Variational Learning for Finite Dirichlet Mixture Models and Applications

Wentao Fan, Nizar Bouguila, *Senior Member, IEEE*, and Djemel Ziou

***Abstract*— In this paper, we focus on the variational learning of finite Dirichlet mixture models. Compared to other algorithms that are commonly used for mixture models (such as expectation-maximization), our approach has several advantages: first, the problem of over-fitting is prevented; furthermore, the complexity of the mixture model (i.e., the number of components) can be determined automatically and simultaneously with the parameters estimation as part of the Bayesian inference procedure; finally, since the whole inference process is analytically tractable with closed-form solutions, it may scale well to large applications. Both synthetic and real data, generated from real-life challenging applications namely image databases categorization and anomaly intrusion detection, are experimented to verify the effectiveness of the proposed approach.**

***Index Terms*— Bayesian estimation, dirichlet distribution, factorized approximation, image databases, intrusion detection, mixture models, unsupervised learning, variational inference.**

I. Introduction

THE problem of learning appropriate statistical models is a fundamental data analysis task, which has been the topic of continuing interest [1]. Much of the early work has focused on finite mixture models, which are widely used in different disciplines, such as machine learning, statistical modeling, computer vision and pattern recognition [2], [3]. Indeed, in recent years there has been a flurry of works on learning finite mixture models from data where the main goal is to provide a concise structure for efficient inference that avoids both over-fitting and under-fitting. The majority of finite mixture learning approaches consider either deterministic or Bayesian techniques [2]. Deterministic techniques aim at optimizing the model likelihood function, are generally implemented within the expectation-maximization (EM) framework, and are well documented [4], [5]. On the other hand, Bayesian techniques have been proposed to avoid drawbacks related to deterministic techniques, such as their suboptimal generalization performance, dependency on initialization, over-fitting and noise-level under-estimation problems of classic likelihood-based inference [6]–[8]. These drawbacks are avoided via the incorporation of prior knowledge (or belief) in a principled way and then marginalizing over parameter uncertainty. Bayesian methods [9], [10] have considered either Laplace's approximation [11] or Markov chain Monte Carlo (MCMC) simulation techniques [12], [13]. While MCMC techniques are computationally expensive, Laplace's approximation is generally imprecise, since it is based on the strong assumption that the likelihood function is unimodal, which is not generally the case for finite mixtures of distributions [14]. Thus, the variational Bayesian framework has been proposed, and has been widely used as an efficient alternative and as a more controllable way to approximate Bayesian learning. The variational learning approach was introduced in the context of the multilayer perceptron in [15] where it was called ensemble learning and developed further in [16]–[18]. The main idea is to approximate the model posterior distribution by minimizing the Kullback–Leibler (KL) divergence between the exact (or true) posterior and an approximating distribution.

The variational framework has received a lot of attention, and has provided good generalization performance and computational tractability in various applications including finite mixtures learning [19]–[29]. For instance, the authors in [30]–[33] have developed comprehensive frameworks for variational learning, in the case of Gaussian mixture models, which have been shown to provide the better parameter estimates than the maximum likelihood (ML) approach. Although finite Gaussian mixtures have been widely used due to their approximation properties, recent works have shown that other models may provide better modeling capabilities in the case of non-Gaussian data. For instance, we have shown in our previous research efforts that the Dirichlet mixture can be a better alternative in several applications especially those involving proportional data [34], [35]. These works have been performed using ML estimation within an EM-based framework. In this paper, we go a step further by proposing a variational Bayes framework, for finite Dirichlet mixture models, which provides analytical calculations of posterior distributions using an approximation to conjugate priors.

A challenging key task, when considering finite mixture models, is the automatic determination of a suitable and accurate number of components. It is well known that the ML alone fails to account for model complexity and has to be considered in conjunction with model selection criteria [e.g., minimum description length, Bayesian information criterion (BIC), minimum message length, Akaike's information

Manuscript received March 20, 2011; revised February 22, 2012; accepted February 26, 2012. Date of publication March 22, 2012; date of current version May 2, 2012. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

W. Fan is with the Department of Electrical and Computer Engineering, Concordia University, Montreal, QC H3G 1T7, Canada (e-mail: wenta_fa@encs.concordia.ca).

N. Bouguila is with the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC H3G 1T7, Canada (e-mail: bouguila@ciise.concordia.ca).

D. Ziou is with the Département d'Informatique, Université de Sherbrooke, Sherbrooke, QC J1K 2R1, Canada (e-mail: djemel.ziou@usherbrooke.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2012.2190298

criterion, etc.,] to determine the number of components [2], [35]. This approach requires the evaluation of a given selection criterion for several numbers of mixture components, which is time consuming ([35], for more details and discussions about model selection in the case of finite Dirichlet mixtures). Thus, a significant bonus of the variational framework that we will propose is that, apart from the elegant way to estimate the parameters, the number of components is automatically determined as part of the Bayesian inference procedure.

The outline of the rest of this paper is the following. In Section II, we briefly review the finite Dirichlet mixture model. In Section III, we present the variational learning framework. The experimental results are shown in Section IV. We conclude with a few summary remarks in Section V.

II. Finite Dirichlet Mixture Model

A finite mixture of Dirichlet distributions with M components is represented by [34]

$$p(\vec{X}|\vec{\pi},\vec{\alpha}) = \sum_{j=1}^{M} \pi_j \mathrm{Dir}(\vec{X}|\vec{\alpha}_j) \tag{1}$$

where $\vec{\pi} = (\pi_1, \ldots, \pi_M)$ denotes the mixing coefficients, which are positive and sum to one, $\sum_{j=1}^{M} \pi_j = 1$, $\vec{\alpha} = (\vec{\alpha}_1, \ldots, \vec{\alpha}_M)$, and $\mathrm{Dir}(\vec{X}|\vec{\alpha}_j)$ is the Dirichlet distribution of component j with its own positive parameters $\vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jD})$

$$\mathrm{Dir}(\vec{X}|\vec{\alpha}_j) = \frac{\Gamma\left(\sum_{l=1}^{D} \alpha_{jl}\right)}{\prod_{l=1}^{D} \Gamma(\alpha_{jl})} \prod_{l=1}^{D} X_l^{\alpha_{jl}-1} \tag{2}$$

where $\vec{X} = (X_1, \ldots, X_D)$, D is the dimensionality of $\vec{X}$ and $\sum_{l=1}^{D} X_l = 1$, $0 \le X_l \le 1$ for $l = 1, \ldots, D$. It is noteworthy that the Dirichlet distribution is used here as a parent distribution to model directly the data and not as a prior to the multinomial.

Consider a set of N independent identically distributed vectors $\mathcal{X} = \{\vec{X}_1, \ldots, \vec{X}_N\}$ assumed to be generated from the mixture distribution in 1. The likelihood function of the Dirichlet mixture model is given by

$$p(\mathcal{X}|\vec{\pi},\vec{\alpha}) = \prod_{i=1}^{N} \left\{ \sum_{j=1}^{M} \pi_j \mathrm{Dir}(\vec{X}_i|\vec{\alpha}_j) \right\}. \tag{3}$$

It is convenient to interpret the finite mixture model in 1 as a latent variable model. Thus, for each vector $\vec{X}_i$, we introduce a M-dimensional binary random vector $\vec{Z}_i = \{Z_{i1}, \ldots, Z_{iM}\}$, such that $Z_{ij} \in \{0, 1\}$, $\sum_{j=1}^{M} Z_{ij} = 1$ and $Z_{ij} = 1$ if $\vec{X}_i$ belong to component j and 0, otherwise. For the latent variables $\mathcal{Z} = \{\vec{Z}_1, \ldots, \vec{Z}_N\}$, which are actually hidden variables that do not appear explicitly in the model, the conditional distribution of $\mathcal{Z}$ given the mixing coefficients $\vec{\pi}$ is defined as

$$p(\mathcal{Z}|\vec{\pi}) = \prod_{i=1}^{N} \prod_{j=1}^{M} \pi_j^{Z_{ij}}. \tag{4}$$

Then, the likelihood function with latent variables, which is actually the conditional distribution of data set $\mathcal{X}$ given the class labels $\mathcal{Z}$ can be written as

$$p(\mathcal{X}|\mathcal{Z},\vec{\alpha}) = \prod_{i=1}^{N} \prod_{j=1}^{M} \mathrm{Dir}(\vec{X}_i|\vec{\alpha}_j)^{Z_{ij}}. \tag{5}$$

Given the data set $\mathcal{X}$, an important problem is the learning of the mixture parameters. By learning, we mean both the estimation of the parameters and the selection of the number of components M. In the following, we describe a variational Bayesian approach, for finite Dirichlet mixture models, that can handle these two issues simultaneously.

III. Variational Learning

A. Variational Approximation

In order to estimate the parameters of the finite Dirichlet mixture model and to select the number of components correctly, we adopt the variational inference methodology proposed in [32] for finite Gaussian mixtures. The main idea of this framework is based on the estimation of the mixing coefficients $\vec{\pi}$ by maximizing the ML $p(\mathcal{X}|\vec{\pi})$ given by

$$p(\mathcal{X}|\vec{\pi}) = \sum_{\mathcal{Z}} \int p(\mathcal{X}, \mathcal{Z}, \vec{\alpha}|\vec{\pi}) d\vec{\alpha} \tag{6}$$

where $p(\mathcal{X}, \mathcal{Z}, \vec{\alpha}|\vec{\pi})$ is the joint distribution of all the mixture model random variables conditioned on the mixing coefficients as

$$p(\mathcal{X}, \mathcal{Z}, \vec{\alpha}|\vec{\pi}) = p(\mathcal{X}|\mathcal{Z},\vec{\alpha}) p(\mathcal{Z}|\vec{\pi})\, p(\vec{\alpha}). \tag{7}$$

An important step now is to define a conjugate prior $p(\vec{\alpha})$ over the $\vec{\alpha}$ parameters. Since the Dirichlet belongs to the exponential family of distributions [36], a conjugate prior can be derived as follows [37]:

$$p(\vec{\alpha}_j) = f(\nu,\lambda) \left[\frac{\Gamma\left(\sum_{l=1}^{D} \alpha_{jl}\right)}{\prod_{l=1}^{D} \Gamma(\alpha_{jl})} \right]^{\nu} \prod_{l=1}^{D} e^{-\lambda_l(\alpha_{jl}-1)} \tag{8}$$

where $f(\nu, \lambda)$ is a normalization coefficient and (ν, λ) are hyperparameters. Unfortunately, this formal conjugate prior for the Dirichlet distribution is intractable, mainly because of the difficulty to evaluate the normalization coefficient, and cannot be applied for the variational inference directly as will be clearer later. We decided, *faut de mieux*, to tackle this problem in a similar way to [38] where the authors proposed a conjugate prior for the Beta distribution (i.e., 1-D Dirichlet) within a variational framework. We assume that the Dirichlet parameters are statistically independent and for each parameter α_{jl}, the Gamma distribution is adopted to approximate the conjugate prior

$$p(\alpha_{jl}) = \mathcal{G}(\alpha_{jl}|u_{jl}, v_{jl}) = \frac{v_{jl}^{u_{jl}}}{\Gamma(u_{jl})} \alpha_{jl}^{u_{jl}-1} e^{-v_{jl}\alpha_{jl}} \tag{9}$$

where $\boldsymbol{u} = \{u_{jl}\}$ and $\boldsymbol{v} = \{v_{jl}\}$ are hyperparameters, subject to the constraints $u_{jl} > 0$ and $v_{jl} > 0$. Therefore

$$p(\vec{\alpha}) = \prod_{j=1}^{M} \prod_{l=1}^{D} p(\alpha_{jl}). \tag{10}$$

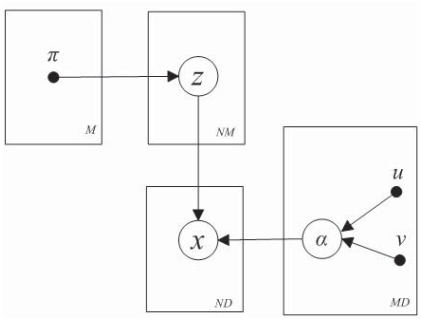


Fig. 1. Graphical model representation of the finite Dirichlet mixture. Symbols in circles denote random variables; otherwise, they denote model parameters. Plates indicate repetition (with the number of repetitions in the lower right), and arcs describe conditional dependencies between variables.

