A STUDY ON ENTROPY-BASED VARIATIONAL LEARNING FOR MIXTURE MODELS

Mohammad Sadegh Ahmadzadeh

A thesis
in
The Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied (Electrical and Computer Engineering)
Concordia University
Montréal, Québec, Canada

December 2020

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared

By: **Mohammad Sadegh Ahmadzadeh**

Entitled: **A STUDY ON ENTROPY-BASED VARIATIONAL LEARNING FOR MIXTURE MODELS**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair and Internal Examiner

Dr. Dongyu Qiu

_____ External Examiner, BCEE

Dr. Bruno Lee

_____ Supervisor

Dr. Nizar Bouguila

Approved _____

Dr. Akshay Kumar Rathore, Graduate Program Director

_____ 17th Dec 2020 _____ _____

Dr. Mourad Debbabi, Dean

Faculty of Engineering and Computer Science

Abstract

A STUDY ON ENTROPY-BASED VARIATIONAL LEARNING FOR MIXTURE MODELS

Mohammad Sadegh Ahmadzadeh

Nowadays, we observe a rapid growth of complex data in all formats due to the technological development. Thanks to the field of machine learning, we can automatically analyze and infer useful information from these data. In particular, data clustering is regarded as one of the most famous data analysis tools aiming at grouping data with similar patterns into the same cluster. Among existing clustering techniques, finite mixture models have shown great flexibility in data modeling. Mixture models are a common unsupervised learning technique that have been widely used to statistically approximate and analyse heterogenous data. The goal of using mixture models is to fit the data into an appropriate distribution. A crucial point is to estimate the prefect parameters of the distribution and the suitable number of clusters in the data. To do so, an entropy-based variational learning algorithm is proposed for the model selection (i.e. determination of the optimal number of components). We investigate if a given component is genuinely distributed according to a mixture model to select the optimal number of components that better suits our data.

In our work we have used the variational inference framework that overcomes the over-fitting problem of maximum likelihood approaches and at the same time convergence is guaranteed. In addition, it decreases the computational complexity of purely Bayesian approaches. In recent researches the main concern when deploying mixture models has been the choice of distributions. The effectiveness of Dirichlet family of distributions has been proved in recent studies especially for non-Gaussian data.

In this thesis, an effective mixture model-based approach for clustering and modeling purposes has been proposed. Our contribution is the application of an entropy-based variational inference algorithm to learn the mixture models, namely, generalized

inverted Dirichlet and inverted Beta-Liouville mixture models. The performance of the proposed model is evaluated on multiple real-world applications such as human activity recognition, images, texture and breast cancer datasets, where in each case we compare our results with popular and similar models.

Acknowledgments

First of all, I want to reach out to my supervisor Prof Dr. Nizar Bouguila, to express my deepest gratitude for all the support and guidance he has given me during the last two years. His supervision truly had a positive impact in my life and I can never thank him enough. Despite my slow start, he gave me the motivation to find myself and believe in my abilities. He has thought me not only knowledge but things for the long run. Thank you for the opportunity and patient. Thank you for opening my eyes to machine learning. I can not show my gratitude enough with words. I can only say from the bottom of my heart THANK YOU and I will never forget all you have thought me.

I would like to show my gratitude to Narges Manouchehri, she has always been by my side, especially in the beginning where she patiently thought me all that I needed to know before starting my thesis. The completion of my thesis would have been very hard without her support.

A special thanks to Dr. Wentao Fan and Dr. Manar Amayri who supported me during my studies.

I would like to thank Hafsa Ennajari for her help in my publications. I have learned so much from her and I will always be grateful.

I would like to give a special thanks to Kamal Maanicshah, he has helped more than I asked. his unconditional help will never be forgotten. Thank you for being there for me in the times of struggle.

I would like to thank my friends who have been by my side in ups and downs. They have always kept me motivated in my studies and my job. Thank you Shakiba, Mohammad, Kiana, Pegah, Ali, and my other friends.

I consider myself very lucky to work beside my lab mates. Their hard work and passion have motivated me to become a better version of myself. Would like to thank Maryam, Pantea, Mahsa, Behnam, Azin, Hieu, Meeta, and other lap mates, for their

help and for being beside me all along.

Last but not least, I would like to express my deepest gratitude to my parents, Dr. Ahmadzadeh, and Mahboubeh for their support and help, financially and emotionally. The only thing that kept me going was to make them proud. They have always encouraged me to follow my dreams. I am truly blessed to have them. I would like to thank my two beautiful and kind sisters, Reyhaneh and Sara, for their love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Cluster Analysis via Finite Mixture Models

Nowadays, large amounts of complex data in various formats (e.g., image, text, speech) are generated increasingly at a bottleneck speed. This increase motivated data scientists to develop tactical models in order to automatically analyze and infer useful knowledge from these data [1].

Data mining approaches have been used to gain useful information from data by using computational models. In general, data mining models can be grouped roughly in two categorizes: predictive and descriptive models [2]. Descriptive models define the relationships within the data with pattern discovery [3], and predictive models, predict the future behaviour of data as opposed to giving information about known behavior [4]. One of the most used methods of data mining is clustering [5]. The main definition of clustering data, is gathering similar data in one cluster, therefore resulting multiple clusters where each cluster holds one group of data but is distinguished from each-other in other words, patterns in one cluster should be more similar to each other than patterns of other clusters [6]. It is noteworthy to mention that clustering could be confused with classification. The difference between these two methods is that clustering is an unsupervised learning method and classification is considered as a supervised one. In clustering, the prior information about the clusters are unknown. A critical challenge in clustering is selecting the number of clusters [7].

In this context, statistical modeling plays a significant role in helping machines to interpret data with statistics. An essential approach in statistical modeling is finite

mixture models that are effectively used for clustering purposes, separating heterogeneous data into homogeneous groups [8]. The usefulness of mixture models has been widely demonstrated in many application areas including pattern recognition, text and image analysis, and smart buildings [9, 10, 11]. However, there exist several challenges to address when working with mixture models: (1) In finite mixture models data samples are described by a mixture of several components, where each component is assumed to come from one specific distribution (e.g., Gaussian distribution). The objective is to estimate the unknown parameters of the distribution which will properly suit the data [12]. Most existing related works assume that data samples are mainly drawn from a Gaussian distribution [12, 13]. However, this assumption has made the applicability of Gaussian mixture models very limited as this type of distribution is not suitable for all kinds of data. Lately, multiple studies have shown that other non-Gaussian statistical models (e.g., scaled Dirichlet [14], generalized inverted Dirichlet [15], Beta [16], inverted Beta-Liouville [17], etc.) are effective in modeling data. Thus, choosing a suitable probability distribution that better describes the nature and the properties of the observed data is crucial to the assessment of the validity of the model. For instance, the inverted Dirichlet mixture, has good flexibility in accepting different symmetric and asymmetric forms that results in better generalization capabilities. But, the model usually supposes that the features of the vectors are positively correlated, and that is not always applicable for real-life applications. (2) In most cases, the mixture model fitting is not straightforward and analytically intractable. Methods like expectation-maximization (EM) and maximum likelihood [1] are widely used in this context, but they remain impractical as they are sensitive to initialization and usually lead to over-fitting [18]. An alternative approach to solve these problems is Bayesian learning, particularly, variational inference has made the parameter estimation process more computationally efficient. Variational learning method has been proposed and tested in different models [19, 20]. The main idea of the variational method is that it proceeds to approximate the true posterior distribution rather than computing it. Therefore, convergence is guaranteed as the complexity of the model is reduced [21]. In the following chapters the theory of variational parameter estimation will be explored. (3) The selection of the number of components is an important issue to consider in the design of mixture models, because a high number of components may lead to learning the data too much, whereas

inference under a model with a small number of components can be biased. To this end, multiple effective methods have been proposed, like minimum message length criterion [22].

Furthermore, we propose an entropy-based variational learning algorithm to select the optimal number of mixture components. Initially, we start with one component, and continue incrementally to find the perfect number of components. We proceed to define the model complexity and initiate a comparison between the theoretical entropy and estimated one to approximate the prefect number of components [23]. Moreover, if a component is found to be unsuitable for our data, we proceed to split it into two new components. This method has shown to be effective due to the fact that it follows the variational learning approach. This method was proposed first in [24] and followed in [25]. A full description and detailed algorithms of the entropy-based variational approach are explained in Chapter 2 and Chapter 3. In [26] and [25], the authors have studied the entropy-based variational learning on Beta-Liouville and generalized Dirichlet mixture models, respectively. Our goal is to study the entropy-based model when applied to generalized inverted Dirichlet (GID) and inverted Beta-Liouville (IBL) mixture models. We have studied these models in multiple applications like image categorization, breast cancer and human activity recognition in smart buildings.

1.2 Contribution

The main purpose of this thesis is to study the efficiency of GID and IBL models when combined with the entropy-based variational learning algorithm. The contributions are listed as follows:

- ☞ **Entropy-based Variational Learning of Generalized Inverted Dirichlet Mixture Model**

  We propose a finite generalized inverted Dirichlet mixture model for unsupervised learning using entropy-based variational approximation procedure. We validate our model on some real-world applications including breast cancer and image categorization. This work has been submitted to the 22^{nd} *IEEE International Conference on Industrial Technology.*

☞ **Entropy-based Variational Learning of Inverted Beta-Liouville Mixture Model**

A finite inverted Beta-Liouville mixture model merged with a splitting process known as the entropy-based variational learning method has been proposed. The evaluation of our model is performed by some challenging applications, namely, human activity recognition and image categorization. This work has been submitted to the 34^{th} *International FLAIRS conference*.

1.3 Thesis Overview

- ❒ Chapter 1 is delegated to introducing the idea of mixture models and clustering, In addition we briefly overview several concepts that are related to our work.
- ❒ Chapter 2 is delegated to the explanation of entropy-based variational learning of generalized inverted Dirichlet mixture model. The model has been challenged with two applications, namely, breast cancer Wisconsin (diagnostic) dataset and image categorization.
- ❒ In chapter 3 we extend our research on entropy-based variational for learning the inverted Beta-Liouville mixture models. At the end, we have shown in details the results of our experiments on human activity recognition and image categorization applications.
- ❒ In conclusion, we briefly summarize our contributions.

Chapter 2

Entropy-based Variational Learning of Finite Generalized Inverted Dirichlet Mixture Model

2.1 Model Specification

In this chapter, we present our discoveries when entropy-based variational algorithm is applied for learning a finite generalized inverted Dirichlet mixture model. This will help us to study and resolve the parameter estimation and model selection problems for a higher quality fitting. At the end we conclude the chapter by showing our results on applications, namely, breast cancer Wisconsin (diagnostic) dataset and image categorization that demonstrate the superior performance of our proposed model.

2.1.1 Finite Generalized Inverted Dirichlet Mixture Model

Lets us assume $\vec{Y} = (\vec{Y}_1, \ldots, \vec{Y}_N)$ is a set of N independent identically distributed vectors, where every single $\vec{Y}_i$ is defined as $\vec{Y}_i = (Y_{i1}, \ldots, Y_{iD})$, where D is the dimensionality of the vector. We are assuming that each $\vec{Y}_i$ follows a mixture of GIDs, where the probability density function of the GID is given by [27],[28]:

$$p(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_{jd} + \beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})} \frac{Y_{id}^{\alpha_{jd}-1}}{(1 + \sum_{l=1}^{d} Y_{il})^{\gamma_{jd}}} \tag{1}$$

where $\vec{\alpha}_j$ and $\vec{\beta}_j$ are the parameters of the GID, and they are defined as $\vec{\alpha}_j = (\alpha_{j1}, \ldots, \alpha_{jd})$ and $\vec{\beta}_j = (\beta_{j1}, \ldots, \beta_{jd})$ with constraints $\alpha_{jd} > 0$ and $\beta_{jd} > 0$. We can find γ_{id} according to $\gamma_{id} = \beta_{jd} + \alpha_{jd} - \beta_{j(d+1)}$. Supposing that the model consists of M different components [1], we are able to define the GID mixture model as follows:

$$p(\vec{Y}_i \mid \vec{\pi}, \vec{\alpha}, \vec{\beta}) = \sum_{j=1}^{M} \pi_j p(\vec{Y}_i \mid \vec{\alpha}_j, \vec{\beta}_j) \tag{2}$$

where $\vec{\pi}$ represents its mixing coefficients correlated with the components, where, $\vec{\pi} = (\pi_1, \ldots, \pi_M)$ with constrains $\pi_j \geq 0$ and $\sum_{j=1}^{M} \pi_j = 1$, and the shape parameters of the distribution are denoted as $\vec{\alpha} = (\vec{\alpha}_1, \ldots, \vec{\alpha}_M)$, $\vec{\beta} = (\vec{\beta}_1, \ldots, \vec{\beta}_M)$ and $j = 1, \ldots, M$. According to [15], we can replace the GID distribution with a product of D inverted Beta distributions, considering that it does not change the model, therefore, equation (2) can be rewritten as:

$$p(\mathcal{X} \mid \pi, \alpha, \beta) = \prod_{i=1}^{N} \left(\sum_{j=1}^{M} \pi_j \prod_{l=1}^{D} p_{iBeta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) \right) \tag{3}$$

