Clustering and classification of fuzzy data using the fuzzy EM algorithm

Benjamin Quost[1] and Thierry Denœux
UMR CNRS 7253 Heudiasyc
Université de Technologie de Compiègne
BP 20529 - F-60205 Compiègne cedex - France

Abstract

In this article, we address the problem of clustering imprecise data using a finite mixture of Gaussians. We propose to estimate the parameters of the model using the fuzzy EM algorithm. This extension of the EM algorithm allows us to handle imprecise data represented by fuzzy numbers. First, we briefly recall the principle of the fuzzy EM algorithm. Then, we provide closed-forms for the parameter estimates in the case of Gaussian fuzzy data. We also describe a Monte-Carlo procedure for estimating the parameter updates in the general case. Experiments carried out on synthetic and real data demonstrate the interest of our approach for taking into account attribute and label uncertainty.

1. Introduction

Gaussian mixture models (GMMs) are very powerful tools for modeling multivariate distributions [1]. This model assumes the data to arise from a random sample, whose distribution is a finite mixture of Gaussians. The major difficulty is to estimate the parameters of the model. Generally, these estimates are computed using the maximum-likelihood (ML) approach, through an iterative procedure known as the expectation-maximization (EM) algorithm [2]. Once the parameter values are known, the posterior probabilities of each data point may be computed, thus giving a fuzzy partition of the data. Then, a crisp partition may be obtained by assigning a label to each instance according to the Maximum A Posteriori (MAP) rule.

In some applications, the precise value taken by the variables may be difficult or even impossible to assess. For example, in acoustical engineering, flaws can be detected on storage tanks by pressurizing the device and measuring the resulting acoustical emissions; this technique provides locations of acoustic events associated with imprecision degrees [3]. As stressed in [4], epidemiological data, such as address of the patients, hospital admission times, etc, often suffer from

[1]Corresponding author. E-mail: benjamin.quost@hds.utc.fr. Fax: +33 (0)3 44 23 44 77.

imprecision. The interest of taking into account the uncertainty of the measurements has been demonstrated [5]. For this purpose, various formalisms allowing to quantify and propagate imprecision have been proposed, among which fuzzy sets [6, 7, 8, 9]. Some consider that the data at hand are intrinsically fuzzy, a position that has been known as the *physical* or *ontic interpretation* of fuzziness [10, 11]. Here, we rather adopt an *epistemic interpretation*, in which a fuzzy number "imperfectly specifies a value that is existing and precise, but not measurable with exactitude under the given observation conditions" [6]. In this setting, partial knowledge of the actual precise value taken by a random variable of interest is represented by a possibility distribution.

Several recent papers have addressed the problem of clustering imprecise data, for which both the attributes and the class information may be partially known. For instance, in [12], the imprecise data at hand are represented by hyperspheres, and the clustering is achieved using a K-means-based procedure. In [13], the imprecision of the data is represented by convex hulls, and hierarchical clustering is used to partition the data. In [14], a density-based clustering algorithm, DBSCAN, is employed to cluster data in presence of noise. The work presented in [15] is of notable importance: it addresses the problem of clustering interval-valued data in the probabilistic setting of GMM parameter estimation. It should be stressed that the EM algorithm is particularly well-suited to this kind of imprecision. Indeed, handling such *censored* data dates back to the seminal article by Dempster *et al* [2], and is still the subject of investigations [16]. Many articles also address handling probabilistic uncertainty over the attributes: the partial knowledge of the value of a variable of interest is described by a probability density function (pdf). The data may then be partitioned in various ways, such as hierarchical clustering [17], or (non-parametric) density-based clustering [18]. Some authors introduced an extension of the K-means algorithm, called Uncertain K-means (UK-means) [19, 20, 21]. In [22], K-means, K-median, and K-centre strategies are used. We may also mention the work in [23], which considers micro-clustering for compressing data, and [24], which addresses the problem of subspace clustering from imprecise data. Reviews may be found in [25, 26, 27]. Eventually, many authors considered clustering uncertain data described by fuzzy numbers or belief functions. Almost all of them [28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39] perform clustering using a geometric approach related to the Fuzzy C-means (FCM) algorithm. The centers of the model and the membership degrees of the data to the clusters are computed iteratively so as to optimize a within-cluster variance criterion. In [40], an approach referred to as Belief K-modes was proposed in the belief functions framework. A notable exception is [4], where the author considers statistical tests to partition the data. Clustering multivariate fuzzy data using mixtures of distributions has been addressed in [41].

Additionally, the related problem of learning from imprecisely labeled instances has been widely addressed in the literature. Semi-supervised learning [42, 43] considers that a subset of instances have been associated with a (precise) label, the labels of the other instances being unknown. Partially supervised learning [44, 45, 46, 47, 48] encompasses this approach, by making it possible

to associate each instance with a set of (equally) plausible labels. Besides, the problem of noisy labels has been addressed in a probabilistic framework [49, 50, 51]. More recently, the general framework of belief functions has been successfully employed to estimate GMMs from imprecise and uncertain labels [52, 53, 54, 55]. Finally, it should be stressed out that the literature dedicated to fuzzy GMM estimation, or more generally imprecise or uncertain clustering, actually refers to associating precise instances with several classes in a imprecise way. For example, [56] addresses the problem of image segmentation where pixels may be mixed (e.g., in ground segmentation, the pixel may belong to both "water" and "forest" categories). As well, [57] proposes to make GMM estimation more flexible in order to fit curve manifolds, by using a trade-off between GMM and fuzzy C-means estimation according to the geometric shape of the dataset considered. Again, this work does not address the case where the data at hand are uncertain. In [58], an algorithm for performing hierarchical clustering of a set of precise instances is developed in the framework of belief functions. Finally, in [59], the authors consider parameter uncertainty, rather than attribute or class uncertainty. Once a GMM has been estimated using the precise data, membership functions are defined on the corresponding parameters, and the instances are linked to the classes by sets of likelihoods instead of probabilities.

In this paper, we propose to fit a GMM defined by precise parameters to uncertain data, where both the attribute vector and the class information may be partially known. This paper distinguishes thus from most of the works mentioned above in that the approach is parametric: we postulate a probabilistic model underlying the data, the parameters of which have to be estimated. Flexibility is a notable advantage of the method: in presence of scarce or low-quality data (or when it is justified by background knowledge), the model may be simplified via additional assumptions on the parameters to estimate, such as explained in [60]. We adopt here possibility distributions as mathematical tools for representing partial knowledge of the instance values or weak class information. In this case, the likelihood of the sample may be computed using Zadeh's definition of the probability of a fuzzy event [61]. Then, the fuzzy EM algorithm may be used to estimate the parameters maximizing this likelihood. This procedure was recently proposed by Denœux as an extension of the EM algorithm for imprecise data represented by fuzzy numbers [62] and by belief functions [63] [2]. Thus, unlike in [52, 53, 55], we consider here both attribute and class uncertainty in our estimation process. Note that [15] considered uncertain attribute values represented by intervals; besides, the approach described in [54] made it possible to take into account the fact that some attribute information were missing. In our approach, we propose instead a generic strategy for incorporating partial attribute information and class information. Any kind of partial knowledge may thus be integrated in the estimation process, as long as

[2] It should be noted that, due to the strong relationships between both formalisms, the work presented in the fuzzy case may be straightforwardly extended to the credal case.

the appropriate possibility function is specified. Under the epistemic interpretation of uncertainty adopted here, fuzzy numbers represent information that should have been precise under ideal conditions in the measurement process. Then, two philosophies may be considered. Following [59], the uncertainty may be propagated through the estimation process. In this case, the range of the family of models estimated accounts for the imperfection of the data. Alternatively, one may look for the best model, according to a particular criterion, given the data at hand [64]. This is the approach considered in this paper, where a generalized likelihood is used to measure the agreement between the model and the fuzzy observations available.

The article is organized as follows. Section 2 recalls the model. Then, we show how the FEM algorithm may be used in order to estimate the parameters of a GMM. In Section 3, we first concentrate on the particular case of multivariate Gaussian possibility distributions: then, closed forms may be obtained for the update equations of the model parameters. For computational reasons however, this may not be possible in the general case (e.g., for trapezoidal possibility distributions). Then, we present in Section 4 an efficient and versatile Monte-Carlo approximation procedure to overcome this problem. Section 5 addresses the problem of avoiding degenerate parameter estimates using Bayesian priors. Section 6 presents various experiments on synthetic and real data, in both clustering and classification settings. Eventually, Section 7 concludes the paper.

2. Model

In this section, we recall basic knowledge on Gaussian mixture models. Then, we formalize the estimation problem when the data at hand are fuzzy.

2.1. Data and generative model

From now on, we assume the existence of a random vector $\mathbf{Y} \in \mathcal{Y}$, referred to as the *complete data* vector, which describes the result of a random experiment. In our case, it takes the form of an iid sample: $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)$. The ith element $\mathbf{y}_i = (\mathbf{x}_i, \mathbf{z}_i)$ of this sample is composed of two pieces of information:

- an attribute vector $\mathbf{x}_i$, supposed to be the realization of a random vector $\mathbf{X} \in \mathbb{R}^p$ drawn from a mixture of K Gaussian distributions with proportions $\pi_1, \ldots, \pi_g$. The marginal distribution of the attribute vectors thus writes

$$g_X(\mathbf{x}_1, \ldots, \mathbf{x}_n, \Psi) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \, g(\mathbf{x}_i; \nu_k, \Sigma_k), \tag{1a}$$

  $g(\cdot; \nu_k, \Sigma_k)$ being the pdf of the multivariate Gaussian distribution with mean vector ν_k and covariance matrix Σ_k:

$$g(\mathbf{x}; \nu_k, \Sigma_k) = (2\pi)^{p/2} |\Sigma_k|^{-\frac{1}{2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \nu_k)^t \Sigma_k^{-1} (\mathbf{x} - \nu_k) \right). \tag{1b}$$

In the following, for the sake of clarity, we will write $g_k(\cdot) = g(\cdot; \nu_k, \Sigma_k)$.

- a vector $\mathbf{z}_i$ of realizations of Bernoulli random variables Z_{ik}, indicating which component of the GMM generated $\mathbf{x}_i$ ($z_{ik} = 1$ if $\mathbf{x}_i$ was drawn from the component ω_k of the mixture, and $z_{ik} = 0$ otherwise). Note that in clustering problems, these variables are typically unknown and must be estimated. In discriminant analysis, however, the class indicators are known, in which case the joint density of the sample can be written as

$$f(\mathbf{y}) = \prod_{i=1}^{n} \prod_{k=1}^{K} \left(\pi_k g_k(\mathbf{x}_i)\right)^{z_{ik}} . \tag{1c}$$

  In the latter case, the ML estimates of the model parameters can be easily computed by maximizing the *complete likelihood* defined by Equation (1c). In the former case, however, proceeding with the *observed likelihood* (1a) is difficult; the EM algorithm [1, 2] may then be used for this purpose.

In this paper, we address the case where the realizations are not precisely observed. We only have a partial knowledge of the actual values $\mathbf{x}_i$ and $\mathbf{z}_i$, in the form of fuzzy subsets $\tilde{\mathbf{y}}_i = (\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i)$ with possibility distribution $\mu_{\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i} = \mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})\mu_{\tilde{\mathbf{z}}_i}(\mathbf{z})$. The value $\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})$ may be interpreted as a degree of possibility that the actual realization of the random vector $\mathbf{X}$ is $\mathbf{x}$. For example, assuming a Gaussian possibility distribution with mean vector $\mathbf{m}_i$ and covariance matrix S_i for $\mathbf{x}_i$ gives:

$$\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{m}_i)^t S_i^{-1}(\mathbf{x} - \mathbf{m}_i)\right). \tag{2}$$

The imprecise class information is represented by a vector $\tilde{\mathbf{z}}_i = (\delta_{i1}, \ldots, \delta_{ig})$, where $\delta_{ik} \in [0, 1]$ is the degree of possibility that $\mathbf{x}_i$ was actually drawn from ω_k. Thus, the degree of possibility that $\mathbf{z}_i$ represents the actual component from which $\mathbf{x}_i$ was generated is

$$\mu_{\tilde{\mathbf{z}}_i}(\mathbf{z}_i) = \prod_{k=1}^{K} \left(\delta_{ik}\right)^{z_{ik}} . \tag{3}$$

Note that complete ignorance of the actual class of an instance corresponds to $\delta_{ik} = 1$ for all $k = 1, \ldots, K$; while full certainty that its actual class of $\mathbf{x}_i$ is ω_k corresponds to $\delta_{ik} = 1$ and $\delta_{i\ell} = 0$ for all $\ell \neq k$.

