Learning from partially supervised data using mixture models and belief functions.

Etienne Côme, Latifa Oukhellou, Thierry Denoeux, Patrice Aknin

Learning from partially supervised data using mixture models and belief functions

E. Côme [a,c] L. Oukhellou [a,b] T. Denœux [c] P. Aknin [a]

[a] *Institut National de Recherche sur les Transports et leur Sécurité (INRETS) - LTN*
2 av. Malleret-Joinville, 94114 Arcueil Cedex, France
[b] *Université Paris XII - CERTES,*
61 av. du Général de Gaulle, 94100 Créteil, France
[c] *Université de Technologie de Compiègne - HEUDIASYC,*
Centre de Recherches de Royallieu, B.P. 20529, 60205 Compiègne Cedex, France

Abstract

This paper addresses classification problems in which the class membership of training data is only partially known. Each learning sample is assumed to consist in a feature vector $\mathbf{x}_i \in \mathcal{X}$ and an imprecise and/or uncertain "soft" label m_i defined as a Dempster-Shafer basic belief assignment over the set of classes. This framework thus generalizes many kinds of learning problems including supervised, unsupervised and semi-supervised learning. Here, it is assumed that the feature vectors are generated from a mixture model. Using the Generalized Bayesian Theorem, an extension of Bayes' theorem in the belief function framework, we derive a criterion generalizing the likelihood function. A variant of the EM algorithm dedicated to the optimization of this criterion is proposed, allowing us to compute estimates of model parameters. Experimental results demonstrate the ability of this approach to exploit partial information about class labels.

Key words: Dempster-Shafer theory, Transferable Belief Model, Mixture models, EM algorithm, Classification, Clustering, Partially supervised learning, Semi-supervised learning.

Email address: come@inrets.fr Tel: +33 1 47 40 73 49 (E. Côme).

1. Introduction

Machine learning classically deals with two different problems: supervised learning (classification) and unsupervised learning (clustering). However, in recent years, new paradigms have emerged to mix these two approaches in order to extend the applicability of machine learning algorithms.

The paradigm that emerged first is *semi-supervised learning* [1, 2], where the learning set $\mathbf{X}^{ss} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_M, y_M), \mathbf{x}_{M+1}, \ldots, \mathbf{x}_N\}$ is composed of two different parts. In the first part, the true class labels y_i are specified, whereas in the second part only the feature vectors $\mathbf{x}_i$ are given. The importance for such problems comes from the fact that labelled data are often difficult to obtain, while unlabelled ones are easily available. Using unlabelled data may thus be a means to enhance the performances of supervised algorithms with low additional cost. The recent publication of a collected volume [3] shows the important activity around this issue in the Machine Learning field. Recent approaches to semi-supervised learning fall into two main categories:

– An important class of methods is based on the hypothesis that the decision boundary should be located in low density areas. Methods in this category aim at deriving a regularizer of the conditional log-likelihood, taking into account the unlabelled data to bias the decision boundary towards low density areas [4, 5]. The Transductive Support Vector Machine [6] uses a margin-based criterion to achieve a similar goal. All these methods suffer from the problem of local maxima, although some relaxation schemes lead to a convex optimization problem in the case of the Transductive Support Vector Machine [7].
– Other methods are based on the assumption that the high-dimensional input data lie near a low dimensional manifold. Unlabelled data are then useful as they help in estimating this manifold. Methods relying on the manifold assumption are typically based on unsupervised dimensionality reduction techniques such as PCA or Kernel-PCA, or on label propagation in a graph [8, 9].

Other paradigms have also been proposed to take into account more sophisticated information on class labels. For example, *partially supervised learning* [10, 11, 12, 13, 14] deals with constraints on the possible classes of samples. In this case, the learning set has the following form $\mathbf{X}^{ps} = \{(\mathbf{x}_1, C_1), \ldots, (\mathbf{x}_N, C_N)\}$, where C_i is a set of possible classes for learning example i. If all classes are possible, the example is not labelled. Conversely, the example is perfectly labelled if only one class is specified ($|C_i| = 1$). Between these two extreme cases, this approach may also handle situations where some examples are known to belong to any subset of classes. In this case, they are considered as *partially* or *imprecisely* labeled. This framework is thus more general than the semi-supervised learning problem.

A completely different paradigm is based on the notion of *label noise* and assumes that the class labels may be pervated by random errors. In this case, class labels are thus *precise*, but *uncertain*. Recent contributions along these lines can by found in References [15, 16, 17]. In the first two papers, a generative model of label noise is assumed. It is then proposed to model the label noise

process by conditional distributions specifying the probabilities that samples labelled as belonging to one class, were in fact drawn from from another class. The parameters of such model are then learnt by maximizing the likelihood of the observations knowing the labels [15] or are optimized using a Classification Maximum Likelihood approach [16]. A kernelized version of this kind of approach has been proposed in [15, 18].

The investigations reported in this paper provide a solution to deal with imprecise and/or uncertain class labels, and can therefore be seen as addressing a more general issue than in the above paradigms. Our approach is based on the theory of belief functions [19, 20], a framework known to be well suited to represent imprecise and uncertain information. In this paper, we explore its use to represent knowledge on class membership of learning examples, in order to extend the partially supervised framework. In this way, both the uncertainty and the imprecision of class labels may be handled. The considered training sets are of the form $\mathbf{X}^{iu} = \{(\mathbf{x}_1, m_1), \ldots, (\mathbf{x}_N, m_N)\}$, where m_i is a basic belief assignment, or Dempster-Shafer mass function [19] encoding our knowledge about the class of example i. The m_is (hereafter referred to as "soft labels") may represent different kinds of knowledge, from precise to imprecise and from certain to uncertain. Thus, previous problems are special cases of this general formulation. Other studies have already proposed solutions in which class labels are expressed by possibility distributions or belief functions [21, 22, 23, 24]. These labels are interesting when they are supplied by one or several experts and when crisp assignments are hard to obtain. In such cases, the elicitation of experts' opinions regarding the class membership of objects under consideration, in term of possibility or belief functions, can be of interest [21, 25].

In this article, we present a new approach to solve learning problems of this type, based on a preliminary study by Vannoorenberghe and Smets [26, 27]. This solution is based on mixture models, and therefore assumes a generative model for the data. Generative models have already proved their efficiency in a lot of applications [28]. Their flexibility offers also a good way to benefit from domain specific knowledge, as shown, for example, in text classification [29]. Finally, the adaptability of the Expectation Maximization (EM) algorithm, which may easily handle specific constraints, is an advantage of generative models. Note that the approach introduced in [26] and [30] to apply the EM algorithm to data with soft labels, although based on strong intuitions, was only imperfectly formalized. It was not clear, in particular, what was the equivalent of the log-likelihood function in this case, and if the proposed extension of the EM algorithm converged at all. Precise answers to these questions are provided here.

This article is organized as follows. Background material on belief functions and on the estimation of parameters in mixture models using the EM algorithm will first be recalled in Sections 2 and 3, respectively. The problem of learning from data with soft labels will then be addressed in Section 4, which constitutes the core of the paper. A criterion extending the usual likelihood criterion will first be derived in Section 4.1, and a version of the EM algorithm that optimizes this criterion will be introduced in Section 4.2. Practical considerations and a general discussion will be presented

in Sections 4.3 and 4.4, respectively. Finally, simulation results illustrating the advantages of this approach will be reported in Section 5, and Section 6 will conclude the paper.

2. Background on Belief Functions

2.1. *Belief Functions on a Finite Frame*

The theory of belief functions was introduced by Dempster [31] and Shafer [19]. The interpretation adopted throughout this paper will be that of the Transferable Belief Model (TBM) introduced by Smets [20]. The first building block of belief function theory is the *basic belief assignment* (bba), which models the beliefs held by an agent regarding the actual value of a given variable taking values in a finite domain (or *frame of discernment*) Ω, based on some body of evidence. A bba m^Ω is a mapping from 2^Ω to $[0,1]$ verifying:

$$\sum_{\omega \subseteq \Omega} m^\Omega(\omega) = 1. \tag{1}$$

Each mass $m^\Omega(\omega)$ is interpreted as the part of the agent's belief allocated to the hypothesis that the variable takes some value in ω [19, 20]. The subsets ω for which $m^\Omega(\omega) > 0$ are called the *focal sets*. A *categorical* bba has only one focal set. A *simple* bba has at most two focal sets, including Ω. A *Bayesian* bba is a bba whose focal sets are singletons. A bba is said to be *consonant* if its focal sets are nested.

A bba is in one to one correspondence with other representations of the agent's belief, including the plausibility function defined as:

$$pl^\Omega(\omega) \triangleq \sum_{\alpha \cap \omega \neq \emptyset} m^\Omega(\alpha), \quad \forall \omega \subseteq \Omega. \tag{2}$$

The quantity $pl^\Omega(\omega)$ is thus equal to the sum of the basic belief masses assigned to propositions that are not in contradiction with ω; it corresponds to the maximum degree of support that could be given to ω, if further evidence became available. The plausibility function associated to a Bayesian bba is a probability measure. If m^Ω is consonant, then pl^Ω is a possibility measure: it verifies $pl^\Omega(\alpha \cup \beta) = \max(pl^\Omega(\alpha), pl^\Omega(\beta))$, for all $\alpha, \beta \subseteq \Omega$.

2.2. *Conditioning and Combination*

Given two bbas m_1^Ω, m_2^Ω supported by two distinct bodies of evidence, we may build a new bba $m_{1\textcircled{\cap}2}^\Omega = m_1^\Omega \textcircled{\cap} m_2^\Omega$ that corresponds to the conjunction of these two bodies of evidence:

$$m_{1\textcircled{\cap}2}^\Omega(\omega) \triangleq \sum_{\alpha_1 \cap \alpha_2 = \omega} m_1^\Omega(\alpha_1) m_2^\Omega(\alpha_2), \quad \forall \omega \subseteq \Omega. \tag{3}$$

This operation is usually referred to as the *unnormalized Dempster's rule*, or the *TBM conjunctive rule*. Any positive mass assigned to the empty set during the combination process is interpreted as indicating partial conflict between the two bodies of evidence. If the frame of discernment is

supposed to be exhaustive, this mass is usually reallocated to other subsets, leading to the definition of the normalized Demspter's rule $\oplus$ defined as:

$$m^{\Omega}_{1\oplus 2}(\omega) = \begin{cases} 0 & \text{if } \omega = \emptyset \\ \frac{m^{\Omega}_{1\bigcirc\!\!\!\cap 2}(\omega)}{1-m^{\Omega}_{1\bigcirc\!\!\!\cap 2}(\emptyset)} & \text{if } \omega \subseteq \Omega, \omega \neq \emptyset, \end{cases} \tag{4}$$

which is well defined provided $m^{\Omega}_{1\bigcirc\!\!\!\cap 2}(\emptyset) \neq 1$. Note that, if m^{Ω}_1 (or m^{Ω}_2) is Bayesian, then $m^{\Omega}_{1\oplus 2}(\omega)$ is also Bayesian.

The combination of a bba m^{Ω} with a categorical bba focused on $\alpha \subseteq \Omega$ using the TBM conjunctive rule is called (unnormalized) *conditioning*. The resulting bba is denoted $m^{\Omega}(\omega|\alpha)$. Probabilistic conditioning is recovered when m^{Ω} is Bayesian, and normalization is performed. Using this definition, we may rewrite the conjunctive combination rule:

$$m^{\Omega}_{1\bigcirc\!\!\!\cap 2}(\omega) = \sum_{\alpha\subseteq\Omega} m^{\Omega}_1(\alpha) m^{\Omega}_2(\omega|\alpha), \quad \forall \omega \subseteq \Omega, \tag{5}$$

which is a counterpart of the total probability theorem in probability theory [32, 33]. This expression shows more clearly the link with probability calculus and provides a shortcut to perform marginal calculations on a product space when conditional bbas are available [33]. Consider two frames Ω and Θ, and a set of conditional belief functions $m^{\Theta|\Omega}(\cdot|\omega)$ for all $\omega \subseteq \Omega$. Each conditional bba $m^{\Theta|\Omega}(\cdot|\omega)$ represents the agent's belief on Θ in a context where ω holds. The combination of these conditional bbas with a bba m^{Ω} on Ω yields the following bba on Θ:

$$m^{\Theta}(\theta) = \sum_{\omega\subseteq\Omega} m^{\Omega}(\omega) m^{\Theta|\Omega}(\theta|\omega), \quad \forall \theta \subseteq \Theta. \tag{6}$$

A similar formula holds for the plausibility function:

$$pl^{\Theta}(\theta) = \sum_{\omega\subseteq\Omega} m^{\Omega}(\omega) pl^{\Theta|\Omega}(\theta|\omega), \quad \forall \theta \subseteq \Theta. \tag{7}$$

This property bears some resemblance with the total probability theorem, except that the sum is taken over the power set of Ω and not over Ω. We will name it the *total plausibility theorem*.