By considering that $\mathcal{X} = (\vec{X}_1, \ldots, \vec{X}_N)$ where $\vec{X}_i = (X_{i1}, \ldots, X_{iD})$, we have $X_{il} = Y_{il}$ and $X_{il} = \frac{Y_{il}}{1+\sum_{k=1}^{l-1} Y_{ik}}$ for $l > 1$. The inverted Beta distribution is defined by $P_{iBeta}(X_{il}|\alpha_{jl}, \beta_{jl})$ with the parameters α_{jl} and β_{jl} and given by:

$$p_{iBeta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) = \frac{\Gamma(\alpha_{jl} + \beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} \frac{{X_{il}}^{\alpha_{jl}-1}}{(1 + X_{il})^{\alpha_{jl}+\beta_{jl}}} \tag{4}$$

In proportion to this design, we are able to estimate the parameters from equation (3) instead of the equation (2). We define the latent variables as $\mathcal{Z} = (\vec{Z}_1, \ldots, \vec{Z}_N)$ where $\vec{Z}_i = (Z_{i1}, \ldots, Z_{iM})$ with the conditions $Z_{ij} \in \{0, 1\}$ that Z_{ij} is equal to 1 if $\vec{X}_i$ is assigned to cluster j and zero otherwise, and $\sum_{j=1}^{M} Z_{ij} = 1$ [27]. The conditional probability for the latent variables $\mathcal{Z}$ given $\vec{\pi}$ can be written as:

$$p(\mathcal{Z} \mid \vec{\pi}) = \prod_{i=1}^{N} \prod_{j=1}^{M} {\pi_j}^{Z_{ij}} \tag{5}$$

We write the probability of the observed data vectors $\mathcal{X}$ given the latent variable and component parameters as:

$$p(\mathcal{X} \mid \mathcal{Z}, \vec{\alpha}, \vec{\beta}) = \prod_{i=1}^{N} \prod_{j=1}^{M} \Big(\prod_{l=1}^{D} p_{iBeta}(X_{il} \mid \alpha_{jl}, \beta_{jl}) \Big)^{Z_{ij}} \tag{6}$$

By assuming that the parameters are independent and positive, we can suppose that the priors of these parameters are Gamma distributions $\mathcal{G}(.)$. According to [29], we can describe them as:

$$p(\alpha_{jl}) = \mathcal{G}(\alpha_{jl} \mid u_{jl}, \nu_{jl}) = \frac{\nu_{jl}^{u_{jl}}}{\Gamma(u_{jl})} {\alpha_{jl}}^{u_{jl}-1} e^{-\nu_{jl}\alpha_{jl}} \tag{7}$$

$$p(\beta_{jl}) = \mathcal{G}(\beta_{jl} \mid g_{jl}, h_{jl}) = \frac{h_{jl}^{g_{jl}}}{\Gamma(g_{jl})} {\beta_{jl}}^{g_{jl}-1} e^{-h_{jl}\beta_{jl}} \tag{8}$$

A graphical representation of GID model is shown in Fig 1. We define the joint distribution including all random variables, as follows:

$$p(\mathcal{X}, \mathcal{Z}, \vec{\alpha}, \vec{\beta} \mid \vec{\pi}) = p(\mathcal{X} \mid \mathcal{Z}, \vec{\alpha}, \vec{\beta}) p(\mathcal{Z} \mid \vec{\pi}) p(\vec{\alpha}) p(\vec{\beta}) \tag{9}$$

$$\begin{aligned} p(\mathcal{X}, \mathcal{Z}, \vec{\alpha}, \vec{\beta} \mid \vec{\pi}) = & \prod_{i=1}^{N}\prod_{j=1}^{M} \left(\prod_{l=1}^{D} \frac{\Gamma(\alpha_{jl}+\beta_{jl})}{\Gamma(\alpha_{jl})\Gamma(\beta_{jl})} \frac{X_{il}^{\alpha_{jl}-1}}{(1+X_{il})^{\alpha_{jl}+\beta_{jl}}} \right)^{Z_{ij}} \left(\prod_{i=1}^{N}\prod_{j=1}^{M} \pi_j^{Z_{ij}} \right) \\ & \prod_{j=1}^{M}\prod_{l=1}^{D} \left(\frac{\nu_{jl}^{u_{jl}}}{\Gamma(u_{jl})} {\alpha_{jl}}^{u_{jl}-1} e^{-\nu_{jl}\alpha_{jl}} \times \frac{h_{jl}^{g_{jl}}}{\Gamma(g_{jl})} {\beta_{jl}}^{g_{jl}-1} e^{-h_{jl}\beta_{jl}} \right) \end{aligned} \tag{10}$$

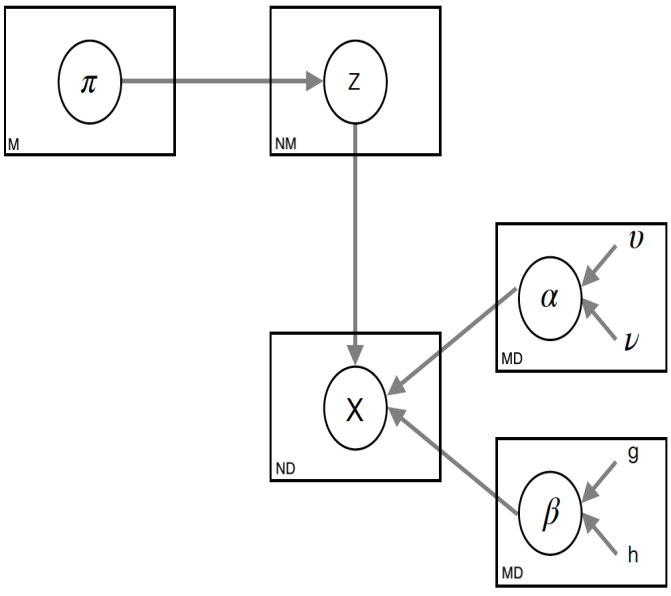

Figure 1: A graphical representation of the GID mixture model. The circles symbolize the model parameters and random variables, and plates indicate the repetitions shown in bottom left corners. The arcs symbolize the conditional dependencies of the variables.

2.2 Model Learning with Variational Inference

The GID mixture model contains hidden variables that can not be estimated directly. In order to estimate them, we apply the variational inference method, in which we aim to find an approximation of the posterior probability distribution of $p(\Theta|\mathcal{X}, \vec{\pi})$ by having $\Theta = \{\mathcal{Z}, \vec{\alpha}, \vec{\beta}\}$. Inspired by [25], we introduce $Q(\Theta)$ as an approximation of the true posterior distribution $p(\Theta|\mathcal{X}, \vec{\pi})$. We make use of the Kullback-Leibler (KL) divergence in order to minimize the difference between the true posterior distribution and the approximated one, which can be expressed as follows:

$$KL\big(Q \mid\mid P\big) = -\int Q\big(\Theta\big) \ln\left(\frac{p\big(\Theta \mid \mathcal{X}, \vec{\pi}\big)}{Q\big(\Theta\big)}\right) \mathrm{d}\Theta = \ln\ p\big(\mathcal{X} \mid \vec{\pi}\big) - \mathcal{L}\big(Q\big) \tag{11}$$

where $\mathcal{L}(Q)$ is defined as:

$$\mathcal{L}\big(Q\big) = \int Q\big(\Theta\big) \ln\left(\frac{p\big(\mathcal{X}, \Theta \mid \vec{\pi}\big)}{Q\big(\Theta\big)}\right) \mathrm{d}\Theta \tag{12}$$

Starting from the fact that $\mathcal{L}(Q) \leq \ln p(\mathcal{X}|\vec{\pi})$, we can see that $\mathcal{L}(Q)$ is the lower bound of the log likelihood. Thus, we have to maximize $\mathcal{L}(Q)$ in order to minimize the KL divergence. We assume a factorization assumption around $Q(\Theta)$ to apply it in variational inference. This assumption is called the Mean Field Approximation. We can factorize the posterior distribution $Q(\Theta)$ as $Q(\Theta) = Q(\mathcal{Z})Q(\vec{\alpha})Q(\vec{\beta})Q(\vec{\pi})$ [30], [31]. In order to obtain a variational solution for the lower bound with respect to all the model parameters, we consider an optimal solution for a fix parameter s that is defined as $\ln Q_s^*\big(\Theta_s\big) = \langle \ln p(\mathcal{X}, \Theta)\rangle_{i\neq s}$ where $\langle\cdot\rangle_{i\neq s}$ refers to the expectation with respect to all the parameters apart from Θ_s, if an exponential is taken from both sides, the normalized equation is as follows.

$$Q_s\big(\Theta_s\big) = \frac{exp\big\langle \ln\ p\big(\mathcal{X}, \Theta\big)\big\rangle_{i\neq s}}{\int exp\big\langle \ln\ p\big(\mathcal{X}, \Theta\big)\big\rangle_{i\neq s} \mathrm{d}\Theta} \tag{13}$$

We obtain the optimal variational posteriors solution that are formulated as:

$$Q\big(\mathcal{Z}\big) = \prod_{i=1}^{N}\prod_{j=1}^{M} r_{ij}^{Z_{ij}} \tag{14}$$

$$Q\big(\vec{\alpha}\big) = \prod_{j=1}^{M}\prod_{l=1}^{D} \mathcal{G}\big(\alpha_{jl} \mid u_{jl}^*, \nu_{jl}^*\big)\ ,\ Q\big(\vec{\beta}\big) = \prod_{j=1}^{M}\prod_{l=1}^{D} \mathcal{G}\big(\beta_{jl} \mid g_{jl}^*, h_{jl}^*\big) \tag{15}$$

$$r_{ij} = \frac{\tilde{r}_{ij}}{\sum_{j=1}^{M} \tilde{r}_{ij}} \tag{16}$$

$$\ln \tilde{r}_{ij} = \ln \pi_j + \sum_{l=1}^{D} \tilde{R}_{jl} + (\bar{\alpha}_{jl} - 1) \ln X_{il} - (\bar{\alpha}_{jl} + \bar{\beta}_{jl}) \ln(1 + X_{il}) \tag{17}$$

$$\begin{aligned} \tilde{R} = \ln \frac{\Gamma(\bar{\alpha} + \bar{\beta})}{\Gamma(\bar{\alpha})\Gamma(\bar{\beta})} + \bar{\alpha}[\psi(\bar{\alpha} + \bar{\beta}) - \psi(\bar{\alpha})](\langle \ln \beta \rangle - \ln \bar{\beta}) + 0.5\alpha^2[\psi'(\bar{\alpha} + \bar{\beta}) \\ - \psi'(\bar{\alpha}))]\langle(\ln \alpha - \ln \bar{\alpha})^2\rangle + 0.5\beta^2[\psi'(\bar{\alpha} + \bar{\beta}) - \psi'(\bar{\beta}))]\langle(\ln \beta - \ln \bar{\beta})^2\rangle \\ + \bar{\alpha}\bar{\beta}\psi'(\bar{\alpha} + \bar{\beta})(\langle \ln \alpha \rangle - \ln \bar{\alpha})(\langle \ln \beta \rangle - \ln \bar{\beta}) \end{aligned} \tag{18}$$

$$u^*_{jl} = u_{jl} + \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_{jl} \left[\psi(\bar{\alpha}_{jl} + \bar{\beta}_{jl}) - \psi(\bar{\alpha}_{jl}) + \bar{\beta}_{jl}\psi'(\bar{\alpha}_{jl} + \bar{\beta}_{jl})(\langle \ln \beta_{jl} \rangle - \ln \bar{\beta}_{jl}) \right] \tag{19}$$

$$\nu^*_{jl} = \nu_{jl} - \sum_{i=1}^{N} \langle Z_{ij} \rangle \ln \frac{X_{il}}{1 + X_{il}} \tag{20}$$

$$g^*_{jl} = g_{jl} + \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\beta}_{jl} \left[\psi(\bar{\alpha}_{jl} + \bar{\beta}_{jl}) - \psi(\bar{\beta}_{jl}) + \bar{\alpha}_{jl}\psi'(\bar{\alpha}_{jl} + \bar{\beta}_{jl})(\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl}) \right] \tag{21}$$

$$h^*_{jl} = h_{jl} - \sum_{i=1}^{N} \langle Z_{ij} \rangle \ln \frac{1}{1 + X_{il}} \tag{22}$$

Furthermore $\psi(.)$ and $\psi^{'}(.)$ are representing the Digamma and Trigamma functions, respectively. As $R = \langle \ln \frac{\Gamma(\bar{\alpha}+\bar{\beta})}{\Gamma(\bar{\alpha})\Gamma(\bar{\beta})} \rangle$ is intractable, we have used the second order Taylor expansion for its approximation. The expected values of the above equations are as follows:

$$\langle Z_{ij} \rangle = r_{ij} \tag{23}$$

$$\bar{\alpha}_{jl} = \langle \alpha_{jl} \rangle = \frac{u^*_{jl}}{\nu^*_{jl}} \ , \ \langle \ln \alpha_{jl} \rangle = \psi(u^*_{jl}) - \ln \nu^*_{jl} \tag{24}$$

$$\bar{\beta}_{jl} = \langle \beta_{jl} \rangle = \frac{g^*_{jl}}{h^*_{jl}} \ , \ \langle \ln \beta_{jl} \rangle = \psi(g^*_{jl}) - \ln h^*_{jl} \tag{25}$$

$$\Big\langle (\ln \alpha_{jl} - \ln \bar{\alpha}_{jl})^2 \Big\rangle = [\psi(u^*_{jl}) - \ln u^*_{jl}]^2 + \psi^{'}(u^*_{jl}) \tag{26}$$

$$\Big\langle (\ln \beta_{jl} - \ln \bar{\beta}_{jl})^2 \Big\rangle = [\psi(g^*_{jl}) - \ln g^*_{jl}]^2 + \psi^{'}(g^*_{jl}) \tag{27}$$

$$\pi_j = \frac{1}{N} \sum_{i=1}^{N} r_{ij} \tag{28}$$

2.3 Entropy-based Variational Model Learning

In this section, we develop an entropy-based variational inference to learn the generalized inverted Dirichlet mixture model, that is mainly motivated by [23]. The core idea is to evaluate the quality of fitting of a component of our mixture model. Hence, we do a comparison between the theoretical maximum entropy and the MeanNN entropy [32]. In case of a significant difference, we proceed with a splitting process to fit the component, which consists in splitting the component into two new clusters.