2.2. Generalized likelihood of fuzzy data

Let us assume that a fuzzy sample $\tilde{\mathbf{y}} = (\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n)$ has been observed. According to Zadeh's definition of the probability of a fuzzy event [65], the probability of this fuzzy sample is the expectation of its possibility distribution:

$$\mathbb{P}(\tilde{\mathbf{y}}; \Psi) = \mathbb{E}_\Psi\left[\mu_{\tilde{\mathbf{y}}}(\mathbf{y})\right] = \int_{\mathcal{Y}} \mu_{\tilde{\mathbf{y}}}(\mathbf{y}) g_Y(\mathbf{y}; \Psi) d\mathbf{y}. \tag{4}$$

We assume here that the fuzzy observations are cognitively independent:

$$\mu_{\tilde{\mathbf{y}}}(\mathbf{y}) = \prod_{i=1}^{n} \mu_{\tilde{\mathbf{y}}_i}(\mathbf{y}_i). \tag{5}$$

Thus, the likelihood of the imprecisely observed data is [62]:

$$\begin{aligned} L(\Psi; \tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n) &= \mathbb{P}(\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n; \Psi) = \prod_{i=1}^{n} \mathbb{E}_\Psi \left[\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})\, \mu_{\tilde{\mathbf{z}}_i}(\mathbf{z})\right], & \text{(6a)} \\ &= \prod_{i=1}^{n} \sum_{k=1}^{K} \mathbb{P}(Z_{ik} = 1) \mathbb{E}_\Psi \left[\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})\, \mu_{\tilde{\mathbf{z}}_i}(\mathbf{z}) | Z_{ik} = 1\right], & \text{(6b)} \\ &= \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k\, \delta_{ik}\, \mathbb{E}_\Psi \left[\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}) | Z_{ik} = 1\right]. & \text{(6c)} \end{aligned}$$

Maximizing this *observed log-likelihood* in order to compute an estimate of the parameter vector Ψ is difficult. As mentioned in Section 2, computing these estimates would be straightforward, should the data at hand be precise and complete. Then, the model may be easily estimated by maximizing the complete likelihood (1c), or equivalently the *complete log-likelihood*:

$$\begin{aligned} \log L(\Psi; \mathbf{y}_1, \ldots, \mathbf{y}_n) = &\sum_{k=1}^{K} \left(\log \pi_k \sum_{i=1}^{n} z_{ik} \right) - \frac{p}{2} \log(2\pi) \sum_{k=1}^{K} \sum_{i=1}^{n} z_{ik} \\ &- \frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{n} z_{ik} \left(\log |\Sigma_k| + (\mathbf{x}_i - \nu_k)^t \Sigma_k^{-1} (\mathbf{x}_i - \nu_k) \right). \end{aligned} \tag{7}$$

3. Gaussian fuzzy numbers: estimation via the FEM algorithm

As mentioned in Section 2.2, computing parameter estimates by maximizing the observed likelihood is difficult. It is possible to overcome this problem by using an extension of the expectation-maximization (EM) algorithm for fuzzy data, known as the fuzzy EM (FEM) algorithm [62], which we briefly recall. Then, we assume that fuzzy instances are described by Gaussian possibility distributions. Although any kind of distribution may be used, in this particular case, closed forms may be obtained for the update equations of the parameters in the FEM algorithm.

3.1. The FEM algorithm

The FEM algorithm proceeds iteratively with the complete log-likelihood $\log L_c(\Psi)$, alternating between two steps. At iteration q, the E-step consists in computing the expectation $Q(\Psi; \Psi^{(q)})$ of the complete log-likelihood with respect to the imprecisely observed data. The M-step is similar to that of the

classical EM algorithm, and requires the maximization of $Q(\Psi;\Psi^{(q)})$ with respect to Ψ. The FEM algorithm alternatively repeats the E- and M- steps until the relative increase of the log-likelihood becomes smaller than some threshold. In [62], it is shown that the FEM algorithm converges towards a local maximum of the observed log-likelihood. The proof is similar to that used in [2] for assessing the convergence of the EM algorithm.

3.2. E-step

At iteration q, the E-step of the FEM algorithm consists in computing the expectation $Q(\Psi;\Psi^{(q)})$ of the complete log-likelihood (7) with respect to the imprecisely observed data:

$$Q(\Psi;\Psi^{(q)}) = \mathbb{E}_{\Psi^{(q)}}[\log L(\Psi;\mathbf{y}_1,\ldots,\mathbf{y}_n)|\tilde{\mathbf{y}}_1,\ldots,\tilde{\mathbf{y}}_n], \tag{8}$$

where $\Psi^{(q)}$ is the current estimate of the parameter vector Ψ. We may remark that this expectation needs to be computed with respect to the imprecise observations of both the instances and class labels. For this purpose, we recall the following definitions of the conditional density $g_X(\cdot|\tilde{\mathbf{x}})$ and conditional expectation $\mathbb{E}[X|\tilde{\mathbf{x}}]$ of a random vector X with respect to a fuzzy event $\tilde{\mathbf{x}}$:

$$g_X(\mathbf{x}|\tilde{\mathbf{x}}) = \frac{\mu_{\tilde{\mathbf{x}}}(\mathbf{x})g_X(\mathbf{x})}{\mathbb{P}(\tilde{\mathbf{x}})}, \tag{9a}$$

$$\mathbb{E}[X|\tilde{\mathbf{x}}] = \int \mathbf{x}\; g_X(\mathbf{x}|\tilde{\mathbf{x}})\; d\mathbf{x}. \tag{9b}$$

Using Equations (9a)-(9b) in Equation (1b) gives

$$\begin{aligned} Q(\Psi,\Psi^{(q)}) = &\sum_{k=1}^{K}\log\pi_k\sum_{i=1}^{n}t_{ik}^{(q)} - \frac{1}{2}\sum_{k=1}^{K}\log|\Sigma_k|\sum_{i=1}^{n}t_{ik}^{(q)} - \frac{np}{2}\log(2\pi) \\ &- \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{K}t_{ik}^{(q)}\underbrace{\mathbb{E}_{\Psi^{(q)}}[(\mathbf{x}_i-\nu_k)^t\Sigma_k^{-1}(\mathbf{x}_i-\nu_k)|\tilde{\mathbf{x}}_i,\tilde{\mathbf{z}}_i,Z_{ik}=1]}_{\mathrm{EQF}^{(q)}}. \end{aligned} \tag{10}$$

Let us define

$$p_{ik}^{(q)} = \mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{x}}_i,\tilde{\mathbf{z}}_i|Z_{ik}=1) = \gamma_{ik}^{(q)}\delta_{ik}^{(q)}, \tag{11a}$$

$$p_{i}^{(q)} = \mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{x}}_i,\tilde{\mathbf{z}}_i) = \sum_{k=1}^{K}\pi_k^{(q)}p_{ik}^{(q)}, \tag{11b}$$

where $\gamma_{ik}^{(q)} = \mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{x}}_i|Z_{ik}=1)$ and $\delta_{ik}^{(q)} = \mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{z}}_i|\tilde{\mathbf{x}}_i,Z_{ik}=1)$. Then, the

quantity $t_{ik}^{(q)}$ in Equation (10) may be computed using Bayes' theorem:

$$\begin{aligned} t_{ik}^{(q)} &= \mathbb{E}_{\Psi^{(q)}}[Z_{ik}|\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i] = \mathbb{P}_{\Psi^{(q)}}(Z_{ik} = 1|\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i), && \text{(12a)} \\ &= \frac{\mathbb{P}_{\Psi^{(q)}}(Z_{ik} = 1)\mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i|Z_{ik} = 1)}{\mathbb{P}_{\Psi^{(q)}}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_i)}, && \text{(12b)} \\ &= \frac{\pi_k^{(q)}\gamma_{ik}^{(q)}\delta_{ik}^{(q)}}{p_i^{(q)}} = \frac{\pi_k^{(q)}\gamma_{ik}^{(q)}\delta_{ik}^{(q)}}{\sum_{l=1}^{K}\pi_l^{(q)}\gamma_{il}^{(q)}\delta_{il}^{(q)}}. && \text{(12c)} \end{aligned}$$

Equation (10) explicitly involves the expression of the possibility distributions describing our imprecise knowledge of the instances at hand. In this section, we assume these fuzzy numbers to be Gaussian (Equation (2)). Let us adopt the following notations:

$$\begin{aligned} \mathcal{A}(\tilde{\mathbf{x}}_i) &= \int \mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})d\mathbf{x}, && \text{(13a)} \\ \mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x}) &= \frac{\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x})}{\mathcal{A}(\tilde{\mathbf{x}}_i)}. && \text{(13b)} \end{aligned}$$

It is obvious that $\mu_{\tilde{\mathbf{x}}_i}^*$ is a pdf, provided that $\mathcal{A}(\tilde{\mathbf{x}}_i)$ is finite: it is positive, and its integral equals 1 by construction. Moreover, if $\mu_{\tilde{\mathbf{x}}_i}$ satisfies (2), then $\mu_{\tilde{\mathbf{x}}_i}^*$ is the pdf of a multivariate Gaussian, and

$$\mathcal{A}(\tilde{\mathbf{x}}_i) = (2\pi)^{p/2}|S_i|^{\frac{1}{2}} \int \mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x})d\mathbf{x} = (2\pi)^{p/2}|S_i|^{\frac{1}{2}}. \quad (14)$$

Based on these quantities, we may provide simple expressions for $\gamma_{ik}^{(q)}$ and $t_{ik}^{(q)}$, using Equation (4) and the key property that the product of two Gaussians is itself a Gaussian up to a normalization factor [66, page 200]:

$$\mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x})g_k(\mathbf{x}) = \kappa_{ik}g_{ik}(\mathbf{x}), \quad (15a)$$

where g_{ik} stands for the multivariate Gaussian pdf with mean vector $\mathbf{m}_{ik}$ and covariance matrix Σ_{ik}, and where these parameters, as well as the normalization term κ_{ik}, are defined by

$$\begin{aligned} \Sigma_{ik} &= \left(S_i^{-1} + \Sigma_k^{-1}\right)^{-1}, && \text{(15b)} \\ \mathbf{m}_{ik} &= \Sigma_{ik}\left(S_i^{-1}\mathbf{m}_i + \Sigma_k^{-1}\nu_k\right), && \text{(15c)} \\ \kappa_{ik} &= |2\pi(S_i + \Sigma_k)|^{-\frac{1}{2}}\exp(-\frac{1}{2}(\mathbf{m}_i - \nu_k)^t(S_i + \Sigma_k)^{-1}(\mathbf{m}_i - \nu_k)). && \text{(15d)} \end{aligned}$$

Let $g_k^{(q)}$ stand for the estimate of g_k obtained using the current fits $\nu_k^{(q)}$ and

$\Sigma_k^{(q)}$. Then,

$$\gamma_{ik}^{(q)} = \mathcal{A}\left(\tilde{\mathbf{x}}_i\right) \int \mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x}) g_k^{(q)}(\mathbf{x}) d\mathbf{x} = \mathcal{A}\left(\tilde{\mathbf{x}}_i\right) \kappa_{ik}^{(q)} \int g_{ik}^{(q)}(\mathbf{x}) d\mathbf{x} = \mathcal{A}\left(\tilde{\mathbf{x}}_i\right) \kappa_{ik}^{(q)}, \tag{16a}$$

whence:

$$t_{ik}^{(q)} = \frac{\pi_k^{(q)}\, \delta_{ik}^{(q)}\, \kappa_{ik}^{(q)}}{\sum_l \pi_l^{(q)}\, \delta_{il}^{(q)}\, \kappa_{il}^{(q)}}. \tag{16b}$$

In the following, the current fits of these quantities at iteration q will be indicated by the adequate superscript as before.