By substituting (4), (5), and (10) into (7), we obtain the joint distribution of all the random variables, conditioned on the mixing coefficients as

$$p(\mathcal{X},\mathcal{Z},\vec{\alpha}|\vec{\pi}) = \prod_{i=1}^{N}\prod_{j=1}^{M}\left[\pi_j \frac{\Gamma\left(\sum_{l=1}^{D}\alpha_{jl}\right)}{\prod_{l=1}^{D}\Gamma(\alpha_{jl})}\prod_{l=1}^{D}X_{il}^{\alpha_{jl}-1}\right]^{Z_{ij}} \times \prod_{j=1}^{M}\prod_{l=1}^{D}\frac{v_{jl}^{u_{jl}}}{\Gamma(u_{jl})}\alpha_{jl}^{u_{jl}-1}e^{-v_{jl}\alpha_{jl}}. \quad (11)$$

A directed graphical representation of this model is illustrated in Fig. 1. Since the marginalization in (6) is intractable, a variational approach is then adopted to find a tractable lower bound on $p(\mathcal{X}|\vec{\pi})$. To simplify the notation without loss of generality, we define $\Theta = \{\mathcal{Z},\vec{\alpha}\}$. By applying Jensen's inequality, lower bound $\mathcal{L}$ of the logarithm of the ML $p(\mathcal{X}|\vec{\pi})$ can be found as

$$\ln p(\mathcal{X}|\vec{\pi}) = \ln\int p(\mathcal{X},\Theta|\vec{\pi})d\Theta = \ln\int Q(\Theta)\frac{p(\mathcal{X},\Theta|\vec{\pi})}{Q(\Theta)}d\Theta \geq \int Q(\Theta)\ln\frac{p(\mathcal{X},\Theta|\vec{\pi})}{Q(\Theta)}d\Theta = \mathcal{L}(Q) \quad (12)$$

where $Q(\Theta)$ is an approximation to the true posterior distribution $p(\Theta|\mathcal{X},\vec{\pi})$. From 12, we can easily show that

$$\ln p(\mathcal{X}|\vec{\pi}) = \mathcal{L}(Q) - \underbrace{\int Q(\Theta)\ln\frac{p(\Theta|\mathcal{X},\vec{\pi})}{Q(\Theta)}d\Theta}_{KL(Q(\Theta)||p(\Theta|\mathcal{X},\vec{\pi}))}. \quad (13)$$

According to the previous equation, we can notice that the lower bound $\mathcal{L}(Q)$ is maximized when the KL divergence equals zero (i.e., $Q(\Theta) = p(\Theta|\mathcal{X},\vec{\pi})$). However, in practice, the true posterior distribution is computationally intractable and cannot be directly used for variational inference. Thus, a restricted family of distributions $Q(\Theta)$ needs to be considered. In this paper, we adopt an approach that uses a factorization assumption for restricting the form of $Q(\Theta)$ [39], which results in interesting simplifications as we shall notice. This approximation framework is known as *mean field theory* [40]–[42] and was developed in the field of physics [43]. With the factorized approximation, the posterior distribution $Q(\Theta)$ can be factorized into disjoint tractable distributions as follows:

$$Q(\Theta) = \prod_j Q_j(\Theta_j). \quad (14)$$

Note that this is the only assumption about the distribution, and no restriction is placed on the functional forms of the individual factors $Q_j(\Theta_j)$. In order to maximize the lower bound $\mathcal{L}(Q)$, we need to make a variational optimization of $\mathcal{L}(Q)$ with respect to each of the distributions $Q_j(\Theta_j)$ in turn. For a specific factor $Q_s(\Theta_s)$ in a standard variational inference approach, the general expression for its optimal solution is given by [32]

$$Q_s(\Theta_s) = \frac{\exp\langle\ln p(\mathcal{X},\Theta)\rangle_{j\neq s}}{\int\exp\langle\ln p(\mathcal{X},\Theta)\rangle_{j\neq s}d\Theta} \quad (15)$$

where $\langle\cdot\rangle_{j\neq s}$ denotes an expectation with respect to all the distributions $Q_j(\Theta_j)$ except for $j = s$. Since the expression for the optimal solution $Q_s(\Theta_s)$ depends on calculating the expectations with respect to the other factors $Q_j(\Theta_j)$ for $j \neq s$, we need to cycle through all the factors for finding the maximum of the lower bound. In general, in order to perform the variational inference, all the factors $Q_j(\Theta_j)$ need to be suitably initialized first. Then, each factor is updated in turn with a revised value obtained by (15) using the current values for all of the other factors. Convergence is guaranteed since bound is convex with respect to each of the factors $Q_j(\Theta_j)$ [44]. By using the factorization $Q(\Theta) = Q(\mathcal{Z})Q(\vec{\alpha}) = \left[\prod_{i=1}^{N}\prod_{j=1}^{M}Q(Z_{ij})\right]\left[\prod_{j=1}^{M}\prod_{l=1}^{D}Q(\alpha_{jl})\right]$, we obtain the following optimal solutions (see Appendix A):

$$Q(\mathcal{Z}) = \prod_{i=1}^{N}\prod_{j=1}^{M}r_{ij}^{Z_{ij}} \quad (16)$$

$$Q(\vec{\alpha}) = \prod_{j=1}^{M}\prod_{l=1}^{D}\mathcal{G}(\alpha_{jl}|u_{jl}^{*},v_{jl}^{*}) \quad (17)$$

where

$$r_{ij} = \frac{\rho_{ij}}{\sum_{j=1}^{M}\rho_{ij}} \quad (18)$$

$$\rho_{ij} = \exp\left\{\ln\pi_j + \widetilde{\mathcal{R}}_j + \sum_{l=1}^{D}(\bar{\alpha}_{jl}-1)\ln X_{il}\right\} \quad (19)$$

$$\begin{aligned}\widetilde{\mathcal{R}}_j = {} & \ln\frac{\Gamma\left(\sum_{l=1}^{D}\bar{\alpha}_{jl}\right)}{\prod_{l=1}^{D}\Gamma(\bar{\alpha}_{jl})} + \sum_{l=1}^{D}\bar{\alpha}_{jl}\left[\Psi\left(\sum_{l=1}^{D}\bar{\alpha}_{jl}\right)-\Psi(\bar{\alpha}_{jl})\right] \times\left[\langle\ln\alpha_{jl}\rangle - \ln\bar{\alpha}_{jl}\right] \\ & + \frac{1}{2}\sum_{l=1}^{D}\bar{\alpha}_{jl}^{2}\left[\Psi'\left(\sum_{l=1}^{D}\bar{\alpha}_{jl}\right)-\Psi'(\bar{\alpha}_{jl})\right] \times\left\langle(\ln\alpha_{jl}-\ln\bar{\alpha}_{jl})^{2}\right\rangle \\ & + \frac{1}{2}\sum_{a=1}^{D}\sum_{b=1,a\neq b}^{D}\bar{\alpha}_{ja}\bar{\alpha}_{jb}\left\{\Psi'\left(\sum_{l=1}^{D}\bar{\alpha}_{jl}\right)\right.\end{aligned}$$

$$\times \left(\langle \ln \alpha_{ja} \rangle - \ln \bar{\alpha}_{ja}\right) \times \left(\langle \ln \alpha_{jb} \rangle - \ln \bar{\alpha}_{jb}\right) \Bigg\} \quad (20)$$

$$u_{jl}^* = u_{jl} + \varphi_{jl} \qquad v_{jl}^* = v_{jl} - \vartheta_{jl} \quad (21)$$

$$\varphi_{jl} = \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_{jl} \left[\Psi\left(\sum_{k=1}^{D} \bar{\alpha}_{jk}\right) - \Psi(\bar{\alpha}_{jl}) + \sum_{k \neq l}^{D} \Psi'\left(\sum_{k=1}^{D} \bar{\alpha}_k\right) \times \bar{\alpha}_k \left(\langle \ln \alpha_k \rangle - \ln \bar{\alpha}_k\right) \right] \quad (22)$$

$$\vartheta_{jl} = \sum_{i=1}^{N} \langle Z_{ij} \rangle \ln X_{il} \quad (23)$$

where $\Psi(\cdot)$ and $\Psi'(\cdot)$ are the digamma and trigamma functions, respectively. The expected values in the above formulas are

$$\langle Z_{ij} \rangle = r_{ij}, \qquad \bar{\alpha}_{jl} = \langle \alpha_{jl} \rangle = \frac{u_{jl}}{v_{jl}} \quad (24)$$

$$\langle \ln \alpha_{jl} \rangle = \Psi(u_{jl}) - \ln v_{jl} \quad (25)$$

$$\langle (\ln \alpha_{jl} - \ln \bar{\alpha}_{jl})^2 \rangle = [\Psi(u_{jl}) - \ln u_{jl}]^2 + \Psi'(u_{jl}). \quad (26)$$

B. Determining the Number of Components

In our work, the mixing coefficients $\vec{\pi}$ are treated as parameters, and point estimations of their values are evaluated by maximizing the variational likelihood bound $\mathcal{L}(Q)$. Setting the derivative of this lower bound with respect to $\vec{\pi}$ to zero gives

$$\pi_j = \frac{1}{N} \sum_{i=1}^{N} r_{ij}. \quad (27)$$

Note that this maximization is interleaved with the variational optimizations for $Q(\mathcal{Z})$ and $Q(\vec{\alpha})$. Indeed, components that provide insufficient contribution to explain the data would have their mixing coefficients driven to zero during the variational optimization, and so they can be effectively eliminated from the model through *automatic relevance determination* [45]. Thus, by starting with a relatively large initial value of M and then removing the redundant components after convergence, we can obtain the optimal number of components in a single training run. It is noteworthy that according to (13), we can see that the variational Bayes objective function penalizes implicitly complex models. Indeed, it can be interpreted as a penalized likelihood where the first term can be viewed as an average log-likelihood and the second one as a function that penalizes complex models[1] [i.e., as the number of parameters increases, the KL divergence, which is always nonnegative, increases] [31]. It is also noteworthy that some works have shown that the variational objective is reduced to the well-known BIC as $N \to \infty$ [31], [30], which justifies the fact that the variational Bayesian approach is more accurate than BIC for model selection (i.e., determination of the optimal number of mixture components) in practical settings [21], [23].