2.3. *Independence*

The usual independence concept of probability theory does not easily find a counterpart in belief function theory, where different notions must be used instead. The simplest form of independence defined in the context of belief functions is *cognitive independence* [19, p. 149]. Frames Ω and Θ are said to be cognitively independent with respect to $pl^{\Omega\times\Theta}$ iff we have

$$pl^{\Omega\times\Theta}(\omega \times \theta) = pl^{\Omega}(\omega)\, pl^{\Theta}(\theta), \quad \forall \omega \subseteq \Omega, \forall \theta \subseteq \Theta. \tag{8}$$

Cognitive independence boils down to probabilistic independence when $pl^{\Omega\times\Theta}$ is a probability measure. However, the concept of cognitive independence does not inherit all of the properties of the probabilistic notion of independence and may be seen as a weak form of independence. See [34, 35] for an in-depth analysis of independence concepts in the belief function theory.

2.4. *Belief Functions on the Real Line*

The theory presented in the previous section can easily be extended to continuous belief functions on the real line, assuming focal sets to be real intervals [36]. In this context, the concept of bba is replaced by that of *basic belief density* (bbd), defined as a mapping $m^{\mathbb{R}}$ from the set of closed real intervals to $[0, +\infty)$ such that

$$\int_{-\infty}^{+\infty} \int_{x}^{+\infty} m^{\mathbb{R}}([x, y])dydx \leq 1. \tag{9}$$

By convention, the one's complement of the integral in the left-hand side of (9) is allocated to $\emptyset$. As in the discrete case, $pl^{\mathbb{R}}([a, b])$ is defined as a sum over all intervals whose intersection with $[a, b]$ is non-empty:

$$pl^{\mathbb{R}}([a, b]) \triangleq \iint_{[x,y]\cap[a,b]\neq\emptyset} m^{\mathbb{R}}([x, y])dydx. \tag{10}$$

Further extension of these definitions to $\mathbb{R}^d, d > 1$ is possible [37]. It is also possible to define belief functions on mixed product spaces involving discrete and continuous frames (see, e.g., [38]).

The last tool from belief function theory which is of interest here is the Generalized Bayesian Theorem (GBT), introduced by Smets [39, 33].

2.5. *Generalized Bayesian Theorem*

The Bayes' theorem of probability theory is replaced in the framework of belief function by the Generalized Bayesian Theorem (GBT), [39, 33, 40, 41]. This theorem provides a way to reverse conditional belief functions without any prior knowledge. Let us consider two spaces, $\mathcal{X}$ the observation space and Θ the parameter space. Assume that our knowledge is encoded by a set of conditional bbas $m^{\mathcal{X}|\Theta}(.|\theta_i), \theta_i \in \Theta$, which express our belief in future observations conditionally on each θ_i, and we observe a realization $x \subseteq \mathcal{X}$. The question is: given this observation and the set of conditional bbas, what is our belief on the value of Θ? The answer is given by the GBT and states that the resulting plausibility function on Θ has the following form:

$$pl^{\Theta|\mathcal{X}}(\theta|x) = pl^{\mathcal{X}|\Theta}(x|\theta) = 1 - \prod_{\theta_i\in\theta} (1 - pl^{\mathcal{X}|\Theta}(x|\theta_i)). \tag{11}$$

When a prior bba m_0^{Θ} on Θ is available, it should be combined conjunctively with the bba defined by (11). The classical Bayes' theorem is recovered when the conditional bbas $m^{\mathcal{X}|\Theta}(.|\theta_i)$ and the prior bba m_0^{Θ} are Bayesian.

After this review of some tools from belief functions theory, the next part is dedicated to the probabilistic formulation of the clustering and classification problems in terms of mixture model.

3. Mixture Models and the EM Algorithm

The Expectation Maximization (EM) algorithm provides a general solution to problems involving missing data [42]. Here, we are interested in its most classical application, which concerns mixture

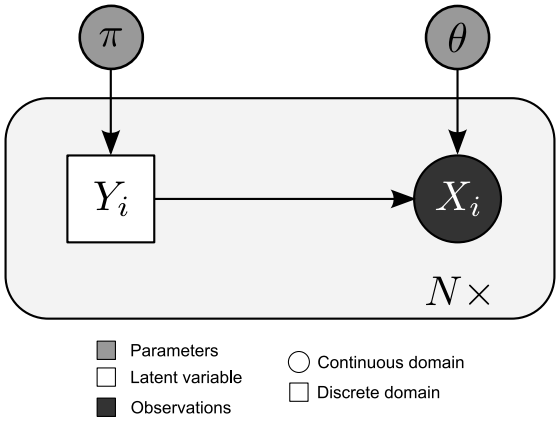


Fig. 1. Graphical model representation of the mixture model (unsupervised learning)

estimation problems.

3.1. *Mixture Models*

Mixture models suppose the following data generation scheme:

– The true class labels $\{y_1, \ldots, y_N\}$ of data points are realizations of independent and identically distributed (i.i.d) random variables $Y_1, \ldots, Y_N \sim Y$ taking their values in the set of all K classes $\mathcal{Y} = \{c_1, \ldots, c_K\}$ and distributed according to a multinomial distribution $\mathcal{M}(1, \pi_1, \ldots, \pi_K)$:

$$\mathbb{P}(Y = c_k) = \pi_k, \quad \forall k \in \{1, \ldots, K\}. \tag{12}$$

The π_k are thus the class proportions; they verify $\sum_{k=1}^{K} \pi_k = 1$.

– The observed values $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ are drawn using the class conditional density in relation with the class label. More formally, $X_1, \ldots, X_N \sim X$ are continuous random variables taking values in $\mathcal{X}$, with conditional probability density functions:

$$f(\mathbf{x}|Y = c_k) = f(\mathbf{x}; \boldsymbol{\theta}_k), \quad \forall k \in \{1, \ldots, K\}. \tag{13}$$

The parameters of this generative model are therefore the proportions $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ and the parameters of the class conditional densities $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K$. To simplify the notations, the vector of all model parameters is denoted:

$$\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_K, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K).$$

This generative model can be represented by an oriented graph, as shown in Figure 1.

The simplest case corresponds to the supervised learning problem where both the observations and their classes are known. The learning set is then:

$$\mathbf{X}^s = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}. \tag{14}$$

It may be noted, equivalently, $\mathbf{X}^s = \{(\mathbf{x}_1, \mathbf{z}_1), \ldots, (\mathbf{x}_N, \mathbf{z}_N)\}$, where $\mathbf{z}_i \in \{0, 1\}^K$ are binary variables encoding the class membership of each data point, such that $z_{ik} = 1$ if $y_i = c_k$, and $z_{ik} = 0$ otherwise. In this case, the complete data log-likelihood can be written:

$$L(\mathbf{\Psi};\mathbf{X}^s) = \sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\ln(\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)). \tag{15}$$

In this case, the Maximum Likelihood Estimator is generally easy to compute because optimization problems are decoupled for each class.

In unsupervised learning problems, however, the available data are only the i.i.d realizations of X, $\mathbf{X}^u = \{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$, provided by the generative model. To learn the parameters and the associated partition of the data, the log-likelihood must be computed according to the marginal density $\sum_{k=1}^{K}\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)$ of X_i. We then have

$$L(\mathbf{\Psi};\mathbf{X}^u) = \sum_{i=1}^{N}\ln\left(\sum_{k=1}^{K}\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)\right). \tag{16}$$

3.2. *EM Algorithm*

The log-likelihood function defined by (16) is difficult to optimize and may lead to a set of different local maxima. The EM algorithm [42, 43] is nowadays the classical solution to this problem. The missing data of the clustering problem are the true class labels y_i of learning examples. The EM algorithm brings them back in the optimization problem defined by (16), and uses them to build an iterative ascent strategy which, given an initial parameter estimate $\mathbf{\Psi}^{(0)}$, alternates two steps: expectation (E) and maximization (M), until a local maximum is found.

The basis of the EM algorithm can be found in the link between the log-likelihood $L(\mathbf{\Psi};\mathbf{X}^u)$ and the complete log-likelihood $L(\mathbf{\Psi};\mathbf{X}^s)$. The following relations hold:

$$L(\mathbf{\Psi};\mathbf{X}^u) = L(\mathbf{\Psi};\mathbf{X}^s) - \sum_{i=1}^{N}\ln(\mathbb{P}(\mathbf{z}_i|\mathbf{x}_i)) \tag{17}$$

$$= \sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\ln(\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)) - \sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\ln\left(\frac{\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K}\pi_{k'} f(\mathbf{x}_i;\boldsymbol{\theta}_{k'})}\right). \tag{18}$$

The observation labels z_{ik} are unknown, but they can be replaced by their expectation given the current parameters estimates $\mathbf{\Psi}^{(q)}$ at iteration q and the observed values $\mathbf{x}_1,\ldots,\mathbf{x}_N$. Relation (18) remains valid and we obtain:

$$L(\mathbf{\Psi};\mathbf{X}^u) = \underbrace{\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)}\ln\left(\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)\right)}_{Q(\mathbf{\Psi},\mathbf{\Psi}^{(q)})} - \underbrace{\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)}\ln\left(\frac{\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K}\pi_{k'} f(\mathbf{x}_i;\boldsymbol{\theta}_{k'})}\right)}_{H(\mathbf{\Psi},\mathbf{\Psi}^{(q)})}, \tag{19}$$

with:

$$t_{ik}^{(q)} = \mathbb{E}_{\mathbf{\Psi}^{(q)}}[z_{ik}|\mathbf{x}_i] = \mathbb{P}(z_{ik}=1|\mathbf{\Psi}^{(q)},\mathbf{x}_i) = \frac{\pi_k^{(q)} f(\mathbf{x}_i;\boldsymbol{\theta}_k^{(q)})}{\sum_{k'=1}^{K}\pi_{k'}^{(q)} f(\mathbf{x}_i;\boldsymbol{\theta}_{k'}^{(q)})}. \tag{20}$$

Such a decomposition is useful to define an iterative ascent strategy. The difference between the log-likelihood evaluated at iterations $q+1$ and q is:

$$L(\mathbf{\Psi}^{(q+1)};\mathbf{X}^u) - L(\mathbf{\Psi}^{(q)};\mathbf{X}^u) = \left(Q(\mathbf{\Psi}^{(q+1)},\mathbf{\Psi}^{(q)}) - Q(\mathbf{\Psi}^{(q)},\mathbf{\Psi}^{(q)})\right) + \left(H(\mathbf{\Psi}^{(q)},\mathbf{\Psi}^{(q)}) - H(\mathbf{\Psi}^{(q+1)},\mathbf{\Psi}^{(q)})\right). \tag{21}$$

The second term in the right-hand side of (21) is:

$$H(\boldsymbol{\Psi}^{(q)}, \boldsymbol{\Psi}^{(q)}) - H(\boldsymbol{\Psi}^{(q+1)}, \boldsymbol{\Psi}^{(q)}) = \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln(t_{ik}^{(q)}) - \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln(t_{ik}^{(q+1)}) \tag{22}$$

$$= -\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(\frac{t_{ik}^{(q+1)}}{t_{ik}^{(q)}}\right). \tag{23}$$

We thus have $H(\boldsymbol{\Psi}^{(q)}, \boldsymbol{\Psi}^{(q)}) - H(\boldsymbol{\Psi}^{(q+1)}, \boldsymbol{\Psi}^{(q)}) \geq 0$, as a consequence Jensen's inequality. Therefore, the log-likelihood will increase between iterations q and $q+1$ if we find new parameter estimates $\boldsymbol{\Psi}^{(q+1)}$ such that:

$$Q(\boldsymbol{\Psi}^{(q+1)}, \boldsymbol{\Psi}^{(q)}) - Q(\boldsymbol{\Psi}^{(q)}, \boldsymbol{\Psi}^{(q)}) > 0. \tag{24}$$

Consequently, the maximization of the auxiliary function Q is sufficient to improve the likelihood. Furthermore, because the sum over the classes is outside the logarithm, the optimization problems are decoupled and the maximization is simpler. The EM algorithm starts with initial estimates $\boldsymbol{\Psi}^{(0)}$ and alternates two steps (called the E and M steps) to define a sequence of parameters estimates with increasing likelihood values. These steps are recalled in more detail below.

