect. In contrast, t-EM and the F-EM algorithm highly outperform them, as can be seen in Tables 6, 7 and 8. This can be emphasized thanks to results of Figure 9, where label-colored two-dimensional embeddings of the data based on the classification produced by the different methods are shown. The effect of

extreme light values seems to be palliated by the robustness properties of the estimators.

Finally, the *20newsgroup* data set is a bag of words constructed from a corpus of news. Each piece of news is classified by topic modeling into twenty groups. Once again, we compare the performance of our methods with the ones of k-means, EM, t-EM, TCLUST , RIMLE and spectral clustering algorithms after applying PCA. The corresponding results are also presented in Tables 6, 7 and 8. One can see that k-means, TCLUST, RIMLE and spectral clustering perform poorly, while GMM-EM and t-EM outperform them. Nevertheless, the proposed F-EM algorithm has strongly better results than the others. It is not clear why spectral clustering is performing so badly on this data set, it could be due to the lack of separation between clusters and/or the presence of noise that breaks the performance. Finally, the very poor capability of the RIMLE algorithm in this dataset is explained by the choice of the parameter that highly over estimates the noise.

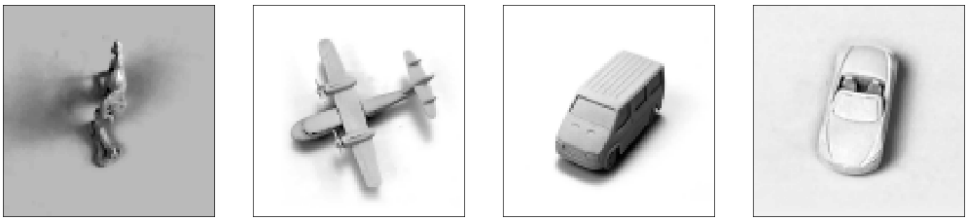

Figure 8: Four samples of the small NORB data set from the 4 considered categories. Differences in brightness between the pictures can be appreciated.

4 Concluding Remarks

In this paper we presented a robust clustering algorithm that outperforms several state of the art algorithms for both synthetic and real diverse data. Its advantages stem from a general model for the data distribution, where each data point is generated by its own elliptical symmetric distribution. The good theoretical properties of this proposal have been studied and supported by simulations. The flexibility of this model makes it particularly suitable for analyzing heavy-tailed distributed and/or noise-contaminated data. Interestingly, under mild assumptions on the data, the estimated probabilities of membership do not depend on the data distributions, making the algorithm simpler (no need to re-estimate the likelihood at each step), flexible and robust. Moreover, the original approach of estimating one scale parameter for each data point makes the algorithm competitive in relatively high-dimensional settings.

On simulated data, we obtained accurate estimations and good classification rates. Of course, the best model is the one that perfectly coincides with the distribution of the data, *e.g.*, when the mixture is actually Gaussian, GMM-EM outperforms all other methods, including ours, but only marginally, and our method performs well on all considered scenarios.
For the real data sets that we considered, We have shown that the proposed method offers better results compared to k-means, GMM-EM and t-EM. It is also competitive with spectral clustering, TCLUST and RIMLE and it still delivers very good results in situations where both HDBSCAN and spectral clustering completely break down.

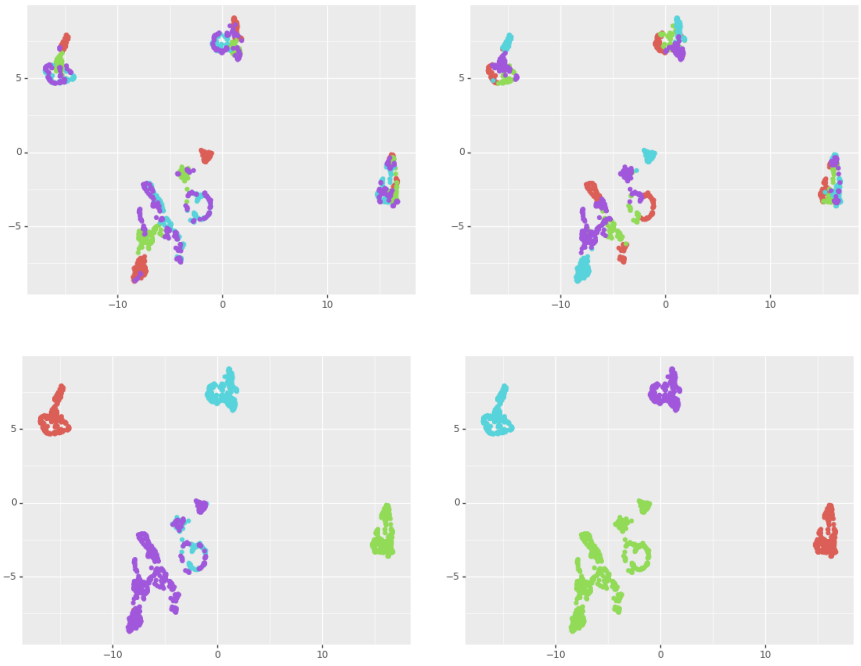

Figure 9: NORB's UMAP embedding colored with relative labels. On the top, the real ground truth labels on the left and the F-EM clustering labels on the right. On the bottom, the k-means clustering labels on the left and the spectral clustering labels on the right.

Concerning future works, we consider studying in depth the convergence of the algorithm with a data-driven approach. Besides, it would be very interesting to study the impact of the τ parameters in the model when using them for classification and / or outlier rejection. Finally, we consider that including a sparse regularization in the scatter estimation would be very useful to take advantage of the fact that the τ parameters are better estimated when the dimension increases with the number of observations.

A Proofs

This Appendix contains the different proofs of propositions provided in this paper (Section 2).