C. Complete Variational Learning Algorithm

In variational learning, it is possible to trace the convergence systematically by monitoring the variational lower bound during the re-estimation step [31]. Indeed, at each step of the iterative re-estimation procedure, the value of this bound should never decrease. Specifically, we evaluate the bound $\mathcal{L}(Q)$ at each iteration and terminate optimization if it does not increase significantly. For the variational Dirichlet mixture model, the lower bound in (12) is evaluated as

$$\begin{aligned}
\mathcal{L}(Q) &= \sum_{\mathcal{Z}} \int Q(\mathcal{Z}, \vec{\alpha}) \ln \left\{ \frac{p(\mathcal{X}, \mathcal{Z}, \vec{\alpha} | \vec{\pi})}{Q(\mathcal{Z}, \vec{\alpha})} \right\} d\vec{\alpha} \\
&= \langle \ln p(\mathcal{X} | \mathcal{Z}, \vec{\alpha}) \rangle + \langle \ln p(\mathcal{Z} | \vec{\pi}) \rangle + \langle \ln p(\vec{\alpha}) \rangle - \langle \ln Q(\mathcal{Z}) \rangle \\
&\quad - \langle \ln Q(\vec{\alpha}) \rangle \\
&= \sum_{i=1}^{N} \sum_{j=1}^{M} r_{ij} [\widetilde{\mathcal{R}}_j + \sum_{l=1}^{D} (\bar{\alpha}_{jl}) \ln X_{il}] + \sum_{i=1}^{N} \sum_{j=1}^{M} r_{ij} \ln \pi_j \\
&\quad + \sum_{j=1}^{M} \sum_{l=1}^{D} \Big\{ u_{jl} \ln v_{jl} - \ln \Gamma(u_{jl}) + (u_{jl} - 1) \langle \ln \alpha_{jl} \rangle \\
&\quad - v_{jl} \bar{\alpha}_{jl} \Big\} - \sum_{i=1}^{N} \sum_{j=1}^{M} r_{ij} \ln r_{ij} - \sum_{j=1}^{M} \sum_{l=1}^{D} \Big\{ u_{jl}^* \ln v_{jl}^* \\
&\quad - \ln \Gamma(u_{jl}^*) + (u_{jl}^* - 1) \langle \ln \alpha_{jl} \rangle - v_{jl}^* \bar{\alpha}_{jl} \Big\}. \quad (28)
\end{aligned}$$

Since the solutions for the variational posterior Q and the value of the lower bound depend on $\vec{\pi}$, the optimization of the variational Dirichlet mixture model can be solved using an EM-like algorithm with a guaranteed convergence ([21], [30] for an empirical study and [33], [47] for a theoretical one). Indeed, local convergence has been formally and analytically proven in the case of the exponential family models with missing values [47] to which the finite Dirichlet mixture belongs. This local convergence is due to the convexity property of the exponential family of distributions. It is noteworthy that in our algorithm the first-order and second-order Taylor approximations are employed to obtain the lower bounds for estimating variational factors. The convexity of the resulting approximated functions is proved in Appendix B. The complete algorithm can be summarized in Algorithm 1.[2]

Algorithm 1 Variational Learning Algorithm

1) Initialization
   a) Choose the initial number of components and the initial values for hyper-parameters $\{u_{jl}\}$ and $\{v_{jl}\}$.
   b) Initialize the value of r_{ij} by K-Means algorithm.
2) The variational E-step: Update the variational solutions for $Q(\mathcal{Z})$ (16) and $Q(\vec{\alpha})$ (17).
3) The variational M-step: maximize lower bound $\mathcal{L}(Q)$ with respect to the current value of $\vec{\pi}$ (27).
4) Repeat steps 2 and 3 until convergence [i.e., stabilization of the variational lower bound in (28)].
5) Detect M by eliminating the components with small mixing coefficients (less than 10^{-5}).
6) Re-estimate the model parameters [(16), (17), and (27)].

[1] Called also Occam factor in the literature [45], [46].

[2] The complete source code of this paper is available upon request.

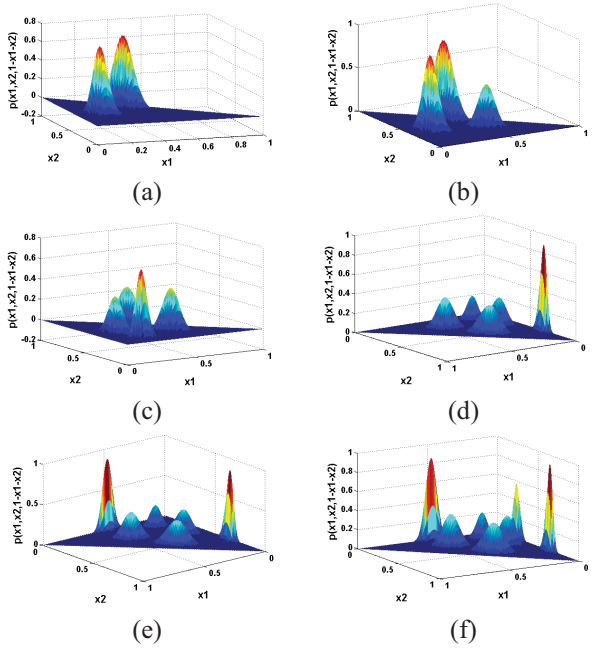


Fig. 2. Mixture densities for the synthetic data sets. (a) Data set 1. (b) Data set 2. (c) Data set 3. (d) Data set 4. (e) Data set 5. (f) Data set 6.

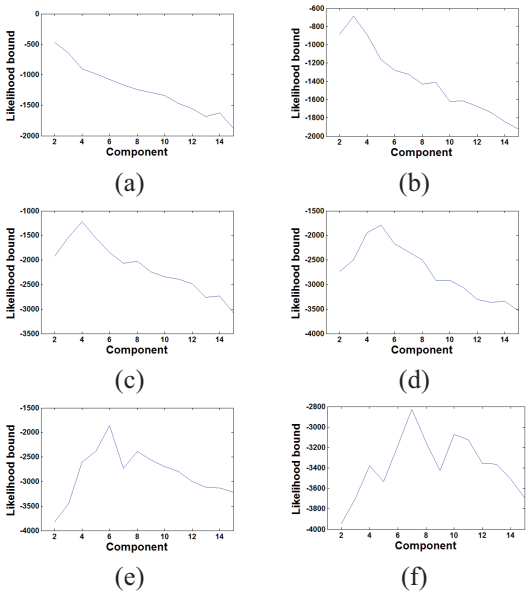


Fig. 3. Variational likelihood bound as a function of the fixed assumed number of mixture components for the different generated data sets. (a) Data set 1. (b) Data set 2. (c) Data set 3. (d) Data set 4. (e) Data set 5. (f) Data set 6.

IV. Experimental Results

In this section, we describe results that evaluate and indicate the effectiveness of the proposed approach using both synthetic and real applications namely images categorization and the challenging problem of anomaly intrusion detection. While the goal of the synthetic data is to investigate the accuracy of the variational approach as compared to the deterministic technique proposed in [34], the goal of the real applications is to compare the performances of finite Dirichlet and finite Gaussian mixture models both learned in a variational way. In our experiments, we initialize the number of components with large value (15 in this paper) with equal mixing coefficients. It is worth mentioning that multiple maxima in the variational bound may exist, and therefore running the optimization several times with different initializations is helpful for discovering a good maximum in principle [32]. In practice we have perceived that, for the experiments involved in this article, poor initialization values of the hyperparameters $\{u_{jl}\}$ and $\{v_{jl}\}$ considerably slow down the convergence speed. Based on our experiments, a good choice of the initial values of the hyperparameters $\{u_{jl}\}$ and $\{v_{jl}\}$ is to set them to 1 and 0.01, respectively. We have also considered hyperparameters initialization strategy previously proposed in [48] in the case of finite Gaussian mixture models. This approach is based on estimating the hyperparameters using ML estimation of the parameters that result from successive runs of the EM algorithm. However, we have not observed, according to our experiments, significant improvement or influence on the learning process.

A. Synthetic Data

We first present the performance of our variational algorithm (varDM) in terms of estimation and selection, on six 3-D synthetic data. Note that, here we choose $D = 3$ purely for ease of representation. We tested the effectiveness of our algorithm for estimating the mixture's parameters and selecting the number of components on generated data sets with different parameters. Table I shows the real and estimated parameters of each data set using both our variational algorithm and the deterministic approach (DM) proposed in [34]. Fig. 2 represents the resultant mixtures with different shapes (symmetric and asymmetric modes). First, we perform our variational optimization on a fixed number of components (i.e., without components elimination). Thus, the variational likelihood bound becomes a model selection score. As shown in Fig. 3, we ran our algorithm by varying the number of mixture components from 2 to 15. According to this figure, it is clear that for each data set, the variational likelihood bound is maximum at the correct number of components, which indicates that the variational likelihood bound can be used as an efficient criterion for model selection. Second, we applied directly our algorithm on these data sets (by starting with 15 components). Fig. 4 shows the estimated mixing coefficients of the different components for the different generated data sets after convergence. According to this figure, we can see that redundant components have estimated mixing coefficients close to zero after convergence. By removing the components with very small mixing coefficients, we obtain the correct number of components for each generated data set. Fig. 5 illustrates the value of the variational likelihood bound in each iteration, and shows that the likelihood bound increases at each iteration and in most cases it increases very fast when one of the mixing coefficients is close to zero (i.e., shall be removed). Moreover, we have performed a comparison between the numerical complexity of the proposed variational algorithm and the DM, in terms of overall computation time and number

TABLE I

PARAMETERS OF THE DIFFERENT GENERATED DATA SETS. N DENOTES THE TOTAL NUMBER OF ELEMENTS, n_j DENOTES THE NUMBER OF ELEMENTS IN CLUSTER j. $\alpha_{j1}, \alpha_{j2}, \alpha_{j3}$, AND π_j ARE THE REAL PARAMETERS. $\hat{\alpha}_{j1}, \hat{\alpha}_{j2}, \hat{\alpha}_{j3}$, AND $\hat{\pi}_j$ ARE THE ESTIMATED PARAMETERS BY VARIATIONAL INFERENCE. $\breve{\alpha}_{j1}, \breve{\alpha}_{j2}, \breve{\alpha}_{j3}$, AND $\breve{\pi}_j$ ARE THE ESTIMATED PARAMETERS USING [34]. WE CAN OBSERVE THAT BOTH ALGORITHMS ARE ABLE TO ESTIMATE UNKNOWN PARAMETERS, YET THE VARIATIONAL ALGORITHM ALWAYS GIVES MORE ACCURATE VALUES