2.3.1 Differential Entropy Estimation

The probability density function of an observation $\vec{X}_i = (\vec{X}_1, \ldots, \vec{X}_D)$ is defined as $p(\vec{X}_i)$, with a set of N samples $\{\vec{X}_i, \ldots, \vec{X}_N\}$, the differential entropy can be defined as:

$$H(\vec{X}_i) = - \int p(\vec{X}_i) log_2 \ p(\vec{X}_i) d\vec{X}_i \tag{29}$$

We introduce the maximum differential entropy of the GID as follows:

$$\begin{aligned} H_{GID}(\vec{X}_i \mid \alpha_j, \beta_j) = \sum_{l=1}^{D} \Big[& - \ln \Gamma(\alpha_{jl} + \beta_{jl}) + \ln \Gamma(\alpha_{jl}) + \ln \Gamma(\beta_{jl}) \\ & - (\alpha_{jl} - 1)\big[- \psi(\alpha_{jl} + \beta_{jl}) + \psi(\alpha_{jl}) \big] + (\alpha_{jl} + \beta_{jl})[-\psi(\alpha_{jl} + \beta_{jl})] \Big] \end{aligned} \tag{30}$$

2.3.2 MeanNN Entropy Estimator

In order to make sure that the specified component is indeed distributed according to a generalized inverted Dirichlet distribution, we choose the MeanNN entropy estimator [23], to estimate $H(\vec{X}_i)$ for random variable $\vec{X}_i$ with D dimensions, that has

an unknown density function $P(\vec{X}_i)$ [33]. By considering the fact that the Shannon entropy estimator in (29) can be considered equal to the average of $-\log P(\vec{X}_i)$, we can exploit an unbiased estimator by estimating $\log P(\vec{X}_i)$ [32], [33]. We assume that $\vec{X}_i$ is the center of a ball with diameter ϵ, and that there is a point within the distance $[\epsilon, \epsilon + d_\epsilon]$ from $\vec{X}_i$. We have $\hat{k} - 1$ points in a smaller distance, and the other $N - \hat{k} - 1$ points are within a large distance from $\vec{X}_i$. Consequently, we can define the probability of the distances and the k-th nearest neighbor as follows:

$$p_{i\hat{k}}(\epsilon) = \frac{(N-1)!}{\left(\hat{k}-1\right)!\left(N-\hat{k}-1\right)!} \frac{dp_i(\epsilon)}{d\epsilon} p_i^{\hat{k}-1} (1-p_i)^{N-\hat{k}-1} \tag{31}$$

where $p_i(\epsilon)$ denotes the mass of the ϵ-ball centered on $\vec{X}_i$:

$$p_i(\epsilon) = \int_{||\vec{X}-\vec{X}_i||<\epsilon} p(\vec{X}_i) d\vec{X}_i \tag{32}$$

We can easily define the expectation of $\log p_i(\epsilon)$ with respect to $p_i(\epsilon)$ as mentioned in equation (33):

$$\mathbb{E}\big(\log p_i(\epsilon)\big) = \int_0^\infty p_{i\hat{k}} \log p_i(\epsilon) d\epsilon = \psi(\hat{k}) - \psi(N) \tag{33}$$

Imagine $P(\vec{X}_i)$ is unchanging in the center of the ϵ-ball, we have $p_i(\epsilon) \simeq V_d \epsilon^d p(\vec{X}_i)$, where d corresponds to the dimension of $\vec{X}_i$, and V_d is the unit ball volume calculated by $V_d = \pi^{\frac{d}{2}} \Gamma(1 + d/2)$. Now, we are able to approximate $-\log p(\vec{X}_i)$ by substituting (32) into (33) we can get the equation (34). Hence, we get the unbiased K-NN estimator of the differential entropy, expressed in (35):

$$-\log p(\vec{X}_i) \simeq \psi(N) - \psi(\hat{k}) + dE(\log \epsilon) + \log V_d \tag{34}$$

$$H_{\hat{k}}\left(\vec{X}\right) = \psi(N) - \psi\left(\hat{k}\right) + \frac{d}{N} \sum_{i=1}^{N} log\epsilon_i + \log V_d \tag{35}$$

To reduce the high computational expenses of the K-NN estimator, we use an extension of the K-NN estimator called MeanNN, proposed in [24]. The main idea behind the MeanNN entropy estimator is to average the $\hat{k}$ nearest neighbor statics for all feasible values of order k in the range of $[1, N-1]$. The MeanNN estimator for the differential entropy is calculated according to (36).

$$H_M\left(\vec{X}\right) = \frac{1}{N-1}\sum_{\hat{k}=1}^{N-1} H_{\hat{k}}\left(\vec{X}\right) = \log V_d + \psi\left(N\right) + \frac{1}{N-1}\sum_{\hat{k}=1}^{N-1}\left[\frac{d}{N}\sum_{i=1}^{N}\log\epsilon_{i,\hat{k}} - \psi\left(\hat{k}\right)\right] \tag{36}$$

where $\epsilon_{i,\hat{k}}$ determines the $\hat{k}$-th nearest neighbor of $\vec{X}_i$. To find the maximum differential entropy of each individual cluster, we use:

$$H_{GID} = \sum_{j=1}^{M}\pi_j H_{GID}(j) \tag{37}$$

At this point, we are able to give an accurate evaluation of the model fitting, by evaluating and comparing the MeanNN and the theoretical maximum differential entropy [24]. Afterwards, we define Ω_{GID}, which is the normalized weighted sum of the difference between the theoretical and the estimated entropy of every component correlated with the generalized inverted Dirichlet mixture model, as expressed bellow:

$$\Omega_{GID} = \sum_{j=1}^{M}\pi_j\Big[\frac{H_{GID}(j) - H_M(j)}{H_{GID}(j)}\Big] = \sum_{j=1}^{M}\pi_j\Big[1 - \frac{H_M\left(j\right)}{H_{GID}(j)}\Big] \tag{38}$$

The normalized weight Ω_{GID} operates in the range of $[0, 1]$ and it is equal to zero, only if the data was genuinely distributed. The splitting process is performed by choosing the cluster j^* with the highest Ω_{GID} according to equation (39), and split the chosen component j^* into two new components.

$$j^* = \arg\max_j\Big[\Omega_{GID}(j)\Big] = \arg\max_j\left[\pi_j\frac{H_{GID}(j) - H_M(j)}{H_{GID}(j)}\right] \tag{39}$$

The overall entropy-based variational learning algorithm of the GID mixture model is illustrated in Algorithm 1.

Algorithm 1 Entropy-based variational learning of GID mixture models

1. Initialization
   - Set $M = 1$, $j^* = M$, $\pi_1 = 1$
   - Initialize hyperparameters u_{jl},ν_{jl},g_{jl},h_{jl}
2. The splitting process
   - Split j^* into two new components j_1 and j_2 with equal proportion $\pi^*/2$.
   - Set $M = M + 1$.
   - Initialize the parameters of j_1 and j_2 using the same parameters of j^*.
3. Apply standard variational Bayes until convergence.
4. Determine the number of components through the evaluation of the mixing coefficients π_j according to (28).
5. If $\pi_j \approx 0$. where $j \in 1, \ldots, M$ then set $M = M - 1$ and terminate the program.
6. Else evaluate Ω_{MD}, choose j^* according to (39) and go back to the splitting process in step 2.

2.4 Experimental Results

In order to demonstrate the effectiveness of the proposed model, Entropy-Based Variational Learning of Finite Generalized Inverted Dirichlet Mixture Model (EV-GIDMM), we conduct several experiments on two real-world challenging applications, including breast cancer detection and image categorization. In the first one, we used the standard breast cancer Wisconsin dataset with numerical features, whereas in the second one, we run our experiments on two other popular data sets, namely, Caltech101 and Describable Texture Dataset (DTD). To validate the performance of our model, we compared our proposed EV-GIDMM against three unsupervised state-of-the-art mixture models, including the Entropy-based variational inference on Multivariate Beta Mixture Model (EV-MBMM) [23], variational Dirichlet Mixture Model (varDMM) [29] and Entropy-based variational on Dirichlet Mixture Model

(EDMM) [25].

2.4.1 Breast Cancer

The first application that we considered to evaluate the performance of our proposed model is breast cancer detection. According to the WHO (World Health Organization), breast cancer has been declared as the most frequent cancer among women that affects about 2.1 million women every year. Machine learning techniques can be of great help in this context, in early detection of women breast cancer, thus, they can have a great impact on the breast cancer treatment. To this end, we applied our proposed model on the breast cancer Wisconsin dataset that is publicly available[1]. This dataset includes 569 data samples of patients seen by Dr. Wolberg, that have been diagnosed with either malignant or benign cancer. The number of patients having a benign tumor is 357, whereas 212 cases with malignant tumor cancer. This data set was obtained by applying the Fine Needle Aspiration (FNA) method [34], [35], and it contains cases showing invasive breast cancer and no sign of distant metastases. The first 30 features describe the characteristics of each nuclei cell in the images of the tissue. Table 1 shows the experimental results of our model as well as the baseline methods for the breast cancer detection task. We can see that our proposed EV-GIDMM successfully achieved the best accuracy on this task.

Table 1: Accuracy performance of our model and the baselines on the breast cancer dataset

Method	**Accuracy(%)**
EV-GIDMM	**93.1**
EV-MBMM	90.8
EDMM	89.7
varDMM	63.5

Furthermore, we have represented the confusion matrix for the breast cancer dataset by using the EV-GIDMM in Fig. 2. From the confusion matrix it can be inferred that in the case of malignant class our model is showing lower accuracy in

[1]https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

comparison to the benign class. However, we have misclassification percentage, low as 6.9%.

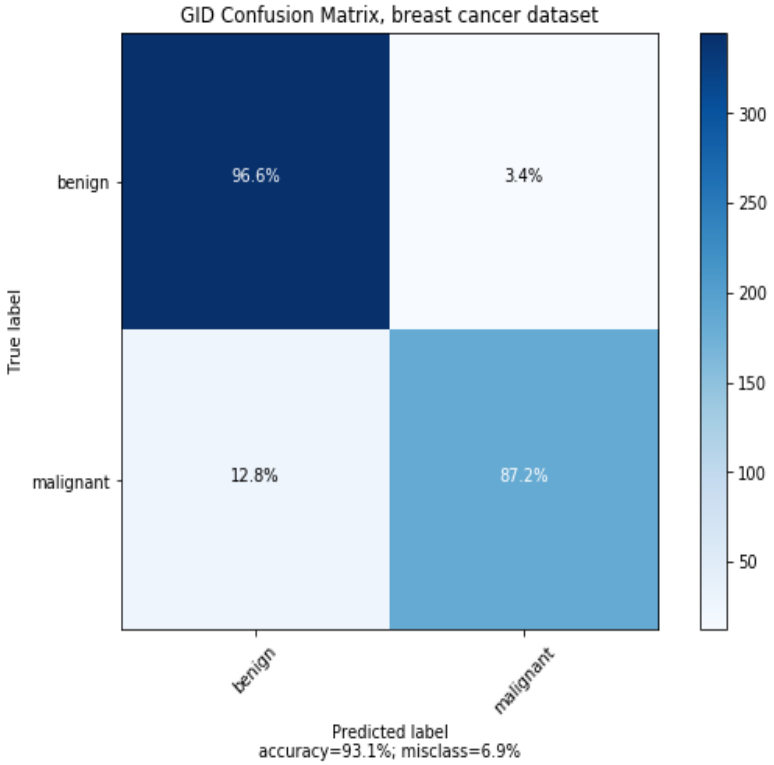

Figure 2: Confusion matrix of breast cancer dataset with EV-GIDMM.

2.4.2 Image analysis

We are now ready to evaluate the performance of the proposed approach on the image categorization task, which is a significant research topic and aims at classifying images into their corresponding category. To do so, we used two popular image data sets, namely, Caltech101 and Describable Texture Dataset (DTD). In this experiment, we first considered the Caltech101 image data set[2] [36], which originally contains a set of images depicting objects belonging to 101 classes, from which we selected three main object categories: Airplane, Sea Horse and Brain. Some sample images from this data set are illustrated in Fig. 3.