A.1 Proof of Proposition 1

Proof 8 *Let us define* $s_{ik} = \frac{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}}$. *Then we can rewrite the expression*

$$E_{Z|\boldsymbol{x},\theta^*}[l(Z, \boldsymbol{x}; \theta)] = \sum_{k=1}^{K} l_{0k}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) + \sum_{k=1}^{K} \sum_{i=1}^{n} l_{ik}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}\right), \tag{13}$$

where the terms of the sum are

$$l_{0k}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) = \sum_{i=1}^{n} p_{ik}[\log(\pi_k) + \log(A_{ik}) + \frac{1}{2}\log(|\boldsymbol{\Sigma}_k^{-1}|) - \frac{m}{2}\log\left((\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)\right)], \tag{14}$$

and

$$l_{ik}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}\right) = p_{ik}\log(s_{ik}^{m/2} g_{i,k}(s_{ik})). \tag{15}$$

If we fix the parameters $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ *and we maximize* $l_{ik}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}\right)$ *w.r.t.* τ_{ik}*, one obtains that*

$$\widehat{\tau}_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k)}{a_{ik}}, \tag{16}$$

where $a_{ik} = \arg\sup_t \{t^{m/2} g_{i,k}(t)\}$*. As we supposed that* $\int t^{m/2} g_{i,k}(t)dt < \infty$*, it implies that* $t^{m/2} g_{i,k}(t) \to 0$*, when* $t \to \infty$ *so that the* a_{ik} *is finite. Now, observing that*

$$\sum_{k=1}^{K}\sum_{i=1}^{n} l_{ik}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \widehat{\tau}_{ik}\right) = \sum_{i=1}^{n}\sum_{k=1}^{K} a_{ik}^{m/2} g_{i,k}(a_{ik}) p_{ik},$$

and since $p_{ik} = P_{i,\theta^*}(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)$ *and* a_{ik} *do not depend on* θ*, then* $\sum_{k=1}^{K}\sum_{i=1}^{n} l_{ik}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \widehat{\tau}_{ik}\right)$ *does not depend on the parameters* $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ *for any* $k \in 1, ..., K$*. Thus, estimating those parameters will only rely on the first term of the expected likelihood, i.e.,*

$$S_0 = \sum_{k=1}^{K} l_{0k}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right). \tag{17}$$

Note that S_0 *involves density functions that are proportional to the Angular Gaussian p.d.f. [Ollila et al., 2012].*

A.2 Proof of Proposition 2

Proof 9 *Let us maximize* $E_{Z|\boldsymbol{x},\theta^*}[l(Z, \boldsymbol{x}; \theta)]$ *with respect to* $\theta_k = (\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$*, for* $k = 1, \ldots, K$*. Note that the optimization problem is solved under the constraint on the* $\{\pi_k\}_{k=1}^K$*, which enforces to use a Lagrange multiplier. Cancelling the gradient of the expected conditional log-likelihood thus leads to the following system of equations*

$$\frac{\partial[E_{Z|\boldsymbol{x},\theta^*}[l(Z, \boldsymbol{x}; \theta)] - \lambda(1 - \sum_{j=1}^{K}\pi_j)]}{\partial \pi_k} = \sum_{i=1}^{n}\frac{p_{ik}}{\pi_k} + \lambda = 0\,, \quad \forall 1 \leq k \leq K,$$

together with the conditions $\sum_{j=1}^{K}\pi_j = 1$ *and* $\sum_{j=1}^{K} p_{ij} = 1$*. This is equivalent to*

$$\pi_k = -\frac{1}{\lambda}\sum_{i=1}^{n} p_{ik}.$$

Taking the summation over k, together with the constraints, leads to $\lambda = -n$, proving the expression given in Eq. (6).

Let us now consider the derivative of the expected conditional log-likelihood with respect to $\boldsymbol{\mu}_k$. One obtains, for $k = 1, \ldots, K$,

$$\begin{aligned}\frac{\partial[E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)] - \lambda(1-\sum_{j=1}^K \pi_j)]}{\partial \boldsymbol{\mu}_k} &= \frac{\partial l_{0k}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\partial \boldsymbol{\mu}_k} \\ &= \frac{m}{2}\sum_{i=1}^n p_{ik}\frac{\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)},\end{aligned}$$

where l_{0k} is given in Eq. (14).
Then, setting the previous expression to zero leads to

$$\boldsymbol{\mu}_k \sum_{i=1}^n \frac{p_{ik}}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)} = \sum_{i=1}^n \frac{p_{ik}}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)}\boldsymbol{x}_i,$$

providing the result of Eq. (7).

Now, in order to estimate $\boldsymbol{\Sigma}_k$, we differentiate the expected conditional log-likelihood w.r.t. $\boldsymbol{\Sigma}_k^{-1}$. One obtains, for $k = 1, \ldots, K$,

$$\begin{aligned}\frac{\partial[E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)] - \lambda(1-\sum_{j=1}^K \pi_j)]}{\partial \boldsymbol{\Sigma}_k^{-1}} = \frac{\partial l_{0k}\left(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^{-1}\right)}{\partial \boldsymbol{\Sigma}_k^{-1}} &= \\ \frac{\partial\left\{\sum_{i=1}^n p_{ik}\left[\frac{1}{2}\log|\boldsymbol{\Sigma}_k^{-1}| - \frac{m}{2}\log\left((\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}\right)\right]\right\}}{\partial \boldsymbol{\Sigma}_k^{-1}} &= \\ \sum_{i=1}^n p_{ik}\left[\boldsymbol{\Sigma}_k - m\frac{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T}{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)}\right] &.\end{aligned}$$

Equating the latter expression to zero leads to Eq. (8) *and concludes the proof.*

A.3 Proof of Proposition 3

Proof 10 *Similarly to the proof given by Dempster et al. [1977], we can decompose $E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)]$ as follows:*

$$E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)] = l(\boldsymbol{x};\theta) + H(\theta,\theta^*), \tag{18}$$

where we define

$$H(\theta,\theta^*) = E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)] - l(\boldsymbol{x};\theta).$$