	n_j	j	α_{j1}	α_{j2}	α_{j3}	π_j	$\hat{\alpha}_{j1}$	$\hat{\alpha}_{j2}$	$\hat{\alpha}_{j3}$	$\hat{\pi}_j$	$\breve{\alpha}_{j1}$	$\breve{\alpha}_{j2}$	$\breve{\alpha}_{j3}$	$\breve{\pi}_j$
Data set 1 ($N = 400$)	200	1	12	30	45	0.5	12.59	31.29	45.56	0.50	11.08	31.33	45.28	0.482
	200	2	32	50	16	0.5	33.58	50.20	15.64	0.50	31.27	50.64	16.38	0.518
Data set 2 ($N = 500$)	200	1	12	30	45	0.4	13.91	35.40	51.07	0.398	13.96	32.41	48.53	0.327
	200	2	32	50	16	0.4	32.68	51.71	16.81	0.401	32.53	48.96	16.79	0.451
	100	3	55	28	35	0.2	50.43	25.99	31.95	0.201	51.66	30.03	37.85	0.222
Data set 3 ($N = 800$)	200	1	12	30	45	0.25	13.07	31.96	46.63	0.247	13.58	28.85	46.54	0.225
	200	2	25	18	90	0.25	24.02	17.76	85.44	0.253	25.96	17.69	93.51	0.231
	200	3	55	28	35	0.25	54.89	27.73	34.13	0.249	56.43	29.72	33.93	0.286
	200	4	32	50	16	0.25	31.63	48.73	14.45	0.251	34.68	51.34	14.18	0.258
Data set 4 ($N = 1000$)	200	1	12	30	45	0.2	11.46	27.97	41.98	0.198	11.28	32.59	46.84	0.231
	100	2	25	18	90	0.1	25.16	19.23	93.36	0.098	23.13	19.50	87.92	0.145
	300	3	55	28	35	0.3	54.45	28.58	34.40	0.300	53.57	29.08	36.77	0.286
	200	4	32	50	16	0.2	36.23	55.47	18.04	0.198	35.31	53.09	19.61	0.174
	200	5	3	118	60	0.2	3.22	130.15	65.89	0.206	2.84	109.37	63.32	0.164
Data set 5 ($N = 900$)	200	1	12	30	45	0.22	12.21	31.24	47.06	0.223	12.50	28.96	46.89	0.258
	200	2	32	50	16	0.22	36.86	57.47	18.96	0.222	34.58	52.67	18.71	0.204
	200	3	55	28	35	0.22	55.83	28.75	34.84	0.221	57.62	27.04	36.18	0.237
	100	4	3	118	60	0.11	3.03	124.93	63.46	0.111	3.19	122.75	58.14	0.125
	100	5	25	18	90	0.11	25.72	17.96	90.71	0.112	26.15	18.03	88.96	0.092
	100	6	75	2	80	0.11	68.48	1.69	74.98	0.111	72.54	2.37	83.28	0.084
Data set 6 ($N = 1000$)	200	1	12	30	45	0.2	13.60	33.37	49.51	0.199	11.13	32.66	47.35	0.218
	200	2	32	50	16	0.2	33.72	53.26	16.67	0.201	35.68	47.52	17.31	0.207
	200	3	80	130	5	0.2	86.35	139.96	5.21	0.199	84.93	136.49	3.98	0.179
	100	4	3	118	60	0.1	2.98	124.84	64.67	0.100	3.50	115.03	66.37	0.081
	100	5	25	18	90	0.1	21.57	15.43	79.37	0.100	22.86	16.71	83.92	0.092
	100	6	75	2	80	0.1	64.15	1.69	67.23	0.101	81.63	2.67	70.38	0.135
	100	7	6	50	118	0.1	5.84	49.48	115.82	0.100	8.19	53.75	128.17	0.088

TABLE II

RUN TIME (IN SECONDS) AND NUMBER OF ITERATIONS REQUIRED BEFORE CONVERGENCE FOR VARDM AND DM

varDM			**DM**	
Data set	Run time	No. iterations	Run time	No. iterations
1	4.81	278	10.62	364
2	4.73	269	10.85	395
3	4.08	191	10.19	282
4	3.95	189	9.83	257
5	3.64	143	9.17	243
6	4.72	265	10.78	386

of iterations before convergence. The corresponding results are shown in Table II. It is obvious that, for each data set, the proposed variational algorithm requires fewer iterations to converge, and has a smaller computational time than the deterministic one. It is noteworthy that the overall computational complexity for one iteration of varDM is $\mathcal{O}(NMD)$, which is the same as the Gaussian case.

B. Images Categorization

In this part, we consider the problem of images categorization, which is a fundamental problem in vision that has recently drawn considerable interest and seen great progress [49]. Applications include the automatic understanding of images, object recognition, image databases browsing and content-based images suggestion, recommendation and retrieval [50]–[52]. As with the majority of computer vision tasks, an important step for accurate images categorization is the extraction of good descriptors (i.e., discriminative and invariant at the same time) to represent these images. Recently, methods based on the bag-of-features approach have been shown to give excellent results [53], [54]. In this section, we therefore follow this class of methods and in particular the one proposed in [54]. First, key points in the images are detected using one of the various detectors and local descriptors, which should be invariant to image transformation, occlusions and variations of illumination are extracted. Then, these local descriptors are grouped into $\mathcal{W}$ homogenous clusters, using a clustering or vector quantization algorithm such as K-means. Therefore, each cluster center is treated as a visual word and a visual vocabulary is built with $\mathcal{W}$ visual words. Applying the paradigm of *bag-of-words*, a $\mathcal{W}-$dimensional histogram representing the frequency of each visual word is calculated for each image. Finally, the probabilistic latent semantic analysis (pLSA) model [55] is applied to reduce the dimensionality of the resulting histograms allowing the representation of images as proportional vectors. Thus, our variational Dirichlet mixture modeling

TABLE III
CLUSTERING RESULTS WITH VARDM MODEL AND VARGM MODEL. M^* DENOTES THE AVERAGE NUMBER OF CLUSTERS

	Data set A		Data set B	
	varDM	**varGM**	**varDM**	**varGM**
M^*	4.85 ± 0.19	4.56 ± 0.31	4.03 ± 0.14	4.41 ± 0.52
Accuracy (%)	74.93 ± 1.62	65.26 ± 1.38	78.01 ± 1.56	68.34 ± 1.29
Run time (s)	195.36 ± 1.03	189.57 ± 0.97	155.13± 1.13	150.95± 1.09

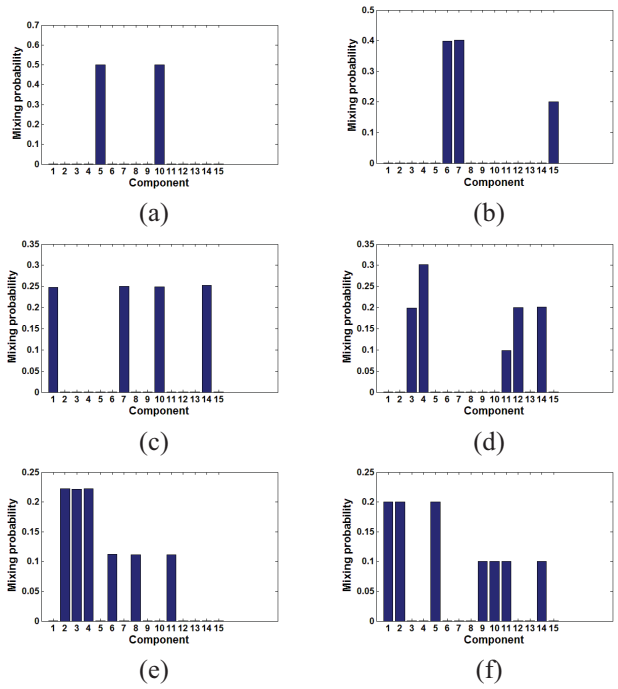


Fig. 4. Mixing probabilities of components found for the different generated data sets after convergence. (a) Data set 1. (b) Data set 2. (c) Data set 3. (d) Data set 4. (e) Data set 5. (f) Data set 6.

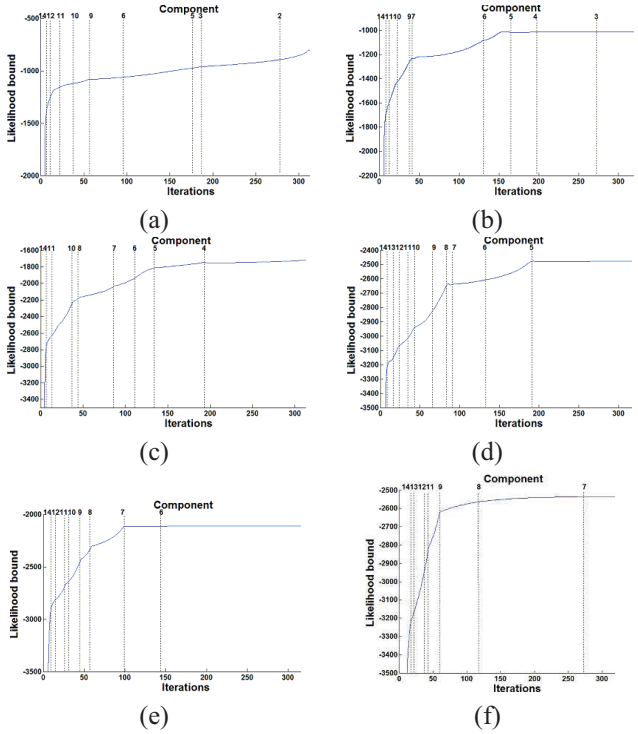


Fig. 5. Variational likelihood bound for each iteration of the different generated data sets. The initial number of components is 15. Vertical dash lines indicate cancelation of components. (a) Data set 1. (b) Data set 2. (c) Data set 3. (d) Data set 4. (e) Data set 5. (f) Data set 6.

framework provides a natural setting in which to address the categorization task.

For our experiments, we have considered the FeiFei's sports event data set containing eight categories of sports scenes: rowing (250 images), badminton (200 images), polo (182 images), bocce (137 images), snow boarding (190 images), croquet (236 images), sailing (190 images), and rock climbing (194 images). Thus, the data set contains 1579 images in total. We normalize each image into a size of 256 × 256 pixels. Examples of images from each category are shown in Fig. 6. In our experiments, the key points of each image are detected using the difference-of-Gaussian interest point detector [56] and described using scale-invariant feature transform (SIFT) resulting in a 128-dimensional vector for each key point [56]. Then, an accelerated version of the K-Means algorithm [57] is used to cluster all the SIFT vectors into a visual vocabulary of 700 visual words. Note that the number of visual words is user-specified. Based on our experiments, the best results have been obtained when $\mathcal{W} \in [600, 800]$. Then, the new representation for each image is calculated through the pLSA model by considering 35 aspects. Two data sets are used for testing our algorithm. Data set A consists of 750 images from five

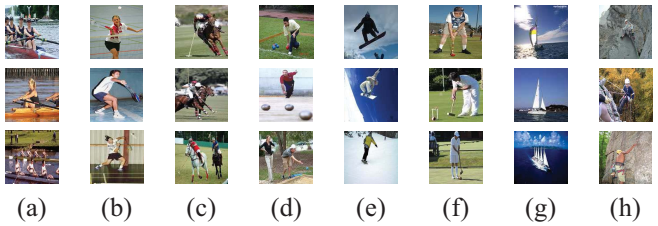


Fig. 6. Sample images from each group of sports event data. (a) Rowing. (b) Badminton. (c) Polo. (d) Bocce. (e) Snow boarding. (f) Croquet. (g) Sailing. (h) Rock climbing (available at http://vision.stanford.edu/lijiali/event_dataset).

categorizes of the sports event data set: rowing, badminton, sailing, croquet, and rock climbing. Data set B consists of 600 images from four different categorizes of the sports event data set: rowing, polo, snow boarding, and bocce. Table III shows the average number of clusters and the average classification accuracies using both Dirichlet and Gaussian mixture (varGM) models learned by running their respective variational algorithms 20 times. Tables IV and V show the confusion

TABLE IV
AVERAGE ROUNDED CONFUSION MATRIX USING THE VARDM MODEL TO CATEGORIZE DATA SET A

	Rowing	Badminton	Sailing	Croquet	Rock
Rowing	**109**	5	28	3	5
Badminton	8	**116**	0	10	18
Sailing	19	3	**122**	2	4
Croquet	9	25	1	**104**	11
Rock	8	18	3	10	**111**

TABLE V
AVERAGE ROUNDED CONFUSION MATRIX USING THE VARDM MODEL TO CATEGORIZE DATA SET B

	Rowing	Polo	Snow	Bocce
Rowing	**115**	17	8	10
Polo	6	**124**	13	7
Snow	21	6	**109**	14
Bocce	5	3	15	**127**

matrices when applying the Dirichlet mixture model for data sets A and B, respectively. According to the obtained results, we can clearly see that the varDM outperforms the varGM in terms of computational cost, categorization accuracy, and selection of the optimal number of image categories.