A closed form for the conditional expectation of the quadratic form, denoted as $\mathrm{EQF}^{(q)}$ in Equation (10), may also be computed. First, note that this expectation needs only be conditioned with respect to $\tilde{\mathbf{x}}_i$, since it is conditional to the event $Z_{ik} = 1$. Then, according to Equations (9a)-(9b),

$$\mathrm{EQF}^{(q)} = \frac{1}{\gamma_{ik}^{(q)}} \int (\mathbf{x}_i - \nu_k^{(q)})^t {\Sigma_k^{(q)}}^{-1} (\mathbf{x}_i - \nu_k^{(q)}) \mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}) g_k^{(q)}(\mathbf{x}) d\mathbf{x}, \tag{17a}$$

which we may rewrite using Equations (13a-16a) as

$$\mathrm{EQF}^{(q)} = \int (\mathbf{x} - \nu_k^{(q)})^t {\Sigma_k^{(q)}}^{-1} (\mathbf{x} - \nu_k^{(q)})\, g_{ik}^{(q)}(\mathbf{x}) d\mathbf{x}. \tag{17b}$$

Thus, $\mathrm{EQF}^{(q)}$ is the expectation of a quadratic function of a random vector X following a multivariate Gaussian distribution $g_{ik}^{(q)}$ with expectation $\mathbf{m}_{ik}^{(q)}$ and covariance matrix $\Sigma_{ik}^{(q)}$ (note that these values are computed by using the current fits for $\pi_k^{(q)}$, $\nu_k^{(q)}$,and $\Sigma_k^{(q)}$ in the appropriate equations). Therefore:

$$\mathrm{EQF}^{(q)} = (\mathbf{m}_{ik}^{(q)} - \nu_k)^t {\Sigma_k^{(q)}}^{-1} (\mathbf{m}_{ik}^{(q)} - \nu_k) + \mathrm{trace}\left(\Sigma_k^{-1} \Sigma_{ik}^{(q)}\right). \tag{18}$$

Using this new expression, we may give an explicit formulation for Equation (10):

$$\begin{aligned} Q(\Psi, \Psi^{(q)}) = & \sum_{k=1}^{K} \log \pi_k \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{1}{2} \sum_{k=1}^{K} \log |\Sigma_k| \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{np}{2} \log(2\pi) \\ & - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{K} t_{ik}^{(q)} \left((\mathbf{m}_{ik}^{(q)} - \nu_k)^t \Sigma_k^{-1} (\mathbf{m}_{ik}^{(q)} - \nu_k) + \mathrm{trace}(\Sigma_k^{-1} \Sigma_{ik}^{(q)}) \right). \end{aligned} \tag{19}$$

3.3. M-step

In the M-step, new estimates of the parameters are computed so as to maximize the expectation (19). The update equation of the parameters are obtained by setting the corresponding partial derivatives of (19) to zero. We detail hereafter the computation of these update equations.

Prior probabilities π_k

The first-order derivative of $Q(\Psi, \Psi^{(q)})$ with respect to π_k is:

$$\frac{\partial Q(\Psi, \Psi^{(q)})}{\partial \pi_k} = \frac{1}{\pi_k} \sum_{i=1}^{n} t_{ik}^{(q)}. \tag{20a}$$

Thus, taking into account the constraint that the proportions π_k sum to 1, it may easily be shown that

$$\pi_k^{(q+1)} = \frac{1}{n} \sum_{i=1}^{n} t_{ik}^{(q)}. \tag{20b}$$

Expectations ν_k

Taking the derivative of Equation (19) with respect to ν_k and setting it equal to zero, we get:

$$\sum_{i=1}^{n} \Sigma_k^{-1} t_{ik}^{(q)} (\mathbf{m}_{ik}^{(q)} - \nu_k) = 0, \tag{21a}$$

whence the update equation for the parameter ν_k:

$$\nu_k^{(q+1)} = \frac{\sum_{i=1}^{n} t_{ik}^{(q)} \mathbf{m}_{ik}^{(q)}}{\sum_{i=1}^{n} t_{ik}^{(q)}}. \tag{21b}$$

Covariance matrices Σ_k

Let us first recall some background notions of derivation with respect to elements of matrices. Let A be a matrix with entries a_{ij} $(i, j \in \{1, \ldots, p\})$, and let $f(A)$ be a function of A. For convenience, we will define the derivative of $f(A)$ with respect to A, written $\partial f(A)/\partial A$, as the matrix with $(i,j)^{\text{th}}$ entry $(\partial f(A)/\partial a_{ij})$. First, let us recall that $\mathbf{x}^t A \mathbf{x} = \text{trace}(AB)$, with $B = \mathbf{x}\mathbf{x}^t$, which makes it possible to re-write Equation (19):

$$\begin{aligned} Q(\Psi, \Psi^{(q)}) = \sum_{k=1}^{K} \log \pi_k \sum_{i=1}^{n} t_{ik}^{(q)} + \frac{1}{2} \sum_{k=1}^{K} \log |\Sigma_k^{-1}| \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{np}{2} \log(2\pi) \\ - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{K} t_{ik}^{(q)} \, \text{trace}\left(\Sigma_k^{-1} \, B_{ik}^{(q)}\right), \end{aligned} \tag{22}$$

where

$$B_{ik}^{(q)} = (\mathbf{m}_{ik}^{(q)} - \nu_k)(\mathbf{m}_{ik}^{(q)} - \nu_k)^t + \Sigma_{ik}^{(q)}. \tag{23}$$

Furthermore, we have:

$$\frac{\partial \, \text{trace}(AB)}{\partial A} = B, \tag{24a}$$

$$\frac{\partial \log |A|}{\partial A} = (A^{-1})^t. \tag{24b}$$

Taking the derivative of Equation (22) with respect to Σ_k^{-1} thus gives:

$$\frac{\partial Q(\Psi, \Psi^{(q)})}{\partial \Sigma_k^{-1}} = \frac{1}{2}\Sigma_k \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{1}{2}\sum_{i=1}^{n} t_{ik}^{(q)} B_{ik}^{(q)}. \tag{25a}$$

Setting this partial derivative to zero gives the following update equation for the covariance matrix Σ_k:

$$\Sigma_k^{(q+1)} = \frac{\sum_{i=1}^{n} t_{ik}^{(q)} B_{ik}^{(q)}}{\sum_{i=1}^{n} t_{ik}^{(q)}}. \tag{25b}$$

3.4. Computational complexity

We conclude this section with a brief discussion on the computational complexity of our approach. More particularly, we compare our algorithm to the classical EM algorithm for GMM estimation. Let us stress out that at each iteration, both algorithms estimate the same parameters. However, for this purpose, the FEM algorithm requires to perform some additional computations, that we list below.

First of all, taking into account the partial class information when computing the posterior probability estimates $t_{ik}^{(q)}$ necessits to perform $n \times K$ additional multiplications. The FEM algorithm also requires to compute the parameters $\mathbf{m}_{ik}^{(q)}$ and $\Sigma_{ik}^{(q)}$. The former necessits K inversions of a $p \times p$ matrix and $(n+1) \times K$ multiplications of a $p \times 1$ vector by a $p \times p$ matrix. The latter involves K inversions and K additions of a $p \times p$ matrix. Note that the quantities S_i^{-1} and $S_i^{-1} m_i$ may be computed outside the main loop of the optimization procedure of the FEM algorithm and stored.

The complexity of these additional operations is $\mathcal{O}(K \times p^2 \times (n+p))$, and can be further reduced if optimized algorithms are used for performing matrix inversions. This is comparable to the complexity of computing the $n \times K$ multivariate Gaussian densities. Overall, the computational complexities of the EM and FEM procedures thus have the same order of magnitude.

4. General case: the Monte-Carlo FEM algorithm

In this section, we address the case where the possibility distributions representing the partial knowledge of the instance values are not Gaussian. In this general case, computing $\text{EQF}^{(q)}$ (Equation (10)) may be intractable: thus, it may not be possible to obtain closed-forms for the update equations of the parameters. Therefore, we propose an efficient Monte-Carlo estimation technique to approximate this quantity, making it possible to proceed with any kind of weak knowledge of the instance values.

4.1. Monte-Carlo strategy

Monte-Carlo estimation consists in replacing the expectation of a function $\varphi(\mathbf{X})$ by an average of M terms $\varphi(\mathbf{x}^{(\ell)})$, $\ell = 1, \ldots, M$, where $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(M)}$ are

sampled according to the distribution of $\mathbf{X}$. In our case, a straightforward application of this principle would lead, at iteration q, to sample the data according to $g_k^{(q)}$, yielding the following estimate for $\mathrm{EQF}^{(q)}$:

$$\widehat{\mathrm{EQF}}^{(q)} = \frac{1}{\widehat{\gamma_{ik}}^{(q)}} \frac{1}{M} \sum_{\ell=1}^{M} (\mathbf{x}^{(\ell)} - \nu_k)^t \Sigma_k^{-1} (\mathbf{x}^{(\ell)} - \nu_k)\, \mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}^{(\ell)}), \quad (26a)$$

$$\widehat{\gamma_{ik}}^{(q)} = \frac{1}{M} \sum_{\ell=1}^{M} \mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}^{(\ell)}). \quad (26b)$$

However, we may remark the following points. First, the support of each possibility distribution $\mu_{\tilde{\mathbf{x}}_i}$ is likely to cover a very small part of the input space. Hence, the number M of terms generated according to $g_k^{(q)}$ should be large enough to ensure that a sufficient amount of these terms belong to this support; otherwise, the Monte-Carlo strategy would yield poor estimates of the parameters. This strategy thus becomes intractable in high dimension, due to the curse of dimensionality.[3] In addition, we may remark that, since the vector of parameters Ψ is updated at each iteration of the EM algorithm, so should be the sample.

Therefore, from a computational point of view, it seems much more efficient to sample the data according to the normalized possibility distributions $\mu^*_{\tilde{\mathbf{x}}_i}$ characterizing the fuzzy observations. Indeed, with such a strategy, data are sampled only once, before the parameter estimates are iteratively computed. The Monte-Carlo strategy thus defined is also generic: the sampling step being dissociated from the estimation step, any kind of possibility distribution may be used, provided that the adequate sampler is available. Finally, all the data points generated in this fashion obviously fall within the support of the densities g_k. This Monte-Carlo strategy, applied to the FEM algorithm, will hereafter be referred to as the MCFEM algorithm.