We can re-write this expression as

$$\begin{aligned} H(\theta,\theta^*) &= E_{Z|\boldsymbol{x},\theta^*}[l(Z,\boldsymbol{x};\theta)] - l(\boldsymbol{x};\theta) \\ &= \sum_{i=1}^{n} E_{Z_i|\boldsymbol{x}_i,\theta^*}\left[\log\left(\frac{f_{i,\theta}(Z_i,\boldsymbol{x}_i)}{f_i(\boldsymbol{x}_i)}\right)\right] \\ &= \sum_{i=1}^{n} E_{Z_i|\boldsymbol{x}_i,\theta^*}[\log(P_{i,\theta}(Z_i|\boldsymbol{x}_i)], \end{aligned} \tag{19}$$

where $f_{i,\theta}(Z_i,\boldsymbol{x}_i) = \sum_{k=1}^{K} \mathbb{1}_{Z_i=k} f_{i,\theta_k}(\boldsymbol{x}_i)$, $\mathbb{1}_\Omega$ *is the indicator function equal to 1 on* Ω *and 0 elsewhere. Moreover,* $P_{i,\theta}(a|b) = \frac{f_{i,\theta}(a,b)}{f_{i,\theta}(b)}$. *At this point, we use the fact that* $\theta^{(t+1)}$, *the set of estimations computed in iteration* $t+1$ *and derived in Proposition 2, fulfills the following equality:*

$$E_{Z|\boldsymbol{x},\theta^{(t)}]}[l(Z,\boldsymbol{x};\theta^{(t+1)})] = \max_{\theta} E_{Z|\boldsymbol{x},\theta^{(t)}}[l(Z,\boldsymbol{x};\theta)]. \tag{20}$$

Of course, we need to assume the convergence of the fixed-point equation system. Thus, using equation (18) *and the fact that* $E_{Z|\boldsymbol{x},\theta^{(t)}}[l(Z,\boldsymbol{x};\theta^{(t+1)})] \geq E_{Z|\boldsymbol{x},\theta^{(t)}}[l(Z,\boldsymbol{x};\theta^{(t)})]$ *from* (20), *we derive the following inequality:*

$$\begin{aligned} l(\boldsymbol{x};\theta^{(t+1)}) - l(\boldsymbol{x};\theta^{(t)}) \geq H(\theta^{(t)},\theta^{(t)}) - H(\theta^{(t+1)},\theta^{(t)}) &= \\ \sum_{i=1}^{n} E_{Z_i|\boldsymbol{x},\theta^{(t)}}\left[\log\left(\frac{P_{i,\theta^{(t)}}(Z_i|\boldsymbol{x}_i)}{P_{i,\theta^{(t+1)}}(Z_i|\boldsymbol{x}_i)}\right)\right] &\geq \\ -\sum_{i=1}^{n} \log\left[E_{Z|\boldsymbol{x},\theta^{(t)}}\left[\frac{P_{i,\theta^{(t+1)}}(Z_i|\boldsymbol{x}_i)}{P_{i,\theta^{(t)}}(Z_i|\boldsymbol{x}_i)}\right]\right], \end{aligned}$$

where in the inequality we applied the Jensen inequality for the $-\log$ *function. As the expectation is one and in consequence the sum is zero, then* $l(\boldsymbol{x};\theta^{(t+1)}) \geq l(\boldsymbol{x};\theta^{(t)})$ *and that concludes the proof.*

A.4 Proof of Proposition 4

Proof 11 *By definition, one has* $p_{ik} = P_\theta(Z_i = k|\mathbf{x}_i = \boldsymbol{x}_i)$. *Using the Bayes theorem, one obtains, for* $i = 1,\ldots,n$ *and* $k = 1,\ldots,K$*:*

$$p_{ik} = \frac{\pi_k f_{i,\theta_k}(\boldsymbol{x}_i)}{\sum_{j=1}^{K} \pi_j f_{i,\theta_j}(\boldsymbol{x}_i)}$$

Now, the estimated conditional probability can be written by replacing unknown parameters by their previously derived estimators as

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k f_{i,\widehat{\theta}_k}(\mathbf{x}_i)}{\sum_{j=1}^{K} \widehat{\pi}_j f_{i,\widehat{\theta}_j}(\mathbf{x}_i)} \tag{21}$$

$$= \frac{\widehat{\pi}_k A_{ik} \widehat{\tau}_{ik}^{-m/2} |\widehat{\mathbf{\Sigma}}_k|^{-1/2} \, g_i\left(\frac{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\mathbf{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}{\widehat{\tau}_{ik}}\right)}{\sum_{j=1}^{K} \widehat{\pi}_j A_{ik} \widehat{\tau}_{ij}^{-m/2} |\widehat{\mathbf{\Sigma}}_j|^{-1/2} \, g_i\left(\frac{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)^T \widehat{\mathbf{\Sigma}}_j^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)}{\widehat{\tau}_{ij}}\right)}. \tag{22}$$

Finally, using the expression of $\widehat{\tau}_{ik}$ given by Eq. (5) obtained in Proposition 1, one obtains

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\mathbf{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)\right)^{-m/2} |\widehat{\mathbf{\Sigma}}_k|^{-1/2} \max_t \left(a_i \, t^{m/2} \, g_i(t)\right)}{\sum_{j=1}^{K} \widehat{\pi}_j \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)^T \widehat{\mathbf{\Sigma}}_j^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)\right)^{-m/2} |\widehat{\mathbf{\Sigma}}_j|^{-1/2} \max_t \left(a_i \, t^{m/2} \, g_i(t)\right)},$$

where we use that $a_i^{m/2} \, g_i(a_i) = \max_t \left(t^{m/2} \, g_i(t)\right)$, by definition of the $a_i = \arg\sup_t \{t^{m/2} g_i(t)\}$. Thus one finally obtains

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\mathbf{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)\right)^{-m/2} |\widehat{\mathbf{\Sigma}}_k|^{-1/2}}{\sum_{j=1}^{K} \widehat{\pi}_j \left((\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)^T \widehat{\mathbf{\Sigma}}_j^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_j)\right)^{-m/2} |\widehat{\mathbf{\Sigma}}_j|^{-1/2}}. \tag{23}$$

A.5 Proof of Proposition 5

Proof 12 *We re-write the $t-$distribution density as*

$$f_k(\boldsymbol{x}_i) = A_k L_{0ik} s_{ik}^{m/2} g_k(s_{ik}),$$

with $s_{ik} = \frac{(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}}$, $A_k = \pi^{-m/2}$,

$$g_k(t) = \frac{\Gamma(\frac{\nu_k + m}{2})}{\Gamma(\frac{\nu_k}{2})} \nu_k^{-m/2} \left[1 + \frac{t}{\nu_k}\right]^{-(\nu_k + m)/2}, \tag{24}$$

and the distribution-free factor

$$L_{0ik} = |\mathbf{\Sigma}_k|^{-1/2} [(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T \mathbf{\Sigma}_k^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)]^{-m/2}. \tag{25}$$