C. Anomaly Intrusion Detection

Nowadays, intrusion detection systems (IDSs) are becoming more and more important as computer security vulnerabilities and flaws are being discovered everyday [58]–[61]. The main goal is to establish approaches, which can scan network activities and detect suspicious patterns that may have been derived from intrusion attacks. Intrusion detection is based on the assumption that intrusive activities are noticeably diverse from normal system activities and hence detectable. According to the analysis methods, IDSs can be classified into two main categories: *misuse* detection and *anomaly* detection systems. In misuse detection systems, pre-defined attack patterns and signatures are used for detecting known attacks. Alternatively, anomaly detection systems detect unknown attacks by observing deviations from normal activities of the system. Anomaly detection has the advantage of detecting new types of intrusions. In our work, we first use our mixture model to learn patterns of normal and intrusive activities from training data. Then, we detect and classify intrusive activities which are deviated from the normal activities in a testing data set.

1) Data Set Description: The well-known KDD Cup 1999 Data[3] is used to investigate our mixture model. This data set (tcpdump file) was collected at MIT Lincoln Laboratory for the 1998 DARPA intrusion detection evaluation program by simulating attacks on a typical U.S. Air Force Lan. Each data instance in the data set is a connection record obtained from the simulated intrusions with 41 features (such as duration, dst_bytes, etc.). A connection is a sequence of TCP packets starting and ending at some well defined times, between which data flows to and from a source IP address to a target IP address under some well-defined protocol. The training data consist of 494 021 data instances of which 97 277 are normal, and 396 744 are attacks. The testing set contains 311 029 data instances of which 60 593 are normal, and 250 436 are attacks. All of these attacks fall into one of the following four categories: DOS: denial-of-service (e.g., syn flood); R2L: unauthorized access from a remote machine (e.g., guessing password); U2R: unauthorized access to local superuser (root) privileges (e.g., buffer overflow attack); and Probing: surveillance and other probing (e.g., port scanning).

[3]Available at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

TABLE VI
CONFUSION MATRIX FOR INTRUSION DETECTION WITH VARIATIONAL DIRICHLET MIXTURE MODEL

	Normal	DOS	R2L	U2R	Probe
Normal	**49081**	1169	9012	1042	289
DOS	38859	**181372**	562	309	8751
R2L	3617	169	**9657**	243	95
U2R	185	63	137	**2185**	66
Probe	401	185	62	149	**3369**

TABLE VII
INTRUSION DETECTION RESULTS USING DIFFERENT APPROACHES

Algorithm	varDM	DM	varGM	GM
Accuracy (%)	78.75	75.53	73.34	71.29

2) Results: Since the features are on quite different scales in the data set, we need to normalize the data set such that one feature would not dominant the others in our algorithm. In our data set, each data instance contains 41 features of which 34 are numeric and seven are symbolic. In our experiments, only the 34 numeric features are used (i.e., each data are then represented as a 34-D vector). By finding the maximum and minimum values of a given feature X_l in a data instance $\vec{X}$, we can transform the feature into the range of $[0, 1]$ by $X_l = (X_L - \min(X_l))/(\max(X_l) - \min(X_l))$, where X_l is set to a small value if the maximum is equal to the minimum. The resulting feature vectors are then normalized to the unit sum. Table VI shows the obtained confusion matrix using our varDM. According to this matrix, the detection rate is 78.75%. A summary of the detection results by applying other approaches namely the DM, the varGM, and the Gaussian mixture learned in a deterministic way (GM) are given in Table VII. According to these results, we can say that the varDM outperforms significantly, according to a student's t-test, the other approaches. Moreover, our algorithm outperforms three other methods namely the KNN (70.13%) and the SVM (75.81%) approaches proposed in [62], and a neural networks (73.68%) approach described in [63].

V. CONCLUSION

In this paper, we have presented an efficient attractive procedure for the variational learning of finite Dirichlet mixture models. Our procedure was based on the construction and the optimization of a lower bound on the model's likelihood by choosing completely factorized conditional distributions over

the model's variables. The proposed framework can be viewed as a compromise between ML estimation, which prefers complex models and then causes overfitting and pure Bayesian techniques that penalize complex models, but unfortunately require intensive computations and were generally intractable. Indeed, unlike pure Bayesian methods that require sampling, the proposed variational approach approximates posterior distributions over model parameters analytically thanks to the accurate choice of specific conjugate priors. Through extensive experiments, we have shown that proposed variational framework allows the automatic and simultaneous adjusting of the mixture parameters and the number of components. It was noteworthy that the ability of our variational approach to lead to a model with the correct number of components has been based solely on empirical evidence via our experiments. These experiments have involved both synthetic and real challenging problems, such as image databases categorization and intrusion detection. Future works can be devoted to the inclusion of a feature selection component within the proposed framework or the extension of the proposed model to the infinite case.

APPENDIX A
PROOF OF (16) AND (17)

According to 15, the general expression for the variational solution $Q_s(\Theta_s)$ can be written as

$$\ln Q_s(\Theta_s) = \left\langle \ln p(\mathcal{X}, \Theta) \right\rangle_{j \neq s} + \text{const} \tag{29}$$

where any terms that are independent of $Q_s(\Theta_s)$ are absorbed into the additive constant. Using the previous equation and the logarithm of the joint distribution in (11), $p(\mathcal{X}, \Theta|\vec{\pi})$, we develop the following variational solutions for $Q(\mathcal{Z})$ and $Q(\vec{\alpha})$.

A. Proof of (16): Variational Solution to $Q(\mathcal{Z})$

$$\ln Q(Z_{ij}) = Z_{ij} \left[\ln \pi_j + \mathcal{R}_j + \sum_{l=1}^{D} (\bar{\alpha}_{jl} - 1) \ln X_{il} \right] + \text{const} \tag{30}$$

where

$$\mathcal{R}_j = \left\langle \ln \frac{\Gamma\left(\sum_{l=1}^{D} \alpha_{jl}\right)}{\prod_{l=1}^{D} \Gamma(\alpha_{jl})} \right\rangle_{\alpha_{j1}, \ldots, \alpha_{jD}}, \qquad \bar{\alpha}_{jl} = \langle \alpha_{jl} \rangle = \frac{u_{jl}}{v_{jl}}. \tag{31}$$

Unfortunately a closed-form expression cannot be found for $\mathcal{R}_j$, so the standard variational inference cannot be applied directly. Therefore, we need to propose a lower bound approximation to obtain a closed-form expression. The second-order Taylor series expansion has been successfully applied in variational inference for providing tractable approximations [38], [64] and we shall use it here. Indeed, we approximate the function $\mathcal{R}_j$ using a second-order Taylor expansion about the expected values of the parameters $\vec{\bar{\alpha}}_j$. Let us define $\widetilde{\mathcal{R}}_j$ to denote the approximation of $\mathcal{R}_j$, and $(\bar{\alpha}_{j1}, \ldots, \bar{\alpha}_{jD})$ to represent the expected values of $\vec{\alpha}_j$. This lower bound approximation is given by (20) and is proved in Appendix B. Then, the optimization in (30) becomes tractable after replacing $\mathcal{R}_j$ by $\widetilde{\mathcal{R}}_j$.

From (30), it is straightforward to see that the optimal solution to $\mathcal{Z}$ has the logarithmic form of (4) except for the normalization constant. Thus, $\ln Q(\mathcal{Z})$ can be written as

$$\ln Q(\mathcal{Z}) = \sum_{i=1}^{N} \sum_{j=1}^{M} z_{ij} \ln \rho_{ij} + \text{const} \tag{32}$$

$$\ln \rho_{ij} = \ln \pi_j + \widetilde{\mathcal{R}}_j + \sum_{l=1}^{D} (\bar{\alpha}_{jl} - 1) \ln X_{il}. \tag{33}$$

Note that, any terms that do not depend on Z_{ij} can be absorbed into the constant part. If we take the exponential of both sides in (32), we obtain

$$Q(\mathcal{Z}) \propto \prod_{i=1}^{N} \prod_{j=1}^{M} \rho_{ij}^{Z_{ij}}. \tag{34}$$

This distribution needs to be normalized, which can be performed as follows:

$$Q(\mathcal{Z}) = \prod_{i=1}^{N} \prod_{j=1}^{M} r_{ij}^{Z_{ij}}, \qquad r_{ij} = \frac{\rho_{ij}}{\sum_{j=1}^{M} \rho_{ij}}. \tag{35}$$

Note that the $\{r_{ij}\}$ are nonnegative and sum to one. Therefore, we can obtain the standard result for $Q(\mathcal{Z})$ as

$$\langle Z_{ij} \rangle = r_{ij} \tag{36}$$

where $\{r_{ij}\}$ are playing the role of responsibilities as in the conventional EM algorithm.

B. Proof of (17): Variational Solution to $Q(\vec{\alpha})$

Since there are M components in the mixture model by considering the assumption that the parameters α_{jl} are independent, $Q(\vec{\alpha})$ can be factorized as

$$Q(\vec{\alpha}) = \prod_{j=1}^{M} \prod_{l=1}^{D} Q(\alpha_{jl}). \tag{37}$$

Let us consider the variational optimization regarding the specific factor $Q(\alpha_{js})$. The logarithm of the optimized factor is given by

$$\ln Q(\alpha_{js}) = \sum_{i=1}^{N} r_{ij} \mathcal{J}(\alpha_{js}) + \alpha_{js} \sum_{i=1}^{N} r_{ij} \ln X_{is} + (u_{js} - 1) \ln \alpha_{js} - v_{js} \alpha_{js} + \text{const} \tag{38}$$

where

$$\mathcal{J}(\alpha_{js}) = \left\langle \ln \frac{\Gamma\left(\alpha_s + \sum_{l \neq s}^{D} \alpha_{jl}\right)}{\Gamma(\alpha_s) \prod_{l \neq s}^{D} \Gamma(\alpha_{jl})} \right\rangle_{\Theta \neq \alpha_{js}} \tag{39}$$

where $\mathcal{J}(\alpha_{js})$ is defined as a function of α_{js}, and is unfortunately analytically intractable. Therefore, similarly to $\mathcal{R}_j$ in the previous section, we need to find a lower bound to approximate $\mathcal{J}(\alpha_{js})$ which we obtain via a first-order Taylor

expansion [38], [1, ch. 10] about $\bar{\alpha}_{js}$ (the expected value of α_{js}) (see Appendix B)

$$\mathcal{J}(\alpha_{js}) \geq \bar{\alpha}_{js} \ln \alpha_{js} \left\{ \Psi\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) - \Psi(\bar{\alpha}_{js}) + \sum_{l \neq s}^{D} \bar{\alpha}_{jl} \times \Psi'\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) \left(\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl}\right) \right\} + \text{const.} \quad (40)$$