E Step: During this step, the expectation of the complete data log-likelihood according to the current value of the parameter is computed. This expectation defines the auxiliary function $Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)})$:

$$Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) = \mathbb{E}_{\boldsymbol{\Psi}^{(q)}}\left[L(\boldsymbol{\Psi}; \mathbf{X}^s) | \mathbf{x}_1, \ldots, \mathbf{x}_N\right] = \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)\right). \tag{25}$$

For this, the posterior probabilities $t_{ik}^{(q)}$ are computed from the current estimates of the parameters using (20). When these posterior probabilities have been computed, we can optimize the auxiliary function to find the new parameters estimates.

M Step: During this step, the parameters are updated. The maximization of the auxiliary function with respect to $\boldsymbol{\Psi}$ provides the new estimate:

$$\boldsymbol{\Psi}^{(q+1)} = \arg\max_{\boldsymbol{\Psi}} Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}). \tag{26}$$

As mentioned previously, this estimate has a higher likelihood than the previous one. This maximization problem has an analytic solution for the mixing proportions: the maximum of

$$\sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln(\pi_k)$$

under the constraint $\sum_{k=1}^{K} \pi_k = 1$ yields

$$\pi_k^{(q+1)} = \frac{1}{N}\sum_{i=1}^{N} t_{ik}^{(q)}. \tag{27}$$

Moreover, if classical parametric models are assumed for the conditional densities (Multivariate normal, Exponential, ...), the optimization problem solution often has a closed form. For example, if multivariate normal densities functions are considered, $f(x; \boldsymbol{\theta}_k) = \phi(x; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, all the parameters are updated using the following formulas:

$$\boldsymbol{\mu}_k^{(q+1)} = \frac{1}{\sum_{i=1}^N t_{ik}^{(q)}} \sum_{i=1}^N t_{ik}^{(q)} \mathbf{x}_i \tag{28}$$

$$\boldsymbol{\Sigma}_k^{(q+1)} = \frac{1}{\sum_{i=1}^N t_{ik}^{(q)}} \sum_{i=1}^N t_{ik}^{(q)} (\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)})(\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)})'. \tag{29}$$

The Expectation and the Maximization steps are repeated until convergence, which can be detected when the increase of the log-likelihood does not exceed a given threshold.

The mixture model setting and the EM algorithm can be adapted to handle specific learning problems such as the semi-supervised and the partially supervised cases, as we will see in the next subsection.

3.3. *Semi-supervised and Partially Supervised Learning of Mixture Models*

In semi-supervised learning, the component origins of the samples are known only for the M first observations. Consequently the log-likelihood can be decomposed in two parts corresponding, respectively, to the supervised and unsupervised learning examples:

$$L(\boldsymbol{\Psi}, \mathbf{X}^{ss}) = \sum_{i=1}^{M} \sum_{k=1}^{K} z_{ik} \ln(\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)) + \sum_{i=M+1}^{N} \ln(\sum_{k=1}^{K} \pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)), \tag{30}$$

It is easy to see that an EM algorithm can be adapted to optimize this function. Function Q can be rewritten as:

$$Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(q)}) = \sum_{i=1}^{M} \sum_{k=1}^{K} z_{ik} \ln(\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)) + \sum_{i=M+1}^{N} \sum_{k=1}^{K} t_{ik}^{(q)} \ln(\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)). \tag{31}$$

This form of Q leads to a new version of the EM algorithm dedicated to semi-supervised learning problems. In fact, the modification affects only the E step, where the t_{ik} are only computed for unlabelled observations. During the M step, the z_{ik} are used instead of the t_{ik} for labelled data.

In the context of partially known labels, the available information on y_i is a subset of possible classes $C_i \subseteq \mathcal{Y}$. The dataset $\mathbf{X}^{ps} = \{(\mathbf{x}_1, C_1), \ldots, (\mathbf{x}_N, C_N)\}$ under consideration contains independent realizations $(\mathbf{x}_i, Y_i \in C_i)$. The likelihood must therefore be computed from the joint distribution of $(\mathbf{x}_i, Y_i \in C_i)$:

$$\mathbb{P}(\mathbf{x}_i, Y_i \in C_i) = \sum_{\{k: c_k \in C_i\}} \mathbb{P}(\mathbf{x}_i, Y_i = c_k) = \sum_{k=1}^{K} \mathbb{P}(Y_i \in C_i \cap \{c_k\}) f(\mathbf{x}_i | Y_i = c_k) \tag{32}$$

where:

$$\mathbb{P}(Y_i \in C_i \cap \{c_k\}) = \begin{cases} 0 & \text{if } c_k \notin C_i, \\ \pi_k & \text{if } c_k \in C_i. \end{cases} \tag{33}$$

Introducing the notation $\mathbf{l}_i = (l_{i1}, \ldots, l_{iK}) \in \{0,1\}^K$ which is simply an indicator of the subset $C_i \subseteq \mathcal{Y}$ corresponding to the partial label ($l_{ik} = 1$ if $c_k \in C_i$, $l_{ik} = 0$ otherwise), we may write:

$$\mathbb{P}(Y_i \in C_i \cap \{c_k\}) = l_{ik} \pi_k. \tag{34}$$

Substituting $\mathbb{P}(Y_i \in \{C_i \cap c_k\})$ by $l_{ik}\pi_k$ in (32), we have:

$$\mathbb{P}(\mathbf{x}_i, Y_i \in C_i) = \sum_{k=1}^{K} l_{ik}\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k) \tag{35}$$

The log-likelihood becomes:

$$L(\boldsymbol{\Psi}, \mathbf{X}^{ps}) = \sum_{i=1}^{N} \ln \left(\sum_{k=1}^{K} l_{ik}\pi_k f(\mathbf{x}_i, \boldsymbol{\theta}_k) \right). \tag{36}$$

The EM algorithm may also be used in this context, with few modifications to produce parameter estimates [11]. The modifications affect only the E step where the posterior probabilities of each class are computed according to:

$$t_{ik}^{(q)} = \frac{l_{ik}\pi_k^{(q)} f(\mathbf{x}_i; \boldsymbol{\theta}_k^{(q)})}{\sum_{k'=1}^{K} l_{ik}\pi_{k'}^{(q)} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'}^{(q)})}. \tag{37}$$

The M step is not affected by taking into account the additional information and is computed as usually.

4. Extension to Imprecise and Uncertain Labels

Our method extends the approach described above to handle *imprecise* and *uncertain* class labels defined by belief functions.

4.1. *Derivation of a Generalized Likelihood Criterion*

In this section, we shall assume the learning set under consideration to be of the form:

$$\mathbf{X}^{iu} = \{(\mathbf{x}_1, m_1^{\mathcal{Y}}), \ldots, (\mathbf{x}_N, m_N^{\mathcal{Y}})\}, \tag{38}$$

where each $m_i^{\mathcal{Y}}$ is a bba on the set $\mathcal{Y}$ of classes, encoding all available information about the class of example i. As before, the $\mathbf{x}_i$ will be assumed to have been generated according to the mixture model defined in Section 3.1. Our goal is to extend the previous method to estimate the model parameters from dataset (38). For that purpose, an objective function generalizing the likelihood function needs to be defined.

The concept of likelihood function has strong relations with that of possibility and, more generally, plausibility, as already noted by several authors [44, 45, 46, 47, 48]. Furthermore, selecting the simple hypothesis with highest plausibility given the observations $\mathbf{X}^{iu}$ is a natural decision strategy in the belief function framework [49]. We thus propose as an estimation principle to search for the value of parameter $\boldsymbol{\psi}$ with maximal conditional plausibility given the data:

$$\hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\psi}} pl^{\boldsymbol{\Psi}}(\{\boldsymbol{\psi}\}|\mathbf{X}^{iu}). \tag{39}$$

To avoid cumbersome notations, we shall not distinguish between the singleton $\{\boldsymbol{\psi}\}$ and the value $\boldsymbol{\psi}$; the notation $pl^{\boldsymbol{\Psi}}(\{\boldsymbol{\psi}\}|\mathbf{X}^{iu})$ will thus be simplified to $pl^{\boldsymbol{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu})$.

The correctness of the intuition leading to this choice of (39) as an estimation principle seems to be confirmed by the fact that the logarithm of $pl^{\boldsymbol{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu})$ is an immediate generalization of criteria (16), (30) and (36), as shown by the following proposition.

Proposition 4.1 *The logarithm of the conditional plausibility of* $\boldsymbol{\Psi}$ *given* $\mathbf{X}^{iu}$ *is given by*

$$\ln\left(pl^{\boldsymbol{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu})\right) = \sum_{i=1}^{N} \ln\left(\sum_{k=1}^{K} pl_{ik}.\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)\right) + \nu, \tag{40}$$

where the pl_{ik} *are the plausibilities of each class* k *for each sample* i *according to soft labels* m_i *and* ν *is a constant independent of* $\boldsymbol{\psi}$*.*

Proof. Using the GBT (11), the plausibility of parameters can be expressed from the plausibility of the observed values:

$$pl^{\boldsymbol{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu}) = pl^{\mathcal{X}_1\times\ldots\times\mathcal{X}_N}(\mathbf{x}_1,\ldots,\mathbf{x}_N|\boldsymbol{\psi}). \tag{41}$$

By making the conditional cognitive independence assumption (8), this plausibility can be decomposed as:

$$pl^{\boldsymbol{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu}) = \prod_{i=1}^{N} pl^{\mathcal{X}_i}(\mathbf{x}_i|\boldsymbol{\psi}). \tag{42}$$

Using the Total Plausibility Theorem (7), we may express the plausibility $pl^{\mathcal{X}_i}(\mathbf{x}_i|\boldsymbol{\psi})$ of an observation $\mathbf{x}_i$ knowing the parameter value as:

$$pl^{\mathcal{X}_i}(\mathbf{x}_i|\boldsymbol{\psi}) = \sum_{C\subseteq\mathcal{Y}} m^{\mathcal{Y}_i}(C|\boldsymbol{\psi}) pl^{\mathcal{X}_i|\mathcal{Y}_i}(\mathbf{x}_i|C,\boldsymbol{\psi}), \tag{43}$$

where $m^{\mathcal{Y}_i}(.|\boldsymbol{\psi})$ is a bba representing our beliefs regarding the class Z_i of example i. This bba comes from the combination of two information sources: the "soft" label $m_i^{\mathcal{Y}}$ and the proportions $\boldsymbol{\pi}$, which induce a Bayesian bba $m^{\mathcal{Y}}(\cdot|\boldsymbol{\pi})$ with $m^{\mathcal{Y}}(\{c_k\}|\boldsymbol{\pi}) = \pi_k$ for all $c_k \in \mathcal{Y}$. As these two sources are supposed to be distinct, they can be combined using the conjunctive rule (3):

$$m^{\mathcal{Y}_i}(\cdot|\boldsymbol{\psi}) = m_i^{\mathcal{Y}} \textcircled{\cap} m^{\mathcal{Y}}(\cdot|\boldsymbol{\pi}).$$