[2]http://www.vision.caltech.edu/Image_Datasets/Caltech101/

Figure 3: Sample images of each cluster from the Caltech101 dataset.

In order to use our model for the selected data set, we need to form a bag of visual words model (BoVW) [37]. Before applying the BoVW, we first need to apply some descriptor extraction method, that, we choose SIFT [38]. Therefore we extract the features with the help of SIFT and then apply K-means clustering on the descriptors extracted with SIFT from the image. As a result a BoVW feature vector is formed for each image. Our experiments revealed that the SIFT method is more suitable for our selected data set, resulting in more discriminative descriptors. After applying SIFT to all images, we obtain a matrix that serves as an input for our model. We report the results of this experiment in Table 5, which shows that our proposed model outperformed all the baseline methods in image clustering, with a considerable accuracy margin of almost 6.7%.

Table 2: Accuracy comparison of our proposed model and the baseline methods on the Caltech101 data set.

Method	**Accuracy(%)**
EV-GIDMM	**91**
EV-MBMM	84.3
EDMM	74.9
varDMM	40.3

In Fig. 4 we have illustrated the confusion matrix of Caltech101 dataset for the EV-GIDMM, with the classes: Airplane, Brain, Sea Horse. As we can see the misclassification is mostly concentrated in the Sea Horse class and the reason is that images in this class hold objects and scenes in the background that could be mistaken with airplane and brain. We observed that the Sea Horse is a noisy class, however, our model has achieved a great accuracy.

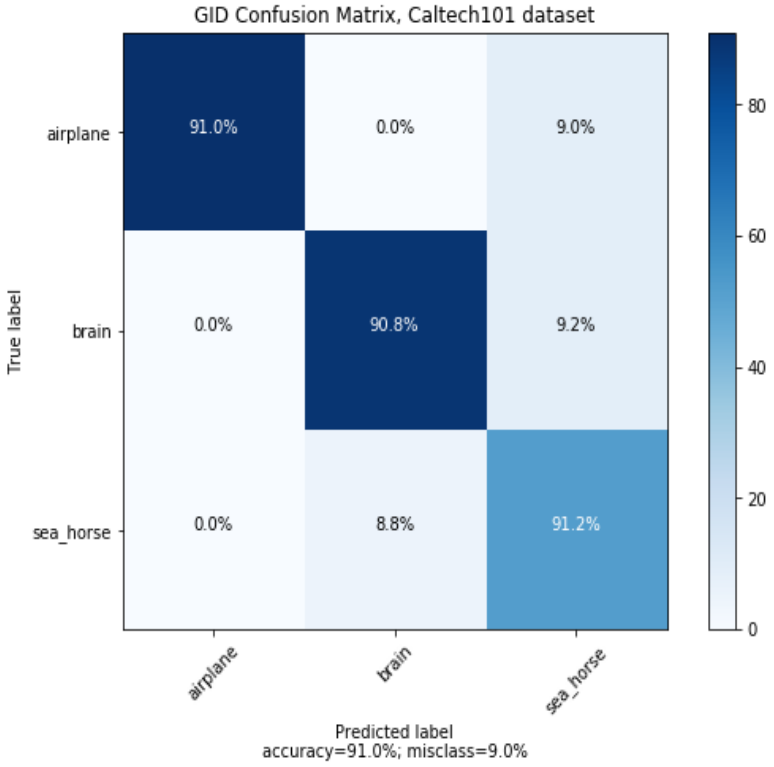

Figure 4: Confusion matrix of Caltech101 data set with EV-GIDMM.

In the second part of our experiments, we focus on texture differentiation. This dataset will be a good challenge for our model as images are very similar. In order to show how machines are becoming more capable of detecting and recognizing fine-grained images, in this experiment, we chose to use the Describable Texture Data set[3] that includes 120 images per class where each class consists of different types of textures. We have chosen Dotted, Frilly and Meshed image categories to evaluate our model as illustrated in Fig. 5.

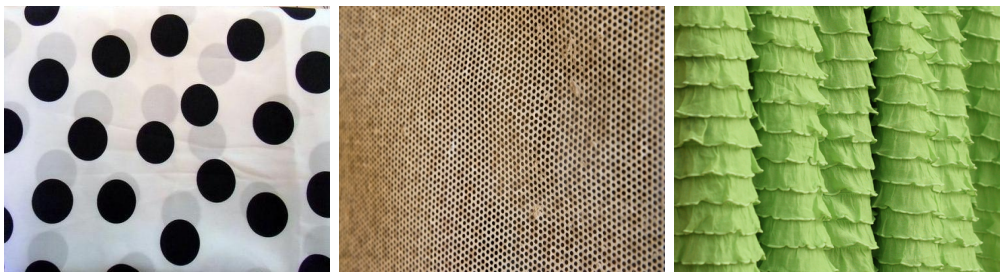

Figure 5: Sample images of each cluster from the DTD data set.

[3]https://www.robots.ox.ac.uk/ vgg/data/dtd/

Similarly, we performed the BoVW and used SIFT, to generate a discriminative input for our EV-GIDMM. The results of clustering evaluation on DTD are listed in Table 3. From this table it can be confirmed that our proposed mixture model achieves the best accuracy performance among all the other mixture models.

Table 3: Accuracy comparison of our EV-GIDMM approach and the baseline methods on the DTD data set.

Method	**Accuracy(%)**
EV-GIDMM	**85.6**
EV-MBMM	65.3
EDMM	65.8
varDMM	71.9

In addition, we have shown the confusion matrix of the DTD dataset on EV-GIDMM. From Fig. 6 it can be concluded that the Meshed class, due to its texture, has caused misclassification between meshed and the two other classes. However, our model has shown greater accuracy in comparison to other mixture models.

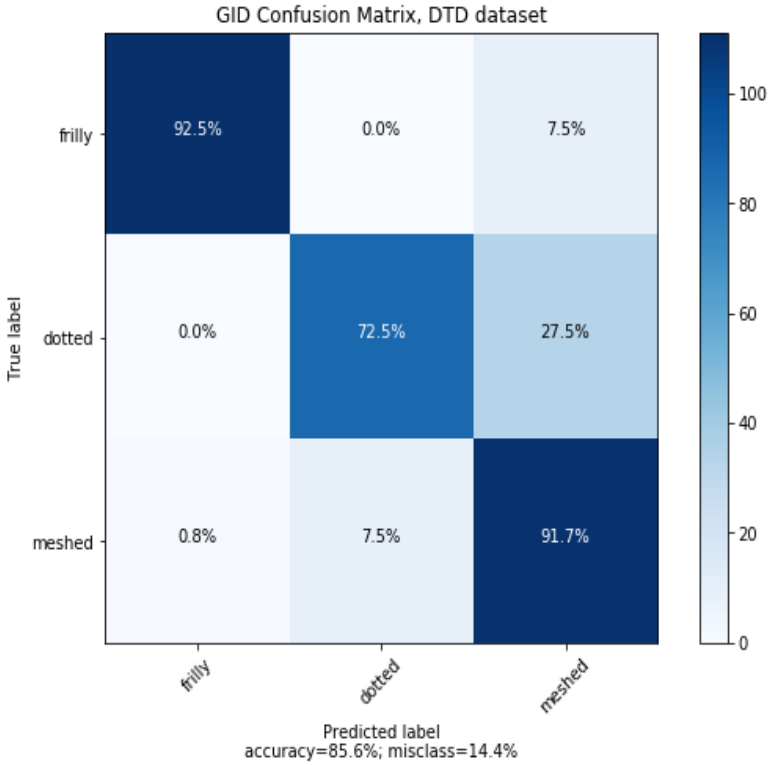

Figure 6: Confusion matrix of DTD data set with EV-GIDMM.

Chapter 3

Entropy-based Variational Learning of Finite Inverted Beta-Liouville Mixture Model

3.1 Model Specification

In this chapter, we introduce an unsupervised entropy-based variational learning of finite inverted Beta-Liouville mixture model for clustering positive data. We assess our proposed algorithm on three real-world applications, human activity recognition, breast cancer and image categorization. Furthermore, we compare the results of the proposed model with two popular mixture models.

3.1.1 Finite Inverted Beta-Liouville Mixture Model

Let $\vec{X}_i = (X_{i1}, \ldots, X_{iD})$ be a D dimensional vector generated from a set of N independently identically distributed data samples $\mathcal{X} = (\vec{X}_1, \ldots, \vec{X}_N)$, drawn from an inverted Beta-Liouville distribution. According to [39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49], the probability density function of the inverted Beta-Liouville is defined as:

$$p(\vec{X}_i \mid \alpha_{j1}, \ldots, \alpha_{jD}, \alpha_j, \beta_j) = \frac{\Gamma(\sum_{l=1}^{D} \alpha_{jl})\Gamma(\alpha_j + \beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \times \prod_{l=1}^{D} \frac{X_{il}^{\alpha_{jl}-1}}{\Gamma(\alpha_{jl})} \Big(\sum_{l=1}^{D} X_{il} \Big)^{\alpha_j - \sum_{l=1}^{D} \alpha_{jl}} \Big(1 + \sum_{l=1}^{D} X_{il} \Big)^{-(\alpha_j + \beta_j)} \tag{40}$$

The parameters of the probability density function for each component j are $\theta_j = (\alpha_{j1}, \ldots, \alpha_{jD}, \alpha_j, \beta_j)$. The mean, variance and covariance of the inverted Beta-Liouville distribution are as follows:

$$E(X_{il}) = \frac{\alpha}{\beta - 1} \frac{\alpha_l}{\sum_{l=1}^{D} \alpha_l} \tag{41}$$

$$Var(X_{il}) = \frac{\alpha(\alpha+1)}{(\beta-1)(\beta-2)} \frac{\alpha_l(\alpha+1)}{\sum_{l=1}^{D} \alpha_l(\sum_{l=1}^{D} \alpha_l + 1)} \frac{\alpha^2}{(\beta-1)^2} \frac{{\alpha_l}^4}{(\sum_{l=1}^{D} \alpha_l)^4} \tag{42}$$

$$Cov(X_{im}, X_{in}) = \frac{\alpha_m \alpha_n}{\sum_{l=1}^{D} \alpha_l} \Big[\frac{\alpha(\alpha+1)}{(\beta-1)(\beta-2)(\sum_{l=1}^{D} \alpha_l + 1)} - \frac{\alpha^2}{(\beta-1)^2(\sum_{l=1}^{D} \alpha_l)} \Big] \tag{43}$$

By assuming that each $\vec{X}_i$ is generated from a mixture of inverted Beta-Liouville distributions, we can define the mixture model as:

$$p(\mathcal{X} \mid \vec{\pi}, \Theta) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j p(\vec{X}_i \mid \theta_j) \tag{44}$$

where $p(\vec{X}_i \mid \theta_j)$ refers to the conditional probability of the data samples with respect to each component, $\Theta = (\theta_1, \ldots, \theta_M)$ and $\vec{\pi} = (\pi_1, \ldots . \pi_M)$ is defined as the set of mixing coefficients with the constraints $\sum_{j=1}^{M} \pi_j = 1$ and $0 \leq \pi_j \leq 1$. Subsequently, we define an indicator matrix $\mathcal{Z} = (\vec{Z}_1, \ldots, \vec{Z}_N)$, where $\vec{Z}_i = (Z_{i1}, \ldots, Z_{iM})$ is a binary latent vector associated with every data sample $\vec{X}_i$, with constraints $Z_{ij} \in \{0, 1\}$ and $\sum_{j=1}^{M} Z_{ij} = 1$. We assume that Z_{ij} is equal to 1 if $\vec{X}_i$ belongs to the component j, and zero otherwise. The conditional probability distribution of the indicator variable $\mathcal{Z}$ is given by:

$$p(\mathcal{Z} \mid \vec{\pi}) = \prod_{i=1}^{N} \prod_{j=1}^{M} {\pi_j}^{Z_{ij}} \tag{45}$$

From the equation above, we can define the conditional distribution of a dataset $\mathcal{X}$ with respect to the latent variable $\mathcal{Z}$ and components parameters as:

$$p(\mathcal{X} \mid \mathcal{Z}, \Theta) = \prod_{i=1}^{N} \prod_{j=1}^{M} p(\vec{X}_i \mid \theta_j)^{Z_{ij}} \tag{46}$$