If we substitute this lower bound back into (38), we obtain a new optimal solution to α_{js} as

$$\begin{aligned} \ln Q(\alpha_{js}) &= \sum_{i=1}^{N} r_{ij} \bar{\alpha}_{js} \ln \alpha_{js} \left[\Psi\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) - \Psi(\bar{\alpha}_{js}) \right. \\ &\quad \left. + \sum_{l \neq s}^{D} \Psi'\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) \bar{\alpha}_{jl} \left(\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl}\right) \right] \\ &\quad + \alpha_{js} \sum_{i=1}^{N} r_{ij} \ln X_{is} + (u_{js} - 1) \ln \alpha_{js} - v_{js} \alpha_{js} \\ &\quad + \text{const} \\ &= \ln \alpha_{js} (u_{js} + \varphi_{js} - 1) - \alpha_{js} (v_{js} - \vartheta_{js}) + \text{const} \end{aligned} \quad (41)$$

where

$$\varphi_{js} = \sum_{i=1}^{N} r_{ij} \bar{\alpha}_{js} \left[\Psi\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) - \Psi(\bar{\alpha}_{js}) + \sum_{l \neq s}^{D} \Psi'\left(\sum_{l=1}^{D} \bar{\alpha}_{jl}\right) \times \bar{\alpha}_{jl} \left(\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl}\right) \right] \quad (42)$$

$$\vartheta_{js} = \sum_{i=1}^{N} r_{ij} \ln X_{is}. \quad (43)$$

We can see that (41) has the logarithmic form of a Gamma distribution. Taking the exponential of both sides, we obtain

$$Q(\alpha_{js}) \propto \alpha_{js}^{u_{js} + \varphi_{js} - 1} e^{-(v_{js} - \vartheta_{js}) \alpha_{js}}. \quad (44)$$

Therefore, we can obtain the optimal solutions to the hyperparameters u_{js} and v_{js} as

$$u_{js}^{*} = u_{js} + \varphi_{js}, \qquad v_{js}^{*} = v_{js} - \vartheta_{js}. \quad (45)$$

APPENDIX B
PROOF OF (20) AND (40)

A. Lower Bound of $\mathcal{R}_j$: Proof of (20)

The function $\mathcal{R}_j$ in (31) is analytically intractable, but a non-linear approximation of the lower bound can be obtained by using the second-order Taylor expansion as done in [38], where the authors have used the first and second Taylor expansions to approximate lower bounds for variational Beta mixture model. In our work, first, we define the function

$$\mathcal{H}(\vec{\alpha}_j) = \mathcal{H}(\alpha_{j1}, \ldots, \alpha_{jD}) = \ln \frac{\Gamma\left(\sum_{l=1}^{D} \alpha_{jl}\right)}{\prod_{l=1}^{D} \Gamma(\alpha_{jl})} \quad (46)$$

where $\alpha_{jl} > 1$. The lower bound of $\mathcal{H}(\vec{\alpha}_j)$ can be obtained by using the second-order Taylor expansion for $\ln \vec{\alpha}_j = (\ln \alpha_{j1}, \ldots, \ln \alpha_{jD})$ at $\ln \vec{\alpha}_{j,0} = (\ln \alpha_{j1,0}, \ldots, \ln \alpha_{jD,0})$ as

$$\begin{aligned} \mathcal{H}(\vec{\alpha}_j) &\geq \mathcal{H}(\vec{\alpha}_{j,0}) + (\ln \vec{\alpha}_j - \ln \vec{\alpha}_{j,0})^T \nabla \mathcal{H}(\vec{\alpha}_{j,0}) \\ &\quad + \frac{1}{2!} (\ln \vec{\alpha}_j - \ln \vec{\alpha}_{j,0})^T \nabla^2 \mathcal{H}(\vec{\alpha}_{j,0}) (\ln \vec{\alpha}_j - \ln \vec{\alpha}_{j,0}) \end{aligned} \quad (47)$$

where $\nabla \mathcal{H}(\vec{\alpha}_{j,0})$ represents the gradient of $\mathcal{H}$ evaluated at $\vec{\alpha}_j = \vec{\alpha}_{j,0}$ and $\nabla^2 \mathcal{H}(\vec{\alpha}_{j,0})$ is the Hessian matrix. This gives

$$\begin{aligned} \mathcal{H}(\vec{\alpha}_j) &\geq \mathcal{H}(\vec{\alpha}_{j,0}) + \sum_{l=1}^{D} \frac{\partial \mathcal{H}(\vec{\alpha}_j)}{\partial \ln \alpha_{jl}} |_{\vec{\alpha}_j = \vec{\alpha}_{j,0}} (\ln \alpha_{jl} - \ln \alpha_{jl,0}) \\ &\quad + \frac{1}{2} \sum_{a=1}^{D} \sum_{b=1}^{D} \frac{\partial^2 \mathcal{H}(\vec{\alpha}_j)}{\partial \ln \alpha_{ja} \partial \ln \alpha_{jb}} |_{\vec{\alpha}_j = \vec{\alpha}_{j,0}} (\ln \alpha_{ja} - \ln \alpha_{ja,0}) \\ &\quad \times (\ln \alpha_{jb} - \ln \alpha_{jb,0}). \end{aligned} \quad (48)$$

Then, the lower bound of the function $\mathcal{R}_j$ can be obtained by taking the expectation of (48) with respect to $\vec{\alpha}_j$ as

$$\begin{aligned} \mathcal{R}_j \geq \widetilde{\mathcal{R}}_j &= \ln \frac{\Gamma\left(\sum_{l=1}^{D} \alpha_{jl,0}\right)}{\prod_{l=1}^{D} \Gamma(\alpha_{jl,0})} \\ &\quad + \sum_{l=1}^{D} \alpha_{jl,0} \left[\Psi\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) - \Psi(\alpha_{jl,0}) \right] \\ &\quad \times \left[\langle \ln \alpha_{jl} \rangle - \ln \alpha_{jl,0} \right] \\ &\quad + \frac{1}{2} \sum_{l=1}^{D} \alpha_{jl,0}^2 \left[\Psi'\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) - \Psi'(\alpha_{jl,0}) \right] \\ &\quad \times \left\langle (\ln \alpha_{jl} - \ln \alpha_{jl,0})^2 \right\rangle \\ &\quad + \frac{1}{2} \sum_{a=1}^{D} \sum_{b=1 (a \neq b)}^{D} \left\{ \alpha_{ja,0} \alpha_{jb,0} \Psi'\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) \left(\langle \ln \alpha_{ja} \rangle \right.\right. \\ &\quad \left.\left. - \ln \alpha_{ja,0}\right) \left(\langle \ln \alpha_{jb} \rangle - \ln \alpha_{jb,0}\right) \right\}. \end{aligned} \quad (49)$$

In order to prove that the second-order Taylor expansion of $\mathcal{H}(\vec{\alpha}_j)$ is indeed a lower bound of $\mathcal{H}(\vec{\alpha}_j)$, we need to show

$$Hess = \begin{bmatrix} \alpha_{j1}[\Psi(\sum_{l=1}^{D} \alpha_{jl}) - \Psi(\alpha_{j1})] & & \\ +\alpha_{j1}^2[\Psi'(\sum_{l=1}^{D} \alpha_{jl}) - \Psi'(\alpha_{j1})] & \cdots & \alpha_{j1}\alpha_{jD}\Psi'(\sum_{l=1}^{D} \alpha_{jl}) - \bar{\alpha}_{j1}\bar{\alpha}_{jD}\Psi'(\sum_{l=1}^{D} \bar{\alpha}_{jl}) \\ -\bar{\alpha}_{j1}^2[\Psi'(\sum_{l=1}^{D} \bar{\alpha}_{jl}) - \Psi'(\bar{\alpha}_{j1})] & & \\ \vdots & \ddots & \vdots \\ & & \alpha_{jD}[\Psi(\sum_{l=1}^{D} \alpha_{jl}) - \Psi(\alpha_{jD})] \\ \alpha_{j1}\alpha_{jD}\Psi'(\sum_{l=1}^{D} \alpha_{jl}) - \bar{\alpha}_{j1}\bar{\alpha}_{jD}\Psi'(\sum_{l=1}^{D} \bar{\alpha}_{jl}) & \cdots & +\alpha_{jD}^2[\Psi'(\sum_{l=1}^{D} \alpha_{jl}) - \Psi'(\alpha_{jD})] \\ & & -\bar{\alpha}_{jD}^2[\Psi'(\sum_{l=1}^{D} \bar{\alpha}_{jl}) - \Psi'(\bar{\alpha}_{jD})] \end{bmatrix} \quad (50)$$

that $\Delta\mathcal{H}(\vec{\alpha}_j) \geq 0$, where $\Delta\mathcal{H}(\vec{\alpha}_j)$ denotes the difference between $\mathcal{H}(\vec{\alpha}_j)$ and its second-order Taylor expansion. The Hessian of $\Delta\mathcal{H}(\vec{\alpha}_j)$ with respect to $(\ln\alpha_{j1},\ldots,\ln\alpha_{jD})$ is given by (50), shown at the bottom of the previous page.

By substituting $(\ln\alpha_{j1},\ldots,\ln\alpha_{jD})$ with the critical point $(\ln\alpha_{j1,0},\ldots,\ln\alpha_{jD,0})$, (50) is reduced to a positive-definite diagonal matrix. Since $(\ln\alpha_{j1,0},\ldots,\ln\alpha_{jD,0})$ is the only critical point, and $\Delta\mathcal{H}(\vec{\alpha}_j)$ is continuous and differentiable through all α_{jl} (for $\alpha_{jl} > 1$), the critical point $(\ln\alpha_{j1,0},\ldots,\ln\alpha_{jD,0})$ is also the global minimum of $\Delta\mathcal{H}(\vec{\alpha}_j)$. The global minimum value 0 is reached when $(\ln\alpha_{j1},\ldots,\ln\alpha_{jD}) = (\ln\alpha_{j1,0},\ldots,\ln\alpha_{jD,0})$. Therefore, the second-order Taylor expansion is indeed a lower bound.