With this factorization of the density function, we can work on the two decoupled factors that let us write

$$\widehat{p}_{ik} = \frac{\widehat{\pi}_k A_k \widehat{L}_{0ik} \sup_t \{t^{m/2} g_k(t)\}}{\sum_{j=1}^{K} \widehat{\pi}_j A_j \widehat{L}_{0ij} \sup_t \{t^{m/2} g_j(t)\}}. \tag{26}$$

First, we compute the derivative of $t^{m/2} g_k(t)$ to get $\sup_{t \geq 0} \{t^{m/2} g_j(t)\}$

$$\frac{d}{dt}\left(\frac{\Gamma(\frac{\nu_k+m}{2})}{\Gamma(\frac{\nu_k}{2})}\nu_k^{-m/2}t^{m/2}\left[1+\frac{t}{\nu_k}\right]^{-(\nu_k+m)/2}\right) = \tag{27}$$

$$\left(\frac{m}{2}t^{m/2-1}\left[1+\frac{t}{\nu_k}\right]^{(\nu_k+m)/2} - \frac{t^{m/2}}{\nu_k}\frac{\nu_k+m}{2}\left(1+\frac{t}{\nu_k}\right)^{-(\nu_k+m)/2-1}\right) = \tag{28}$$

$$t^{m/2-1}\left(1+\frac{t}{\nu_k}\right)^{-(\nu_k+m)/2-1}\left[\frac{m}{2}\left(1+\frac{t}{\nu_k}\right) - \frac{\nu_k+m}{2}\frac{t}{\nu_k}\right]. \tag{29}$$

Equating the latter to 0, we get two possible solutions: $t = 0$ and $t = m$. Then, m maximizes $t^{m/2}g_j(t)$ and the maximum is reached and is

$$\frac{\Gamma(\frac{\nu_k+m}{2})}{\Gamma(\frac{\nu_k}{2})}\left(\frac{\nu_k}{m}\right)^{-m/2}\left[1+\frac{m}{\nu_k}\right]^{-(\nu_k+m)/2}.$$

Case large ν_k's and fixed m:
For ν_k tending to infinity and fixed m, one retrieves the Gaussian case as follows

$$\frac{\Gamma(\frac{\nu_k+m}{2})}{\Gamma(\frac{\nu_k}{2})}\left(\frac{\nu_k}{m}\right)^{-m/2}\left[1+\frac{m}{\nu_k}\right]^{-(\nu_k+m)/2} \approx \tag{30}$$

$$\frac{\Gamma(\frac{\nu_k}{2})(\frac{\nu_k}{2})^{-m/2}}{\Gamma(\frac{\nu_k}{2})}\left(\frac{\nu_k}{m}\right)^{-m/2}\left[\left[1+\frac{1}{\nu_k/m}\right]^{\nu_k/m}\right]^{-m(\nu_k+m)/(2\nu_k)} \approx \tag{31}$$

$$m^{m/2}(2e)^{-m/2}, \tag{32}$$

where the $2^{-m/2}$ factor corresponds to the normalizing constant of the Gaussian case together with $A_k = \pi^{-m/2}$.

Case large ν_k's and m:
If ν_k are fixed and m tends to infinity, as in Gaussian case, one notice that this part of the likelihood diverges. Consequently, we assume ν_k and m are large and of the same rate $\frac{\nu_k}{m} \to c_k$.

Then, using the Stirling approximation of the Gamma function

$$\Gamma(z) = \sqrt{\frac{2\pi}{z}}\left(\frac{z}{e}\right)^z\left(1+O\left(\frac{1}{z}\right)\right),$$

we derive the following approximations.

$$
\begin{aligned}
\frac{\Gamma(\frac{\nu_k+m}{2})}{\Gamma(\frac{\nu_k}{2})}\left(\frac{\nu_k}{m}\right)^{-m/2}\left[1+\frac{m}{\nu_k}\right]^{-(\nu_k+m)/2} &= \\
\frac{\Gamma\left(\frac{(1+c_k)m}{2}\right)}{\Gamma\left(\frac{c_k m}{2}\right)}c_k^{-m/2}\left[1+\frac{1}{c_k}\right]^{-(1+c_k)m/2} &= \\
\frac{\sqrt{\frac{4\pi}{(1+c_k)m}}\left(\frac{(1+c_k)m}{2e}\right)^{(1+c_k)m/2}\left(1+O\left(\frac{1}{m}\right)\right)}{\sqrt{\frac{4\pi}{c_k m}}\left(\frac{c_k m}{2e}\right)^{c_k m/2}\left(1+O\left(\frac{1}{m}\right)\right)}c_k^{-m/2}\left[1+\frac{1}{c_k}\right]^{-(1+c_k)m/2} &= \\
\sqrt{\frac{c_k}{1+c_k}}(2e)^{-\frac{m}{2}}\left(\frac{1+c_k}{c_k}\right)^{(1+c_k)m/2} m^{\frac{m}{2}}\frac{\left(1+O\left(\frac{1}{m}\right)\right)}{\left(1+O\left(\frac{1}{m}\right)\right)}\left[\frac{1+c_k}{c_k}\right]^{-(1+c_k)m/2} &= \\
\sqrt{\frac{c_k}{1+c_k}}(2e)^{-\frac{m}{2}} m^{\frac{m}{2}}\frac{\left(1+O\left(\frac{1}{m}\right)\right)}{\left(1+O\left(\frac{1}{m}\right)\right)}.
\end{aligned}
$$