4.2. GMM estimation via the MCFEM algorithm

E-step

Our goal here is to estimate the integral in Equation (17a), which is generally intractable. For this purpose, let us rewrite Equation (10) using Equations (13a)

[3] Note that we may restrict the sampling process to regions of the space where the fuzzy instances lie, e.g., using truncated multivariate Gaussian distributions [67]. However, the use of such distributions with reasonable complexity is still an open problem. Besides, when the number of features p is arbitrary, computing a closed-form of the support of the possibility distributions (in order to truncate the Gaussians) is also intractable.

and (13b):

$$Q(\Psi, \Psi^{(q)}) = \sum_{k=1}^{K} \log \pi_k \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{1}{2} \sum_{k=1}^{K} \log |\Sigma_k| \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{np}{2} \log(2\pi)$$
$$- \frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{n} t_{ik}^{(q)} \frac{\mathcal{A}(\tilde{\mathbf{x}}_i)}{\gamma_{ik}^{(q)}} \int (\mathbf{x} - \nu_k)^t \Sigma_k^{-1} (\mathbf{x} - \nu_k) \, \mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x}) \, g_k^{(q)}(\mathbf{x}) \, d\mathbf{x}. \quad (27)$$

As stressed in Section 3, the normalized possibility distributions $\mu_{\tilde{\mathbf{x}}_i}^*$ satisfy the requirements of a probability distribution. Thus, they may be used to sample data, which makes it possible to estimate $\mathrm{EQF}^{(q)}$ by

$$\mathrm{EQF}^{(q)} \simeq \frac{1}{M} \sum_{\ell=1}^{M} (\mathbf{x}_i - \nu_k)^t {\Sigma_k^{(q)}}^{-1} (\mathbf{x}_i - \nu_k) \, g_k^{(q)}(\mathbf{x}_i^{(\ell)}), \quad (28)$$

where the M instances $\mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(\ell)}, \ldots, \mathbf{x}_i^{(M)}$ are sampled according to $\mu_{\tilde{\mathbf{x}}_i}^*$. This leads to the following Monte-Carlo approximation to Equation (10):

$$Q(\Psi, \Psi^{(q)}) \simeq \sum_{k=1}^{K} \log \pi_k \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{1}{2} \sum_{k=1}^{K} \log |\Sigma_k| \sum_{i=1}^{n} t_{ik}^{(q)} - \frac{np}{2} \log(2\pi)$$
$$- \frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{n} t_{ik}^{(q)} \frac{\mathcal{A}(\tilde{\mathbf{x}}_i)}{\gamma_{ik}^{(q)}} \frac{1}{M} \sum_{\ell=1}^{M} (\mathbf{x}_i^{(\ell)} - \nu_k)^t \Sigma_k^{-1} (\mathbf{x}_i^{(\ell)} - \nu_k) \, g_k^{(q)}(\mathbf{x}_i^{(\ell)}). \quad (29)$$

Note that this expression may be further simplified using the following approximation to γ_{ik}:

$$\frac{\gamma_{ik}}{\mathcal{A}(\tilde{\mathbf{x}}_i)} = \mathcal{A}(\tilde{\mathbf{x}}_i) \int \mu_{\tilde{\mathbf{x}}_i}^*(\mathbf{x}) \, g_k^{(q)}(\mathbf{x}) d\mathbf{x} \simeq \frac{1}{M} \sum_{f\ell=1}^{M} g_k(\mathbf{x}_i^{(\ell)}). \quad (30)$$

By dividing both the numerator and denominator in Equation (12c) by $\mathcal{A}(\tilde{\mathbf{x}}_i)$, and using Equation (30), we get the following update equation for t_{ik}:

$$t_{ik}^{(q)} \simeq \frac{\pi_k^{(q)} \, \delta_{ik} \sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})}{\sum_k \pi_k^{(q)} \, \delta_{ik} \sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})}. \quad (31)$$

M-step: update equations of the parameters

The M-step is the same as in the case of Gaussian fuzzy numbers. Setting the partial derivatives of Equation (29) with respect to the parameters to zero gives the same update equations of the parameters, defined by Equations (20b),

(21b) and (25b):

$$\pi_k^{(q+1)} = \frac{1}{n}\sum_{i=1}^{n} t_{ik}^{(q)}, \quad \nu_k^{(q+1)} = \frac{\sum_{i=1}^{n} t_{ik}^{(q)} \mathbf{m}_{ik}^{(q)}}{\sum_{i=1}^{n} t_{ik}^{(q)}}, \quad \Sigma_k^{(q+1)} = \frac{\sum_{i=1}^{n} t_{ik}^{(q)} B_{ik}^{(q)}}{\sum_{i=1}^{n} t_{ik}^{(q)}}, \tag{32a}$$

with $\mathbf{m}_{ik}^{(q)}$ and $B_{ik}^{(q)}$ being now defined by:

$$\mathbf{m}_{ik}^{(q)} = \frac{\sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})\, \mathbf{x}_i^{(\ell)}}{\sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})}, \tag{32b}$$

$$B_{ik}^{(q)} = \frac{\sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})\, (\mathbf{x}_i^{(\ell)} - \nu_k^{(q+1)})(\mathbf{x}_i^{(\ell)} - \nu_k^{(q+1)})^t}{\sum_{\ell=1}^{M} g_k^{(q)}(\mathbf{x}_i^{(\ell)})}. \tag{32c}$$

5. Bayesian priors

5.1. Motivations

As mentioned previously, the convergence of the FEM algorithm to a local maximum for the observed log-likelihood has been proved in [62]. Under some conditions on the initial values of the parameters, L is bounded from above. Since the observed log-likelihood increases at each iteration of the algorithm [2], the convergence is ensured. In practice, the algorithm is stopped when the relative increase of the observed likelihood is less than a given threshold ϵ. As noted in [68, page 85], in many practical applications, the EM algorithm converges to a local maximizer of the observed log-likelihood. However, it is underlined that this convergence towards nontrivial solutions relies on the compactness of the parameter space. This assumption may not hold in certain cases. For example, when computing ML estimators of the parameters in a mixture of Gaussians, setting the mean of a class to be one of the data points and letting its variance tend to zero will let $L(\Psi)$ tend to infinity. In practice, such a behaviour may be observed particularly when few data are available. It is interesting to note that in our case, this behaviour will more generally depend on the *informational content* of the data at hand. Indeed, the instances that are characterized by a high degree of imprecision (e.g., Gaussian fuzzy numbers with a large covariance matrix) will carry little information: the actual value for this instance may be located in a large area of the input space. Our experiments showed that a sample composed mostly of such imprecise instances is comparable to a sample with few precise instances.

To avoid degenerate solutions in the estimation process, we propose to integrate prior knowledge on the actual values of the parameters, using an adequate distribution $p(\Psi)$ [69, 70]. Then, the maximum *a posteriori* (MAP) estimate of the vector parameter Ψ may be computed so as to maximize the log (incomplete) posterior density

$$\log p(\Psi|\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n) = \log L(\Psi|\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n) + \log p(\Psi). \tag{33}$$

The log posterior density may be maximized using the FEM algorithm. The E-step consists in computing the expectation of Equation (33) with respect to the imprecisely observed data, using the current fits of the parameters. Since $p(\Psi)$ does not depend on $\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n$, we have

$$\mathbb{E}_{\Psi^{(q)}}\left[\log p(\Psi|\tilde{\mathbf{y}}_1, \ldots, \tilde{\mathbf{y}}_n)\right] = Q(\Psi; \Psi^{(q)}) + \log p(\Psi). \tag{34}$$

In the M-step, new estimates are computed by setting the partial derivatives of this expectation to zero.

5.2. Conjugate priors

The analytic formulation for the update equations of the parameter estimates is simpler if $p(\Psi)$ is a *conjugate prior* for the distribution of the model. Indeed, in this case, the derivatives of the log posterior density with respect to the parameters lead to simple update equations for the parameters.

In the case of a mixture of Gaussians, the normal-inverse-Wishart prior is commonly used to integrate background knowledge on the expectation vectors and on the covariance matrices [69]. More particularly, for the kth component, we have

$$\log p(\nu_k, \Sigma_k|\nu_{k0}, n_{k0}^{(1)}, \Sigma_{k0}, n_{k0}^{(2)}) = \log g(\nu_k|\nu_{k0}, \Sigma_k/n_{k0}^{(1)}) + \log\ h(\Sigma_k|\Sigma_{k0}, n_{k0}^{(2)}), \tag{35a}$$

where g is given by Equation (1b); and where h is defined, for any $p\times p$ symmetric positive-definite matrix, by:

$$h(\mathbf{\Sigma}; \Phi, \tau) = \frac{|\Phi|^{\frac{\tau}{2}}}{2^{\frac{\tau p}{2}}\Gamma_p(\tau/2)}|\Sigma|^{-\frac{\tau+p+1}{2}} \exp\left(-\frac{1}{2}\operatorname{trace}\Phi\Sigma^{-1}\right), \tag{35b}$$

with Φ and τ being the parameters of this density function, and Γ_p the multivariate Gamma function.

Thus, in our case, a Gaussian prior g with mean ν_{k0} and covariance matrix $\Sigma_k/n_{k0}^{(1)}$ is put on the expectation vector m_k, and an inverse-Wishart prior h with scale matrix Σ_{k0} and parameter $n_{k0}^{(2)}$ is put on the covariance matrix Σ_k. Here, $n_{k0}^{(1)}$ and $n_{k0}^{(2)}$ are the numbers of degrees of freedom associated with the Gaussian and inverse-Wishart distributions, respectively. Note that Bayesian regularization for multivariate GMM is discussed in [69, 70], in which many details on the calculation of the parameter estimates may be found.

5.3. Update equations

The log priors write

$$\begin{aligned}\log g(\nu_k|\nu_{k0}, \Sigma_k/n_{k0}^{(1)}) = -\frac{p}{2}\log(2\,\pi) + \frac{p}{2}\log n_{k0}^{(1)} + \frac{1}{2}\log|\Sigma_k^{-1}| \\ - \frac{n_{k0}^{(1)}}{2}\left(\nu_k - \nu_{k0}\right)^t \Sigma_k^{-1}\left(\nu_k - \nu_{k0}\right),\end{aligned} \tag{36a}$$

$$\log h(\Sigma_k|\Sigma_{k0}, n_{k0}^{(2)}) = \frac{n_{k0}^{(2)}}{2}\log|\Sigma_{k0}| + \frac{n_{k0}^{(2)}+p+1}{2}\log|\Sigma_k^{-1}| - \frac{1}{2}\operatorname{trace}\left(\Sigma_{k0}\Sigma_k^{-1}\right) - \frac{n_{k0}^{(2)}p}{2}\log(2) - \log\Gamma_p(\frac{n_{k0}^{(2)}}{2}). \quad (36b)$$

Expectations m_k

Since the inverse-Wishart distribution does not depend on ν_k, the partial derivative of the overall log prior with respect to ν_k is

$$\frac{\partial \log g(\nu_k|\nu_{k0}, \Sigma_k/n_{k0}^{(1)})}{\partial \nu_k} = \frac{n_{k0}^{(1)}}{2}\Sigma_k^{-1}(\nu_k - \nu_{k0}). \quad (37a)$$

Thus, adding this term to the partial derivative of $Q(\Psi;\Psi^{(q)})$ with respect to ν_k gives the following update equation for the expectation vector when a Gaussian prior is employed:

$$\nu_k^{(q+1)} = \frac{\sum_{i=1}^n t_{ik}^{(q)}\mathbf{m}_{ik}^{(q)} + n_{k0}^{(1)}\nu_{k0}}{\sum_{i=1}^n t_{ik}^{(q)} + n_{k0}^{(1)}}, \quad (37b)$$

with $\mathbf{m}_{ik}^{(q)}$ being defined either by Equation (15c) in the case of Gaussian fuzzy numbers, or by Equation (32b) when using the MCFEM algorithm.

Covariance matrices Σ_k

The partial derivative of the overall log prior with respect to Σ_k^{-1} writes as

$$\frac{\partial \log p(\nu_k, \Sigma_k|\nu_{k0}, n_{k0}^{(1)}, \Sigma_{k0}, n_{k0}^{(2)})}{\partial \Sigma_k^{-1}} = \frac{1}{2}\Sigma_k - \frac{n_{k0}^{(1)}}{2}(\nu_k - \nu_{k0})(\nu_k - \nu_{k0})^t + \frac{n_{k0}^{(2)}+p+1}{2}\Sigma_k - \frac{1}{2}\Sigma_{k0}. \quad (38a)$$

This leads to the following update equation for the covariance matrix Σ_k, when the likelihood is regularized using a normal-inverse-Wishart prior:

$$\Sigma_k^{(q+1)} = \frac{\sum_{i=1}^n t_{ik}^{(q)}B_{ik}^{(q)} + n_{k0}^{(1)}\left(\nu_k^{(q+1)} - \nu_{k0}\right)\left(\nu_k^{(q+1)} - \nu_{k0}\right)^t + \Sigma_{k0}}{\sum_{i=1}^n t_{ik}^{(q)} + n_{k0}^{(2)} + p + 2}, \quad (38b)$$

where $B_{ik}^{(q)}$ is defined by Equation (23) when the data at hand are Gaussian fuzzy numbers, or by Equation (32c) for the MCFEM algorithm.