B. Lower Bound of $\mathcal{J}(\alpha_{js})$: Proof of (40)

Since the first-order Taylor expansion of a convex function is a tangent line of that function at a specific value, the lower bound of $\mathcal{J}(\alpha_{js})$ in (39) can be approximated by a first-order Taylor expansion. In [38], the authors evaluate the lower bound of the Log-inverse-Beta function by using the first-order Taylor expansion. In our work, we extent this idea to the multivariate case. Let us define the function $\mathcal{F}(\alpha_{js})$ as

$$\mathcal{F}(\alpha_{js}) = \ln\frac{\Gamma\left(\alpha_{js}+\sum_{l\neq s}^{D}\alpha_{jl}\right)}{\Gamma(\alpha_{js})\prod_{l\neq s}^{D}\Gamma(\alpha_{jl})}. \tag{51}$$

1) Convexity of $\mathcal{F}(\alpha_{js})$: It is not straightforward to show directly that $\mathcal{F}(\alpha_{js})$ is a convex function of α_{js}. Yet, by adopting the *relative convexity* as in [38] and [65], we can show that $\mathcal{F}(\alpha_{js})$ is convex relative to $\ln\alpha_{js}$. A function is considered to be convex on an interval if and only if its second derivative is nonnegative there. The first derivative of $\mathcal{F}(\alpha_{js})$ with respect to $\ln\alpha_{js}$ is

$$\frac{\partial\mathcal{F}(\alpha_{js})}{\partial\ln\alpha_{js}} = \left[\Psi\left(\alpha_{js}+\sum_{l\neq s}^{D}\alpha_{jl}\right)-\Psi(\alpha_{js})\right]\alpha_{js}. \tag{52}$$

Then, the second derivative with respect to $\ln\alpha_{js}$ is

$$\begin{aligned}\frac{\partial^2\mathcal{F}(\alpha_{js})}{\partial(\ln\alpha_{js})^2} &= \left[\Psi\left(\alpha_{js}+\sum_{l\neq s}^{D}\alpha_{jl}\right)-\Psi(\alpha_{js})\right]\alpha_{js}\\ &\quad+\left[\Psi'\left(\alpha_{js}+\sum_{l\neq s}^{D}\alpha_{jl}\right)-\Psi'(\alpha_{js})\right]\alpha_{js}^2\\ &= \alpha_{js}\int_0^\infty\frac{1-e^{-\left(\sum_{l\neq s}^{D}\alpha_{jl}\right)t}}{1-e^{-t}}e^{-\alpha_{js}t}(1-\alpha_{js}t)dt\end{aligned} \tag{53}$$

where the integral representations of $\Psi(x)$ and $\Psi'(x)$ are defined by

$$\Psi(x) = \int_0^\infty\left(\frac{e^{-t}}{t}-\frac{e^{-xt}}{1-e^{-t}}\right)dt \tag{54}$$

$$\Psi'(x) = \int_0^\infty\frac{te^{-xt}}{1-e^{-t}}dt. \tag{55}$$

We can rewrite (53) as

$$\frac{\partial^2\mathcal{F}(\alpha_{js})}{\partial(\ln\alpha_{js})^2} = \alpha_{js}\int_0^\infty f_1(t)f_2(t)dt \tag{56}$$

$$f_1(t) = \frac{1-e^{-\left(\sum_{l\neq s}^{D}\alpha_{jl}\right)t}}{1-e^{-t}}\quad f_2(t) = e^{-\alpha_{js}t}(1-\alpha_{js}t). \tag{57}$$

By analyzing (57), we can find that when $\sum_{l\neq s}^{D}\alpha_{jl} > 1$: if $t > 1/\alpha_{js}$, then $f_1(t) < f_1(1/\alpha_s)$ and $f_2(t) < 0$; if $t < 1/\alpha_{js}$, then $f_1(t) > f_1(1/\alpha_{js})$ and $f_2(t) > 0$. Hence, we can rewrite (56) as

$$\begin{aligned}\frac{\partial^2\mathcal{F}(\alpha_{js})}{\partial(\ln\alpha_{js})^2} &= \alpha_{js}\left\{\int_0^{\frac{1}{\alpha_{js}}}f_1(t)f_2(t)dt+\int_{\frac{1}{\alpha_{js}}}^\infty f_1(t)f_2(t)dt\right\}\\ &> \alpha_{js}\left\{\int_0^{\frac{1}{\alpha_{js}}}f_1\left(\frac{1}{\alpha_{js}}\right)f_2(t)dt+\int_{\frac{1}{\alpha_{js}}}^\infty f_1\left(\frac{1}{\alpha_{js}}\right)f_2(t)dt\right\}\\ &= \alpha_{js}f_1\left(\frac{1}{\alpha_{js}}\right)\int_0^\infty f_2(t)dt\\ &= \alpha_{js}f_1\left(\frac{1}{\alpha_{js}}\right)\lim_{t\to\infty}te^{-\alpha_{js}t} = 0.\end{aligned} \tag{58}$$

Therefore, when $\sum_{l\neq s}^{D}\alpha_{jl} > 1$, the convexity of $\mathcal{F}(\alpha_{js})$ relative to $\ln\alpha_{js}$ is proved.

2) Evaluating Lower Bound by the First-Order Taylor Expansion: Since $\mathcal{F}(\alpha_{js})$ is a convex function relative to $\ln\alpha_{js}$, its lower bound can be obtained by applying the first-order Taylor expansion of $\mathcal{F}(\alpha_{js})$ for $\ln\alpha_{js}$ at $\ln\alpha_{js,0}$ as follows:

$$\begin{aligned}\mathcal{F}(\alpha_{js}) &\geq \mathcal{F}(\alpha_{js,0})+\frac{\partial\mathcal{F}(\alpha_{js})}{\partial\ln\alpha_{js}}|_{\alpha_{js}=\alpha_{js,0}}(\ln\alpha_{js}-\ln\alpha_{js,0})\\ &= \mathcal{F}(\alpha_{js,0})+\frac{\partial\mathcal{F}(\alpha_{js})}{\partial\alpha_{js}}\frac{\partial\alpha_{js}}{\partial\ln\alpha_{js}}|_{\alpha_{js}=\alpha_{js,0}}\\ &\quad\times(\ln\alpha_{js}-\ln\alpha_{js,0})\\ &= \ln\frac{\Gamma\left(\alpha_{js,0}+\sum_{l\neq s}^{D}\alpha_{jl}\right)}{\Gamma(\alpha_{js,0})\prod_{l\neq s}^{D}\Gamma(\alpha_{jl})}\\ &\quad+\left[\Psi\left(\alpha_{js,0}+\sum_{l\neq s}^{D}\alpha_{jl}\right)-\Psi(\alpha_{js,0})\right]\\ &\quad\times\alpha_{js,0}(\ln\alpha_{js}-\ln\alpha_{js,0}).\end{aligned} \tag{59}$$

Note that we reach the equality when $\alpha_{js} = \bar{\alpha}_{js}$. By substituting (59) into (39), we obtain

$$\begin{aligned}\mathcal{J}(\alpha_{js}) &\geq \left\langle\ln\frac{\Gamma(\alpha_{js,0}+\sum_{l\neq s}^{D}\alpha_{jl})}{\Gamma(\alpha_{js,0})\prod_{l\neq s}^{D}\Gamma(\alpha_{jl})}+\left[\Psi\left(\alpha_{js,0}+\sum_{l\neq s}^{D}\alpha_{jl}\right)\right.\right.\\ &\quad\left.\left.-\Psi(\alpha_{js,0})\right]\alpha_{js,0}\left(\ln\alpha_{js}-\ln\alpha_{js,0}\right)\right\rangle_{\vec{\alpha}\neq\alpha_{js}}\\ &= \ln\alpha_{js}\alpha_{js,0}\left\{\left\langle\Psi\left(\alpha_{js,0}+\sum_{l\neq s}^{D}\alpha_{jl}\right)\right\rangle_{\vec{\alpha}\neq\alpha_{js}}-\Psi(\alpha_{js,0})\right\}\\ &\quad+\text{const}.\end{aligned} \tag{60}$$

We can notice that in (60), the calculation of the expectation $\left\langle \Psi(\alpha_{js,0} + \sum_{l\neq s}^{D} \alpha_{jl}) \right\rangle_{\vec{\alpha}\neq\alpha_{js}}$ is also analytically intractable. Using a similar proof to that in Appendix B-B.1, it is straightforward to conclude that $\Psi(\alpha_{js,0} + \sum_{l\neq s}^{D} \alpha_{jl})$ is a convex function relative to $\ln \alpha_{jl,0}$, for $l = \{1, \cdots, D\}$ and $l \neq s$. We can apply a first-order Taylor expansion for the function $\Psi(\sum_{i=1}^{n} x_i + y)$ at $\ln \hat{\mathbf{x}}$, where $\hat{\mathbf{x}} = (\hat{x}_1, \cdots, \hat{x}_n)$, to obtain its lower bound as

$$\Psi\left(\sum_{i=1}^{n} x_i + y\right) \geq \Psi\left(\sum_{i=1}^{n} \hat{x}_i + y\right) + \sum_{i=1}^{n} (\ln x_i - \ln \hat{x}_i) \Psi'\left(\sum_{i=1}^{n} \hat{x}_i + y\right) \hat{x}_i. \quad (61)$$

Using the previous equation, the approximation lower bound of expectation $\left\langle \Psi(\alpha_{jl,0} + \sum_{l\neq s}^{D} \alpha_{jl}) \right\rangle_{\vec{\alpha}\neq\alpha_{js}}$ is given by

$$\left\langle \Psi\left(\sum_{l\neq s}^{D} \alpha_{jl} + \alpha_{js,0}\right) \right\rangle_{\vec{\alpha}\neq\alpha_{js}} \geq \Psi\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) + \sum_{l\neq s}^{D} \alpha_{jl,0} \Psi'\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) \left(\left\langle \ln \alpha_{jl} \right\rangle - \ln \alpha_{jl,0}\right). \quad (62)$$

Finally, the lower bound of $\mathcal{J}(\alpha_{js})$ can be calculated by substituting (62) back into (60)

$$\mathcal{J}(\alpha_{js}) \geq \ln \alpha_{js} \alpha_{js,0} \left\{ \Psi\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) - \Psi(\alpha_{js,0}) + \sum_{l\neq s}^{D} \alpha_{jl,0} \Psi'\left(\sum_{l=1}^{D} \alpha_{jl,0}\right) \left(\left\langle \ln \alpha_{jl} \right\rangle - \ln \alpha_{jl,0}\right) \right\} + \text{const.} \quad (63)$$

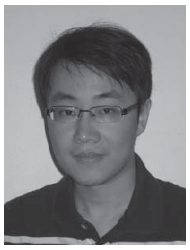
ACKNOWLEDGMENT

The authors would like to thank the anonymous referees and the Associate Editor for their comments.

