Nonparametric Bayesian Clustering Ensembles

Pu Wang[1], Carlotta Domeniconi[1], and Kathryn Blackmond Laskey[2]

[1] Department of Computer Science
[2] Department of Systems Engineering and Operations Research
George Mason University
4400 University Drive, Fairfax, VA 22030 USA

Abstract. Forming consensus clusters from multiple input clusterings can improve accuracy and robustness. Current clustering ensemble methods require specifying the number of consensus clusters. A poor choice can lead to under or over fitting. This paper proposes a nonparametric Bayesian clustering ensemble (NBCE) method, which can discover the number of clusters in the consensus clustering. Three inference methods are considered: collapsed Gibbs sampling, variational Bayesian inference, and collapsed variational Bayesian inference. Comparison of NBCE with several other algorithms demonstrates its versatility and superior stability.

1 Introduction

Clustering ensemble methods operate on the output of a set of base clustering algorithms to form a consensus clustering. Clustering ensemble methods tend to produce more robust and stable clusterings than the individual solutions [28]. Since these methods require only the base clustering results and not the raw data themselves, clustering ensembles provide a convenient approach to privacy preservation and knowledge reuse [31]. Such desirable aspects have generated intense interest in cluster ensemble methods.

A variety of approaches have been proposed to address the clustering ensemble problem. Our focus is on statistically oriented approaches. Topchy et al. [28] proposed a mixture-membership model for clustering ensembles. Wang et al. [31] applied a Bayesian approach to discovering clustering ensembles. The Bayesian clustering ensemble model has several desirable properties [31]: it can be adapted to handle missing values in the base clusterings; it can handle the requirement that the base clusterings reside on a distributed collection of hosts; and it can deal with partitioned base clusterings in which different partitions reside in different locations. Other clustering ensemble algorithms, such as the cluster-based similarity partitioning algorithm (CSPA) [25], the hypergraph partitioning algorithm (HGPA) [25], or k-means based algorithms [18] can handle one or two of these cases; however, none except the Bayesian method can address them all.

Most clustering ensemble methods have the disadvantage that the number of clusters in the consensus clustering must be specified *a priori*. A poor choice can lead to under- or over-fitting. Our approach, nonparametric Bayesian clustering ensembles (NBCE), can discover the number of clusters in the consensus

J.L. Balcázar et al. (Eds.): ECML PKDD 2010, Part III, LNAI 6323, pp. 435–450, 2010.

clustering from the observations. Because it is also a Bayesian approach, NBCE inherits the desirable properties of the Bayesian clustering ensembles model [31]. Similar to the mixture modeling approach [28] and the Bayesian approach [31], NBCE treats all base clustering results for each object as a feature vector with discrete feature values, and learns a mixed-membership model from this feature representation.

The NBCE model is adapted from the Dirichlet Process Mixture (DPM) model [22]. The following sections show how the DPM model can be adapted to the clustering ensemble problem, and examine three inference methods: collapsed Gibbs sampling, standard variational Bayesian inference, and collapsed variational Bayesian inference. These methods are compared in theory and practice. Our empirical evaluation demonstrates the versatility and superior stability and accuracy of NBCE.

2 Related Work

A clustering ensemble technique is characterized by two components: the mechanism to generate diverse partitions, and the consensus function to combine the input partitions into a final clustering. Diverse partitions are typically generated by using different clustering algorithms [1], or by applying a single algorithm with different parameter settings [10,16,17], possibly in combination with data or feature sampling [30,9,20,29].

One popular methodology to build a consensus function utilizes a co-association matrix [10,1,20,30]. Such a matrix can be seen as a similarity matrix, and thus can be used with any clustering algorithm that operates directly on similarities [30,1]. As an alternative to the co-association matrix, voting procedures have been considered to build consensus functions in [7]. Gondek et al. [11] derive a consensus function based on the Information Bottleneck principle: the mutual information between the consensus clustering and the individual input clusterings is maximized directly, without requiring approximation.