6. Experiments

In this section, we first present results obtained by clustering synthetic data, in order to assess the robustness of our algorithm to noise and to the lack of information. Then, we present experiments realized on classical real datasets, in which quadratic discriminant analysis is used in presence of corrupted data.

Table 1: Parameters of the components of the Gaussian mixture.

	comp. 1	comp. 2	comp. 3
π_k	0.4	0.25	0.35
ν_k	$(2,1)^t$	$(0,-2)^t$	$(-2,1)^t$
Σ_k	$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1.25 \end{pmatrix}$	$\begin{pmatrix} 1.5 & 0.75 \\ 0.75 & 1.5 \end{pmatrix}$

6.1. Synthetic data

Noisy instances

First, we ran an experiment using synthetic two-dimensional data in order to evaluate the performance of our algorithms in presence of data with corrupted instance values. More precisely, we generated the data so that the level of imprecision on the instances depends on the magnitude of the instance values. First, we drew a sample of $n = 1000$ realizations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of a Gaussian mixture of $K = 3$ components with the parameters given in Table 1. Then, we introduced noise in the data as follows. For each instance $\mathbf{x}_i$, we randomly generated a covariance matrix W_i according to an inverse-Wishart distribution with 4 degrees of freedom, and we set then $S_i = \lambda W_i$. The scalar $\lambda \in \{2, 4, \ldots, 20\}$ makes it possible to set the degree of imprecision introduced in the data. For each instance $\mathbf{x}_i$, a noisy value $\mathbf{m}_i$ was then generated from a multivariate Gaussian with mean $\mathbf{x}_i$ and covariance matrix S_i.

We thus created a set of Gaussian fuzzy numbers with mean value $\mathbf{m}_i$ and covariance matrix S_i. We also defined a set of trapezoidal fuzzy numbers, such that the support and the core of the ith element are the squares containing respectively 95% and 50% of the Gaussian distribution with mean $\mathbf{w}_i$ and covariance matrix S_i. The parameters (a_{ij}, d_{ij}) and (b_{ij}, c_{ij}), that respectively define the support and the core along the jth dimension ($j \in \{1, 2\}$), are thus

$$\begin{array}{rclcrcl} a_{ij} & = & \tilde{w}_{ij} + \tilde{\sigma}_i \, \Phi^{-1}(\frac{1-\sqrt{0.95}}{2}), & & d_{ij} & = & \tilde{w}_{ij} + \tilde{\sigma}_i \, \Phi^{-1}(\frac{1+\sqrt{0.95}}{2}), \\ b_{ij} & = & \tilde{w}_{ij} + \tilde{\sigma}_i \, \Phi^{-1}(\frac{1-\sqrt{0.5}}{2}), & & c_{ij} & = & \tilde{w}_{ij} + \tilde{\sigma}_i \, \Phi^{-1}(\frac{1+\sqrt{0.5}}{2}); \end{array} \tag{39}$$

where $\tilde{w}_{ij}$ denotes the jth component of $\mathbf{w}_i$, and where Φ stands for the cumulative distribution function of the centered and scaled Gaussian distribution.

We estimated the parameters of a GMM with three components as follows. The original (precise, uncorrupted) data and the precise corrupted ones were processed using the EM algorithm. The Gaussian fuzzy instances were clustered using the Gaussian FEM algorithm. The MCFEM algorithm was used in the case of trapezoidal fuzzy instances: for this purpose, $M = 200$ instances were generated according to each trapezoidal distribution using the procedure described in AppendixA. For each value of λ, we ran each algorithm five times. The starting values were computed using 10% of randomly selected instances in each class (the same for all the algorithms). The labels of these instances were considered as known throughout the estimation process. We stopped iterating

when the relative increase of the log-likelihood becomes small, i.e.

$$\frac{\log L(\Psi^{(q)}) - \log L(\Psi^{(q-1)})}{|\log L(\Psi^{(q-1)})|} \leq 10^{-5}. \tag{40}$$

Then, for each algorithm, we kept the best solution over the five trials in terms of observed log-likelihood. Each instance $\mathbf{x}_i$ was classified so as to maximize the estimated posterior probabilities $t_{ik}^{(q)}$. The adjusted Rand index (ARI) [71] was then computed in order to assess the accuracy of the partition thus obtained. We recall here that the ARI is a corrected-for-chance version of the Rand index, a popular measure frequently employed to compare two partitions. This whole procedure (data generation and corruption, model estimation, and assessment of the accuracy of the partition thus obtained) was repeated 50 times, so that averages of the ARI may be computed.

Figure 1 displays the evolution of the average ARI as a function of the degree of noise λ introduced in the instances. Without surprise, the partitions computed from noisy data exhibit a poor accuracy when the amount of noise increases. However, taking into account the uncertainty on the attribute values improves the robustness of the model, and the partition thus obtained is closer to the actual partition of the data.

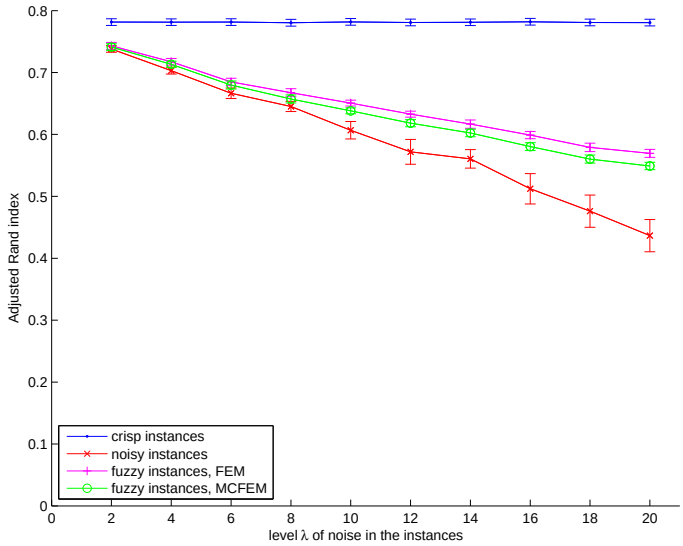

Figure 1: Evolution of the average adjusted Rand index as a function of the degree of noise λ in the data. The values of the average Rand index and the 95% confidence intervals are computed over 50 experiments.

Figure 2 presents the results obtained for one experiment with $\lambda = 20$. The instances are printed with a color depending on the component from which they were generated. The mean vectors thus obtained are displayed, along with the

covariance matrices represented by the 95% confidence ellipsoids. We may see

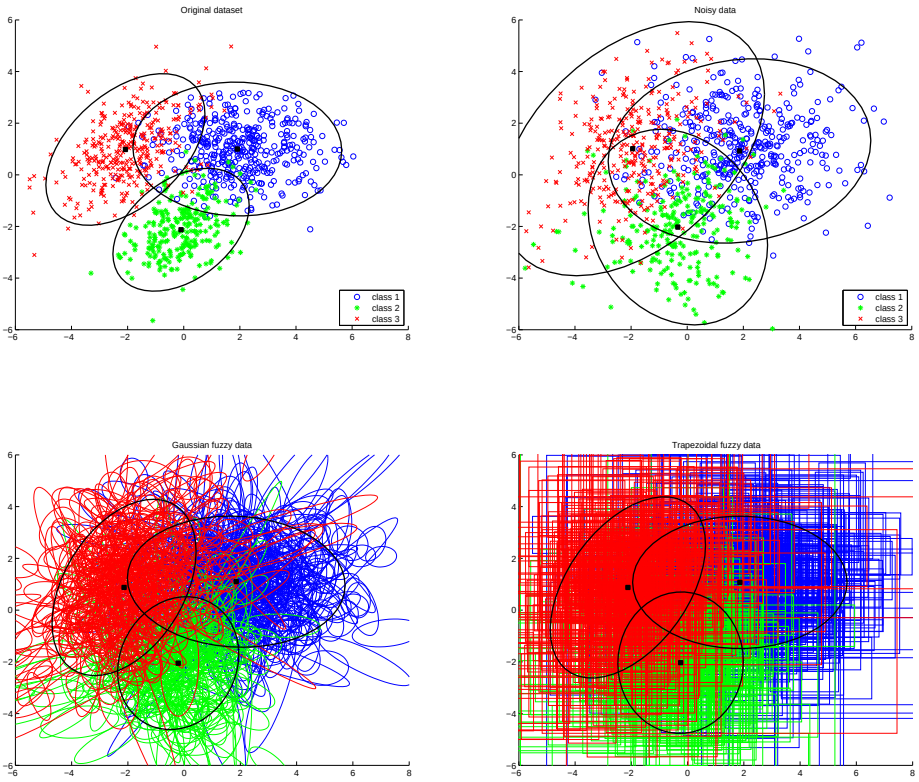

Figure 2: Original data (top left), noisy data (top right), Gaussian (bottom left) and trapezoidal (bottom right) fuzzy data, along with the parameters estimated using the EM algorithm (top) and the fuzzy EM algorithm (bottom).

that the parameters (and in particular the covariance matrices) computed using the crisp EM algorithm are sensitive to the presence of noisy data. However, taking into account the attribute uncertainty using the FEM algorithm yields more robust parameter estimates.

Corrupted labels

We realized an experiment on data with corrupted labels. We generated data as before, with $\lambda = 2$. Then, we introduced noise in the labeling: the actual labels were replaced by a random label with probability η. The probability $\eta \in \{0.1, 0.2, \ldots, 1\}$ thus allowed us to control the level of noise introduced in the labels. Then, we compared the results obtained with two kinds of labeling:

- crisp labeling, which corresponds to a semi-supervised approach, where an instance $\mathbf{x}_i$ is associated with a vector $\mathbf{z}_i$ indicating its (possibly corrupted) associated label;

- imprecise labeling, where the partial knowledge of the actual class of an instance is represented using a possibility distribution.

In this last case, we followed [55, 63] and gave the observed label ω_k a degree of possibility $\delta_{ik} = 1$, and the other plausible labels $\omega_{k'}$, $k' \neq k$ a degree of possibility $pl_{ik'} = \eta$.

As previously, we computed the average ARI over 50 randomly generated datasets. The data without label information and the data with noisy labels were clustered via the EM algorithm. The FEM and MCFEM algorithm were used to process the Gaussian and the trapezoidal fuzzy instances with plausibilistic labels, respectively. Figure 3 displays the ARI as a function of the degree of noise η introduced in the labels. Estimating the parameters of the model us-

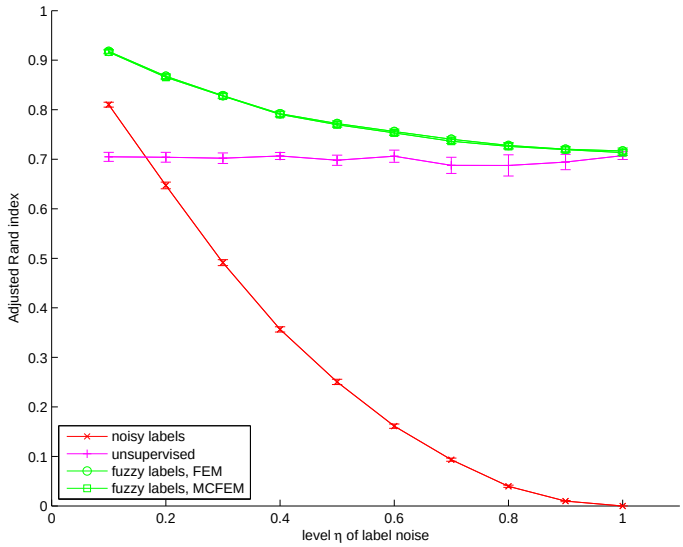

Figure 3: Evolution of the adjusted Rand index as a function of the degree of noise η in the labels. The values of the average Rand index and the 95% confidence intervals are computed over 50 experiments.

ing a semi-supervised labeling gives good results when the amount η of noise is low. As η increases, the accuracy of the obtained partition decreases. The results obtained using imprecise labels via the FEM and MCFEM algorithms are almost identical, and always better than those in the semi-supervised case. Remarkably, their poorest performance is achieved for $\eta = 1$: then, all the components being equally plausible, their accuracy is similar to that of the unsupervised EM algorithm.