When replacing this expression in (26) *we derived the following approximation,*

$$
\begin{aligned}
\widehat{p}_{ik} &= \frac{\widehat{\pi}_k\widehat{L}_{0ik}\sqrt{\frac{c_k}{1+c_k}}(2e)^{-\frac{m}{2}} m^{\frac{m}{2}}\frac{\left(1+O\left(\frac{1}{m}\right)\right)}{\left(1+O\left(\frac{1}{m}\right)\right)}}{\sum_{j=1}^{K}\left(\widehat{\pi}_j\widehat{L}_{0ij}\sqrt{\frac{c_j}{1+c_j}}(2e)^{-\frac{m}{2}} m^{\frac{m}{2}}\frac{\left(1+O\left(\frac{1}{m}\right)\right)}{\left(1+O\left(\frac{1}{m}\right)\right)}\right)} \\
&= \frac{\widehat{\pi}_k\widehat{L}_{0ik}\sqrt{\frac{c_k}{1+c_k}}\left(1+O\left(\frac{1}{m}\right)\right)}{\sum_{j=1}^{K}\left(\widehat{\pi}_j\widehat{L}_{0ij}\sqrt{\frac{c_j}{1+c_j}}\left(1+O\left(\frac{1}{m}\right)\right)\right)} \\
&= \frac{\widehat{\pi}_k\widehat{L}_{0ik}\sqrt{\frac{c_k}{1+c_k}}+O\left(\frac{1}{m}\right)}{\sum_{j=1}^{K}\left(\widehat{\pi}_j\widehat{L}_{0ij}\sqrt{\frac{c_j}{1+c_j}}\right)+O\left(\frac{1}{m}\right)}.
\end{aligned}
\tag{33}
$$

Finally, one obtains for $i=1,\ldots,n$ *and* $k=1,\ldots,K$

$$
\widehat{p}_{ik}=\frac{\widehat{\pi}_k\widehat{L}_{0ik}\sqrt{\frac{c_k}{1+c_k}}}{\sum_{j=1}^{K}\left(\widehat{\pi}_j\widehat{L}_{0ij}\sqrt{\frac{c_j}{1+c_j}}\right)}+O\left(\frac{1}{m}\right).
$$

If $c_k=c$ *for all* k*, we retrieve the same result as the particular case developed in Section 2.2.*

A.6 Proof of Proposition 6

Proof 13 *By Tyler's Theorem that applies to elliptical distributions under the assumptions included in the proposition, we have the convergence of the scatter matrix* $\widehat{\mathbf{\Sigma}}_k$ *to the true* $\mathbf{\Sigma}_k$ *in probability [Tyler, 1987, Theorem 4.1].*

Then, by applying the continuous mapping theorem, it follows that $\widehat{\mathbf{\Sigma}}_k^{-1} \xrightarrow{\mathcal{P}} {\mathbf{\Sigma}_k}^{-1}$. *Given that* $\mathbf{x}_i$, *one has*

$$\begin{aligned}\widehat{\tau}_{ik} &= \frac{(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\mathbf{\Sigma}}_k^{-1} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_k)}{m} \\ &= \frac{(\sqrt{\tau_{ik}}\mathbf{A}_k\mathbf{q}_i + \boldsymbol{\mu}_k - \widehat{\boldsymbol{\mu}}_k)^T \widehat{\mathbf{\Sigma}}_k^{-1} (\sqrt{\tau_{ik}}\mathbf{A}_k\mathbf{q}_i + \boldsymbol{\mu}_k - \widehat{\boldsymbol{\mu}}_k)}{m}.\end{aligned}$$

Combining $\widehat{\mathbf{\Sigma}}_k^{-1} \xrightarrow{\mathcal{P}} {\mathbf{\Sigma}_k}^{-1}$ *and* $\widehat{\boldsymbol{\mu}}_k \xrightarrow{\mathcal{P}} \boldsymbol{\mu}_k$ *and the Slutsky theorem leads to*

$$\widehat{\tau}_{ik} \xrightarrow{\mathcal{P}} \frac{\sqrt{\tau_{ik}}\mathbf{q}_i^T\mathbf{A}_k^T\mathbf{\Sigma}^{-1}\sqrt{\tau_{ik}}\mathbf{A}_k\mathbf{q}}{m} = \frac{\tau_{ik}\mathbf{q}_i^T\mathbf{q}_i}{m}.$$

Furthermore,

$$\frac{\tau_{ik}\mathbf{q}_i^T\mathbf{q}_i}{m} = \frac{\tau_{ik}\sum_{l=1}^m (\mathbf{q}_i)_l^2}{m},$$

with the components $(\mathbf{q}_i)_1^2, ..., (\mathbf{q}_i)_m^2$ *i.i.d. distributed as* $\chi^2(1)$ *because* $\mathbf{q}_i \sim \mathcal{N}(0, \mathbf{I}_m)$. *Thus,* $\widehat{\tau}_{ik}$ *tends to* $\tau_{ik}\frac{\chi^2(m)}{m}$.

Now, to assess the behavior when m *tends to infinity, one has thanks to the Central Limit Theorem that, since* $E[(\mathbf{q}_i)_1^2] = 1$ *and* $V[(\mathbf{q}_i)_1^2] = 2$, *for* m *large enough*

$$\frac{\tau_{ik}\sum_{l=1}^m (\mathbf{q}_i)_l^2}{m} \sim \mathcal{N}(\tau_{ik}, 2\tau_{ik}^2/m),$$

Finally, sequentially combining the approximations and imposing the condition $n > m(2m-1)$ *to ensure the existence and uniqueness of the estimator, one obtains the limiting distribution for* $(\widehat{\tau}_{ik} - \tau_{ik})$.

References

J. D. Banfield and A. E. Raftery. Model-based gaussian and non-gaussian clustering. *Biometrics*, 49(3):803–821, 1993. ISSN 0006341X, 15410420. URL http://www.jstor.org/stable/2532201.

M. Bilodeau and D. Brenner. *Robustness*, pages 206–242. Springer New York, New York, NY, 1999. ISBN 978-0-387-22616-3. doi: 10.1007/978-0-387-22616-3_13. URL https://doi.org/10.1007/978-0-387-22616-3_13.

G. Boente, M. Salibián Barrera, and D. E. Tyler. A characterization of elliptical distributions and some optimality properties of principal components for functional data. *Journal of Multivariate Analysis*, 131:254 – 264, 2014. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2014.07.006. URL http://www.sciencedirect.com/science/article/pii/S0047259X14001638.

C. Bouveyron and C. Brunet-Saumard. Model-based clustering of high-dimensional data: A review. *Computational Statistics & Data Analysis*, 71:52 – 78, 2014. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2012.12.008. URL http://www.sciencedirect.com/science/article/pii/S0167947312004422.

R. P. Browne and P. D. McNicholas. A mixture of generalized hyperbolic distributions. *Canadian Journal of Statistics*, 43(2):176–198, 2015. doi: 10.1002/cjs.11246. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cjs.11246.