A different popular mechanism for constructing a consensus maps the problem onto a graph-based partitioning setting [25,3,12]. In particular, Strehl et al. [25] propose three graph-based approaches: Cluster-based Similarity Partitioning Algorithm (CSPA), HyperGraph Partitioning Algorithm (HGPA), and Meta-Clustering Algorithm (MCLA). The methods use METIS (or HMETIS) [15] to perform graph partitioning. The authors in [23] develop soft versions of CSPA, HGPA, and MCLA which can combine soft partitionings of data.

Another class of clustering ensemble algorithms is based on probabilistic mixture models [28,31]. Topchy et al. [28] model the clustering ensemble as a finite mixture of multinomial distributions in the space of base clusterings. A consensus result is found as a solution to the corresponding maximum likelihood problem using the EM algorithm. Wang et al. [31] proposed Bayesian Cluster Ensembles (BCE), a model that applies a Bayesian approach to protect against the overfitting to which the maximum likelihood method is prone [28]. The BCE model is applicable to some important variants of the basic clustering ensemble problem, including clustering ensembles with missing values, as well as row-distributed or

column-distributed clustering ensembles. Our work extends the BCE model to a nonparametric version, keeping all the advantages thereof, while allowing the number of clusters to adapt to the data.

3 Dirichlet Process Mixture Model

The Dirichlet process (DP) [8] is an infinite-dimensional generalization of the Dirichlet distribution. Formally, let S be a set, G_0 a measure on S, and α_0 a positive real number. The random probability distribution G on S is distributed according to DP with the concentration parameter α_0 and the base measure G_0, if for any finite partition $\{B_k\}_{1\leq k\leq K}$ of S:

$$(G(B_1), G(B_2), \cdots, G(B_K)) \sim \\ Dir(\alpha_0 G_0(B_1), \alpha_0 G_0(B_2), \cdots, \alpha_0 G_0(B_K))$$

Let G be a sample drawn from a DP. Then with probability 1, G is a discrete distribution [8]. In addition, if the first $N-1$ draws from G yield K distinct values $\theta^*_{1:K}$ with multiplicities $n_{1:K}$, then the probability of the N^{th} draw conditioned on the previous $N-1$ draws is given by the Pólya urn scheme [5]:

$$\theta_N = \begin{cases} \theta^*_k, & \text{with prob } \frac{n_k}{N-1+\alpha_0},\ k \in \{1, \cdots, K\} \\ \theta^*_{K+1} \sim G_0, & \text{with prob } \frac{\alpha_0}{N-1+\alpha_0} \end{cases}$$

The DP is often used as a nonparametric prior in Bayesian mixture models [2]. Assume the data are generated from the following generative procedure:

$$G \sim Dir(\alpha_0, G_0) \\ \theta_{1:N} \sim G \\ x_{1:N} \sim \prod_{n=1}^{N} F(\cdot|\theta_n)$$

The $\theta_{1:N}$ typically contains duplicates; thus, some data points are generated from the same mixture component. It is natural to define a cluster as those observations generated from a given mixture component. This model is known as the *Dirichlet process mixture* (DPM) model. Although any finite sample contains only finitely many clusters, there is no bound on the number of clusters and any new data point has non-zero probability of being drawn from a new cluster [22]. Therefore, DPM is known as an "infinite" mixture model.

The DP can be generated via the stick-breaking construction [24]. Stick-breaking draws two infinite sequences of independent random variables, $v_k \sim Beta(1, \alpha_0)$ and $\theta^*_k \sim G_0$ for $k = \{1, 2, \cdots\}$. Let G be defined as:

$$\pi_k = v_k \prod_{j=1}^{k-1}(1 - v_j) \tag{1}$$

$$G(\theta) = \sum_{k=1}^{\infty} \pi_k \delta(\theta, \theta^*_k) \tag{2}$$

where $\boldsymbol{\pi} = \langle \pi_k | k = 1, 2, \cdots \rangle$ are the mixing proportions of the infinite number of components. Then $G \sim Dir(\alpha_0, G_0)$. It is helpful to use an indicator variable z_n to denote which mixture component is associated with x_n. The generative procedure for the DPM model using the stick-breaking construction becomes:

1. Draw $v_k \sim Beta(1, \alpha_0)$, $k = \{1, 2, \cdots\}$ and calculate $\boldsymbol{\pi}$ as in Eq (1).
2. Draw $\theta_k^* \sim G_0$, $k = \{1, 2, \cdots\}$
3. For each data point:
   - Draw $z_n \sim Discrete(\boldsymbol{\pi})$
   - Draw $x_n \sim F(\cdot|\theta_{z_n}^*)$

In practice, the process is typically truncated at level K by setting $v_{K-1} = 1$ [13]; Eq (1) then implies that all π_k for $k > K$ are zero. The truncated process is called truncated stick-breaking (TSB). The resulting distribution, the truncated Dirichlet process (TDP), closely approximates the Dirichlet process when K is sufficiently large. The choice of the truncation level K is discussed in [13]. The joint probability over data items $\boldsymbol{X} = \langle x_n | n \in \{1, \cdots, N\} \rangle$, component assignments $\boldsymbol{Z} = \langle z_n | n \in \{1, \cdots, N\} \rangle$, stick-breaking weights $\boldsymbol{v} = \langle v_k | k \in \{1, \cdots, K\} \rangle$ and component parameters $\boldsymbol{\theta}^* = \langle \theta_k^* | k \in \{1, \cdots, K\} \rangle$ is:

$$p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{v}, \boldsymbol{\theta}^*) = \left[\prod_{n=1}^{N} F(x_n | \theta_{z_n}^*) \pi_{z_n}(\boldsymbol{v}) \right] \left[\prod_{k=1}^{K} G_0(\theta_k^*) Beta(v_k; 1, \alpha_0) \right]$$

Another approach to approximate the DP is to assume a finite but large K-dimensional symmetric Dirichlet prior (FSD) on the mixture proportion $\boldsymbol{\pi}$ [14], which is $\boldsymbol{\pi} \sim Dir(\alpha_0/K, \cdots, \alpha_0/K)$. This results in the joint distribution:

$$p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{\theta}^*) = \left[\prod_{n=1}^{N} F(x_n | \theta_{z_n}^*) \pi_{z_n} \right] \left[\prod_{k=1}^{K} G_0(\theta_k^*) \right] Dir(\boldsymbol{\pi}; \frac{\alpha_0}{K}, \cdots, \frac{\alpha_0}{K})$$

With TSB, the cluster weights differ in expected value, with lower-numbered cluster indices having higher probability. With FSD, the clusters are exchangeable. A detailed comparison of these DP approximations can be found in [19].

4 NBCE Generative Model

Following [28] and [31], we assume there are M base clustering algorithms, each generating a hard partition on the N data items to be clustered. Let J_m denote the number of clusters generated by the m^{th} clustering φ_m, $m \in \{1, \cdots, M\}$, and let $y_{nm} \in \{1, \cdots, J_m\}$ denote the cluster ID assigned to the n^{th} data item x_n by φ_m, $n \in \{1, \cdots, N\}$. The row $\boldsymbol{y}_n = \langle y_{nm} | m \in \{1, \cdots, M\} \rangle$ of the base clustering matrix $\boldsymbol{Y}$ gives a new feature vector representation for the n^{th} data item.

Figure 1 depicts the generative model for $\boldsymbol{Y}$. We assume $\boldsymbol{y}_n$ is generated from a truncated Dirichlet Process mixture model, where α_0 is the concentration parameter, G_0 is the base measure, and K is the truncation level. The probability