As $m^{\mathcal{Y}}(\cdot|\boldsymbol{\pi})$ is Bayesian, the same property holds for $m^{\mathcal{Y}_i}(\cdot|\boldsymbol{\psi})$. We have:

$$m^{\mathcal{Y}_i}(\{c_k\}|\boldsymbol{\psi}) = \sum_{C\cap c_k\neq\emptyset} m_i^{\mathcal{Y}}(C) m^{\mathcal{Y}}(\{c_k\}|\boldsymbol{\pi}) = pl_{ik}\,\pi_k, \quad \forall k\in\{1,\ldots,K\} \tag{44}$$

$$m^{\mathcal{Y}_i}(C|\boldsymbol{\psi}) = 0, \quad \forall C\subseteq\mathcal{Y} \text{ such that } |C|>1. \tag{45}$$

In the right-hand side of (43), the only terms in the sum that need to be considered are those corresponding to subsets C of $\mathcal{Y}$ such that $|C| = 1$. Consequently, we only need to express $pl^{\mathcal{X}_i|\mathcal{Y}_i}(\mathbf{x}_i|c_k,\boldsymbol{\psi})$ for all $k \in \{1,...,K\}$. There is a difficulty at this stage, since $pl^{\mathcal{X}_i|\mathcal{Y}_i}(\cdot|c_k,\boldsymbol{\psi})$ is the continuous probability measure with density function $f(\mathbf{x};\boldsymbol{\theta}_k)$: consequently, the plausibility of any single value would be null if observations $\mathbf{x}_i$ had an infinite precision. However, observations always have a finite precision, so that what we denote by $pl^{\mathcal{X}_i|\mathcal{Y}_i}(\mathbf{x}_i|c_k,\boldsymbol{\psi})$ is in fact the plausibility of a infinitesimal region around $\mathbf{x}_i$ with volume $\mathrm{d}x_{i1}\ldots\mathrm{d}x_{ip}$ (where p is the feature space dimension). We thus have

$$pl^{\mathcal{X}_i|\mathcal{Y}_i}(\mathbf{x}_i|c_k,\boldsymbol{\psi}) = f(\mathbf{x}_i;\boldsymbol{\theta}_k)\mathrm{d}x_{i1}\ldots\mathrm{d}x_{ip}. \tag{46}$$

Using (44) and (46), (43) can be expressed as:

$$pl^{\mathcal{X}_i}(\mathbf{x}_i|\boldsymbol{\psi}) = \left(\sum_{k=1}^{K} pl_{ik}\pi_k f(\mathbf{x}_i;\boldsymbol{\theta}_k)\right)\mathrm{d}x_{i1}\ldots\mathrm{d}x_{ip}. \tag{47}$$

Substituting this expression in (42), we obtain:

$$pl^{\mathbf{\Psi}}(\boldsymbol{\psi}|\mathbf{X}^{iu}) = \prod_{i=1}^{N} \left[\left(\sum_{k=1}^{K} pl_{ik} \pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k) \right) \mathrm{d}x_{i1} \dots \mathrm{d}x_{ip} \right]. \tag{48}$$

The terms $\mathrm{d}x_{ij}$ can be considered as multiplicative constants that do not affect the optimization problem. By taking the logarithm of (48), we get (40), which completes the proof. □

Remark 4.1 *The bba m_i defining the label of sample i does not appear in its bba form in (40), but through the plausibilities pl_{ik}. Therefore, labels sharing the same plausibility profile (i.e., the same plausibilities of the singletons) are handled identically. This invariance comes from the probabilistic nature of the generative model. The full expressive power of belief functions is not used, but only $|\mathcal{Y}|$ parameters (to be compared to $2^{\mathcal{Y}}$) are needed to define a label, which is an advantage when the number of components is high.*

Remark 4.2 *Our estimation principle can be expressed as:*

$$\hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\psi}} L(\mathbf{\Psi}; \mathbf{X}^{iu}), \tag{49}$$

with

$$L(\mathbf{\Psi}; \mathbf{X}^{iu}) = \sum_{i=1}^{N} \ln \left(\sum_{k=1}^{K} pl_{ik}\, \pi_k\, f(\mathbf{x}_i; \boldsymbol{\theta}_k) \right). \tag{50}$$

This choice of this notation is justified by the fact that (50) extends the maximum likelihood criteria (16), (30) and (36):

- *When the m_i are all vacuous, we have $pl_{ik} = 1$, $\forall i, k$, and the unsupervised criterion (16) is recovered;*
- *In the semi-supervised case, we have*

$$pl_{ik} = \begin{cases} z_{ik}, & \forall i \in \{1, \dots, M\},\ \forall k \\ 1, & \forall i \in \{M+1, \dots, N\},\ \forall k. \end{cases}$$

  In that case (50) is equivalent to (30);
- *Finally, the partially supervised criterion (36) is recovered when labels are categorical bbas, in which case we have $pl_{ik} = l_{ik}$, $\forall i, k$.*

Once the criterion is defined, the remaining work concerns its optimization. The next section presents a variant of the EM algorithm dedicated to this task.

4.2. *EM algorithm for Imprecise and Uncertain Labels*

To build an EM algorithm able to optimize $L(\mathbf{\Psi}; \mathbf{X}^{iu})$, we follow a path that parallels the one recalled in Section 3.2.

At iteration q, our knowledge of the class of example i given the current parameter estimates comes from three sources:

(i) the class label $m_i^{\mathcal{Y}}$ of example i;

(ii) the current estimates $\boldsymbol{\pi}^{(q)}$ of the proportions, which induce a Bayesian bba $m^{\mathcal{Y}}(\cdot|\boldsymbol{\pi}^{(q)})$ such that $m^{\mathcal{Y}}(\{c_k\}|\boldsymbol{\pi}^{(q)}) = \pi_k^{(q)}$;

(iii) vector $\mathbf{x}_i$ and the current parameter estimate $\mathbf{\Psi}^{(q)}$. Using (46) and the GBT (11), we have

$$pl^{\mathcal{Y}_i|\mathcal{X}_i}(\{c_k\}|\mathbf{x}_i, \boldsymbol{\psi}) = pl^{\mathcal{X}_i|\mathcal{Y}_i}(\mathbf{x}_i|c_k, \boldsymbol{\psi}) = f(\mathbf{x}_i; \boldsymbol{\theta}_k)\mathrm{d}x_{i1} \ldots \mathrm{d}x_{ip}.$$

By combining these three items of evidence using Dempster's rule (4), we get a Bayesian bba (since $m^{\mathcal{Y}}(\cdot|\boldsymbol{\pi}^{(q)})$ is Bayesian). Let us denote by $t_{ik}^{(q)}$ the mass assigned to $\{c_k\}$ after combination. We have

$$t_{ik}^{(q)} = \frac{pl_{ik}\pi_k^{(q)} f(\mathbf{x}_i; \boldsymbol{\theta}_k^{(q)})}{\sum_{k'=1}^{K} pl_{ik'}\pi_{k'}^{(q)} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'}^{(q)})}, \tag{51}$$

which is quite similar to (20) and (37).

Using this expression, we may decompose the log-likelihood in two parts, as in (17). This is expressed by the following proposition.

Proposition 4.2

$$L(\mathbf{\Psi}; \mathbf{X}^{iu}) = Q(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}) - H(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}), \tag{52}$$

with:

$$Q(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}) = \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(pl_{ik}\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)\right)$$

$$H(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}) = \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln(t_{ik}).$$

Proof: We have

$$\begin{aligned}
Q(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}) - H(\mathbf{\Psi}, \mathbf{\Psi}^{(q)}) &= \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(pl_{ik}\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)\right) - \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} . \ln(t_{ik}) \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(\frac{pl_{ik}\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)}{pl_{ik}\pi_k f(\mathbf{x}_i; \boldsymbol{\theta}_k)} \sum_{k'=1}^{K} pl_{ik'}\pi_{k'} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'})\right) \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} t_{ik}^{(q)} \ln\left(\sum_{k'=1}^{K} pl_{ik'}\pi_{k'} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'})\right) \\
&= \sum_{i=1}^{N} \ln\left(\prod_{k=1}^{K}\left(\sum_{k'=1}^{K} pl_{ik'}\pi_{k'} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'})\right)^{t_{ik}^{(q)}}\right) \\
&= \sum_{i=1}^{N} \ln\left(\left(\sum_{k'=1}^{K} pl_{ik'}\pi_{k'} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'})\right)^{\sum_{k=1}^{K} t_{ik}^{(q)}}\right) \\
&= \sum_{i=1}^{N} \ln\left(\sum_{k'=1}^{K} pl_{ik'}\pi_{k'} f(\mathbf{x}_i; \boldsymbol{\theta}_{k'})\right) \\
&= L(\mathbf{\Psi}; \mathbf{X}^{iu}).
\end{aligned}$$

□

Therefore, using the same argument as for the classical EM algorithm (Section 3.2), an algorithm which alternates between computing t_{ik} using (51) and maximization of Q will increase the log likelihood. The EM algorithm that performs the optimization of our criterion (49) is therefore the classical EM algorithm, except for the E step, where the posterior distributions t_{ik} are weighted

by the plausibility of each class $pl_{ik} = \sum_{C \cap c_k \neq \emptyset} m_i(C)$. Note that the impact of the soft labels in the EM algorithm has a natural interpretation.

4.3. *Practical Considerations*

The theoretical algorithm presented in the previous section can easily be implemented, with the usual precautions to prevent numerical problems such as storage and manipulations of posterior probabilities t_{ik} on log-scale. The algorithm can readily be adapted to deal with different class conditional distributions. As an example, the pseudo-code for the implementation of our algorithm in the classical case of multidimensional Gaussian mixture is given in Algorithm 1.

Algorithm 1 EM algorithm, pseudo-code for Gaussian mixture models with partial labels. The notation $\phi(., \boldsymbol{\mu}, \boldsymbol{\Sigma})$ stands for the multivariate Gaussian probability density with parameters $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Inputs: $\{\mathbf{x}_i\}_{i=1\ldots N}$, $\{pl_{ik}\}_{i=1\ldots N}^{k=1\ldots K}$

Initialize: $\boldsymbol{\mu}_1^{(0)}, \ldots, \boldsymbol{\mu}_K^{(0)}$; $\boldsymbol{\Sigma}_1^{(0)}, \ldots, \boldsymbol{\Sigma}_K^{(0)}$; $\pi_1^{(0)}, \ldots, \pi_K^{(0)}$

while Increment in log likelihood > precision threshold **do**

`//E step`

$$t_{ik}^{(q)} = \frac{pl_{ik}\pi_k^{(q)}\phi(\mathbf{x}_i; \boldsymbol{\mu}_k^{(q)}, \boldsymbol{\Sigma}_k^{(q)})}{\sum_{k'=1}^{K} pl_{ik'}\pi_{k'}^{(q)}\phi(\mathbf{x}_i; \boldsymbol{\mu}_{k'}^{(q)}, \boldsymbol{\Sigma}_{k'}^{(q)})}$$

`//M Step`

$$\pi_k^{(q+1)} = \frac{1}{N}\sum_{i=1}^{N} t_{ik}^{(q)}$$

$$\boldsymbol{\mu}_k^{(q+1)} = \frac{1}{\sum_{i=1}^{N} t_{ik}^{(q)}}\sum_{i=1}^{N} t_{ik}^{(q)}\mathbf{x}_i$$

$$\boldsymbol{\Sigma}_k^{(q+1)} = \frac{1}{\sum_{i=1}^{N} t_{ik}^{(q)}}\sum_{i=1}^{N} t_{ik}^{(q)}(\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)})(\mathbf{x}_i - \boldsymbol{\mu}_k^{(q+1)})'$$

$q \leftarrow q + 1$

end while

Outputs: $\hat{\boldsymbol{\mu}}_k = \boldsymbol{\mu}_k^{(q)}, \hat{\boldsymbol{\Sigma}}_k = \boldsymbol{\Sigma}_k^{(q)}, \hat{\pi}_k = \pi_k^{(q)}, \quad \forall k \in \{1, \ldots, K\}$

Some issues related to the practical implementation of this algorithm are briefly addressed below.