Since these parameters are positive, it would be convenient if we describe the priors with the Gamma distribution $\mathcal{G}(\cdot)$ as follows:

$$p(\alpha_{jl}) = \mathcal{G}(\alpha_{jl} \mid e_{jl}, f_{jl}) = \frac{f_{jl}^{e_{jl}}}{\Gamma(e_{jl})} {\alpha_{jl}}^{e_{jl}-1} e^{-f_{jl}\alpha_{jl}} \tag{47}$$

$$p(\beta_j) = \mathcal{G}(\beta_j \mid g_j, h_j) = \frac{h_j^{g_j}}{\Gamma(g_j)} {\beta_j}^{g_j - 1} e^{-h_j \beta_j} \tag{48}$$

$$p(\alpha_j) = \mathcal{G}(\alpha_j \mid u_j, \nu_j) = \frac{\nu_j^{u_j}}{\Gamma(u_j)} {\alpha_j}^{u_j - 1} e^{-\nu_j \alpha_j} \tag{49}$$

where all the hyperparameters are positive. At this point, we can represent the joint distribution for all the random variables as:

$$p(\mathcal{X}, \mathcal{Z}, \Theta \mid \vec{\pi}) = p(\mathcal{X} \mid \mathcal{Z}, \Theta) p(\mathcal{Z} \mid \vec{\pi}) p(\vec{\alpha}_l) p(\vec{\beta}) p(\vec{\alpha}) \tag{50}$$

$$\begin{aligned} p(\mathcal{X}, \mathcal{Z}, \Theta \mid \vec{\pi}) = & \prod_{i=1}^{N} \prod_{j=1}^{M} \Big[\frac{\Gamma(\sum_{l=1}^{D} \alpha_{jl}) \Gamma(\alpha_j + \beta_j)}{\Gamma(\alpha_j) \Gamma(\beta_j)} \\ & \prod_{l=1}^{D} \frac{X_{il}^{\alpha_{jl} - 1}}{\Gamma(\alpha_{jl})} \Big(\sum_{l=1}^{D} X_{il} \Big)^{\alpha_j - \sum_{l=1}^{D} \alpha_{jl}} \Big(1 + \sum_{l=1}^{D} X_{il} \Big)^{-(\alpha_j + \beta_j)} \Big]^{Z_{ij}} \\ & \times \left(\prod_{i=1}^{N} \prod_{j=1}^{M} \pi_j^{Z_{ij}} \right) \prod_{j=1}^{M} \prod_{l=1}^{D} \Big[\frac{\nu_j^{u_j}}{\Gamma(u_j)} {\alpha_j}^{u_j - 1} e^{-\nu_j \alpha_j} \\ & \times \frac{h_j^{g_j}}{\Gamma(g_j)} {\beta_j}^{g_j - 1} e^{-h_j \beta_j} \times \frac{f_{jl}^{e_{jl}}}{\Gamma(e_{jl})} {\alpha_{jl}}^{e_{jl} - 1} e^{-f_{jl} \alpha_{jl}} \Big] \end{aligned} \tag{51}$$

3.2 Model Learning with Variational Inference

In this section, we explain the variational inference framework that we adopted to accurately learn the finite inverted Beta-Liouville mixture model based on the proposed inference methodology in [50]. We define $Q(\Theta)$ as the approximation of the true posterior $p\big(\Theta \mid \mathcal{X}, \vec{\pi}\big)$. The main goal of variational inference is to minimize the difference between the approximated distribution and the true posterior. The estimation of the true posterior distribution is accomplished with the Kullback-Leibler (KL) divergence between the two distributions. Therefore, the KL divergence between $p\big(\Theta \mid \mathcal{X}, \vec{\pi}\big)$ and $Q(\Theta)$ is defined as follows:

$$\begin{aligned} KL\big(Q \mid\mid P\big) &= -\int Q\big(\Theta\big) \ln \left(\frac{p\big(\Theta \mid \mathcal{X}, \vec{\pi}\big)}{Q\big(\Theta\big)} \right) \mathrm{d}\Theta \\ &= \ln\ p\big(\mathcal{X} \mid \vec{\pi}\big) - \mathcal{L}\big(Q\big) \end{aligned} \tag{52}$$

$$\mathcal{L}\big(Q\big) = \int Q\big(\Theta\big) \ln \left(\frac{p\big(\mathcal{X}, \Theta \mid \vec{\pi}\big)}{Q\big(\Theta\big)} \right) \mathrm{d}\Theta \tag{53}$$

According to the Jensen's inequality $\mathcal{L}(Q) \leq \ln p(\mathcal{X} \mid \vec{\pi})$, $\mathcal{L}(Q)$ acts as the lower bound of the log likelihood. This means that we can minimize the KL divergence by maximizing the lower bound $\mathcal{L}(Q)$ [50]. We adopt the mean field approximation approach in order to find the optimal parameters of the fully factorizable distribution Q, where $Q(\Theta) = Q(\mathcal{Z})Q(\vec{\alpha})Q(\vec{\beta})Q(\vec{\pi})Q(\vec{\alpha_l})$. Now we perform variational optimization with respect to each of the parameters. For a specific parameter $Q_s(\Theta_s)$, we can represent the optimal solution as:

$$Q_s(\Theta_s) = \frac{\exp\left\langle \ln\ p(\mathcal{X}, \Theta)\right\rangle_{i \neq s}}{\int \exp\left\langle \ln\ p(\mathcal{X}, \Theta)\right\rangle_{i \neq s} \mathrm{d}\Theta} \tag{54}$$

where $\langle \cdot \rangle_{i \neq s}$ indicates the expectation with respect to all the parameters except Q_s. We can derive the variational approximations of our model are as shown in Appendix A as:

$$Q(\mathcal{Z}) = \prod_{i=1}^{N} \prod_{j=1}^{M} r_{ij}^{Z_{ij}} \tag{55}$$

$$Q(\vec{\alpha}) = \prod_{j=1}^{M} \mathcal{G}(\alpha_j \mid u_j^*, \nu_j^*) \tag{56}$$

$$Q(\vec{\beta}) = \prod_{j=1}^{M} \mathcal{G}(\beta_j \mid g_j^*, h_j^*) \tag{57}$$

$$Q(\vec{\alpha}_l) = \prod_{j=1}^{M} \prod_{l=1}^{D} \mathcal{G}(\alpha_{jl} \mid e_{jl}^*, f_{jl}^*) \tag{58}$$

$$r_{ij} = \frac{\tilde{r}_{ij}}{\sum_{j=1}^{M} \tilde{r}_{ij}} \tag{59}$$

$$\begin{aligned} \tilde{r}_{ij} = \exp\Big\{ & \ln \pi_j + \tilde{R}_j + \tilde{S}_j + (\bar{\alpha}_j - \sum_{l=1}^{D} \bar{\alpha}_{jl}) \ln(\sum_{l=1}^{D} X_{il}) \\ & + \sum_{l=1}^{D} \Big[(\bar{\alpha}_{jl} - 1) \ln X_{il} \Big] - (\bar{\alpha}_j + \bar{\beta}_j) \ln(1 + \sum_{l=1}^{D} X_{il}) \Big\} \end{aligned} \tag{60}$$

$$\tilde{R}_j = \ln \frac{\Gamma(\sum_{l=1}^{D} \bar{\alpha}_{jl})}{\prod_{l=1}^{D} \Gamma(\bar{\alpha}_{jl})} + \sum_{l=1}^{D} \bar{\alpha}_{jl} \Big[\psi\Big(\sum_{l=1}^{D} \bar{\alpha}_{jl} \Big) - \psi(\bar{\alpha}_{jl}) \Big] \times$$
$$\Big[\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl} \Big] + 0.5 \sum_{l=1}^{D} \bar{\alpha}_{jl}^2 \Big[\psi^{'}\Big(\sum_{l=1}^{D} \bar{\alpha}_{jl} \Big) - \psi^{'}(\bar{\alpha}_{jl}) \Big]$$
$$- \Big\langle (\ln \alpha_{jl} - \ln \bar{\alpha}_{jl})^2 \Big\rangle + 0.5 \sum_{a=1}^{D} \sum_{b=1}^{D} \bar{\alpha}_{ja} \bar{\alpha}_{jb} \Big[\psi^{'}\Big(\sum_{l=1}^{D} \bar{\alpha}_{jl} \Big)$$
$$\times \Big(\langle \ln \alpha_{ja} \rangle - \ln \bar{\alpha}_{ja} \Big) \Big(\langle \ln \alpha_{jb} \rangle - \ln \bar{\alpha}_{jb} \Big) \Big] \tag{61}$$

$$\tilde{S}_j = \ln \frac{\Gamma(\bar{\alpha}_j + \bar{\beta}_j)}{\Gamma(\bar{\alpha}_j)\Gamma(\bar{\beta}_j)} + \bar{\alpha}_j \Big[\psi(\bar{\alpha}_j + \bar{\beta}_j) - \psi(\bar{\alpha}_j) \Big]$$
$$\times \Big(\langle \ln \alpha_j \rangle - \ln \bar{\alpha}_j \Big) + \bar{\beta}_j \Big[\psi(\bar{\alpha}_j + \bar{\beta}_j) - \psi(\bar{\beta}_j) \Big]$$
$$\times \Big(\langle \ln \beta_j \rangle - \ln \bar{\beta}_j \Big) + 0.5 \bar{\alpha}_j^2 \Big[\psi^{'}(\bar{\alpha}_j + \bar{\beta}_j) - \psi^{'}(\bar{\alpha}_j) \Big]$$
$$\times \Big\langle (\ln \alpha_j - \ln \bar{\alpha}_j)^2 \Big\rangle + 0.5 \bar{\beta}_j^2 \Big[\psi^{'}(\bar{\alpha}_j + \bar{\beta}_j) - \psi^{'}(\bar{\beta}_j) \Big]$$
$$\times \Big\langle (\ln \beta_j - \ln \bar{\beta}_j)^2 \Big\rangle + \bar{\beta}_j \bar{\alpha}_j \psi^{'}(\bar{\alpha}_j + \bar{\beta}_j) \Big(\langle \ln \alpha_j \rangle - \ln \bar{\alpha}_j \Big)$$
$$\times \Big(\langle \ln \beta_j \rangle - \ln \bar{\beta}_j \Big) \tag{62}$$

$$e_{jl}^{*} = e_{jl} + \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_{jl} \Big[\psi\Big(\sum_{l=1}^{D} \bar{\alpha}_{jl} \Big) - \psi(\bar{\alpha}_{jl}) + \psi^{'}(\sum_{l=1}^{D} \bar{\alpha}_{jl}) \sum_{d \neq l}^{D} \Big(\langle \ln \alpha_{jl} \rangle - \ln \bar{\alpha}_{jl} \Big) \bar{\alpha}_{jl} \Big] \tag{63}$$

$$f_{jl}^{*} = f_{jl} - \sum_{i=1}^{N} \langle Z_{ij} \rangle \Big[\ln X_{il} - \ln \Big(\sum_{l=1}^{D} X_{il} \Big) \Big] \tag{64}$$

$$u_j^{*} = u_j + \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_j \Big[\psi(\bar{\alpha}_j + \bar{\beta}_j) - \psi(\bar{\alpha}_j) + \bar{\beta}_j \psi^{'}(\bar{\alpha}_j + \bar{\beta}_j) \Big(\langle \ln \beta_j \rangle - \ln \bar{\beta}_j \Big) \Big] \tag{65}$$

$$\nu_j^{*} = \nu_j - \sum_{i=1}^{N} \langle Z_{ij} \rangle \ln(\sum_{l=1}^{D} X_{il}) + \sum_{i=1}^{N} \langle Z_{ij} \rangle \ln(1 + \sum_{l=1}^{D} X_{il}) \tag{66}$$

$$g_j^{*} = g_j + \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\beta}_j \Big[\psi(\bar{\alpha}_j + \bar{\beta}_j) - \psi(\bar{\beta}_j) + \bar{\alpha}_j \psi^{'}(\bar{\alpha}_j + \bar{\beta}_j) \Big(\langle \ln \alpha_j \rangle - \ln \bar{\alpha}_j \Big) \Big] \tag{67}$$

$$h_j^* = h_j + \sum_{i=1}^{N} \langle Z_{ij} \rangle \Big[1 + \sum_{l=1}^{D} X_{il} \Big] \tag{68}$$

In the above equations, $\psi(\cdot)$ and $\psi^{'}(\cdot)$ refer to the Digamma and Trigamma functions, respectively. The terms $S_j = \langle \ln \frac{\Gamma(\bar{\alpha}_j + \bar{\beta}_j)}{\Gamma(\bar{\alpha}_j)\Gamma(\bar{\beta}_j)} \rangle$ and $R_j = \langle \ln \frac{\Gamma(\sum_{l=1}^{D} \bar{\alpha}_{jl})}{\prod_{l=1}^{D} \Gamma(\bar{\alpha}_{jl})} \rangle$ are indeed intractable. To solve this problem, we use the second-order Taylor series to approximate them. The expected values of the aforementioned equations can be written as:

$$\langle Z_{ij} \rangle = r_{ij} \tag{69}$$

$$\bar{\alpha}_j = \langle \alpha_j \rangle = \frac{u_j^*}{\nu_j^*} \ , \ \langle \ln \alpha_j \rangle = \psi(u_j^*) - \ln \nu_j^* \tag{70}$$

$$\bar{\beta}_j = \langle \beta_j \rangle = \frac{g_j^*}{h_j^*} \ , \ \langle \ln \beta_j \rangle = \psi(g_j^*) - \ln h_j^* \tag{71}$$

$$\bar{\alpha}_{jl} = \langle \alpha_{jl} \rangle = \frac{e_{jl}^*}{f_{jl}^*} \ , \ \langle \ln \alpha_{jl} \rangle = \psi(e_{jl}^*) - \ln f_{jl}^* \tag{72}$$

$$\Big\langle (\ln \alpha_{jl} - \ln \bar{\alpha}_{jl})^2 \Big\rangle = \Big[\psi(e_{jl}^*) - \ln e_{jl}^* \Big]^2 + \psi^{'}(e_{jl}^*) \tag{73}$$

$$\Big\langle (\ln \beta_j - \ln \bar{\beta}_j)^2 \Big\rangle = \Big[\psi(g_j^*) - \ln g_j^* \Big]^2 + \psi^{'}(g_j^*) \tag{74}$$

$$\Big\langle (\ln \alpha_j - \ln \bar{\alpha}_j)^2 \Big\rangle = \Big[\psi(u_j^*) - \ln u_j^* \Big]^2 + \psi^{'}(u_j^*) \tag{75}$$

$$\pi_j = \frac{1}{N} \sum_{i=1}^{N} r_{ij} \tag{76}$$

3.3 Entropy-based Variational Model Learning

In this section, we develop an entropy-based variational Bayes for learning the finite inverted Beta-Liouville mixture model (EV-IBLMM). Our main motivation comes from the success of the entropy-based method in [51]. Initially, we start with one

component and incrementally increase the number of components. By comparing the theoretical maximum entropy with the MeanNN entropy [50], we conclude if a given component was genuinely inverted Beta-Liouville distributed. If their difference is phenomenal we proceed to split the component into two new components to fit the component.