of generating a cluster ID $y_{nm} = j_m$ by φ_m for x_n is θ_{nmj_m}, $j_m \in \{1, \cdots, J_m\}$ and $\sum_{j_m=1}^{J_m} \theta_{nmj_m} = 1$. So $\boldsymbol{y}_n = \langle y_{nm} = j_m | m \in \{1, \cdots, M\}\rangle$ is generated with probability $\prod_{m=1}^{M} \theta_{nmj_m}$. We define $\boldsymbol{\theta}_{nm} = \langle \theta_{nmj_m} | j_m \in \{1, \cdots, J_m\}\rangle$. We further assume a prior $G_0^{(m)}$ for $\boldsymbol{\theta}_{\cdot m} = \{\boldsymbol{\theta}_{nm} | n = 1, \cdots, N\}$, where $G_0^{(m)}$ is a symmetric Dirichlet distribution of dimension J_m with hyperparameter β. The base measure G_0 is defined as $G_0 = G_0^{(1)} \times \cdots \times G_0^{(M)}$. We denote $\boldsymbol{\theta}_n = \langle \boldsymbol{\theta}_{nm} | m \in \{1, \cdots, M\}\rangle$. Since the truncation level is K, there are K unique $\boldsymbol{\theta}_n$, denoted as $\boldsymbol{\theta}_k^* = \langle \boldsymbol{\theta}_{km}^* | m \in \{1, \cdots, M\}\rangle$, where $\boldsymbol{\theta}_{km}^* = \langle \theta_{kmj_m}^* | j_m \in \{1, \cdots, J_m\}\rangle$, $\sum_{j_m=1}^{J_m} \theta_{kmj_m}^* = 1$ and $k \in \{1, \cdots, K\}$. We associate with each x_n an indicator variable z_n to indicate which $\boldsymbol{\theta}_k^*$ is assigned to x_n; if $z_n = k$, then $\boldsymbol{\theta}_n = \boldsymbol{\theta}_k^*$. A consensus cluster is defined as a set of data items associated with the same $\boldsymbol{\theta}_k^*$. That is, z_n indicates which consensus cluster x_n belongs to. There are at most K consensus clusters, but some consensus clusters may be empty; we define the total number of consensus clusters to be the number of distinct z_n in the sample.

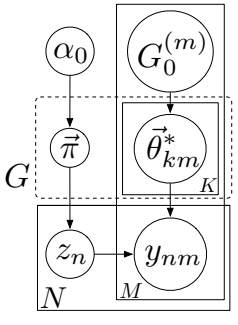

Fig. 1. Nonparametric Bayesian Clustering Ensembles Model

The stick breaking generative process for $\boldsymbol{Y}$ is:

1. Draw $v_k \sim Beta(1, \alpha_0)$, for $k = \{1, \cdots, K\}$ and calculate $\boldsymbol{\pi}$ as in Eq (1)
2. Draw $\boldsymbol{\theta}_k^* \sim G_0$, for $k = \{1, \cdots, K\}$
3. For each x_n:
   - Draw $z_n \sim Discrete(\boldsymbol{\pi})$
   - For each base clustering φ_m, draw $y_{nm} \sim Discrete(\boldsymbol{\theta}_{z_n m}^*)$

Using the symmetric Dirichlet prior, step 1 becomes:

1. Draw $\boldsymbol{\pi} \sim Dir(\frac{\alpha_0}{K}, \cdots, \frac{\alpha_0}{K})$

5 Inference and Learning

This section considers three inference and learning methods: collapsed Gibbs sampling, standard variational Bayesian, and collapsed variational Bayesian inference. Table 1 gives the notation used throughout this section.

The joint probability of observed base clustering results $\boldsymbol{Y} = \langle \boldsymbol{y}_n | n \in \{1, \cdots, N\} \rangle$, indicator variables $\boldsymbol{Z} = \langle z_n | n \in \{1, \cdots, N\} \rangle$, component weights $\boldsymbol{\pi} = \langle \pi_k | k \in \{1, \cdots, K\} \rangle$, and component parameters $\boldsymbol{\theta}^* = \langle \boldsymbol{\theta}_k^* | k \in \{1, \cdots, K\} \rangle$ is given by:

$$
\begin{aligned}
&p(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{\theta}^* | \alpha_0, G_0) = \\
&\left(\prod_{n=1}^{N} p(z_n | \boldsymbol{\pi}) p(\boldsymbol{y}_n | \boldsymbol{\theta}^*, z_n) \right) \cdot p(\boldsymbol{\pi} | \alpha_0) \left(\prod_{k=1}^{K} p(\boldsymbol{\theta}_k^* | G_0) \right) = \\
&\left(\prod_{n=1}^{N} p(z_n | \boldsymbol{\pi}) \prod_{m=1}^{M} p(y_{nm} | \boldsymbol{\theta}_{z_n m}^*) \right) \cdot p(\boldsymbol{\pi} | \alpha_0) \left(\prod_{k=1}^{K} \prod_{m=1}^{M} p(\boldsymbol{\theta}_{km}^* | G_0^{(m)}) \right)
\end{aligned} \tag{3}
$$