REFERENCES

[1] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer-Verlag, 2006.
[2] G. J. McLachlan and D. Peel, *Finite Mixture Models*. New York: Wiley, 2000.
[3] E. López-Rubio and E. J. Palomo, "Growing hierarchical probabilistic self-organizing graphs," *IEEE Trans. Neural Netw.*, vol. 22, no. 7, pp. 997–1008, Jul. 2011.
[4] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. New York: Wiley, 1997.
[5] M.-W. Mak and S.-Y. Kung, "Estimation of elliptical basis function parameters by the EM algorithm with application to speaker verification," *IEEE Trans. Neural Netw.*, vol. 11, no. 4, pp. 961–969, Jul. 2000.
[6] S. Waterhouse, D. MacKay, and T. Robinson, "Bayesian methods for mixtures of experts," in *Proc. Neural Inf. Process. Syst.*, 1995, pp. 351–357.
[7] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.
[8] C. P. Robert, *The Bayesian Choice*. New York: Springer-Verlag, 2001.
[9] C. E. Rasmussen, "A practical Monte Carlo implementation of Bayesian learning," in *Proc. Neural Inf. Process. Syst.*, 1995, pp. 598–604.
[10] D. Barber and B. Schottky, "Radial basis functions: A Bayesian treatment," in *Proc. Neural Inf. Process. Syst.*, 1997, pp. 1–7.
[11] D. Husmeier, "The Bayesian evidence scheme for regularizing probability-density estimating neural networks," *Neural Comput.*, vol. 11, no. 12, pp. 2685–2717, 2000.
[12] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York: Springer-Verlag, 1999.
[13] D. Husmeier, W. D. Penny, and S. J. Roberts, "An empirical evaluation of Bayesian sampling with hybrid Monte Carlo for training neural network classifiers," *Neural Netw.*, vol. 12, nos. 4–5, pp. 677–705, 1999.
[14] S. M. Lewis and A. E. Raftery, "Estimating Bayes factors via posterior simulation with the Laplace-Metropolis estimator," *J. Amer. Stat. Assoc.*, vol. 92, no. 438, pp. 648–655, 1997.
[15] G. E. Hinton and D. van Camp, "Keeping neural network simple by minimizing the description length of the weights," in *Proc. 6th Annu. Conf. Comput. Learn. Theory*, 1993, pp. 5–13.
[16] D. J. C. MacKay, "Developments in probabilistic modelling with neural networks – ensemble learning," in *Proc. 3rd Annu. Symp. Neural Netw.: Artif. Intell. Ind. Appl.*, 1995, pp. 191–198.
[17] D. Barber and C. M. Bishop, "Ensemble learning for multilayer networks," in *Proc. Neural Inf. Process. Syst.*, 1997, pp. 1–7.
[18] D. Barber and C. M. Bishop, "Ensemble learning in Bayesian neural networks," in *Neural Networks and Machine Learning*. New York: Springer-Verlag, 1998, pp. 215–237.
[19] M. N. Gibbs and D. J. MacKay, "Variational Gaussian process classifiers," *IEEE Trans. Neural Netw.*, vol. 11, no. 6, pp. 1458–1464, Nov. 2000.
[20] C. M. Bishop, "Variational learning in graphical models and neural networks," in *Proc. Int. Conf. Artif. Neural Netw.*, 1998, pp. 13–22.
[21] M. J. Beal and Z. Ghahramani, "The variational Bayesian EM algorithm for incomplete data: With application to scoring graphical model structures," in *Bayesian Statistics*, J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, Eds. New York: Oxford Univ. Press, 2003, pp. 453–464.
[22] H. Snoussi and A. Mohammad-Djafari, "Bayesian unsupervised learning for source separation with mixture of Gaussians prior," *J. VLSI Signal Process. Syst. Signal, Image Video Technol.*, vol. 37, nos. 2–3, pp. 263–279, Jun.–Jul. 2004.
[23] A. E. Teschendorff, Y. Wang, N. L. Barbosa-Morais, J. D. Brenton, and C. Caldas, "A variational Bayesian mixture modelling framework for cluster analysis of gene-expression data," *Bioinformatics*, vol. 21, no. 13, pp. 3025–3033, 2005.
[24] N. Bali and A. Mohammad-Djafari, "A variational Bayesian algorithm for BSS problem with hidden Gauss-Markov models for the sources," in *Proc. ICA*, 2007, pp. 137–144.
[25] H. Snoussi and A. Mohammad-Djafari, "Estimation of structured Gaussian mixtures: The inverse EM algorithm," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3185–3191, Jul. 2007.
[26] A. Mohammad-Djafari, "A variational Bayesian algorithm for inverse problem of computed tomography," in *Proc. Math. Methods Biomed. Imaging Intens.-Modul. Radiat. Therapy*, 2008, pp. 231–252.
[27] H. Ayasso and A. Mohammad-Djafari, "Joint image restoration and segmentation using Gauss-Markov-Potts prior models and variational Bayesian computation," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2265–2277, Nov. 2010.
[28] A. Rakotomamonji, R. Flamary, G. Gasso, and S. Canu, "$\ell_p - \ell_q$ penalty for sparse linear and sparse multiple kernel multitask learning," *IEEE Trans. Neural Netw.*, vol. 22, no. 8, pp. 1307–1320, Aug. 2011.
[29] G. Skolidis and G. Sanguinetti, "Bayesian multitask classification with Gaussian process priors," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2011–2021, Dec. 2011.
[30] H. Attias, "Inferring parameters and structure of latent variable models by variational Bayes," in *Proc. UAI*, 1999, pp. 21–30.
[31] H. Attias, "A variational Bayes framework for graphical models," in *Proc. Neural Inf. Process. Syst.*, 1999, pp. 209–215.
[32] A. Corduneanu and C. M. Bishop, "Variational Bayesian model selection for mixture distributions," in *Proc. 8th Int. Conf. Artif. Intell. Stat.*, 2001, pp. 27–34.
[33] B. Wang and D. M. Titterington, "Convergence properties of a general algorithm for calculating variational Bayesian estimates for a normal mixture model," *Bayesian Anal.*, vol. 1, no. 3, pp. 625–650, 2006.
[34] N. Bouguila, D. Ziou, and J. Vaillancourt, "Unsupervised learning of a finite mixture model based on the Dirichlet distribution and its application," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1533–1543, Nov. 2004.
[35] N. Bouguila and D. Ziou, "Unsupervised selection of a finite Dirichlet mixture model: An MML-based approach," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 8, pp. 993–1009, Aug. 2006.

[36] P. Diaconis and D. Ylvisaker, "Conjugate priors for exponential families," *Ann. Stat.*, vol. 7, no. 2, pp. 269–281, 1979.
[37] E. Castillo, A. S. Hadi, and C. Solares, "Learning and updating of uncertainty in Dirichlet models," *Mach. Learn.*, vol. 26, no. 1, pp. 43–63, 1997.
[38] Z. Ma and A. Leijon, "Bayesian estimation of beta mixture models with variational inference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2160–2173, Nov. 2011.
[39] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," in *Learning in Graphical Models*. Norwell, MA: Kluwer, 1998, pp. 105–162.
[40] C. M. Bishop, N. Lawrence, T. Jaakola, and M. I. Jordan, "Approximating posterior distributions in belief networks using mixtures," in *Proc. Neural Inf. Process. Syst.*, 1997, pp. 1–7.
[41] N. D. Lawrence, C. M. Bishop, and M. I. Jordan, "Mixture representations for inference and learning in Boltzmann machines," in *Proc. UAI*, 1998, pp. 320–327.
[42] M. M. Ichir and A. Mohammad-Djafari, "A mean field approximation approach to blind source separation with L_p priors," in *Proc. EUPISCO*, 2005, pp. 1–4.
[43] G. Parisi, *Statistical Field Theory*. Reading, MA: Addison-Wesley, 1988.
[44] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[45] D. J. C. Mackay, "Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks," *Netw.: Comput. Neural Syst.*, vol. 6, no. 3, pp. 469–505, 1995.
[46] H. H. Thodberg, "A review of Bayesian neural networks with an application to near infrared spectroscopy," *IEEE Trans. Neural Netw.*, vol. 7, no. 1, pp. 56–72, Jan. 1996.
[47] B. Wang and D. M. Titterington, "Convergence and asymptotic normality of variational Bayesian approximations for exponential family models with missing values," in *Proc. UAI*, 2004, pp. 577–584.
[48] N. Nasios and A. G. Bors, "Variational learning for Gaussian mixture models," *IEEE Trans. Syst., Man, Cybern. Part B: Cybern.*, vol. 36, no. 4, pp. 849–862, Aug. 2006.
[49] V. Viitaniemi and J. Laaksonen, "Techniques for still image scene classification and object detection," in *Proc. Int. Conf. Artif. Neural Netw.*, 2006, pp. 35–44.
[50] J. S. Bonet and P. Viola, "Structure driven image database retrieval," in *Proc. Neural Inf. Process. Syst.*, 1997, pp. 1–8.
[51] A. Quattoni, M. Collins, and T. Darrell, "Conditional random fields for object recognition," in *Proc. Neural Inf. Process. Syst.*, 2004, pp. 1–8.
[52] S. Boutemedjet, D. Ziou, and N. Bouguila, "Unsupervised feature selection for accurate recommendation of high-dimensional image data," in *Proc. Neural Inf. Process. Syst.*, 2007, pp. 177–184.
[53] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. 8th Eur. Conf. Workshop Stat. Learn. Comput. Vis.*, 2004, pp. 1–16.
[54] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via pLSA," in *Proc. 9th Eur. Conf. Comput. Vis.*, 2006, pp. 517–530.
[55] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Mach. Learn.*, vol. 42, nos. 1–2, pp. 177–196, 2001.
[56] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[57] C. Elkan, "Using the triangle inequality to accelerate K-means," in *Proc. Int. Conf. Mach. Learn.*, 2003, pp. 147–153.
[58] D. E. Denning, "An intrusion-detection model," *IEEE Trans. Softw. Eng.*, vol. 13, no. 2, pp. 222–232, Feb. 2006.
[59] M. Bahrololum and M. Khaleghi, "Anomaly intrusion detection system using hierarchical Gaussian mixture model," *Int. J. Comput. Sci. Netw. Secur.*, vol. 8, no. 8, pp. 264–271, 2008.
[60] S. Northcutt and J. Novak, *Network Intrusion Detection: An Analyst's Handbook*. Indianapolis, IN: New Riders Publishing, 2002.
[61] S. Chebrolu, A. Abraham, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," *Comput. Secur.*, vol. 24, no. 4, pp. 295–307, 2005.
[62] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data," in *Applications of Data Mining in Computer Security*. Norwell, MA: Kluwer, 2002, pp. 77–99.
[63] A. K. Ghosh and A. Schwartzbard, "A study in using neural networks for anomaly and misuse detection," in *Proc. 8th Conf. USENIX Secur. Symp.*, vol. 8. 1999, p. 12.
[64] M. W. Woolrich and T. E. Behrens, "Variational Bayes inference of spatial mixture models for segmentation," *IEEE Trans. Med. Imaging*, vol. 25, no. 10, pp. 1380–1391, Oct. 2006.
[65] J. A. Palmer, "Relative convexity," Dept. Electr. Comput. Eng., Univ. California, San Diego, Tech. Rep., 2003.



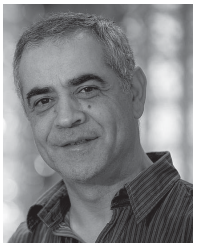
Wentao Fan received the B.A. degree in computer science and the M.Sc. degree in information systems security from Concordia University, Montreal, QC, Canada, in 2007 and 2009, respectively. He is currently pursuing the Ph.D. degree with the same university.

His current research interests include machine learning, data mining, and pattern recognition.

Nizar Bouguila (S'05–M'06–SM'11) received the Engineer degree from the University of Tunis, Tunis, Tunisia, in 2000, and the M.Sc. and Ph.D. degrees in computer science from Sherbrooke University, Sherbrooke, QC, Canada, in 2002 and 2006, respectively.

He is currently an Associate Professor with the Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC. His current research interests include image processing, machine learning, data mining, 3-D graphics, computer vision, and pattern recognition.

Dr. Bouguila received the Best Ph.D. Thesis Award in Engineering and Natural Sciences from Sherbrooke University in 2007. He received the Prestigious Prix d'Excellence de l'Association des Doyens des Études Supèrieures au Quebec (Best Ph.D. Thesis Award in Engineering and Natural Sciences in Quebec), and was a runner-up for the Prestigious Natural Sciences and Engineering Research Council Doctoral Prize.



Djemel Ziou received the B.E. degree in computer science from the University of Annaba, Annaba, Algeria, in 1984, and the Ph.D. degree in computer science from the Institut National Polytechnique de Lorraine, Nancy, France, in 1991.

He served as a Lecturer in several universities in France from 1987 to 1993. He was a Researcher with the Centre de Recherche en Informatique de Nancy and the Institut National de Recherche en Informatique et Automatique, Grenoble, France. He is currently a Full Professor with the Department of Computer Science, Sherbrooke University, Sherbrooke, QC, Canada. He holds the Natural Sciences and Engineering Research Council/Bell Canada Research Chair in personal imaging. His current research interests include image processing, information retrieval, computer vision, and pattern recognition.

Dr. Ziou has served on numerous conference committees as member or chair. He heads the Laboratory MOdélisation en Imagerie, Vision et RÉseaux de neurones (MOIVRE) and the consortium CoRIMedia, which he founded.