3.3.1 Differential Entropy Estimation

Let $p(\vec{X}_i)$ be the probability density function of a random variable $\vec{X}_i = (\vec{X}_1, \ldots, \vec{X}_D)$ belonging to a set of N samples $\{\vec{X}_i, \ldots, \vec{X}_N\}$, $i = 1, \ldots, N$. The differential entropy of the continuous random variable $\vec{X}_i$ is defined by:

$$H\left(\vec{X}_i\right) = -\int p(\vec{X}_i) \log_2 p(\vec{X}_i) d\vec{X}_i \tag{77}$$

The maximum differential entropy of the IBL is given by:

$$\begin{aligned} H_{IBL}\left[p\left(\vec{X}_i \mid \theta\right)\right] &= \ln\left[\frac{\Gamma(\alpha)\Gamma(\beta)(\prod_{l=1}^{D}\Gamma(\alpha_l))}{\Gamma(\alpha+\beta)\Gamma(\sum_{l=1}^{D}\alpha_l)}\right] \\ &+ (\alpha+\beta)(\psi(\beta) - \psi(\alpha+\beta)) + \sum_{l=1}^{D}\Big[(1-\alpha_l)(\psi(\alpha_l) \\ &- \psi\Big(\sum_{l=1}^{D}\alpha_l\Big)\Big] + (D-\alpha)(\psi(\alpha) - \psi(\alpha+\beta)) \end{aligned} \tag{78}$$

3.3.2 MeanNN Entropy Estimator

We propose to adopt a MeanNN entropy estimator [52] to evaluate if a component was genuinely distributed according to the inverted Beta-Liouville distribution. The MeanNN estimator proceeds to find an estimation of $H(\vec{X}_i)$ with an unknown density function $p(\vec{X}_i)$ of a D dimensional random variable $\vec{X}_i$ [53]. Knowing that the Shannon differential entropy in equation (77) can be assumed equal to the average of $-\log\ p(\vec{X}_i)$, we can form an unbiased entropy estimator by estimating $\log\ p(\vec{X}_i)$. We consider a ball with diameter ϵ located at the center of $\vec{X}_i$, there is a point within the distance $[\epsilon, \epsilon + d\epsilon]$ from $\vec{X}_i$. Therefore, there is $\hat{k} - 1$ points that are in shorter distances and $N - \hat{k} - 1$ points that are in greater distances from the $\vec{X}_i$. By considering the above assumptions, the probability of the distances of the variable $\vec{X}_i$ and

its $\hat{k}$-th nearest neighbour is given by:

$$p_{i\hat{k}}\left(\epsilon\right)=\frac{(N-1)!}{(\hat{k}-1)!(N-\hat{k}-1)!}\frac{dp_i(\epsilon)}{d\epsilon}p_i^{\hat{k}-1}(1-p_i)^{N-\hat{k}-1} \tag{79}$$

where $p_i(\epsilon)$ refers to the mass of the ϵ-ball at $\vec{X}_i$, and can be found according to $p_i(\epsilon)=\int_{||\vec{X}-\vec{X}_i||<\epsilon}p(\vec{X}_i)d\vec{X}_i$. The expectation of $\log\ p_i(\epsilon)$ with respect to the term $p_i(\epsilon)$ is given as:

$$E\big(\log p_i(\epsilon)\big)=\int_0^\infty p_{i\hat{k}}\log p_i(\epsilon)d\epsilon=\psi(\hat{k})-\psi(N) \tag{80}$$

If we assume that the $p(\vec{X}_i)$ is unchanged in the center of the ϵ-ball, we have $p_i(\epsilon)\approx V_d\epsilon^d p(\vec{X}_i)$, where d describes the dimension of $\vec{X}_i$ and V_d represents the volume of the unit ball, that can be found according to $V_d=\pi^{d/2}/\Gamma(1+d/2)$. By substituting the approximation of $p_i(\epsilon)$ into the expectation of $\log\ p_i(\epsilon)$ we can find the approximation of $-\log\ p(\vec{X}_i)$ as:

$$-\log p(\vec{X}_i)\simeq\psi(N)-\psi(\hat{k})+dE(\log\epsilon)+\log V_d \tag{81}$$

Furthermore, the unbiased K-NN estimator of the differential entropy can be written as:

$$H_{\hat{k}}\left(\vec{X}\right)=\psi\left(N\right)-\psi\left(\hat{k}\right)+\frac{d}{N}\sum_{i=1}^{N}\log\epsilon_i+\log V_d \tag{82}$$

According to [52] to maintain a lower computational cost of the K-NN estimator, an extension to the K-NN has been proposed, known as the MeanNN entropy estimator. The fundamental idea of the MeanNN is to average the $\hat{k}$-nearest neighbour statistics for all feasible values of order $\hat{k}$ in the range of 1 to $N-1$. The MeanNN estimator of the differential entropy is given by equation (39).

$$\begin{aligned}H_M\left(\vec{X}\right)&=\frac{1}{N-1}\sum_{\hat{k}=1}^{N-1}H_{\hat{k}}\left(\vec{X}\right)=\log V_d+\psi\left(N\right)\\&+\frac{1}{N-1}\sum_{\hat{k}=1}^{N-1}\left[\frac{d}{N}\sum_{i=1}^{N}\log\epsilon_{i,\hat{k}}-\psi\left(\hat{k}\right)\right]\end{aligned} \tag{83}$$

where the $\hat{k}$-th nearest neighbour of $\vec{X}_i$ is represented by $\epsilon_{i,\hat{k}}$. We obtain the maximum entropy of the inverted Beta-Liouville mixture models according to:

$$H_{IBL}=\sum_{j=1}^{M}\pi_j H_{IBL}(j) \tag{84}$$

In (84) $H_{IBL}(j)$ describes the maximum differential entropy of the data in component j. Finally, with the information we have, we are able to perform a comparison between the theoretical maximum differential entropy and the entropy estimated by the MeanNN estimator to examine if a given component was genuinely inverted Beta-Liouville distributed. Motivated by [54], the normalized weighted sum of the difference between the estimated entropy of each component correlated with the IBL mixture model and the theoretical entropy is represented as follows:

$$\Omega_{IBL} = \sum_{j=1}^{M} \pi_j \Big[\frac{H_{IBL}(j) - H_M(j)}{H_{IBL}(j)} \Big] = \sum_{j=1}^{M} \pi_j \Big[1 - \frac{H_M(j)}{H_{IBL}(j)} \Big] \tag{85}$$

where $\Omega_{IBL} \in [0, 1]$ and is equal to zero only if the data was genuinely inverted Beta-Liouville distributed. We choose the target component with the highest weight according to (85) and split it into two new components according to equation (86):

$$j^* = \arg \max_j \Big[\Omega_{IBL}(j) \Big] = \arg \max_j \left[\pi_j \frac{H_{IBL}(j) - H_M(j)}{H_{IBL}(j)} \right] \tag{86}$$

The overall entropy-based variational learning algorithm of the IBL mixture model is illustrated in Algorithm 2.

Algorithm 2 Entropy-based variational learning of IBL mixture models

1. Initialization
   - Set $M = 1$, $j^* = M$, $\pi_1 = 1$
   - Initialize the hyperparameters e_{jl},f_{jl},u_j,ν_j,g_j,h_j.
2. The splitting process.
   - Split j^* into two new components j_1 and j_2 with equal proportion $\pi^*/2$.
   - Set $M = M + 1$.
   - Initialize the parameters of j_1 and j_2 using the same parameters of j^*.
3. Apply the standard variational Bayes until convergence.
4. Determine the number of components through the evaluation of the mixing coefficients π_j according to (76)
5. If $\pi_j \approx 0$. where $j \in 1, \ldots, M$ then set $M = M - 1$ and terminate the program.
6. Else evaluate Ω_{MD}, choose j^* according to (86) and go back to the splitting process in step 2.

3.4 Experimental Results

In this section, we evaluate the performances of our proposed model EV-IBLMM based on real-world challenging data sets for human activity recognition, breast cancer and image categorization applications. We compare the results of our experiments with two other similar models, namely, Entropy-based Variational Dirichlet Mixture Model (EDMM) [55] and Entropy-based Multivariate Beta Mixture Model (EV-MBMM) [56].

3.4.1 Human Activity Recognition (HAR)

Human activity recognition in smart homes is a key factor to achieve home automation especially with the significant advancement in sensing technologies. It enables

the smart applications to automatically react according to the human behaviour. However, automatically recognizing human activities like walking, sleeping and cooking is a challenging task, because human activities are complex by nature. In order to validate the performance of our model on the human activity recognition task, we used a data set proposed in [57]. This data set was collected based on several types of wireless sensors including contact switches, pressure mats, and float sensors. While trying to recognize activities from the sensor there could be some issues.

First of all, the start and end time of an activity is unknown. There is doubtfulness in the observed data, that the sensor has been activated according to which activity. For instants, getting a drink and cooking are two activities that require opening the fridge, but it doesn't shown which item was taken to recognize the designated activity. Activities can be performed in many ways, therefore it makes it harder to draw a general description of the activity. These issues have made human activity recognition a challenging task. It is to note that the recorded data are prone to noise because data might be lost if one of these sensors gets disconnected from the network. The Raw time series data has discredited into T time slices with the length of 5 minutes. It is possible that activities will overlap, for instants, a activity was left somewhere halfway, therefore the activity that has taken up most of the time slice is kept. The Labels of the activities were recorded with a hand written diary or Bluetooth headset. More details about the data set is mentioned in [57] and Appendix B.

Table 4: Accuracy comparison of our EV-IBLMM approach and the baseline methods on the Human Activity Recognition data set.

Method	**Accuracy(%)**
EV-IBLMM	**95.00**
EV-MBMM	93.30
EDMM	92.52

In this study, 20 sensors have been used, where each sensor represents a feature to our model. Since actions can overlap, the action that lasts longer is maintained and kept. Thus, we consider a total of 6851 entries with 4 recorded activities. These activities include eating, sleeping, taking a shower and opening a door. The results of our proposed model and baselines are shown in Table 4. We can see that our model achieves the best accuracy performance among the other mixture models, which

further demonstrates its efficiency for automatic human action recognition.

3.4.2 Image Categorization

Image categorization is considered as one of the important tasks of computer vision, and has witnessed much attention in the last decades. In this part of our experiments, we tested our proposed model on the image clustering task based on the Caltech101 image dataset [58]. This dataset contains images from 101 classes, with about 40-800 images in each category. For evaluation, we select a subset of 2033 data samples from 3 classes, namely, motorbikes, faces and airplanes. Some sample images from the three considered categories are illustrated in Figure 7. In order to test our model on the

Figure 7: Sample images of each category from the considered subset of the Caltech101 dataset.

Caltech101 dataset, we use SIFT [59] to extract the features of the designated images. This method has been shown to be a good choice for this dataset in comparison to other feature extraction methods like, SURF [60] and HOG [61]. Then, we apply the K-means clustering algorithm on the results of the SIFT method, and use the output to create the Bag of Visual Words (BoVW) features. Table 5 illustrates the accuracy performance produced by each model. We observe that our EV-IBLMM model outperforms the EV-MBMM and EDMM models with a considerable margin of 1.8% and 3.2%, respectively, on the Caltech101 dataset. This highlights the effectiveness of our model in terms of model selection and data clustering.

Table 5: Accuracy comparison of our EV-IBLMM approach and the baseline methods on the Caltech101 dataset.

Method	**Accuracy(%)**
EV-IBLMM	**90.20**
EV-MBMM	88.50
EDMM	87.10

In Fig. 8 we have demonstrated the confusion matrix of Caltech101 dataset for the EV-IBLMM. In the case of the Airplane we can see that our model has clustered the data with a 97.2% accuracy and other class are clustered with an accuracy greater than 82%. In the case of EV-MBMM and EDMM they did not perform well clustering the Faces class, therefore, reduced their accuracy.