After marginalizing out the parameters $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$, the complete data likelihood is:

$$
\begin{aligned}
&p(\boldsymbol{Y}, \boldsymbol{Z} | \alpha_0, G_0,) = p(\boldsymbol{Z} | \alpha_0) \\
&\cdot \left(\prod_{m=1}^{M} \prod_{k=1}^{K} \frac{\Gamma(J_m \beta)}{\Gamma(J_m \beta + \mathcal{N}_{z_\cdot = k})} \prod_{j_m=1}^{J_m} \frac{\Gamma(\beta + \mathcal{N}_{z_\cdot = k}^{y_{\cdot m} = j_m})}{\Gamma(\beta)} \right)
\end{aligned} \tag{4}
$$

where for the two DP approximations, $p(\boldsymbol{Z}|\alpha_0)$ is different [19]:

$$
p_{TSB}(\boldsymbol{Z} | \alpha_0) = \prod_{k<K} \frac{\Gamma(1 + \mathcal{N}_{z_\cdot = k}) \Gamma(\alpha_0 + \mathcal{N}_{z_\cdot > k})}{\Gamma(1 + \alpha_0 + \mathcal{N}_{z_\cdot \geq k})}
$$

$$
p_{FSD}(\boldsymbol{Z} | \alpha_0) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_0 + N)} \prod_{k=1}^{K} \frac{\Gamma(\frac{\alpha_0}{K} + \mathcal{N}_{z_\cdot = k})}{\Gamma(\frac{\alpha_0}{K})}
$$

5.1 Collapsed Gibbs Sampling

Collapsed Gibbs sampling [21] speeds up the convergence of Gibbs sampling by marginalizing out the parameters $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$, sampling only the latent indicator variables $\boldsymbol{Z}$ over the so-called collapsed space.

From Eq (4), we can derive the distribution for sampling components of $\boldsymbol{Z}$:

$$
\begin{aligned}
&p(z_n = k | \boldsymbol{Z}_{\neg n}, \boldsymbol{Y}) \propto \\
&p(z_n = k | \boldsymbol{Z}_{\neg n}) \prod_{m=1}^{M} \left(\frac{\prod_{j_m=1}^{J_m} (\beta + \mathcal{N}_{z_\cdot = k, y_{\cdot m} = j_m}^{\neg n})}{J_m \beta + \mathcal{N}_{z_\cdot = k}^{\neg n}} \right)
\end{aligned} \tag{5}
$$

where for the two different DP approximations, $p(z_n = k|\boldsymbol{Z}_{\neg n})$ is different:

$$
p_{TSB}(z_n = k | \boldsymbol{Z}_{\neg n}) = \frac{1 + \mathcal{N}_{z_\cdot = k}^{\neg n}}{1 + \alpha_0 + \mathcal{N}_{z_\cdot \geq k}^{\neg n}} \prod_{h<k} \frac{\alpha_0 + \mathcal{N}_{z_\cdot > h}^{\neg n}}{1 + \alpha_0 + \mathcal{N}_{z_\cdot \geq h}^{\neg n}}
$$

$$
p_{FSD}(z_n = k | \boldsymbol{Z}_{\neg n}) = \frac{\frac{\alpha_0}{K} + \mathcal{N}_{z_\cdot = k}^{\neg n}}{\alpha_0 + N - 1}
$$