N. A. Campbell. Mixture models and atypical values. *Journal of the International Association for Mathematical Geology*, 16(5):465–477, 1984. ISSN 1573-8868. doi: 10.1007/BF01886327. URL https://doi.org/10.1007/BF01886327.

R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Trans. Knowl. Discov. Data*, 10 (1):5:1–5:51, 2015. ISSN 1556-4681. doi: 10.1145/2733381. URL http://doi.acm.org/10.1145/2733381.

G. Celeux and G. Govaert. Gaussian parsimonious clustering models. *Pattern Recognition*, 28 (5):781 – 793, 1995. ISSN 0031-3203. doi: https://doi.org/10.1016/0031-3203(94)00125-6. URL http://www.sciencedirect.com/science/article/pii/0031320394001256.

E. Conte and M. Longo. Characterisation of radar clutter as a spherically invariant random process. *IEE Proceedings F - Communications, Radar and Signal Processing*, 134(2):191–197, 1987. ISSN 0143-7070. doi: 10.1049/ip-f-1.1987.0035.

E. Conte, A. De Maio, and G. Ricci. Covariance matrix estimation for adaptive CFAR detection in compound-gaussian clutter. *IEEE Transactions on Aerospace and Electronic Systems*, 38 (2):415–426, 2002a. ISSN 0018-9251. doi: 10.1109/TAES.2002.1008976.

E. Conte, A. De Maio, and G. Ricci. Recursive estimation of the covariance matrix of a compound-gaussian process and its application to adaptive cfar detection. *IEEE Transactions on Signal Processing*, 50(8):1908–1915, 2002b. ISSN 1053-587X. doi: 10.1109/TSP.2002.800412.

P. Coretto and C. Hennig. Consistency, breakdown robustness, and algorithms for robust improper maximum likelihood clustering. *J. Mach. Learn. Res.*, 18(1):5199–5237, January 2017. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=3122009.3208023.

Pietro Coretto and Christian Hennig. *otrimle: Robust Model-Based Clustering*, 2019. R package version 1.3.

R. Couillet, F. Pascal, and J. W. Silverstein. Robust estimates of covariance matrices in the large dimensional regime. *IEEE Transactions on Information Theory*, 60(11):7269–7278, Nov 2014. ISSN 1557-9654. doi: 10.1109/TIT.2014.2354045.

R. Couillet, F. Pascal, and J. W. Silverstein. The random matrix regime of Maronna's M-estimator with elliptically distributed samples. *Journal of Multivariate Analysis*, 139:56 – 78, 2015. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2015.02.020. URL http://www.sciencedirect.com/science/article/pii/S0047259X15000676.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39:1–38, 1977. URL http://web.mit.edu/6.435/www/Dempster77.pdf.

C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, 2002. ISSN 01621459. URL http://www.jstor.org/stable/3085676.

J. Frontera-Pons, M. Veganzones, F. Pascal, and J-P. Ovarlez. Hyperspectral Anomaly Detectors using Robust Estimators. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)*, 9(2):720–731, february 2016.

L. A. García-Escudero, A. Gordaliza, C. Matrán, and A. Mayo-Iscar. A general trimming approach to robust cluster analysis. *Ann. Statist.*, 36(3):1324–1345, 2008. doi: 10.1214/07-AOS515. URL `https://doi.org/10.1214/07-AOS515`.

I. D. Gebru, X. Alameda-Pineda, F. Forbes, and R. Horaud. EM algorithms for weighted-data clustering with application to audio-visual scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(12):2402–2415, 2016. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2522425. URL `https://doi.org/10.1109/TPAMI.2016.2522425`.

F. Gini and A. Farina. Vector subspace detection in compound-gaussian clutter. part I: survey and new results. *IEEE Transactions on Aerospace and Electronic Systems*, 38(4):1295–1311, 2002. ISSN 0018-9251. doi: 10.1109/TAES.2002.1145751.

F. Gini, M. V. Greco, M. Diani, and L. Verrazzani. Performance analysis of two adaptive radar detectors against non-gaussian real sea clutter data. *IEEE Transactions on Aerospace and Electronic Systems*, 36(4):1429–1439, 2000. ISSN 0018-9251. doi: 10.1109/7.892695.

J. D. Gonzalez. *Métodos de clustering robustos*. PhD thesis, Universidad de Buenos Aires, 2019.

J. D. Gonzalez, V. J. Yohai, and R. H. Zamar. Robust Clustering Using Tau-Scales. *arXiv e-prints*, art. arXiv:1906.08198, Jun 2019.

C. Hennig. Clustering strategy and method selection. In Christian Hennig, Marina Meila, Fionn Murtagh, and Roberto Rocci, editors, *Handbook of Cluster Analysis*, chapter 31. CRC Press, 2015.

John T Kent, David E Tyler, et al. Redescending m-estimates of multivariate location and scatter. *The Annals of Statistics*, 19(4):2102–2119, 1991.

Y. LeCun and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–104 Vol.2, 2004. doi: 10.1109/CVPR.2004.1315150.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. ISSN 0018-9219. doi: 10.1109/5.726791.

S. Lee and G. J. McLachlan. Finite mixtures of multivariate skew t-distributions: some recent and new results. *Statistics and Computing*, 24(2):181–202, March 2014. ISSN 1573-1375. doi: 10.1007/s11222-012-9362-4. URL `https://doi.org/10.1007/s11222-012-9362-4`.

Z. Liao and R. Couillet. A large dimensional analysis of least squares support vector machines. *IEEE Transactions on Signal Processing*, 67:1065–1074, 2017.

R. A. Maronna. Robust M-Estimators of multivariate location and scatter. *The Annals of Statistics*, 4(1):51–67, 1976. ISSN 00905364. URL `http://www.jstor.org/stable/2957994`.

L. McInnes and J. Healy. Accelerated hierarchical density based clustering. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 33–42, 2017. doi: 10.1109/ICDMW.2017.12.

L. McInnes, J. Healy, and S. Astels. hdbscan: Hierarchical density based clustering. *J. Open Source Software*, 2(11):205, 2017.

L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv e-prints*, 2018.