Robustness to lack of data

We produce here the results of an experiment which purpose is to demonstrate how Bayesian priors may ameliorate the estimation of the parameters, as

explained in Section 5. First of all, it should be pointed out that the amount of information contained in a fuzzy sample is directly related to the degree of imprecision of the data. Figure 4 displays the GMM estimated using the FEM algorithm on Gaussian fuzzy data with two levels of imprecision. The data were generated as before, with $\lambda = 2$ in one case and $\lambda = 20$ in the other case (the instances having the same mean values $\mathbf{m}_i$). It may be observed that a high amount of imprecision results in a decrease of the trace of the estimated covariance matrices.

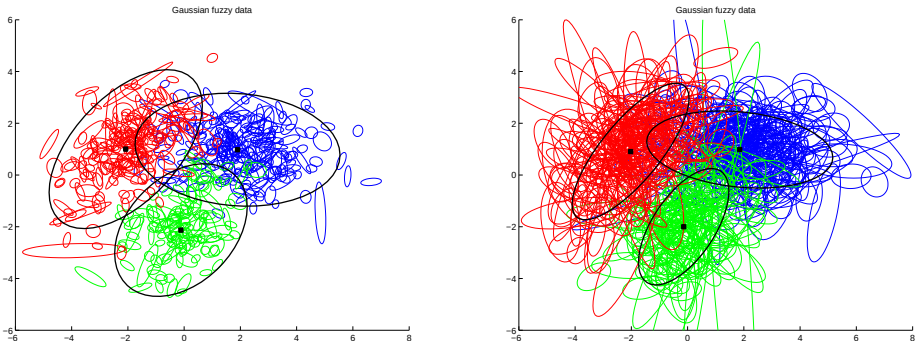

Figure 4: Parameters estimated using the FEM algorithm on Gaussian fuzzy data, for a low (left) and a high (right) amount of imprecision λ in the data. The trace of the estimated covariance matrices decreases when the level of imprecision increases.

To show the interest of using Bayesian priors, we generated Gaussian fuzzy data as previously, except that the mean $\mathbf{m}_i$ of each Gaussian fuzzy number $\tilde{\mathbf{x}}_i$ was set to $\mathbf{x}_i$: the data at hand are thus imprecise, but not noisy. Then, we clustered the imprecise data at hand using three different versions of the Gaussian FEM algorithm:

- without setting any prior on the model parameters;
- using the actual parameter values as prior: in this case, the degrees of freedom $n_{k0}^{(1)}$ and $n_{k0}^{(2)}$ were set to $\widehat{n}_{k0}^{(1)} = 1e6$ and $\widehat{n}_{k0}^{(2)} = 1e-6-2$, respectively;
- using estimated priors obtained as follows: for each class ω_k, we randomly selected a number $\widehat{n_k}$ of precise instances, corresponding to 25% of the instances in the class, which were used to compute estimates $\widehat{\nu_k}$ and $\widehat{\Sigma_k}$ of the class parameters. Then, we set a Gaussian-inverse-Wishart prior on the kth component of the GMM, with mean $\nu_{k0} = \widehat{\nu_k}$, covariance matrix $\Sigma_{k0} = n_{k0}^{(2)} \widehat{\Sigma_k}$, and degrees of freedom $n_{k0}^{(1)} = \widehat{n_k}$ and $n_{k0}^{(2)} = \widehat{n_k} - 2$, respectively.

Figure 5 displays the evolution of the average ARI, computed over 100 experiments, as a function of the degree of imprecision of the instances. We may

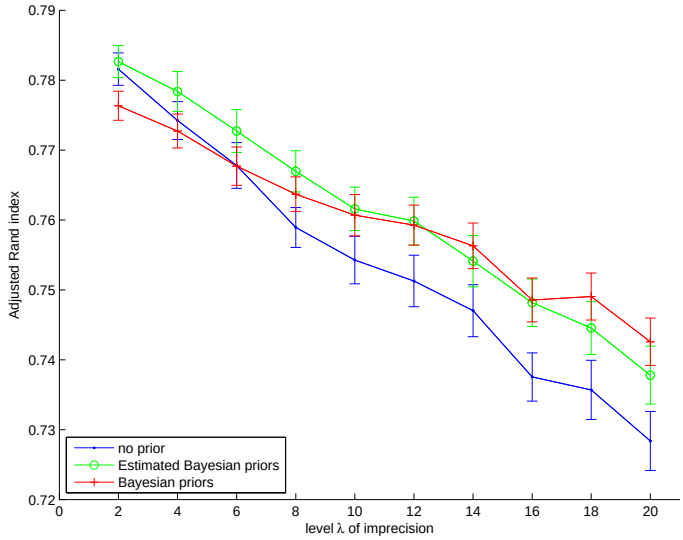

Figure 5: Evolution of the adjusted Rand index as a function of the level of imprecision λ in the instances. The values of the average Rand index and the 95% confidence intervals are computed over 50 experiments.

notice that the ARI decreases while the level of imprecision λ increases, which shows that the accuracy of the model highly depends on the informational content of the data at hand. However, using Bayesian priors makes it possible to guide the algorithm towards more robust parameter estimates.

6.2. Real data

In this Section, we report the results obtained by applying GMM estimation to classification problems. We considered seven real datasets described in Table 2. For each dataset, we randomly selected 66% of the data for learning

Table 2: Characteristics of the real datasets processed.

dataset	number of instances	number of classes	number of variables
Iris	150	3	4
Letter	18200	26	16
Pageblocks	5473	5	10
Pendigits	10992	10	16
Satimage	6435	6	36
Waveform	5000	3	21

the parameters of a GMM. Note that in some datasets, one or several variables

may be constant; such uninformative variables were suppressed to avoid numerical problems. The remaining variables were then centered and scaled. We introduced noise in the instance values as explained before, using a parameter value $\lambda = 1$ (note that the labels were left unchanged). Then, the parameters of each class were estimated from the noisy training instances, and from the fuzzy instances using the FEM algorithm. The remaining test instances were classified using quadratic discriminant analysis (QDA): no additional assumption was made on the covariance matrices of the classes. Conditional densities and then posterior probabilities were computed for each test instance using the parameters estimated, using Equation (16b) in the case of fuzzy instances.

Since the imprecision of the instances may not be known for test labels, we performed two series of experiments. A first estimation of the posterior probabilities was performed using Equation (16b) using the actual covariance matrices generated to corrupt the data. Besides, we also estimated these covariance matrices, by averaging the ones used for the fuzzy training data.

This whole procedure (random selection of training and test sets, introduction of noise, estimation of the parameters, and classification of the test data) was repeated 25 times. Figures 6 to 11 present the classification accuracy obtained. The results obtained by classifying the uncorrupted and noisy test instances with the quadratic classifier trained from noisy data (boxplots 1 and 2, respectively) are referred to as "crisp QDA"; while "fuzzy QDA" refers to processing fuzzy test instances with a classifier estimated from fuzzy training data (boxplot 3 when the actual uncertainty of the test instances is used, and boxplot 4 for the estimated uncertainty).

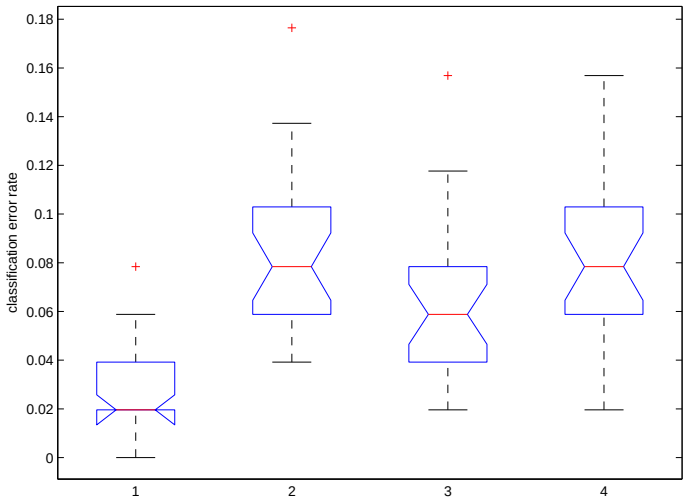

Figure 6: Classification accuracy, Iris dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

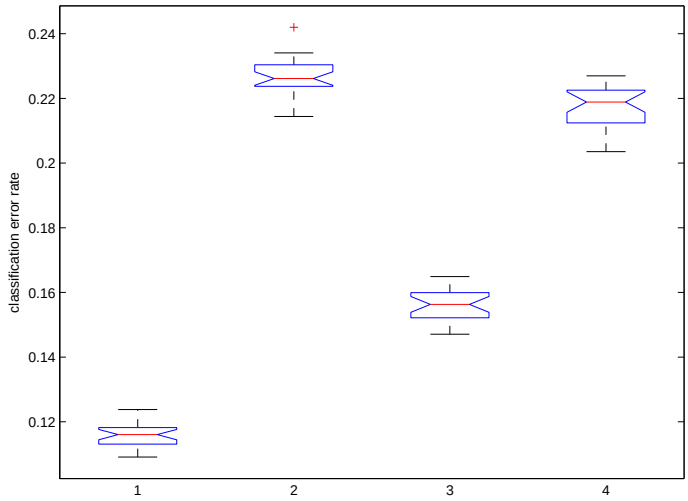

Figure 7: Classification accuracy, Letter dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

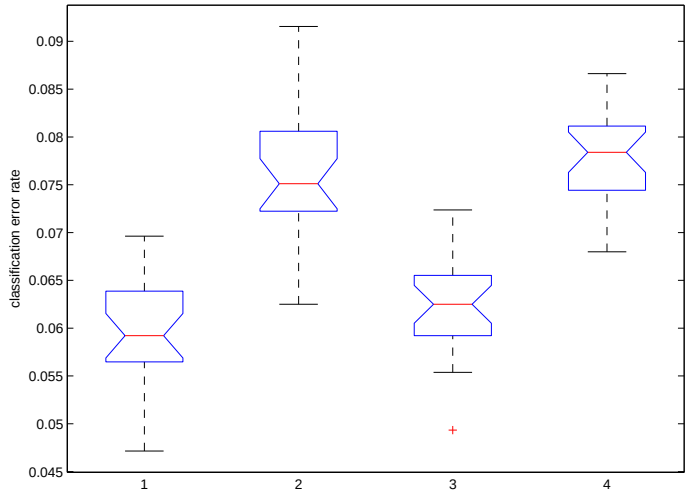

Figure 8: Classification accuracy, Pageblocks dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