4.3.1. *Initialization*

A practical advantage of soft labels can be found in their ability to provide an interesting starting point for the algorithm. For this, we can simply compute the pignistic transform [20] of each label m_i to supply the initial values of the posterior probabilities:

$$t_{ik}^{(0)} = \sum_{C : c_k \in C} \frac{m_i^{\mathcal{Y}}(C)}{|C|}.$$

4.3.2. *Convergence test*

The convergence of the algorithm can be checked in different ways. As proposed in the pseudo-code of Algorithm 1, the increment in log likelihood can be monitored and a test such as:

$$\frac{L(\boldsymbol{\Psi}^{(q)}; \mathbf{X}^{iu}) - L(\boldsymbol{\Psi}^{(q-1)}; \mathbf{X}^{iu})}{|L(\boldsymbol{\Psi}^{(q-1)}; \mathbf{X}^{iu})|} < \epsilon, \tag{53}$$

where ϵ is a precision threshold set to a small value (10^{-6}), can be used to check the convergence. Another convergence test can be based on differences between successive estimates of the parameters or latent variables ($t_{ik}^{(q)}$ and $t_{ik}^{(q+1)}$).

4.4. *Discussion*

To conclude this section, the problem of data dimensionality and the time complexity of our method will now be discussed. The improvement over previous work will also be stressed.

4.4.1. *Curse of Dimensionality*

In real applications, it is crucial to take care of data dimensionality. It is well known that Gaussian Mixture Models may perform poorly in high dimensional feature spaces: consequently, similar problems can be expected with our method. Limiting the number of free parameters is a classical way to cope with this problem. In Ref. [50, 51], different parametrizations of the covariance matrix with varying numbers of free parameters are proposed. Another solution to the dimensionality problem was recently introduced in [52, 53], assuming each class to lie in a low dimensional manifold; constraints are then imposed on the spectral decomposition of the covariances matrix. Finally, Bayesian regularization techniques can also be used to circumvent problems encountered by Gaussian Mixture Models in high dimensional feature spaces [54]. Since all of these solutions affect the M step of the EM algorithm, whereas our integration of soft label information impacts the E step, they can easily be used in conjunction with our method.

4.4.2. *Time Complexity*

The complexity of one iteration of our algorithm is exactly the same as in the classical EM algorithm for mixture model. Differences that may arise come from label imprecision, which influences the number of iterations needed to converge and the problem of local maxima. The experimental study presented in Section 5.1.3 investigates the influence of soft label imprecision on these two aspects. In the practical application of our algorithm, the key problem will certainly be the existence of local maxima. The algorithm must therefore be initialized with different starting points in order to find different local maxima and select the best one. Other solutions developed to cope with this problem in unsupervised learning of mixture model such as the *Deterministic Annealing EM* Algorithm [55] can also be interesting.

4.4.3. *Comparison with Previous Work*

As outlined in Section 1, the idea of adapting the EM algorithm to handle soft labels can be traced back to the work of Vannoorenberghe and Smets [26, 27], which was recently extended to categorical data by Jraidi et al. [30]. These authors proposed a variant of the EM algorithm called

CrEM (Credal EM), based on a modification of the auxiliary function $Q(\mathbf{\Psi}, \mathbf{\Psi}^{(q)})$ defined in (19). However, our method differs from this previous approach in several respects:

– First, the CrEM algorithm was not derived as optimizing a generalized likelihood criterion such as (50); consequently, its interpretation was unclear, the relationship with related work (see Remark 4.2) could not be highlighted and, most importantly, the convergence of the algorithm was not proven.
– In our approach, the soft labels $m_i^{\mathcal{Y}}$ appear in the criterion and in the update formulas for posterior probabilities (51) only in the form of the plausibilities pl_{ik} of the singletons (plausibility profiles). In constrast, the CrEM algorithm uses the $2^{|\mathcal{Y}|}$ values in each bba $m_i^{\mathcal{Y}}$. This fact has an important consequence, as the computations involved in the E step of the CrEM algorithm have a complexity in $O(2^{|\mathcal{Y}|})$ whereas our solution only involves calculations which scale with the cardinality of the set of classes, that is in $O(|\mathcal{Y}|)$. For large numbers of classes, this can be expected to have a major influence on running time.
– Finally, no experimental study demonstrating the interest of soft labels was presented in Ref. [26, 27], since only two simulated examples were used to illustrate the approach. The set of experiments presented in the next section will provide a more complete demonstration of the interest of soft labels.

5. Simulations

To better understand the behaviour of our algorithm in the presence of imprecise and uncertain labels, two sets of experiments were carried out. The first one aimed at giving some clues on the influence of soft labels on the complexity of the optimization problem and on estimation accuracy, as compared to unsupervised clustering. We studied the influence of label imprecision on these two aspects. The second set of experiments was designed to show the benefits of soft labels when labels are uncertain. We will show that information on label reliability encoded in soft labels can be exploited to obtain a significant reduction of classification error rates.

5.1. *Influence of Label Imprecision*

The aim of this section is to show the influence of label precision on the accuracy of parameter estimation. During the next experiments, the assumption that the true class has the highest plausibility is made. This assumption will be abandoned in Subsection 5.2, where a solution to deal with label uncertainty will be presented.

5.1.1. *Imprecise and Uncertain Label Generation*

First of all, we must determine how to quantify the *amount of information* on the true class encoded by a soft label. This question is related to the quantification of label imprecision. Several

uncertainty measures have been proposed to quantify the uncertainty in a belief function [56]. Among them, *nonspecificity* (NS) is related to imprecision and has interesting properties [57]. It is defined as:

$$NS(m^{\mathcal{Y}}) = \sum_{C \subseteq \mathcal{Y}} m^{\mathcal{Y}}(C) \ln(|C|). \tag{54}$$

To obtain learning sets with specific properties, we varied the mean nonspecificity ns of label sets from a quasi-supervised setting ($ns = 0.05$) to a quasi unsupervised setting ($ns = 0.95$) in a two class problem, and we compared the results obtained with these different training sets. To generate a training set with fixed mean nonspecificity, we drew randomly, for each learning sample, a nonspecificity value from a uniform distribution in $[ns - 0.05, ns + 0.05]$ and we computed class plausibilities using this value.

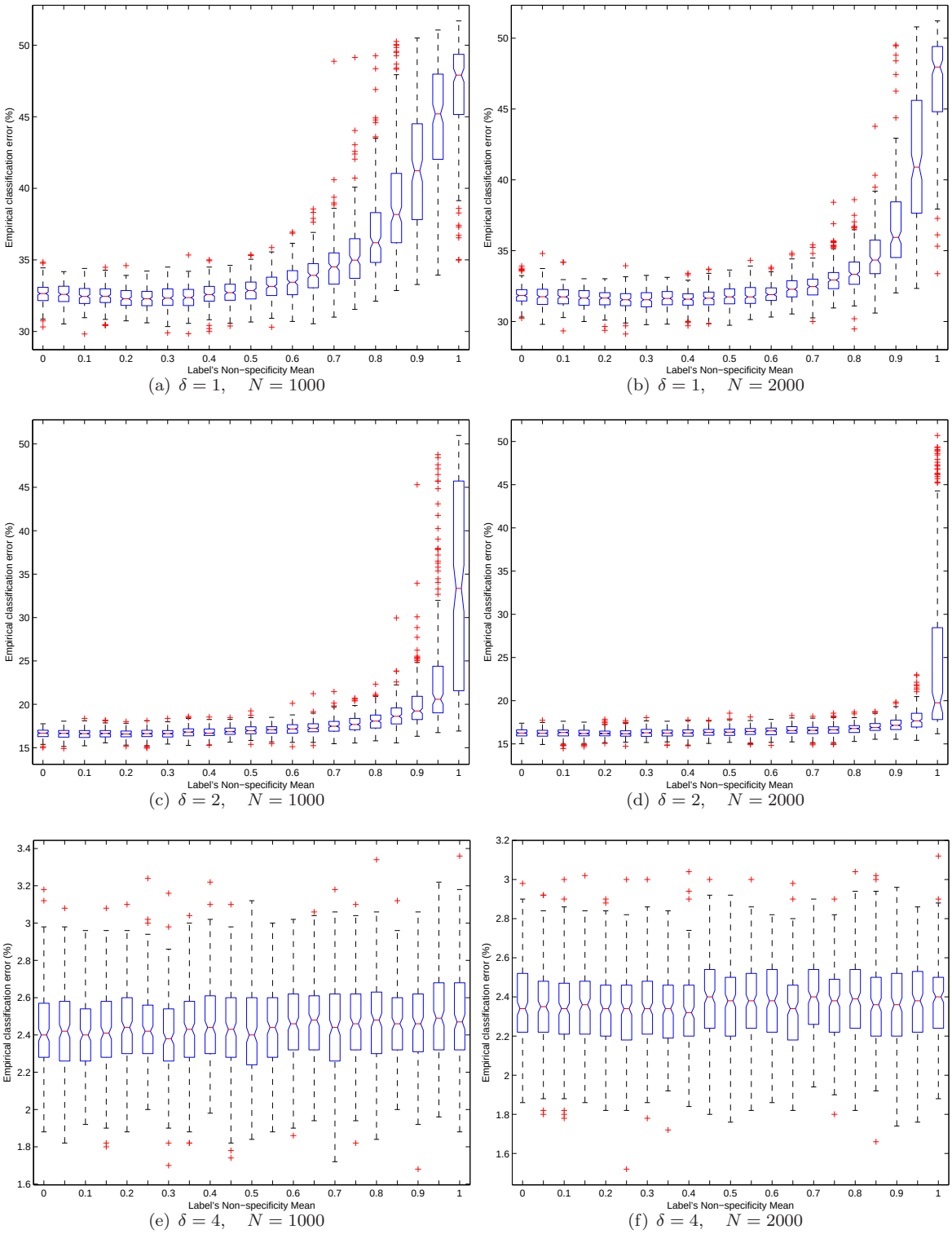
As an example, let us assume that we drew a nonspecificity value equal to $NS = 0.3$, and that the true class of the sample is k^*. As we are only interested in the plausibility of singletons we can constrain belief functions to be consonant; therefore, we have to find the masses assigned to focal sets c_{k^*} and $\mathcal{Y}$, as there are only two elements in the frame. From (54) we find $m^{\mathcal{Y}}(\{c_{k^*}\}) \times 0 + m^{\mathcal{Y}}(\mathcal{Y}) \times 1 = 0.3$, hence $m^{\mathcal{Y}}(\mathcal{Y}) = 0.3$. Using the fact that masses should sum to one, we find the mass of the true class $m^{\mathcal{Y}}(\{c_{k^*}\}) = 1 - 0.3 = 0.7$. Thus, the corresponding plausibilities used as label are equal to $pl_{ik^*} = 1$ and $pl_{ik} = 0.3$ for $k \neq k^*$.

5.1.2. *Influence of Label Imprecision on Estimation Accuracy*

First, we investigated with such principle the influence of labels on estimation accuracy. The following simulations were performed. For each learning set size $N \in \{1000, 2000\}$, N samples in a ten-dimensional feature space were drawn from a two component normal mixture with common identity covariance matrix and balanced proportions. The distance between the two centers $\delta = ||\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2||$ was varied in $\{1, 2, 4\}$, in order to study different cases from strongly overlapping to well separated classes.