G.J. McLachlan. 9 The classification and mixture maximum likelihood approaches to cluster analysis. In *Classification Pattern Recognition and Reduction of Dimensionality*, volume 2 of *Handbook of Statistics*, pages 199 – 208. Elsevier, 1982. doi: https://doi.org/10.1016/S0169-7161(82)02012-4. URL `http://www.sciencedirect.com/science/article/pii/S0169716182020124`.

P.D. McNicholas. *Mixture model-based classification*. 10 2016. doi: 10.1201/9781315373577.

T. M. Mitchell. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997. ISBN 0070428077, 9780070428072.

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, pages 849–856, Cambridge, MA, USA, 2001. MIT Press. URL `http://dl.acm.org/citation.cfm?id=2980539.2980649`.

E. Ollila and D. E. Tyler. Distribution-free detection under complex elliptically symmetric clutter distribution. In *2012 IEEE 7th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 413–416, 2012.

E. Ollila, D.E. Tyler, V. Koivunen, and H.V. Poor. Complex elliptically symmetric distributions: Survey, new results and applications. *Signal Processing, IEEE Transactions on*, 60(11):5597–5625, November 2012. ISSN 1053-587X. doi: 10.1109/TSP.2012.2212433.

F. Pascal, Y. Chitour, J-P. Ovarlez, P. Forster, and P. Larzabal. Covariance structure maximum-likelihood estimates in compound gaussian noise: Existence and algorithm analysis. *Trans. Sig. Proc.*, 56(1):34–48, January 2008. ISSN 1053-587X. doi: 10.1109/TSP.2007.901652. URL `http://dx.doi.org/10.1109/TSP.2007.901652`.

F. Pascal, L. Bombrun, J-Y. Tourneret, and Y. Berthoumieu. Parameter estimation for multivariate generalized gaussian distributions. *IEEE Transactions on Signal Processing*, 61(23):5960–5971, 2013.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research*, 12:2825–2830, 2011.

D. Peel and G. J. McLachlan. Robust mixture modelling using the t distribution. *Statistics and Computing*, 10(4):339–348, Oct 2000. ISSN 1573-1375. doi: 10.1023/A:1008981510081. URL `https://doi.org/10.1023/A:1008981510081`.

V. Roizman, M. Jonckheere, and F. Pascal. Robust clustering and outlier rejection using the mahalanobis distance distribution. In *2020 28th European Signal Processing Conference, EUSIPCO*, 2020. To appear.

P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53 – 65, 1987. ISSN 0377-0427. doi: https://doi.org/10.1016/0377-0427(87)90125-7. URL `http://www.sciencedirect.com/science/article/pii/0377042787901257`.

scikit-learn developers. Clustering–scikit-learn v0.20.3, 2019. URL `https://scikit-learn.org/stable/modules/clustering.html#clustering`.

S. Tadjudin and D. A. Landgrebe. Robust parameter estimation for mixture model. *IEEE Transactions on Geoscience and Remote Sensing*, 38(1):439–445, 2000. ISSN 0196-2892. doi: 10.1109/36.823939.

D. E. Tyler. A distribution-free M-estimator of multivariate scatter. *The Annals of Statistics*, 15(1):234–251, 1987.

L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

N. X. Vinh, J. Epps, and J. Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.*, 11: 2837–2854, December 2010. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1756006.1953024`.

K. Wang, S. Ng, and G. J. McLachlan. Multivariate skew t mixture models: Applications to fluorescence-activated cell sorting data. In *2009 Digital Image Computing: Techniques and Applications*, pages 526–531, 2009.

L. M. Weber and M. D. Robinson. Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. *Cytometry Part A*, 89(12):1084–1096, 2016. doi: 10.1002/cyto.a.23030. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/cyto.a.23030`.

Y. Wei, Y. Tang, and P. D. McNicholas. Mixtures of Generalized Hyperbolic Distributions and Mixtures of Skew-t Distributions for Model-Based Clustering with Incomplete Data. *arXiv e-prints*, Mar 2017.

Chong Wu, Can Yang, Hongyu Zhao, and Ji Zhu. On the convergence of the em algorithm: A data-adaptive analysis, 2016.

K. Yao. A representation theorem and its applications to spherically invariant random processes. *IEEE Trans.-IT*, 19(5):600–608, September 1973.

K. Yu, X. Dang, H. Bart, and Y. Chen. Robust model-based learning via spatial-em algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 27(6):1670–1682, 2015. ISSN 1041-4347. doi: 10.1109/TKDE.2014.2373355.

T. Zhang, X. Cheng, and A. Singer. Marcenko pastur law for Tyler's M-estimator. *Journal of Multivariate Analysis*, 149:114 – 123, 2016. ISSN 0047-259X. doi: https://doi.org/10.1016/j.jmva.2016.03.010. URL http://www.sciencedirect.com/science/article/pii/S0047259X16300069.