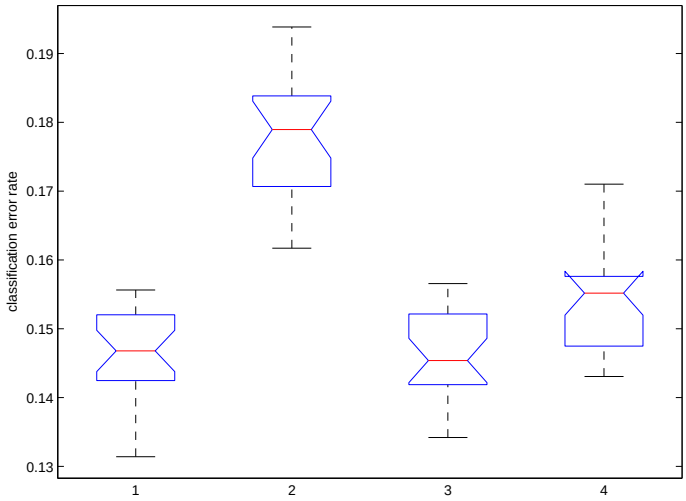

Figure 9: Classification accuracy, Satimage dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

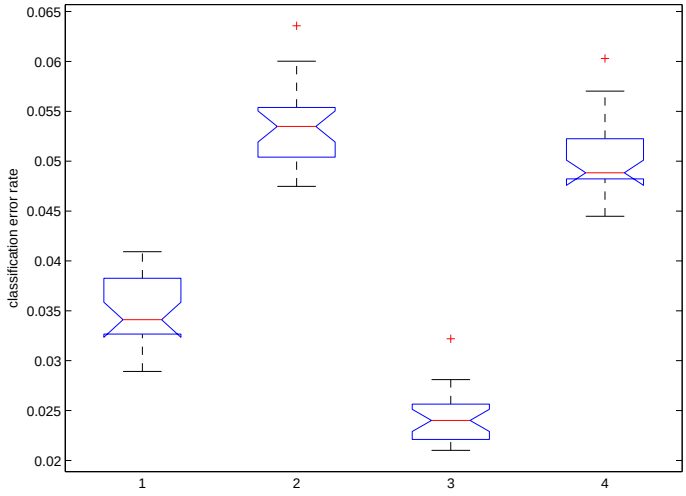

Figure 10: Classification accuracy, Pendigits dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

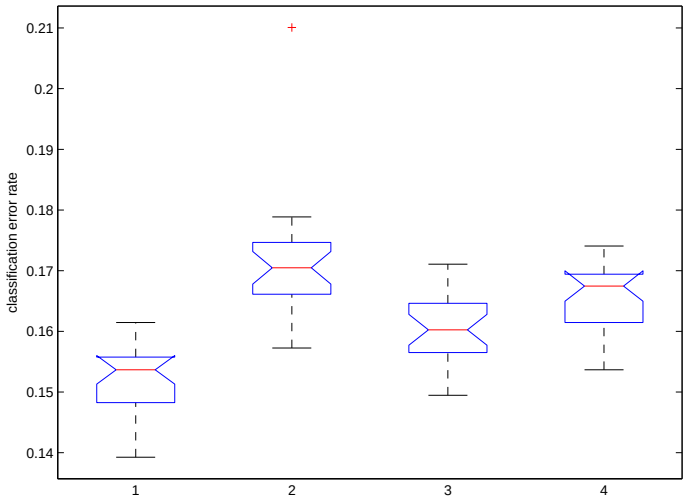

Figure 11: Classification accuracy, Waveform dataset. Boxplot 1: crisp QDA, uncorrupted test data; boxplot 2: crisp QDA, noisy test data; boxplot 3: fuzzy QDA, test data with actual uncertainty; boxplot 4: fuzzy QDA, test data with estimated uncertainty.

The results clearly assess the interest of our approach for estimating the parameters of the model. Indeed, taking into account the uncertainty on the training and test data makes it possible to achieve better results than ignoring this information. In addition, the results obtained with the fuzzy GMM with estimated covariance matrices is significantly better than the results obtained via the crisp GMM in three cases over six. This clearly advocates taking into account the uncertainty when processing data pervaded with noise.

7. Conclusion

In this paper, we addressed the problem of estimating the parameters of a GMM from imprecisely observed data. Our approach is based on an extention of the EM algorithm for fuzzy data proposed by Denœux [62, 63]. Given a sample of fuzzy numbers, the likelihood of a mixture of Gaussians is computed using Zadeh's definition of the probability of a fuzzy event. Then, the estimates maximizing this likelihood is estimated using an iterative procedure. At each iteration, the expectation of the log-likelihood with respect to the fuzzy sample is first computed. In a second step, the parameters of the model are updated so as to maximize this expectation.

We presented two main results. First, we showed that when the data are represented by Gaussian fuzzy numbers, closed-forms of the parameter estimates may be computed. Furthermore, we proposed a Monte-Carlo approach to estimate the parameters in the general case. This Monte-Carlo approach is generic, since it is suitable to any kind of possibility distribution, provided that

an adequate sampler be available. It is also computationally efficient, since the sampling step is made only once, and outside of the estimation procedure.

We conducted experiments on synthetic and real data. The results show that our algorithm makes it possible to estimate accurately the distribution of imprecisely known data. In particular, taking into account all the available information on the data uncertainty makes it possible to compute robust estimates of the parameters in presence of noisy attributes and corrupted labels. When applied to the classification of noisy data via quadratic discriminant analysis, taking into account the uncertainty on the instance values makes it possible to improve the classification accuracy, even when the uncertainty on the test data is estimated from the training set. In conclusion, the results clearly shows the interest of our approach for performing clustering or classification in the presence of noisy or uncertain information.

Acknowledgements

This work was carried out in the framework of the Labex MS2T, which was funded by the French Government, through the program "Investments for the future" by the National Agency for Research (reference ANR-11-IDEX-0004-02).

Bibliography

[1] R. A. Redner, H. F. Walker, Mixture densities, maximum likelihood and the em algorithm, SIAM Review 26 (2) (1984) 195–239.

[2] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society B 39 (1977) 1–38.

[3] H. Hamdan, G. Govaert, CEM algorithm for imprecise data. application to flaw diagnosis using acoustic emission, in: Proceedings of the IEEE International conference on Systems, Man and Cybernetics, Vol. 5, The Hague, Netherlands, 2004, pp. 4774–4779.

[4] G. M. Jacquez, Disease cluster statistics for imprecise space-time locations, Statistics in Medicine 15 (1996) 873–885.

[5] G. Mauris, Expression of measurement uncertainty in a very limited knowledge context: a possibility theory-based approach, IEEE Transactions on Instrumentation and Measurements 56 (3) (2007) 731–735.

[6] J. Gebhardt, M. A. Gil, R. Kruse, Fuzzy set-theoretic methods in statistics, Vol. 1 of Handbook of Fuzzy Sets, Kluwer Academic Publishers, 1998, Ch. 10, pp. 311–347.

[7] M. A. Gil, M. Lopez-Diaz, D. A. Ralescu, Overview on the development of fuzzy random variables, Fuzzy Sets and Systems 157 (19) (2006) 2546–2557.

[8] R. Kruse, K. D. Meyer, Statistics with vague data, Kluwer, 1987.

[9] R. Viertl, Univariate statistical analysis with fuzzy data, Computational Statistics and Data Analysis 51 (1) (2006) 133–147.

[10] D. Dubois, Ontic vs. epistemic fuzzy sets in modeling and data processing tasks, in: K. Madani, J. Kacprzyk, J. Filipe (Eds.), Proceedings of the 2011 International Conference on Neural Computation: Theory and Applications (NCTA'2011), Paris, France, 2011.

[11] I. Couso, D. Dubois, Statistical reasoning with set-valued information: Ontic vs. epistemic views, International Journal of Approximate Reasoning In Press, DOI: 10.1016/j.ijar.2013.07.002.

[12] B. B. Chaudhuri, P. R. Bhowmik, An approach of clustering data with noisy or imprecise feature measurement, Pattern Recognition Letters 19 (1998) 1307–1313.

[13] A. Irpino, V. Tontodonato, Clustering reduced interval data using Hausdorff distance, Computational Statistics 21 (2006) 241–288.

[14] D. Habich, P. B. Volk, W. L. R. Dittman, C. Utzny, Error-aware density-based clustering of imprecise measurement values, in: Proceedings of the 7th International Conference on Data Mining Workshops (ICDM'07), Omaha, Nevada, USA, 2007, pp. 471–476.

[15] H. Hamdan, G. Govaert, Mixture model clustering of uncertain data, in: Proceedings of the IEEE International conference on Fuzzy Systems, Reno, Nevada, USA, 2005, pp. 879–884.

[16] G. Lee, C. Scott, EM algorithms for multivariate gaussian mixture models with truncated and censored data, Computational Statistics and Data Analysis 56 (9) (2012) 2816–2829.

[17] H.-P. Kriegel, M. Pfeifle, Hierarchical density-based clustering of uncertain data, in: Proceedings of the 5th International Conference on Data Mining (ICDM'05), Houston, Texas, USA, 2005, pp. 689–692.

[18] H.-P. Kriegel, M. Pfeifle, Density-based clustering of uncertain data, in: Proceedings of the 11th International Conference on Knowledge Discovery in Data Mining (ACM SIGKDD'11), Houston, Texas, USA, 2005, pp. 672–677.

[19] M. Chau, R. Cheng, B. Kao, J. Ng, Uncertain data mining: An example in clustering location data, in: W. K. Ng, Kitsuregawa, J. Li (Eds.), Proceedings of the 10th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2006), Vol. LNAI-3918 of Lecture Notes in Artificial Intelligence, Springer-Verlag, 2006, pp. 199–204.

[20] W. K. Ngai, B. Kao, C. K. Chui, M. Chau, R. Cheng, K. Y. Yip, Efficient clustering of uncertain data, in: Proceedings of the 6th International Conference on Data Mining (ICDM'06), Hong Kong, China, 2006, pp. 436–445.

[21] B. Kao, S. D. Lee, F. K. F. Lee, D. W. Cheung, W.-S. Ho, Clustering uncertain data using clustering uncertain data using Voronoi diagrams and R-tree index, IEEE Transactions on Knowledge and Data Engineering 22 (9) (2010) 1219–1233.

[22] G. Cormode, A. McGregor, Approximation algorithms for clustering uncertain data, in: Proceedings of the 27th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'08), Vancouver, British Columbia, Canada, 2008, pp. 191–200.

[23] C. C. Aggarwal, On density based transforms for uncertain data mining, in: Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE'07), Istanbul, Turkey, 2007, pp. 866–875.

[24] S. Günnemann, H. Kremer, T. Seidl, Subspace clustering for uncertain data, in: Proceedings of the SIAM International Conference on Data Mining (SDM 2010), Columbus, Ohio, USA, 2010, pp. 385–396.

[25] W. Zhang, X. Lin, J. Pei, Y. Zhang, Managing uncertain data: Probabilistic approaches, in: Proceedings of the 9th International Conference on Web-Age Information Management (WAIM'08), Zhangjiajie, China, 2008, pp. 405–412.

[26] C. C. Aggarwal, Managing and Mining Uncertain Data, no. 35 in Advances in Database Systems, Springer, 2009.

[27] C. C. Aggarwal, P. S. Yu, A survey of uncertain data algorithms and applications, IEEE Transactions on Knowledge and Data Engineering 21 (5) (2009) 609–623.

[28] M. Sato, Y. Sato, Fuzzy clustering model for fuzzy data, in: Proceedings of the IEEE International conference on Fuzzy Systems, Yokohama, Japan, 1995, pp. 2123–2128.

[29] R. J. Hathaway, J. C. Bezdek, W. Pedrycz, A parametric model for fusing heterogeneous fuzzy data, IEEE Transactions on Fuzzy Systems 4 (3) (1996) 1277–1282.

[30] W. Pedrycz, J. C. Bezdek, R. J. Hathaway, G. W. Rogers, Two nonparametric models for fusing heterogeneous fuzzy data, IEEE Transactions on Fuzzy Systems 6 (3) (1998) 411–425.

[31] O. Takata, S. Miyamoto, K. Umayahara, Clustering of data with uncertainties using hausdorff distance, in: Proceedings of the IEEE International conference on Intelligence Processing Systems, 1998, pp. 67–71.