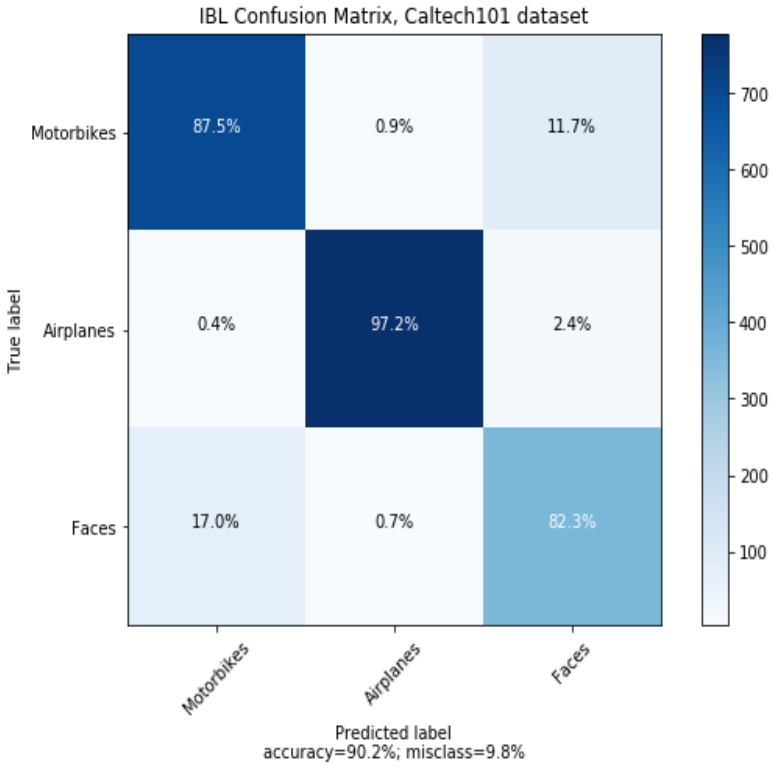

Figure 8: Confusion matrix of Caltech101 data set with EV-IBLMM.

3.4.3 Breast Cancer

In the last part of our experiments we applied the breast cancer data set that we previously applied to the GID mixture model. According to the WHO (World Health Organization), breast cancer has been known as the most frequent cancer among women, this type cancer affects about 2.1 million women every year. However machine learning techniques have shown to be effective in this context, in early detection of women breast cancer, therefore, they can have a great impact on the breast cancer treatment. To this end, we applied our proposed model, EV-IBLMM, on the breast cancer Wisconsin data set that is publicly available[1]. The data set includes the records of Dr. Wolberg patients that have been diagnosed with either malignant or benign cancer. The data set includes 569 data samples of patients that includes 357 benign and 212 malignant cases of tumor cancers. This data set was obtained by applying the Fine Needle Aspiration (FNA) method [34], [35], and it contains cases showing invasive breast cancer and no sign of distant metastases. The characteristics of each nuclei cell in the images of the tissue are the first 30 features of the data. Table 6 shows the experimental results of our model as well as the baseline methods for the breast cancer detection task. We can see that our proposed EV-IBLMM successfully achieved the best accuracy on this task.

Table 6: Accuracy performance of our model and the baselines on the breast cancer data set

Method	**Accuracy(%)**
EV-IBLMM	**91.2**
EV-MBMM	90.8
EDMM	89.7

[1]https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

Chapter 4

Conclusion

Mixture models are considered as a powerful approach for modeling complex data in an unsupervised manner. In this thesis we have studied the entropy-based variational learning for two mixture models and examined its efficiency with challenging data sets. Finally, we compared our results with several popular and similar models to demonstrate the robustness of the proposed models.

In chapter 2, we introduced an unsupervised entropy-based variational framework that effectively learns the finite generalized inverted Dirichlet mixture model. In the proposed method, a splitting technique called entropy was used, where we started by comparing the theoretical maximum entropy and the resulting entropy from MeanNN. Thereafter, we proceeded to split the component that has the highest difference into two smaller components, since it was concluded that the mixture model is not describing the component properly. Our experimental results have demonstrated that EV-GIDMM works very well and has outperformed other models on two real-world applications, namely, breast cancer detection and image categorization, across three different benchmark data sets. Considering the fact that the conducted experiments are under the category of unsupervised learning. The method has approximated accurate number of components in all experiments and has achieved accurate and computationally efficient parameter estimation of the EV-GIDMM. The results indicate that our proposed mixture model is able to produce high quality data clusters. In comparison to similar approaches our model has shown high accuracy with a considerable accuracy margin of 6.7% in the case on Caltech101 dataset and an accuracy margin of 13.6% in the case of DTD dataset. In all of our experiments the EV-GIDMM has

out performed the varDMM, EDMM and EV-MBMM.

In Chapter 3, we proposed an unsupervised entropy-based variational method to learn the finite inverted Beta-Liouville mixture model. In order to select the optimal number of components, we used a novel entropy-based method for the splitting process. The variational inference combined with the entropy-based variational inference has been effective in model selection, and have predicted the correct number of components in all of our tests. The accuracy of our experiments, on breast cancer, image categorization and human activity recognition in smart buildings has shown the robustness of EV-IBLMM in compare to similar mixture models, namely, EV-MBMM and EDMM. In the case of image categorization we decided to test our model on the Caltech101 dataset due to its popularity in related works. As a result we achieved a 90.2% accuracy with accurate number of components. In the second part of our experiments we decided to challenge our model with another real-world application, namely human activity recognition. We have reached a great 95% accuracy in this case.

Both proposed frameworks have shown great results in comparison to similar models, and have achieved high accuracy clustering. We have seen that the EV-IBLMM has shown better performance than the EV-GIDMM in accomplishing great accuracy. Future work could be dedicated to adding feature selection to both models. In addition, future work could be devoted to accelerated variational learning for the GID and IBL mixture models.

Bibliography

[1] Christopher M Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, NY, 2006. Softcover published in 2016.

[2] Graham Williams. *Data mining with Rattle and R: The art of excavating data for knowledge discovery*. Springer Science & Business Media, 2011.

[3] Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery*, 15(1):55–86, 2007.

[4] Riccardo Bellazzi and Blaz Zupan. Predictive data mining in clinical medicine: current issues and guidelines. *International journal of medical informatics*, 77(2):81–97, 2008.

[5] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.

[6] Brian S Everitt, Sabine Landau, and Morven Leese. *Cluster Analysis*. Arnold, 4th edition, 2001.

[7] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

[8] Geoffrey J McLachlan and David Peel. *Finite mixture models*. John Wiley & Sons, 2004.

[9] Tin Kam Ho and Henry S. Baird. Large-scale simulation studies in image pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1067–1079, 1997.

[10] Hieu Nguyen, Maryam Rahmanpour, Narges Manouchehri, Kamal Maanicshah, Manar Amayri, and Nizar Bouguila. A statistical approach for unsupervised occupancy detection and estimation in smart buildings. In *2019 IEEE International Smart Cities Conference, ISC2 2019, Casablanca, Morocco, October 14-17, 2019*, pages 414–419. IEEE, 2019.

[11] Narges Manouchehri, Jaspreet Singh Kalsi, Manar Amayri, and Nizar Bouguila. Finite two-dimensional beta mixture models: Model selection and applications. In *28th IEEE International Symposium on Industrial Electronics, ISIE 2019, Vancouver, BC, Canada, June 12-14, 2019*, pages 1407–1412. IEEE, 2019.

[12] Kenji Fukumizu, Shotaro Akaho, and Shun-ichi Amari. Critical lines in symmetry of mixture models and its application to component splitting. In *Advances in Neural Information Processing Systems*, pages 889–896, 2003.

[13] Wentao Fan and Nizar Bouguila. Non-gaussian data clustering via expectation propagation learning of finite dirichlet mixture models and applications. *Neural processing letters*, 39(2):115–135, 2014.

[14] Bromensele Samuel Oboh and Nizar Bouguila. Unsupervised learning of finite mixtures using scaled dirichlet distribution and its application to software modules categorization. In *2017 IEEE international conference on industrial technology (ICIT)*, pages 1085–1090. IEEE, 2017.

[15] Taoufik Bdiri, Nizar Bouguila, and Djemel Ziou. Variational bayesian inference for infinite generalized inverted dirichlet mixtures with feature selection and its application to clustering. *Applied Intelligence*, 44(3):507–525, 2016.

[16] Nizar Bouguila and Djemel Ziou. Unsupervised learning of a finite discrete mixture model based on the multinomial dirichlet distribution: Application to texture modeling. In *PRIS*, pages 118–127, 2004.

[17] Can Hu, Wentao Fan, Ji-Xiang Du, and Nizar Bouguila. A novel statistical approach for clustering positive data based on finite inverted beta-liouville mixture models. *Neurocomputing*, 333:110–123, 2019.

[18] Kenji Fukumizu and Shun-ichi Amari. Local minima and plateaus in hierarchical structures of multilayer perceptrons. *Neural networks*, 13(3):317–327, 2000.

[19] Yee W Teh, David Newman, and Max Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In *Advances in neural information processing systems*, pages 1353–1360, 2007.

[20] Constantinos Constantinopoulos and Aristidis Likas. Unsupervised learning of gaussian mixtures based on variational component splitting. *IEEE Transactions on Neural Networks*, 18(3):745–755, 2007.

[21] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[22] Nizar Bouguila and Djemel Ziou. High-dimensional unsupervised selection and estimation of a finite generalized dirichlet mixture model based on minimum message length. *IEEE transactions on pattern analysis and machine intelligence*, 29(10):1716–1731, 2007.

[23] Narges Manouchehri, Maryam Rahmanpour, Nizar Bouguila, and Wentao Fan. Learning of multivariate beta mixture models via entropy-based component splitting. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2825–2832. IEEE, 2019.

[24] Antonio Penalver and Francisco Escolano. Entropy-based incremental variational bayes learning of gaussian mixtures. *IEEE transactions on neural networks and learning systems*, 23(3):534–540, 2012.

[25] Wentao Fan, Faisal R Al-Osaimi, Nizar Bouguila, and Jixiang Du. Proportional data modeling via entropy-based variational bayes learning of mixture models. *Applied Intelligence*, 47(2):473–487, 2017.

[26] Wentao Fan, Nizar Bouguila, Sami Bourouis, and Yacine Laalaoui. Entropy-based variational bayes learning framework for data clustering. *IET Image Processing*, 12(10):1762–1772, 2018.

[27] Sami Bourouis, Mohamed Al Mashrgy, and Nizar Bouguila. Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection. *Expert Systems with Applications*, 41(5):2329–2336, 2014.

[28] Kamal Maanicshah, Nizar Bouguila, and Wentao Fan. Variational learning for finite generalized inverted dirichlet mixture models with a component splitting approach. In *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*, pages 1453–1458. IEEE, 2019.

[29] Wentao Fan, Nizar Bouguila, and Djemel Ziou. Variational learning for finite dirichlet mixture models and applications. *IEEE transactions on neural networks and learning systems*, 23(5):762–774, 2012.

[30] David Chandler. Introduction to modern statistical. *Mechanics. Oxford University Press, Oxford, UK*, 1987.

[31] Gilles Celeux, Florence Forbes, and Nathalie Peyrard. Em procedures using mean field-like approximations for markov model-based image segmentation. *Pattern recognition*, 36(1):131–144, 2003.

[32] Lev Faivishevsky and Jacob Goldberger. Ica based on a smooth estimation of the differential entropy. In *Advances in neural information processing systems*, pages 433–440, 2009.

[33] Nikolai Leonenko, Luc Pronzato, Vippal Savani, et al. A class of rényi information estimators for multidimensional densities. *The Annals of Statistics*, 36(5):2153–2182, 2008.

[34] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[35] William H Wolberg, W Nick Street, and Olvi L Mangasarian. Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. *Cancer letters*, 77(2-3):163–171, 1994.

[36] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[37] Teng Li, Tao Mei, In-So Kweon, and Xian-Sheng Hua. Contextual bag-of-words for visual categorization. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(4):381–392, 2010.

[38] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[39] Wentao Fan and Nizar Bouguila. Modeling and clustering positive vectors via nonparametric mixture models of liouville distributions. *IEEE Transactions on Neural Networks and Learning Systems*, 2019.

[40] Manar Amayri, Stéphane Ploix, Nizar Bouguila, and Frédéric Wurtz. Estimating occupancy using interactive learning with a sensor environment: Real-time experiments. *IEEE Access*, 7:53932–53944, 2019.

[41] Taoufik Bdiri and Nizar Bouguila. Bayesian learning of inverted dirichlet mixtures for SVM kernels generation. *Neural Comput. Appl.*, 23(5):1443–1458, 2013.

[42] Mohamed Al Mashrgy, Taoufik Bdiri, and Nizar Bouguila. Robust simultaneous positive data clustering and unsupervised feature selection using generalized inverted dirichlet mixture models. *Knowl. Based Syst.*, 59:182–195, 2014.

[43] Taoufik Bdiri and Nizar Bouguila. An infinite mixture of inverted dirichlet distributions. In Bao-Liang Lu, Liqing Zhang, and James T. Kwok, editors, *Neural Information Processing - 18th International Conference, ICONIP 2011, Shanghai, China, November 13-17, 2011, Proceedings, Part II*, volume 7063 of *Lecture Notes in Computer Science*, pages 71–78. Springer, 2011.

[44] Taoufik Bdiri and Nizar Bouguila. Learning inverted dirichlet mixtures for positive data clustering. In Sergei O. Kuznetsov, Dominik Slezak, Daryl H. Hepting, and Boris G. Mirkin, editors, *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing - 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011. Proceedings*, volume 6743 of *Lecture Notes in Computer Science*, pages 265–272. Springer, 2011.