Setup	Error	GMM-EM	t-EM	F-EM	TCLUST	OTRIMLE
1	$\boldsymbol{\Sigma}_1$	0.0094	0.0070	0.0073	0.0102	0.0101
1	$\boldsymbol{\Sigma}_2$	0.0371	0.0132	0.0151	0.0371	0.0260
1	$\boldsymbol{\Sigma}_3$	0.0128	0.0046	0.0046	0.0090	0.0062
1	$\boldsymbol{\mu}_1$	0.2110	0.1628	0.1670	0.2304	0.1808
1	$\boldsymbol{\mu}_2$	1.6277	0.2100	0.2513	1.2186	0.3079
1	$\boldsymbol{\mu}_3$	1.2004	0.1238	0.1401	0.8618	0.1420
2	$\boldsymbol{\Sigma}_1$	0.0098	0.0104	0.0083	0.0135	0.0372
2	$\boldsymbol{\Sigma}_2$	0.0083	0.0068	0.0076	0.0075	0.0099
2	$\boldsymbol{\Sigma}_3$	0.0103	0.0074	0.0094	0.0088	0.0289
2	$\boldsymbol{\mu}_1$	0.2168	0.2200	0.1853	0.3115	2.1054
2	$\boldsymbol{\mu}_2$	0.1879	0.1379	0.1570	0.1405	0.2337
2	$\boldsymbol{\mu}_3$	0.2063	0.1532	0.2077	0.1895	1.0695
3	$\boldsymbol{\Sigma}_1$	0.0019	0.0025	0.0013	0.0016	0.0011
3	$\boldsymbol{\Sigma}_2$	0.0016	0.0034	0.0014	0.0014	0.0000
3	$\boldsymbol{\Sigma}_3$	0.0022	0.0012	0.0012	0.0011	0.0029
3	$\boldsymbol{\mu}_1$	0.5565	10.1226	0.2967	0.3469	0.3714
3	$\boldsymbol{\mu}_2$	0.3655	6.1910	0.3025	0.3374	4.6289
3	$\boldsymbol{\mu}_3$	0.6081	0.2885	0.3060	0.2781	1.7128
4	$\boldsymbol{\Sigma}_1$	0.0070	0.0144	0.0060	0.0055	0.0055
4	$\boldsymbol{\Sigma}_2$	0.0085	0.0085	0.0064	0.0055	0.0056
4	$\boldsymbol{\Sigma}_3$	0.0254	0.0076	0.0065	0.0057	0.0056
4	$\boldsymbol{\mu}_1$	0.1876	3.1683	0.1249	0.1048	0.1071
4	$\boldsymbol{\mu}_2$	0.7796	0.7783	0.1382	0.1179	0.1167
4	$\boldsymbol{\mu}_3$	3.1758	0.2386	0.1436	0.1080	0.1053
5	$\boldsymbol{\Sigma}_1$	0.0376	0.0384	0.0134	0.0328	0.0117
5	$\boldsymbol{\Sigma}_2$	0.0481	0.0361	0.0191	0.0322	0.0181
5	$\boldsymbol{\Sigma}_3$	0.0000	0.0091	0.0110	0.0093	0.0095
5	$\boldsymbol{\mu}_1$	3.9425	2.0617	0.0751	1.9759	0.1822
5	$\boldsymbol{\mu}_2$	0.6168	1.3857	0.3063	2.2991	0.2996
5	$\boldsymbol{\mu}_3$	5.1633	0.1457	0.1659	0.1434	0.1424

Table 3: Average of the norm of the error in the estimation of the main parameters in the different setups.

Setup	Error	GMM-EM	t-EM	F-EM	TCLUST	OTRIMLE
1	AMI	0.4491	0.7095	0.6809	0.4036	0.4197
1	ARI	0.4373	0.7895	0.7513	0.4293	0.2851
2	AMI	0.8784	0.8843	0.8836	0.7414	0.5342
2	ARI	0.9156	0.9233	0.9208	0.8476	0.4809
3	AMI	0.7753	0.5514	0.9597	0.8377	0.4936
3	ARI	0.7056	0.5624	0.9722	0.9120	0.4497
4	AMI	0.7373	0.6115	0.7836	0.9551	0.9476
4	ARI	0.5709	0.5603	0.8159	0.9690	0.9661
5	AMI	0.1058	0.5479	0.6711	0.5573	0.6426
5	ARI	0.0187	0.5038	0.6946	0.4947	0.7265

Table 4: Average clustering metrics in the different setups.

Set	Set name	m	n	k
1	**MNIST 3-8**	30	1600	2
2	**MNIST 7-1**	30	1600	2
3	**MNIST 3-8-6**	30	1800	3
4	**MNIST 3-8-6 + noise**	30	2080	3
5	**NORB**	30	1600	4
6	**20newsgroup**	100	2000	4

Table 5: Characteristics of the subsets of the data sets that have been used to compare the algorithms. The data sets are variations of the MNIST data set, small NORB and *20newsgroup*.

Set	k-means	GMM	t-EM	F-EM	spectral	TCLUST	RIMLE
1	0.2203	0.4878	0.5520	**0.5949**	0.5839	0.5666	0.3875
2	0.7839	0.8414	**0.8947**	0.8811	0.8852	0.5705	0.3875
3	0.6149	0.7159	0.7847	0.7918	**0.8272**	0.7818	0.6077
4	0.3622	0.4418	0.4596	0.4664	0.3511	**0.6047**	0.3553
5	0.0012	0.0476	0.4370	**0.5321**	~ 0	0.1516	0.2312
6	0.2637	0.3526	0.4496	**0.4873**	0.1665	0.2604	0.0686

Table 6: Median AMI index measuring the performance of k-means, GMM-EM, t-EM, TCLUST, RIMLE, spectral and our algorithm (F-EM) results for variations of the MNIST data set, small NORB and *20newsgroup*.

Set	k-means	GMM	t-EM	F-EM	spectral	TCLUST	RIMLE
1	0.2884	0.5716	0.6397	**0.6887**	0.6866	0.6847	0.2494
2	0.8486	0.8905	**0.9432**	0.9360	0.9384	0.6885	0.2493
3	0.6338	0.7332	0.8262	0.8306	**0.8542**	0.8366	0.4274
4	0.4475	0.4909	0.5296	0.5548	0.3115	**0.6908**	0.1498
5	0.0015	0.0468	0.4223	**0.5067**	~ 0	0.1330	0.1472
6	0.1883	0.2739	0.4426	**0.5114**	0.0987	0.2664	0.0026

Table 7: Median AR index measuring the performance of k-means, GMM-EM, t-EM, TCLUST, RIMLE, spectral and our algorithm (F-EM) results for variations of the MNIST data set, small NORB and *20newsgroup*.

Set	k-means	GMM	t-EM	F-EM	spectral	TCLUST	RIMLE
1	0.7687	0.8781	0.9093	**0.9150**	0.9050	0.8881	0.5193
2	0.9606	0.9718	0.9856	**0.9868**	0.9844	0.8893	0.5193
3	0.8495	0.8976	0.9366	0.9390	**0.9476**	0.9183	0.5157
4	0.8144	0.8700	0.8894	0.8966	0.5444	**0.9247**	0.4988
5	0.2725	0.3487	0.6528	**0.6975**	0.2600	0.4087	0.3887
6	0.5755	0.7100	0.6900	**0.8030**	0.5220	0.5740	0.2970

Table 8: Median accuracy measuring the performance (correct classification rate) of k-means, GMM-EM, t-EM, TCLUST, RIMLE, spectral and our algorithm (F-EM) results for variations of the MNIST data set, small NORB and *20newsgroup*.