Table 1. Notation Description

Symbols	Description
N	the number of data
x_n	the n^{th} data item
M	the number base clusterings
φ_m	the m^{th} base clustering algorithm
y_{nm}	the cluster ID assigned to x_n by φ_m
K	the number of consensus clusters (truncation level)
J_m	the number of clusters in the m^{th} base clustering
j_m	the jth cluster in the m^{th} base clustering
$G_0^{(m)}$	the Dirichlet prior to $\{1, 2, \cdots, J_m\}$ of φ_m
β	the hyperparameter of $G_0^{(m)}$
G_0	$\prod_{m=1}^{M} G_{0m}$
z_n	the indicator variable of x_n to indicate which $\boldsymbol{\theta}_k^*$ assigned to x_n
$\boldsymbol{Z}_{\neg n}$	the indicator variables except for x_n
θ_{nmj_m}	the probability of $y_{nm} = j_m$
$\boldsymbol{\theta}_{nm}$	$\langle \theta_{nmj_m} \vert j_m \in \{1, \cdots, J_m\} \rangle$ and $\sum_{j_m=1}^{J_m} \theta_{nmj_m} = 1$
$\boldsymbol{\theta}_n$	$\langle \boldsymbol{\theta}_{nm} \vert m \in \{1, \cdots, M\} \rangle$
$\theta_{kmj_m}^*$	the probability of $y_{nm} = j_m$ if $z_n = k$
$\boldsymbol{\theta}_{km}^*$	$\langle \theta_{kmj_m}^* \vert j_m \in \{1, \cdots, J_m\} \rangle$ and $\sum_{j_m=1}^{J_m} \theta_{kmj_m}^* = 1$
$\boldsymbol{\theta}_k^*$	$\langle \boldsymbol{\theta}_{km}^* \vert m \in \{1, \cdots, M\} \rangle$, unique parameter value of $\boldsymbol{\theta}_n$
$\boldsymbol{\theta}^*$	$\langle \boldsymbol{\theta}_k^* \vert k \in \{1, \cdots, K\} \rangle$
$\mathcal{N}_{z.=k}$	$\sum_{n=1}^{N} \delta(z_n, k)$
$\mathcal{N}_{z.=k}^{\neg n}$	$\sum_{n'=1, n' \neq n}^{N} \delta(z_{n'}, k)$
$\mathcal{N}_{z.=k}^{y \cdot m = j_m}$	$\sum_{n=1}^{N} \delta(z_n, k) \delta(y_{nm}, j_m)$
$\mathcal{N}_{z.=k, y \cdot m = j_m}^{\neg n}$	$\sum_{n'=1, n' \neq n}^{N} \delta(z_{n'}, k) \delta(y_{n'm}, j_m)$
$\mathcal{N}_{z. \geq k}$	$\sum_{n=1}^{N} \mathbf{1}_{\{z \geq k\}}(z_n)$
$\mathcal{N}_{z. \geq k}^{\neg n}$	$\sum_{n'=1, n' \neq n}^{N} \mathbf{1}_{\{z \geq k\}}(z_{n'})$

5.2 Standard Variational Bayesian Inference

Variational Bayesian inference [4] approximates the posterior distribution by adjusting free parameters of a tractable variational distribution to minimize the KL-divergence between the variational and true distributions. This is equivalent to maximizing a lower bound on the true log-likelihood.

We consider only the FSD prior as the DP approximation for standard variational Bayesian (VB) inference. VB assumes the following variational distributions:

$$\begin{aligned} q(\boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{Z} | \boldsymbol{\xi}, \boldsymbol{\rho}, \boldsymbol{\gamma}) &= q(\boldsymbol{\pi}|\boldsymbol{\xi}) \left(\prod_{k=1}^{K} p(\boldsymbol{\theta}_k^* | \boldsymbol{\rho}_k) \right) \left(\prod_{n=1}^{N} p(z_n | \gamma_n) \right) \\ &= q(\boldsymbol{\pi}|\boldsymbol{\xi}) \left(\prod_{k=1}^{K} \prod_{m=1}^{M} p(\boldsymbol{\theta}_{km}^* | \rho_{km}) \right) \left(\prod_{n=1}^{N} p(z_n | \gamma_n) \right) \end{aligned} \tag{6}$$