For each generated sample, a nonspecificity value NS was drawn randomly from a uniform distribution in $[ns - 0.05, ns + 0.05]$ with $ns \in \{0.05, 0.1, 0.15, \ldots, 0.95\}$ and a soft label with that nonspecificity value was built as explained above. For each learning set size N, mean nonspecificity ns and distance δ between centers, two hundred learning sets were generated. The accuracy measure was the empirical classification error rate of the trained classifier estimated on a test set of 5000 samples. As there is no model bias, differences in empirical classification error are only due to differences in estimation accuracy. Finally, to avoid the problem of local maxima, the EM algorithm was initialized with the parameters of the true distributions: in this way, it is guaranteed to pick among all local maxima of the log-likelihood function the one which is in the basin of attraction of the optimal value.

Figure 2 shows blox-plots of estimated error rates for the 200 learning sets in each configuration (N, ns, δ). It clearly shows the influence of label imprecision when the estimation problem is



(a) $\delta = 1, \quad N = 1000$ (b) $\delta = 1, \quad N = 2000$

(c) $\delta = 2, \quad N = 1000$ (d) $\delta = 2, \quad N = 2000$

(e) $\delta = 4, \quad N = 1000$ (f) $\delta = 4, \quad N = 2000$

Fig. 2. Boxplots (over 200 independent learning sets) of empirical classification error rates (%) as a function of average nonspecificity of learning set labels $ns \in \{0.5, \ldots 0.95\}$, (results of supervised learning $ns = 0$ and unsupervised learning, $ns = 1$ are also shown), for two learning set sizes ($N = 1000$ or $N = 2000$) and different distances between the two centers of the components ($\delta \in \{1, 2, 4\}$).

difficult: when the class overlap is important ($\delta = 1$ or 2) and when the learning set is not too large ($N = 1000$) we can see a significant improvement of supplying partial information on class label. In contrast, when the problem is simple (well separated classes and large number of training samples), we do not gain much from supplying more precise labels. These findings are natural and consistent with theoretical studies on asymptotic efficiency of labelled samples with respect to unlabelled samples [58]. These results show that our solution is able to take advantage of partial information on class labels to enhance estimation accuracy, provided the soft labels are not wrong (in these experiments, the true class always has a plausibility equal to one). We will now see that partial label information also has an effect on the complexity of the optimization problem.

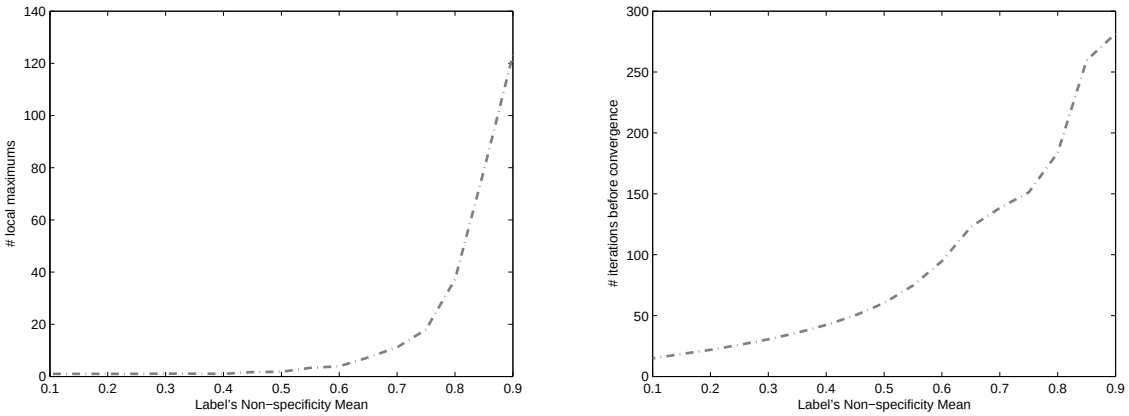
5.1.3. *Influence of Label Precision on the Likelihood Landscape*

The following experiment was designed to show the potential benefit of soft labels in terms of simplification of the optimization problem. Two indicators of optimization problem complexity were investigated: the number of iterations before convergence and the number of different local maxima found.

The behaviour of these two indicators was analyzed with respect to the average nonspecificity of the learning set labels. For that purpose, we drew $N = 1000$ samples from the same mixture distribution as previously with $\delta = 2$, and we simulated different sets of labels with average nonspecificity ranging from 0.1 to 0.9. For all of these learning sets, we ran the EM algorithm from two hundred different random starting points and we counted the number of different local maxima that were found [1], as well as the average number of iterations until convergence. This experiment was repeated ten times with different randomly generated soft labels, and the results were averaged over these ten independent sets. Figure 3 displays these two indicators of optimization problem complexity.

As expected, information on the true class of samples has a great impact on the complexity of the optimization problem. The increase of the number of local maxima seems to be roughly exponential as a function of the nonspecificity of class labels. Furthermore, for $ns < 0.25$ we noted that no switching problem appeared whatever the chosen starting point (the predicted class of any learning sample does not change across the successive runs the EM algorithm). When enough information is available, the problem can be considered as convex since only one maximum is found, as in the limit situation of supervised learning. This consideration has high practical interest as the problem of local maxima is very important in the unsupervised learning context.

[1] Two local maxima were supposed to be distinct if the distance between the vectors containing all the parameters was larger than 0.001.



(a) Number of local maxima (b) Number of iterations before convergence

Fig. 3. (a) Number of local maxima detected over 200 random initializations for the EM algorithm, (b) number of iterations until convergence, as a function of average label nonspecificity.

5.2. *Simulations with Label Noise*

We present here simulation results that highlight the practical interest of soft labels when labelling errors are present. The experiment aimed at using information on class labels simulating expert opinions. As a reasonable setting, we assumed that the expert supplies, for each sample i, his/her more likely label c_k and a measure of doubt p_i. This doubt is represented by a number in $[0, 1]$, which can be seen as the probability that the expert knows nothing about the true label.

To handle this additional information in the belief function framework, it is natural to *discount* the categorical bba associated to the guessed label with a discount rate p_i [19, Page 251]. Thus, the imperfect labels built from expert opinions are simple bbas such that $m_i^{\mathcal{Y}}(\{c_{k^*}\}) = 1 - p_i$ for some k^*, and $m_i^{\mathcal{Y}}(\mathcal{Y}) = p_i$. The corresponding plausibilities are $pl_{ik^*} = 1$ and $pl_{ik} = p_i$ for all $k \neq k^*$. Such labelling can easily be handled by our method.

5.2.1. *Label Simulation*

Simulated and real datasets with known class labels were corrupted as follows: for each training sample i, a number p_i was drawn from a specific probability distribution to define the doubt expressed by a hypothetical expert on the class of that sample. With probability p_i, the label of sample i was changed (to any other class with equal probabilities).

The probability distribution used to draw the p_i specifies the expert's labelling error rate. The expected value of p_i is equal to the asymptotic labelling error rate. For our experiments we used Beta distributions with expected value equal to $\{0.1, 0.15, \ldots, 0.4\}$ and variance kept equal to 0.2.

5.2.2. *Simulated Datasets*

The first results concern simulated data. In that case, there is no modeling bias in these experiments. Four data sets of size $N \in \{500, 1000, 2000, 4000\}$ were generated from a two component

Gaussian mixture with the same parameters as in Section 5.1.2, and with a distance $\delta = 2$ between the two centers. Finally, the labels were corrupted by noise as described previously. The results of our approach with soft labels were compared to:

(i) supervised learning of the Gaussian Mixture Model, using the potentially wrong expert's labels;

(ii) unsupervised learning of the Gaussian Mixture Model, which does not use any information on class label coming from experts;

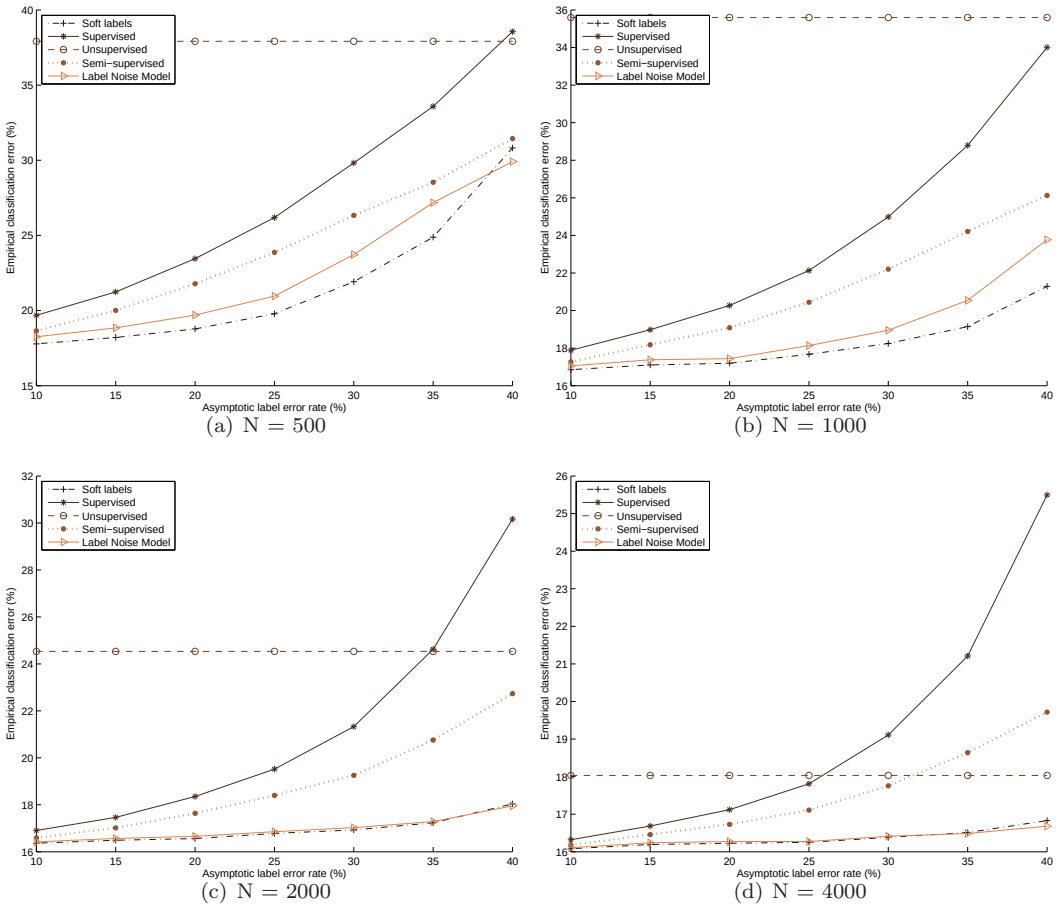
(iii) a strategy based on semi-supervised learning of the Gaussian Mixture Model, which takes into account the reliability of labels supplied by the p_i's. This strategy considers each sample as labeled if the doubt expressed by the expert is moderate ($p_i \leq 0.5$) and as unlabelled otherwise ($p_i > 0.5$). This strategy will be called "adaptive semi-supervised learning";

(iv) the generative probabilistic label noise model introduced in [15, 18], which learns the probabilities of label flipping. This solution to deal with label noise addresses a problem close to this one but did not use information on label reliability coming from the expert. The implementation used was the one proposed in [18], and the mixture model postulated was the same as for all methods, one cluster per class and full covariance matrix.

It should be noted that methods (i)-(iii) above are based on the same EM algorithm as our method, with different class labels. Method (iv) is based on the same generative model, but a specific EM algorithm, as explained in [18].

Table 1 and Figure 4 show the performances of the different classifiers trained with these learning sets. The error rates of the different classifiers were estimated on a test set of 5000 observations according to their real classes, the results were averaged over one hundred randomly chosen independent training sets. For all methods, the EM algorithm was initialized with the true parameter values. A set of paired, two-sided sign test were performed between the best method (according to empirical classification error) and all other methods. If the null hypothesis (zero median for the difference between the two input distributions) was rejected for all the tests at the 5% significance level, this method was considered to perform significantly better than the others. The corresponding error rates are printed in bold in Table 1.