[32] O. Takata, S. Miyamoto, K. Umayahara, Fuzzy clustering of data with uncertainties using minimum and maximum distances based on l1 metric, in: Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vancouver, British Columbia, Canada, 2001, pp. 2511–2516.

[33] M.-S. Yang, H.-H. Liu, Fuzzy clustering procedures for conical fuzzy vector data, Fuzzy Sets and Systems 106 (2) (1999) 189–200.

[34] M.-S. Yang, P.-Y. Hwang, D.-H. Chen, Fuzzy clustering algorithms for mixed feature variables, Fuzzy Sets and Systems 141 (2) (2004) 301–317.

[35] R. Coppi, P. d'Urso, Three-way fuzzy clustering models for lr fuzzy time trajectories, Computational Statistics and Data Analysis 43 (2003) 149–177.

[36] P. d'Urso, P. Giordani, A weighted fuzzy c-means clustering model for fuzzy data, Computational Statistics and Data Analysis 50 (6) (2006) 1496–1523.

[37] W.-L. Hung, M.-S. Yang, Fuzzy clustering on lr-type fuzzy numbers with an application in Taiwanese tea evaluation fuzzy clustering on lr-type fuzzy numbers with an application in taiwanese tea evaluation, Fuzzy Sets and Systems 150 (3) (2005) 561–577.

[38] M. H. F. Zarandi, Z. S. Razaee, A fuzzy clustering model for fuzzy data with outliers, International Journal of Fuzzy System Applications 1 (2) (2011) 29–42.

[39] R. Coppi, P. d'Urso, P. Giordani, Fuzzy and possibilistic clustering for fuzzy data, Computational Statistics and Data Analysis 56 (2012) 915–927.

[40] S. B. Hariz, Z. Elouedi, K. Mellouli, Clustering approach using belief function theory, in: J. Euzenat, J. Domingue (Eds.), Artificial Intelligence: Methodology, Systems, and Applications, Vol. LNAI-4183 of Lecture Notes in Artificial Intelligence, Springer, 2006, pp. 162–171.

[41] B. Quost, T. Denœux, Clustering fuzzy data using the fuzzy em algorithm, in: A. Deshpande, A. Hunter (Eds.), Proceedings of the 4th International Conference on Scalable Uncertainty Management (SUM'2010), Vol. LNAI-6379 of Lecture Notes in Artificial Intelligence, Springer-Verlag, Toulouse, France, 2010, pp. 333–346.

[42] D. W. Hosmer, A comparison of iterative maximum likelihood estimates of the parameters of a mixture of two normal distributions under three different types of samples, Biometrics 29 (1973) 761–770.

[43] G. J. McLachlan, Estimating the linear discriminant function from initial samples containing a small number of unclassified observations, Journal of the American Statistical Association 72 (358) (1977) 403–406.

[44] T. Denœux, A k-nearest neighbor classification rule based on dempster-shafer theory, IEEE Transactions on Systems, Man and Cybernetics 25 (5) (1995) 804–813.

[45] C. Ambroise, G. Govaert, EM algorithm for partially known labels, in: Springer (Ed.), Proceedings of the 7th Conference of the International Federation of Classification Societies (IFCS 2000), Namur, Belgique, 2000, pp. 161–166.

[46] C. Ambroise, T. Denœux, G. Govaert, P. Smets, Learning from an imprecise teacher: probabilistic and evidential approaches, in: Proceedings of ASMDA'01, Compiègne, France, 2001, pp. 100–105.

[47] Y. Grandvalet, Logistic regression for partial labels, in: 9th International Conference on Information Processing and Management of Uncertainty in Knowlege-based Systems (IPMU'02), Vol. 3, Annecy, France, 2002, pp. 1935–1941.

[48] E. Hüllermeier, J. Beringer, Learning from ambiguously labeled examples, in: Proceedings of the 6th International Symposium on Intelligent Data Analysis (IDA 05), Madrid, Spain, 2005, pp. 168–179.

[49] N. D. Lawrence, B. Schölkopf, Estimating a kernel fisher discriminant in the presence of label noise, in: M. Kaufmann (Ed.), Proceedings of the 18th International Conference on Machine Learning, Williamstown, Massachussets, USA, 2001, pp. 306–313.

[50] R. Amini, P. Gallinari, Semi-supervised learning with an imperfect supervisor, Knowledge Information Systems 8 (4) (2005) 385–413.

[51] A. Karmaker, S. Kwek, A boosting approach to remove class label noise, in: Proceedings of Fifth International Conference on Hybrid Intelligent Systems, Rio de Janeiro, Brasil, 2005, pp. 6–9.

[52] P. Vannoorenberghe, P. Smets, Partially supervised learning by a credal em approach, in: L. Godo (Ed.), Proceedings of the Eighth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (Ecsqaru 2005), Vol. LNAI-3571, Springer, 2005, pp. 956–967.

[53] P. Vannoorenberghe, Estimation de modèles de mélanges finis par un algorithm em crédibiliste, Traitement du Signal 24 (2) (2007) 103–113.

[54] I. Jraidi, Z. Elouedi, Belief classification approach based on generalized credal em, in: K. Mellouli (Ed.), Proceedings of the Ninth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (Ecsqaru 2005), Vol. LNAI-4724, Springer, 2007, pp. 524–535.

[55] E. Côme, L. Oukhellou, T. Denœux, P. Aknin, Learning from partially supervised data using mixture models and belief functions, Pattern Recognition 42 (3) (2009) 334–348.

[56] H. Caillol, W. Pieczynski, A. Hillion, Estimation of fuzzy Gaussian mixture and unsupervised statistical image segmentation, IEEE Transactions on Signal Processing 6 (3) (1997) 425–440.

[57] Z. Ju, H. Liu, Fuzzy Gaussian mixture models, Pattern Recognition 45 (2012) 1146–1158.

[58] W. Maalel, K. Zhou, A. Martin, Z. Elouedi, Belief hierarchical clustering, in: F. Cuzzolin (Ed.), Belief Functions: Theory and Applications, Vol. LNAI-8764 of Lecture Notes in Artificial Intelligence, Springer, 2014, pp. 68–76.

[59] J. Zeng, L. Xie, Z.-Q. Liu, Type-2 fuzzy Gaussian mixture models, Pattern Recognition 41 (2008) 3636–3643.

[60] G. Celeux, G. Covaert, Gaussian parsimonious clustering models, International Journal of Pattern Recognition 28 (5) (1995) 781–793.

[61] L. A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.

[62] T. Denœux, Maximum likelihood estimation from fuzzy data using the fuzzy EM algorithm, Fuzzy Sets and Systems 183 (1) (2011) 72–91.

[63] T. Denœux, Maximum likelihood estimation from uncertain data in the belief function framework, IEEE Transactions on Knowledge and Data Engineering 25 (1) (2013) 119–130.

[64] E. Hüllermeier, Learning from imprecise and fuzzy observations: Data disambiguation through generalized loss minimization, International Journal of Approximate Reasoning In Press, DOI: 10.1016/j.ijar.2013.09.003.

[65] L. A. Zadeh, Probability measures of fuzzy events, Journal of Mathematical Analysis and Applications 10 (1968) 421–427.

[66] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2006.

[67] N. Chopin, Fast simulation of truncated Gaussian distributions, Statistics and Computing 21 (2) (2011) 275–288.

[68] G. J. McLachlan, T. Krishnan, The EM Algorithm and Extensions, Wiley, New York, 1997.

[69] C. Fraley, A. E. Raftery, Bayesian regularization for normal mixture estimation and model-based clustering, Journal of Classification 24 (2007) 155–181.

[70] C. Fraley, A. E. Raftery, Bayesian regularization for normal mixture estimation and model-based clustering, Tech. rep., Department of Statistics, University of Washington (2009).

[71] L. Hubert, P. Arabie, Comparing partitions, Journal of Classification 2 (1) (1985) 193–218.

[72] R. N. Kacker, J. F. Lawrence, Trapezoidal and triangular distributions for Type B evaluation of standard uncertainty, Metrologia 44 (2007) 117–127.

AppendixA. Sampling according to trapezoidal possibility distributions

We provide here details about sampling according to trapezoidal-based multivariate fuzzy numbers. We assume here that each multivariate possibility distribution $\mu_{\tilde{\mathbf{x}}_i}$ associated with $\tilde{\mathbf{x}}_i$ may be expressed as the product of the unidimensional possibility distributions $\mu_{\tilde{x}_{ij}}$ of its components $\tilde{x}_{ij}$:

$$\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}) = \prod_{j=1}^{p} \mu_{\tilde{x}_{ij}}(x_j). \tag{A.1}$$

Thus, data may be sampled feature-wise and aggregated. Let us denote by $x_{ij}^{(1)}, \ldots, x_{ij}^{(M)}$ the sample generated according to the univariate trapezoidal distribution $\mu_{\tilde{x}_{ij}}$. The sample $\mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(M)}$ following the multivariate distribution $\mu_{\tilde{\mathbf{x}}_i}$ may then be obtained by concatenating these one-dimensional samples: we obtain $\mathbf{x}_i^{(\ell)} = (x_{i1}^{(\ell)}, \ldots, x_{ip}^{(\ell)})$, for each $\ell = 1, \ldots, M$.

To obtain an univariate sample $x^{(1)}, \ldots, x^{(M)}$ from a parent variable according to a univariate trapezoidal distribution, we propose the following procedure. Let F be the cumulative distribution function of the trapezoidal distribution. First, a random sample $u^{(1)}, \ldots, u^{(M)}$ is generated according to an uniform distribution $\mathcal{U}_{[0;1]}$. Then, the inverse cdf F^{-1} is applied to these numbers to obtain the univariate sample: $x^{(\ell)} = F^{-1}(u^{(\ell)})$, for $\ell = 1, \ldots, M$.

We provide below the expression of the cumulative distribution function $F_{\tilde{x}}$ associated with an univariate trapezoidal fuzzy number $\tilde{x}$ with support $[a; d]$ and core $[b; c]$, as well as of its inverse $F_{\tilde{x}}^{-1}$. Details may be found in [72].

$$F_{\tilde{x}}(x) = \begin{cases} 0 & \text{if} \quad x \leq a, \\ \dfrac{1}{\mathcal{A}(\tilde{x})} \dfrac{(x-a)^2}{2(b-a)} & \text{if} \quad a \leq x \leq b, \\ \dfrac{1}{\mathcal{A}(\tilde{x})} \left(x - \dfrac{a+b}{2} \right) & \text{if} \quad b \leq x \leq c, \\ \dfrac{1}{\mathcal{A}(\tilde{x})} \left(\mathcal{A}(\tilde{x}) + \dfrac{(x-d)^2}{2(c-d)} \right) & \text{if} \quad c \leq x \leq d, \\ 1 & \text{if} \quad d \leq x; \end{cases} \tag{A.2}$$

with

$$\mathcal{A}(\tilde{x}) = \int \mu_{\tilde{x}}(x)\, dw, \tag{A.3}$$

$$= \frac{c+d-a-b}{2}. \tag{A.4}$$

Thus, the inverse cdf $F_{\tilde{x}}^{-1}$ is defined, for all $u \in [0;1]$, by:

$$F_{\tilde{x}}^{-1}(u) = \begin{cases} a + \sqrt{2(b-a)\mathcal{A}(\tilde{x})\,u} & \text{if} \quad 0 \leq u \leq \dfrac{b-a}{2\mathcal{A}(\tilde{x})}; \\ \dfrac{a+b}{2} + \mathcal{A}(\tilde{x})\,u & \text{if} \quad \dfrac{b-a}{2\mathcal{A}(\tilde{x})} \leq u \leq \dfrac{2c-b-a}{2\mathcal{A}(\tilde{x})}, \\ d - \sqrt{2(d-c)\mathcal{A}(\tilde{x})\,u} & \text{if} \quad \dfrac{2c-b-a}{2\mathcal{A}(\tilde{x})} \leq u \leq 1. \end{cases} \tag{A.5}$$