[45] Taoufik Bdiri, Nizar Bouguila, and Djemel Ziou. Visual scenes categorization using a flexible hierarchical mixture model supporting users ontology. In *25th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2013, Herndon, VA, USA, November 4-6, 2013*, pages 262–267. IEEE Computer Society, 2013.

[46] Tarek Elguebaly and Nizar Bouguila. Semantic scene classification with generalized gaussian mixture models. In Mohamed Kamel and Aurélio J. C. Campilho, editors, *Image Analysis and Recognition - 12th International Conference, ICIAR 2015, Niagara Falls, ON, Canada, July 22-24, 2015, Proceedings*, volume 9164 of *Lecture Notes in Computer Science*, pages 159–166. Springer, 2015.

[47] Ali Shojaee Bakhtiari and Nizar Bouguila. An expandable hierarchical statistical framework for count data modeling and its application to object classification. In *IEEE 23rd International Conference on Tools with Artificial Intelligence, ICTAI 2011, Boca Raton, FL, USA, November 7-9, 2011*, pages 817–824. IEEE Computer Society, 2011.

[48] Parisa Tirdad, Nizar Bouguila, and Djemel Ziou. Variational learning of finite inverted dirichlet mixture models and applications. In Yacine Laalaoui and Nizar Bouguila, editors, *Artificial Intelligence Applications in Information and Communication Technologies*, volume 607 of *Studies in Computational Intelligence*, pages 119–145. Springer, 2015.

[49] Wentao Fan and Nizar Bouguila. Topic novelty detection using infinite variational inverted dirichlet mixture models. In Tao Li, Lukasz A. Kurgan, Vasile Palade, Randy Goebel, Andreas Holzinger, Karin Verspoor, and M. Arif Wani, editors, *14th IEEE International Conference on Machine Learning and Applications, ICMLA 2015, Miami, FL, USA, December 9-11, 2015*, pages 70–75. IEEE, 2015.

[50] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[51] Wentao Fan, Nizar Bouguila, Sami Bourouis, and Yacine Laalaoui. Entropy-based variational bayes learning framework for data clustering. *IET Image Processing*, 12(10):1762–1772, 2018.

[52] Lev Faivishevsky and Jacob Goldberger. Ica based on a smooth estimation of the differential entropy. In *Advances in neural information processing systems*, pages 433–440, 2009.

[53] Nikolai Leonenko, Luc Pronzato, Vippal Savani, et al. A class of rényi information estimators for multidimensional densities. *The Annals of Statistics*, 36(5):2153–2182, 2008.

[54] Antonio Penalver and Francisco Escolano. Entropy-based incremental variational bayes learning of gaussian mixtures. *IEEE transactions on neural networks and learning systems*, 23(3):534–540, 2012.

[55] Wentao Fan, Faisal R Al-Osaimi, Nizar Bouguila, and Jixiang Du. Proportional data modeling via entropy-based variational bayes learning of mixture models. *Applied Intelligence*, 47(2):473–487, 2017.

[56] Narges Manouchehri, Maryam Rahmanpour, Nizar Bouguila, and Wentao Fan. Learning of multivariate beta mixture models via entropy-based component splitting. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2825–2832. IEEE, 2019.

[57] Tim LM van Kasteren, Gwenn Englebienne, and Ben JA Kröse. Human activity recognition from wireless sensor network data: Benchmark and software. In *Activity recognition in pervasive intelligent environments*, pages 165–186. Springer, 2011.

[58] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[59] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[60] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[61] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

Appendix A

Appendix

A.1 Proof of equation (51).

From eq (50) we can write the joint PDF as follows:

$$\ln p(\mathcal{X}, \mathcal{Z}) = \sum_{i=1}^{N}\sum_{l=1}^{M} Z_{ij}\Big[\ln\frac{\Gamma(\sum_{l=1}^{D}\alpha_{jl})}{\prod_{l=1}^{D}\Gamma(\alpha_{jl})} + \frac{\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j+\beta_j)} + \sum_{l=1}^{D}(\alpha_{jl}-1)\ln X_{il}$$
$$+ (\alpha_j - \sum_{l=1}^{D}\alpha_{jl})\ln(\sum_{l=1}^{D}X_{il}) - (\alpha_j+\beta_j)\ln(1+\sum_{l=1}^{D}X_{il})\Big] + \sum_{i=1}^{N}\sum_{j=1}^{M} Z_{ij}\ln\pi_j$$
$$+ \sum_{j=1}^{M}\sum_{l=1}^{D} e_{jl}\ln f_{jl} - \ln\Gamma(e_{jl}) + (e_{jl}-1)\ln_{\alpha_{jl}} - f_{jl}\alpha_{jl}$$
$$+ \sum_{j=1}^{M} u_j\ln\nu_j - \ln\Gamma(u_j) + (u_j-1)\ln\alpha_j - \nu_j\alpha_j$$
$$+ \sum_{j=1}^{M} g_j\ln h_j - \ln\Gamma(g_j) + (g_j-1)\ln\alpha_j - h_j\alpha_j$$

In order to find the variational solutions for each parameter, we apply a logarithm with respect to each of the parameters assuming the rest of the parameters are constant. We explain this in the following subsection.

A.1.1 Variational Solution for $Q(Z)$ Eq. (60)

The logarithm with respect to $Q(Z_i)$ on the joint PDF is given by:

$$\ln Q(Z_i) = \sum_{j=1}^{M} Z_{ij} \Big[\ln Z_{ij} + R_j + S_j + \sum_{l=1}^{D} (\alpha_{jl} - 1) \ln X_{jl} + (\alpha_j - \sum_{l=1}^{D} \alpha_{jl}) \ln(\sum_{l=1}^{D} X_{il}) - (\alpha_j + \beta_j) \ln(1 + \sum_{l=1}^{D} X_{il}) \Big] \tag{87}$$

Where,

$$S_j = \Big\langle \ln \frac{\Gamma(\bar{\alpha}_j + \bar{\beta}_j)}{\Gamma(\bar{\alpha}_j)\Gamma(\bar{\beta}_j)} \Big\rangle \ , \ R_j = \Big\langle \ln \frac{\Gamma(\sum_{l=1}^{D} \bar{\alpha}_{jl})}{\prod_{l=1}^{D} \Gamma(\bar{\alpha}_{jl})} \Big\rangle \tag{88}$$

The S_j and R_j are intractable and we have used the second-order Taylor Series approximation. Therefore we find the $\ln \tilde{r}_{ij}$ as follows:

$$\ln \tilde{r}_{ij} = \ln \pi_j + \tilde{R}_j + \tilde{S}_j + (\bar{\alpha}_j - \sum_{l=1}^{D} \bar{\alpha}_{jl}) \ln(\sum_{l=1}^{D} X_{il}) + \sum_{l=1}^{D} \Big[(\bar{\alpha}_{jl} - 1) \ln X_{il} \Big] - (\bar{\alpha}_j + \bar{\beta}_j) \ln(1 + \sum_{l=1}^{D} X_{il}) \tag{89}$$

By taking an exponential from both sides we have:

$$\tilde{r}_{ij} = \exp \Big\{ \ln \pi_j + \tilde{R}_j + \tilde{S}_j + (\bar{\alpha}_j - \sum_{l=1}^{D} \bar{\alpha}_{jl}) \ln(\sum_{l=1}^{D} X_{il}) + \sum_{l=1}^{D} \Big[(\bar{\alpha}_{jl} - 1) \ln X_{il} \Big] - (\bar{\alpha}_j + \bar{\beta}_j) \ln(1 + \sum_{l=1}^{D} X_{il}) \Big\} \tag{90}$$

A.1.2 Proof of equation (58) : variational solution of $Q(\vec{\alpha}_l)$

We can find the logarithm of the variational solution $Q(A_{jl})$ according to:

$$\begin{aligned} \ln Q(\alpha_{jl}) &= \langle \ln p(\mathcal{X}, \Theta) \rangle_{\Theta \neq \alpha_{jl}} \\ &= \sum_{i=1}^{N} \langle Z_{ij} \rangle [\mathcal{J}(\alpha_{jl}) + \alpha_{jl} \ln X_{il} - \alpha_{jl} \ln(\sum_{l=1}^{D} X_{il})] \\ &+ (u_{jl} - 1) \ln \alpha_{jl} - \nu_{jl} \alpha_{jl} + const \end{aligned} \tag{91}$$

Where,

$$\mathcal{J}(\alpha_{jl}) = \left\langle \ln \frac{\Gamma(\alpha_{jl} + \sum_{s\neq l}^{D+1} \alpha_{js})}{\Gamma(\alpha_{jl}) \prod_{s\neq l}^{D+1} \Gamma(\alpha_{js})} \right\rangle_{\Theta\neq\alpha_{jl}} \tag{92}$$

The equation $\mathcal{J}(\alpha_{jl})$ is intractable as well, we solve this problem by finding the lower bound for the equation by calculating the first-order Taylor expansion with respect to $\bar{\alpha}_{jl}$.

$$\mathcal{J}(\alpha_{jl}) \geq \bar{\alpha}_{jl} \ln \alpha_{jl} \Big[\psi(\sum_{l=1}^{D+1} \bar{\alpha}_{jl}) - \psi(\alpha_{jl}) + \sum_{s\neq l}^{D+1} \bar{\alpha}_{js} \times \psi^{'}(\sum_{l=1}^{D+1} \bar{\alpha}_{jl})(\langle \ln \alpha_{js} \rangle - \ln \alpha_{js}) \Big] + const$$

Substituting this equation for lower bound in equation (91) we will have:

$$\begin{aligned} \ln Q(\alpha_{jl}) = & \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_{jl} \ln \alpha_{jl} \Big[\psi(\sum_{l=1}^{D} \bar{\alpha}_{jl}) - \psi(\bar{\alpha}_{jl}) \\ & + \psi^{'}(\sum_{l=1}^{D} \bar{\alpha}_{jl}) \sum_{d\neq l}^{D} (\langle \alpha_{jd} \rangle - \ln \alpha_{jd}) \bar{\alpha}_{jd} \Big] \\ & + \sum_{i=1}^{N} \alpha_{jl} \langle Z_{ij} \rangle \Big[\ln X_{il} - \ln(\sum_{l=1}^{D} X_{il}) \Big] + const \end{aligned} \tag{93}$$

The equation above can be written as:

$$\ln Q(\alpha_{jl}) = \ln \alpha_{jl}(u_{jl} + \varphi_{jl} - 1) - \alpha_{jl}(\nu_{jl} - \vartheta_{jl}) + const \tag{94}$$

Where,

$$\begin{aligned} \varphi_{jl} = & \sum_{i=1}^{N} \langle Z_{ij} \rangle \bar{\alpha}_{jl} \Big[\psi(\sum_{l=1}^{D} \bar{\alpha}_{jl}) - \psi(\bar{\alpha}_{jl}) \\ & + \Big[\psi^{'}(\sum_{l=1}^{D} \bar{\alpha}_{jl}) \sum_{d\neq l}^{D} (\langle \ln \alpha_{jd} \rangle - \ln \bar{\alpha}_{jd}) \bar{\alpha}_{jd} \Big] \psi^{'}(\sum_{l=1}^{D+1} \bar{\alpha}_{jl})(\langle \ln \alpha_{js} \rangle - \ln \bar{\alpha}_{js}) \Big] \end{aligned} \tag{95}$$

$$\vartheta_{jl} = \sum_{i=1}^{N} \langle Z_{ij} \rangle \Big[\ln X_{il} - \ln(\sum_{l=1}^{D} X_{il}) \Big] \tag{96}$$

Therefore, the optimal solution for the hayperparameters e_{jl} and f_{jl} given by:

$$e^*_{jl} = e_{jl} + \varphi_{jl} \ , \ f^*_{jl} = f_{jl} - \vartheta_{jl} \tag{97}$$

We can find the optimal solution for the $Q(\vec{\alpha})$ and $Q(\vec{\beta})$ by following the same procedure.

Appendix B

Human Activity Recognition

B.1 Dataset Details

In this section, we will give some information about the human activity Recognition Data set that helps to understand the data set represented in the paper [57]. In table 7 we are showing the details of the recorded data sets.

Table 7: Details of recorded data sets

	House A	**House B**	**House C**
Age	26	28	57
Gender	Male	Male	Male
Setting	Apartment	Apartment	House
Rooms	3	2	6
Duration	25 days	14 days	19 days
Activities	10	13	16
Annotation	Bluetooth	Diary	Bluetooth

B.2 Feature Representation

The raw data gained from the sensors can be transformed into a different representation form or even used directly. The authors of the paper [57] have experimented with three different feature representations:

Changepoint:

The raw sensor representation is directly using the sensor data. 1 indicated the sensor has been activated and 0 otherwise.

Raw:

The change point representation specifies when a sensor has been took place, It shoes when a senors has changed value. it gives 1 when a sensor changes state (i.e. changes from zero to one or vice verse).

Last-fired:

The last-fired sensor representation specifies which sensor has been fired last. The sensor that has lasted longest in the specifies timescale will continue to give 1 and will change to 0 when another sensor changes state.