As expected, when the expert's labelling error rate increases, the error rate of supervised learning also increases. Our solution based on soft labels does not suffer as much as supervised learning and adaptive semi-supervised learning from label noise. Whatever the dataset size, our solution takes advantage of additional information on the reliability of labels to keep good performances. We can notice that, for $N \in \{1000, 2000, 4000\}$, the results are almost stable even if the expert's labelling error rate increases. Finally, our approach clearly outperforms unsupervised learning, except when the number of samples is high ($N = 4000$). The comparison with the generative label noise model is also in favor of soft labels when the number of training patterns is low and the results are similar for $N \in \{2000, 4000\}$.



(a) N = 500 (b) N = 1000

(c) N = 2000 (d) N = 4000

Fig. 4. Empirical classification error rates estimated by ten-fold cross validation (%) averaged over 100 independent label sets, as a function of the asymptotic error rate of the training labels (%), and for different sample sizes. The results were obtained using soft labels with discounting (-.+), supervised learning (-∗), unsupervised learning (- -o), adaptive semi-supervised learning (..•) and the probabilistic label noise model (-▷).

To conclude about this experiment, our solution outperforms both supervised learning, which is an hazardous approach in this context as some labels are wrong, and unsupervised learning that is a conservative approach because information coming from experts is discarded as it is potentially wrong. It also outperforms semi-supervised learning which makes some errors when choosing between considering points as labelled or not. Finally, for small sample size, our approach yields better results than the probabilistic label noise model. This experiment has also proved the ability of belief function based labels to carry useful information even if some labels are partially wrong (the real class does not have a plausibility equal to one).

Table 1

Empirical classification error rates (%) averaged over one hundred independent datasets, for different asymptotic labelling error rates (%) and learning set sizes. Error rates in bold are significantly lower than other error rates at the 5 % level according to paired two-sided sign tests.

		Expert asymptotic error rate (%)						
		10	*15*	*20*	*25*	*30*	*35*	*40*
$N = 500$	Soft labels	**17.8**	**18.2**	**18.8**	**19.8**	**21.9**	**24.9**	30.8
	Supervised	19.7	21.2	23.4	26.2	29.8	33.6	38.6
	Unsupervised	37.9	37.9	38.0	37.9	37.9	37.9	37.9
	Semi-Supervised	18.7	20.0	21.8	23.9	26.3	28.5	31.4
	Label Noise Model	18.3	18.8	19.7	21.0	23.7	27.2	29.9
$N = 1000$	Soft labels	**16.9**	**17.1**	**17.2**	**17.7**	**18.2**	**19.1**	**21.3**
	Supervised	17.9	19.0	20.3	22.1	25.0	28.8	34.0
	Unsupervised	35.6	35.6	35.6	35.6	35.6	35.7	35.5
	Semi-Supervised	17.3	18.2	19.1	20.4	22.2	24.2	26.1
	Label Noise Model	17.1	17.4	17.4	18.1	19.0	20.5	23.8
$N = 2000$	Soft labels	**16.4**	**16.5**	**16.6**	16.8	**16.9**	17.2	18.0
	Supervised	16.9	17.5	18.4	19.5	21.3	24.6	30.2
	Unsupervised	24.5	24.5	24.5	24.6	24.5	24.5	24.5
	Semi-Supervised	16.6	17.0	17.6	18.4	19.3	20.8	22.7
	Label Noise Model	16.4	16.6	16.7	16.9	17.0	17.3	18.0
$N = 4000$	Soft labels	16.1	**16.2**	16.2	16.3	16.4	16.5	16.8
	Supervised	16.3	16.7	17.1	17.8	19.1	21.2	25.5
	Unsupervised	18.0	18.1	18.1	18.1	18.0	18.0	18.0
	Semi-Supervised	16.2	16.5	16.7	17.1	17.8	18.6	19.7
	Label Noise Model	16.1	16.2	16.3	16.3	16.4	16.5	**16.7**

5.2.3. *Real datasets*

We also investigated the interest of our approach with real datasets. In that case, modeling bias may arise. We used well known datasets available on-line [2] [59], with the characteristics summarized in Table 2. The first dataset is the well known Fisher's *Iris* data. The *Crabs* dataset concerns the recognition of crabs species and sexes in a population of crabs using different morphological measurements. The *Wine* dataset contains the results of a chemical analysis of wines grown in a specific area of Italy. Lastly, the *Breast Cancer Wisconsin* dataset deals with the recognition of breast tumor from 30 features extracted from digitized images of fine needle aspirate (FNA) of breast mass. The features describe the characteristics of the cell nuclei present in the image and the task is to determine if the tumor is malignant or benign.

No preprocessing was performed except the classical centering and variance normalization. The same process for soft label generation as above was used as for simulated data. The classification error was estimated by ten-fold cross-validation, therefore 9/10 of the samples were used for training

[2] http://mlearn.ics.uci.edu/MLRepository.html and http://rweb.stat.umn.edu/R/library/MASS/html/crabs.html.

Table 2

Characteristics of real datasets.

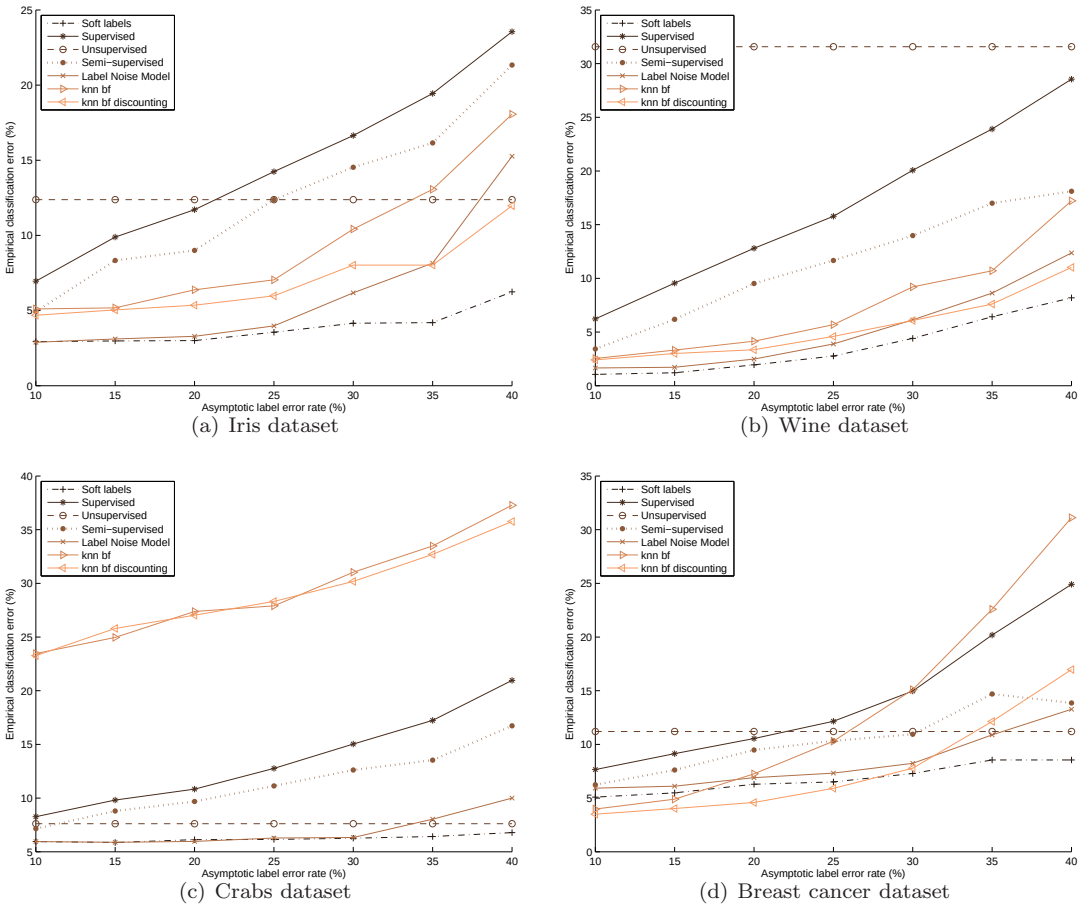
name	# dimensions	# samples	# classes
Iris	4	150	3
Crabs	5	200	4
Wine	13	178	3
Breast Cancer Wisconsin	30	569	2

with noisy labels and the remaining samples were used to estimate classification error using their real classes. Finally, we sampled thirty independent label sets and computed the mean classification error rate over these sets. To place the different strategies in a realistic context, we used for the each variant of the EM algorithm an initialization procedure compatible with the information available by this method. As no class information is available for unsupervised learning, we started from one hundred different random initializations [3] . To solve the label switching problem resulting from these random initializations, we computed the classification error according to the best class permutations. For soft labels and semi-supervised learning the initialization was based on the pignistic transformation of soft labels as explained in Section 4.3. Therefore, only one starting point was used for semi-supervised learning and for soft labels. For these real datasets we also compared our approach with a non parametric approach that can also deal with soft labels, the evidential k nearest neighbor rule proposed in [10, 21]. The results of this method are supplied using the crisp labels given by the virtual expert and the discounted labels. The number of neighbors was fixed a priori to 10.

As for simulated data, the ability of soft labels to adequately account for label noise is visible in Table 3 and Figure 5. For all the problems, results are quite stable even if the expert error rate increases, which means that information on label reliability is correctly exploited to retain good performances. We can also notice that, as with simulated data, the solution based on soft labels significantly outperforms both supervised learning and adaptive semi-supervised learning when the expert error rate increases. It also outperforms unsupervised learning, which is probably affected in some cases by the problem of local maxima. Results of our method are similar to those of label noise model for low label noise level but important differences in favor of soft labels appeared for higher error rates. The k nearest neighbor approach gives also interesting results, except for the dataset *Crabs*, where this method fails. It yields the best results for the *Breast Cancer Wisconsin* dataset for low label noise level and gives good results for the *Wine* and *Iris* datasets even if mixture models with soft labels perform better. Finally we may note that the solution based on mixture modesl outperforms all the other methods in 14 experiments over 28.

In the different experiments, the solution based on discounting expert labels according to their reliability to produce soft labels has demonstrated its ability to yield good parameter estimates even

[3] The centers were drawn according to a Gaussian distribution estimated on the whole population, the covariance matrix were all initialized with the one estimated on the whole dataset and the proportions were set to be equal.



(a) Iris dataset (b) Wine dataset

(c) Crabs dataset (d) Breast cancer dataset

Fig. 5. Empirical classification error rates estimated by ten-fold cross validation (%) averaged over thirty independent label sets, as a function of the asymptotic error rate of the training labels (%), and for different real datasets. The results were obtained using soft labels with discounting (-.+), supervised learning (-∗), unsupervised learning (- -o), adaptive semi-supervised learning (..•) and probabilistic label noise model (-▷).

if the expert's labelling error rate is high. We may conclude that information on label reliability is useful in the context of label noise, and that imprecise and uncertain labels are suitable to deal with such additional information.

6. Conclusions

The approach presented in this paper, based on concepts coming from maximum likelihood estimation and belief function theory, offers an interesting way to deal with imperfect and imprecise labels. It assumes that the data are generated by a mixture model and that the class labels are belief functions. In this context, a likelihood criterion was defined and an EM algorithm dedicated to its optimization was presented. Simulations have shown that, even if the information on class labels is partial, its exploitation with our approach may significantly improve the estimation accuracy and simplify the optimization problem. Moreover, the practical interest of imprecise and imperfect labels, as a solution to deal with label noise, has been highlighted by an experimental study using

Table 3

Empirical classification error rates estimated by ten-fold cross validation (%) over thirty independent label sets, with respect to expert's asymptotic error rate on training labels, and for different real problem. Error rates in bold are significantly lower than other error rates at the 5 % level according to paired two-sided sign tests.

		Expert asymptotic error rate (%)						
		10	*15*	*20*	*25*	*30*	*35*	*40*
Iris	Soft labels	2.9	3.0	3.0	3.6	**4.2**	**4.2**	**6.2**
	Supervised	7.0	9.9	11.7	14.2	16.6	19.4	23.6
	Unsupervised	12.4	12.4	12.4	12.4	12.4	12.4	12.4
	Semi-Supervised	4.9	8.3	9.0	12.4	14.5	16.2	21.3
	Label Noise Model	2.9	3.1	3.3	4.0	6.2	8.2	15.3
	k-nn TBM	5.1	5.2	6.4	7.0	10.4	13.1	18.1
	k-nn TBM discounting	4.7	5.0	5.4	6.0	8.0	8.0	12.0
Wine	Soft labels	1.1	**1.2**	**1.9**	**2.8**	**4.4**	**6.4**	**8.2**
	Supervised	6.2	9.6	12.8	15.8	20.1	23.9	28.6
	Unsupervised	31.6	31.6	31.6	31.6	31.6	31.6	31.6
	Semi-Supervised	3.4	6.2	9.5	11.7	14.0	17.0	18.1
	Label Noise Model	1.6	1.7	2.5	3.9	6.1	8.6	12.4
	k-nn TBM	2.5	3.3	4.1	5.7	9.2	10.7	17.2
	k-nn TBM discounting	2.4	3.0	3.4	4.6	6.1	7.6	11.0
Crabs	Soft labels	6.0	5.9	6.1	6.2	6.3	**6.4**	**6.8**
	Supervised	8.3	9.8	10.8	12.8	15.0	17.2	21.0
	Unsupervised	7.6	7.6	7.6	7.6	7.6	7.6	7.6
	Semi-Supervised	7.2	8.8	9.7	11.1	12.6	13.5	16.7
	Label Noise Model	6.0	5.9	6.0	6.3	6.3	8.0	10.0
	k-nn TBM	23.5	25.0	27.4	27.9	31.0	33.5	37.3
	k-nn TBM discounting	23.3	25.8	27.0	28.3	30.2	32.7	35.8
Breast Cancer Wisconsin	Soft labels	5.1	5.5	6.3	6.5	**7.3**	**8.5**	**8.5**
	Supervised	7.7	9.1	10.5	12.2	15.0	20.2	24.9
	Unsupervised	11.2	11.2	11.2	11.2	11.2	11.2	11.2
	Semi-Supervised	6.2	7.6	9.5	10.3	10.9	14.7	13.9
	Label Noise Model	5.9	6.1	6.9	7.3	8.2	10.9	13.3
	k-nn TBM	4.0	4.9	7.3	10.3	15.1	22.6	31.1
	k-nn TBM discounting	**3.5**	**4.0**	**4.6**	**5.9**	7.8	12.1	17.0

both real and experimental data.

The proposed criterion has a natural expression that is closely related to previous solutions found in the context of probabilistic models, and also has a clear and justified origin in the context of belief functions. Therefore, we hope that the link established by this study between probabilistic methods and belief function-based approaches will be extended to other problems. Furthermore, the long history of mixture models is an important advantage for our approach since a lot of tools developed in the context of probabilistic models, such as the Bayesian Information Criterion or other criteria for model selection, can easily be adapted to our approach.

We are working now on the application of such approach for the diagnosis of a railway signalling system [60]. Soft labels seem to be appropriate to deal with the imprecise knowledge that experts of this area have on the monitoring data of this complex system.

References

[1] 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 sample, Biometrics 29 (1973) 761–770.

[2] 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.

[3] O. Chapelle, B. Schölkopf, A. Zien (Eds.), Semi-Supervised Learning, MIT Press, Cambridge, Ma, 2006.

[4] Y. Bengio, Y. Grandvalet, Semi-supervised learning by entropy minimization, in: Advances in Neural Information Processing Systems 17, MIT Press, 2005, pp. 529–536.

[5] A. Corduneanu, T. Jaakkola, On information regularization, in: Uncertainty in Artificial Intelligence, 2003, pp. 151–158.

[6] T. Joachims, Transductive inference for text classification using support vector machine, in: Sixteenth International Conference on Machine Learning, Morgan Kaufmann, 1999, pp. 202–209.

[7] T. D. Bie, Semi-supervised learning based on kernel methods and graph cut algorithms, Phd thesis, K.U.Leuven (Leuven, Belgium), Faculty of Engineering (May 2005).

[8] D. Zhou, O. Bousquet, T. Lal, B. Schölkopf, Learning with local and global consistency, in: Advances in Neural Information Processing Systems 16, 2004, pp. 321–328.

[9] X. Zhu, Z. Ghahramani, Learning from labelled and unlabelled data with label propagation, Tech. rep., Carnegie Mellon University (2002).

[10] T. Denœux, A k-nearest neighbor classification rule based on Dempster-Shafer theory, IEEE Trans. on Systems, Man and Cybernetics 25 (05) (1995) 804–813.

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

[12] 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.

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

[14] 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.

[15] N. D. Lawrence, B. Schölkopf, Estimating a kernel fisher discriminant in the presence of label noise, in: Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA, 2001, pp. 306–313.

[16] R. Amini, P. Gallinari, Semi-supervised learning with an imperfect supervisor, Knowl. Inf. Syst. 8 (4) (2005) 385–413.

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

[18] Y. Li, W. Lodeswyk, D. De Ridder, M. Reinders, Classification in the presence of class noise using a probabilistic kernel fisher method, Pattern Recognition 40 (2007) 3349–3357.

[19] G. Shafer, A mathematical theory of evidence, Princeton University Press, Princeton, N.J., 1976.

[20] P. Smets, R. Kennes, The Transferable Belief Model, Artificial Intelligence 66 (1994) 191–243.

[21] T. Denœux, L. M. Zouhal, Handling possibilistic labels in pattern classification using evidential reasoning, Fuzzy Sets and Systems 122 (3) (2001) 47–62.

[22] Z. Elouedi, K. Mellouli, P. Smets, Belief decision trees: Theoretical foundations, International Journal of Approximate Reasoning 28 (2001) 91–124.

[23] S. Trabelsi, Z. Elouedi, K. Mellouli, Pruning belief decision tree methods in averaging and conjunctive approaches, International Journal of Approximate Reasoning 46 (3) (2007) 568–595.

[24] I. Jenhani, N. Ben Amor, Z. Elouedi, Decision trees as possibilistic classifiers, International Journal of Approximate ReasoningDoi:10.1016/j.ijar.2007.12.002.

[25] A. Ben Yaghlane, T. Denœux, K. Mellouli, Elicitation of expert opinions for constructing belief functions, in: Proceedings of the 11th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU '06), Vol. I, 2006, pp. 403–411.

[26] P. Vannoorenberghe, P. Smets, Partially supervised learning by a credal EM approach, in: L. Godo (Ed.), Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU '05), Springer, Barcelona, Spain, 2005, pp. 956–967.

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

[28] G. J. McLachlan, D. Peel, Finite Mixture Models, Wiley, New York, 2000.

[29] K. Nigam, A. McCallum, S. Thrun, T. Mitchell, Text classification from labelled and unlabelled documents using EM, Machine Learning 39 (2-3) (2000) 103–134.

[30] I. Jraidi, Z. Elouedi, Belief classification approach based on generalized credal EM, in: K. Mellouli (Ed.), 9th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU '07), Springer, Hammamet, Tunisia, 2007, pp. 524–535.

[31] A. P. Dempster, Upper and lower probabilities induced by a multivalued mapping, Annals of

Mathematical Statistics 38 (1967) 325–339.

[32] D. Dubois, H. Prade, On the unicity of Dempster's rule of combination, International Journal of Intelligent Systems 1 (1986) 133–142.

[33] P. Smets, Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem, International Journal of Approximate Reasoning 9 (1993) 1–35.

[34] B. Ben Yaghlane, P. Smets, K. Mellouli, Belief function independence: I. the marginal case, International Journal of Approximate Reasoning 29 (2002) 47–70.

[35] B. Ben Yaghlane, P. Smets, K. Mellouli, Belief function independence: II. the conditional case, International Journal of Approximate Reasoning 31 (2002) 31–75.

[36] P. Smets, Belief functions on real numbers, International Journal of Approximate Reasoning 40 (3) (2005) 181–223.

[37] F. Caron, B. Ristic, E. Duflos, P. Vanheeghe, Least committed basic belief density induced by a multivariate gaussian: Formulation with applications, International Journal of Approximate Reasoning (in press)Doi:10.1016/j.ijar.2006.10.003.

[38] A. Aregui, T. Denœux, Novelty detection in the belief functions framework, in: Proceedings of IPMU '06, Vol. 1, Paris, 2006, pp. 412–419.

[39] P. Smets, Un modèle mathématico-statistique simulant le processus du diagnostic médical, Ph.D. thesis, Université Libre de Bruxelles, Brussels, Belgium, (in French) (1978).

[40] F. Delmotte, P. Smets, Target identification based on the Transferable Belief Model interpretation of Dempster-Shafer model, IEEE Transactions on Systems, Man and Cybernetics A 34 (4) (2004) 457–471.

[41] T. Denœux, P. Smets, Classification using belief functions: the relationship between the case-based and model-based approaches, IEEE Transactions on Systems, Man and Cybernetics B 36 (6) (2006) 1395–1406.

[42] 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.

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

[44] P. Smets, Possibilistic inference from statistical data, in: Second World Conference on Mathematics at the service of Man, Universidad Politecnica de Las Palmas, 1982, pp. 611–613.

[45] P. Walley, S. Moral, Upper probabilities based on the likelihood function, Journal of the Royal Statistical Society B 161 (1999) 831–847.

[46] P. P. Shenoy, P. H. Giang, Decision making on the sole basis of statistical likelihood, Artificial Intelligence 165 (2) (2005) 137–163.

[47] P. Smets, Numerical representation of uncertainty, in: D. M. Gabbay, P. Smets (Eds.), Handbook of Defeasible reasoning and uncertainty management systems, Vol. 3, Kluwer Academic Publishers, Dordrecht, 1998, pp. 265–309.

[48] P.-A. Monney, A Mathematical Theory of Arguments for Statistical Evidence, Contributions

to Statistics, Physica-Verlag, Heidelberg, 2003.

[49] B. R. Cobb, P. P. Shenoy, On the plausibility transformation method for translating belief function models to probability models, International Journal of Approximate Reasoning 41 (3) (2006) 314–330.

[50] J. Banfield, A. Raftery, Model-based gaussian and non-gaussian clustering, Biometrics 49 (1993) 803–821.

[51] G. Celeux, G. Govaert, Gaussian parsimonious clustering models, Pattern Recognition 28 (5) (1995) 781–793.

[52] C. Bouveyron, S. Girard, C. Schmid, High-dimensional data clustering, Computational Statistics and Data Analysis 52 (2007) 502–519.

[53] C. Bouveyron, S. Girard, C. Schmid, High dimensional discriminant analysis, Communications in Statistics: Theory and Methods 36 (2007) 2596–2607.

[54] C. Fraley, A. Raftery, Bayesian regularization for normal mixture estimation and model-based clustering, Journal of Classification 24 (2) (2007) 155–181.
URL http://dx.doi.org/10.1007/s00357-007-0004-5

[55] N. Ueda, R. Nakano, Deterministic annealing variant of the EM algorithm, in: Advances in Neural Information Processing Systems 7, 1995, pp. 545–552.

[56] G. J. Klir, M. J. Wierman, Uncertainty-Based Information. Elements of Generalized Information Theory, Springer-Verlag, New-York, 1998.

[57] A. Ramer, Uniqueness of information measure in the theory of evidence, Fuzzy Sets and Systems 24 (1987) 183–196.

[58] T. O'Neill, Normal discrimination with unclassified observations, Journal of the American Statistical Association 73 (364) (1978) 821–826.

[59] P. M. Murphy, D. W. Aha, UCI Repository of machine learning databases [Machine-readable data repository], University of California, Department of Information and Computer Science., Irvine, CA, 1994.

[60] E. Côme, L. Oukhellou, and P. Aknin, Diagnosis of complex system by combined use of RKCCA and graphical model, in : CCKM'06 Workshop on Current Challenge in Kernel Methods, Brussels, 2006.