where $\boldsymbol{\xi} = \langle \xi_k | k \in \{1, \cdots, K\} \rangle$, $\boldsymbol{\rho} = \langle \boldsymbol{\rho}_k | k \in \{1, \cdots, K\} \rangle = \langle \rho_{km} | k \in \{1, \cdots, K\}, m \in \{1, \cdots, M\} \rangle$ and $\boldsymbol{\gamma} = \langle \gamma_n | n \in \{1, \cdots, N\} \rangle$ are variational parameters, assumed to be independent. Further, given these variational parameters, the model parameters and indicator variables, $\boldsymbol{\pi}$, $\boldsymbol{\theta}$ and $\boldsymbol{Z}$ are independent of each other.[1] In particular, $\boldsymbol{\xi}$ specifies a K-dimensional Dirichlet distribution

[1] This is a strong assumption: note the dependences between $\boldsymbol{\pi}$ and $\boldsymbol{Z}$, $\boldsymbol{\theta}$ and $\boldsymbol{Z}$, $\boldsymbol{\theta}$ and $\boldsymbol{\pi}$ depicted in Figure 1.

for $\boldsymbol{\pi}$, ρ_{km} specifies a J_m-dimensional Dirichlet distribution for $\boldsymbol{\theta}^*_{km}$, and γ_n specifies an N-dimensional multinomial distribution for the indicator z_n of x_n.

A lower bound $\mathcal{L}_{VB}$ for the log-likelihood is given by:

$$\log p(\boldsymbol{Y}|\alpha_0, G_0) \geq \tag{7}$$
$$E_q[\log p(\boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{\theta}^*|\alpha_0, G_0)] - E_q[\log q(\boldsymbol{\pi}, \boldsymbol{\theta}^*, \boldsymbol{Z}|\boldsymbol{\xi}, \boldsymbol{\rho}, \boldsymbol{\gamma})] =$$
$$\left(\sum_{n=1}^{N}\sum_{m=1}^{M} E_q[\log p(y_{nm}|\boldsymbol{\theta}^*_{z_n m})]\right) - E_q[\log p(\boldsymbol{\pi}|\alpha_0)] + \left(\sum_{n=1}^{N} E_q[\log p(z_n|\boldsymbol{\pi})]\right) +$$
$$\left(\sum_{k=1}^{K} E_q[\log p(\boldsymbol{\theta}^*_k|G_0)]\right) - E_q[\log q(\boldsymbol{\pi}|\boldsymbol{\xi})] - E_q[\log q(\boldsymbol{\theta}^*|\boldsymbol{\rho})] - E_q[\log q(\boldsymbol{Z}|\boldsymbol{\gamma})]$$

See the Appendix for the expansion of Eq (7).

A local optimum is found by setting the partial derivatives of $\mathcal{L}_{VB}$ with respect to each variational parameter to be zero. This gives rise to the following first-order conditions:

$$\gamma_{nk} \propto \exp\left\{\left(\sum_{m=1}^{M}\sum_{j_m=1}^{J_m} \delta(y_{nm}, j_m)\log\rho_{km j_m}\right) + \Psi(\xi_k) - \Psi(\sum_{h=1}^{K}\xi_h)\right\}$$
$$\rho_{km j_m} = \beta + \sum_{n=1}^{N}\sum_{j_m=1}^{J_m}\gamma_{nk}\delta(y_{nm}, j_m)$$
$$\xi_k = \frac{\alpha_0}{K} + \sum_{n=1}^{N}\gamma_{nk}.$$

As for the remaining parameters α_0 and β, we first write the parts of $\mathcal{L}_{VB}$ involving α_0 and β as:

$$\mathcal{L}^{[\alpha_0]}_{VB} = \log\Gamma(\alpha_0) - K\log\Gamma(\frac{\alpha_0}{K}) + \left(\frac{\alpha_0}{K} - 1\right)\sum_{k=1}^{K}\left[\Psi(\xi_k) - \Psi(\sum_{h=1}^{K}\xi_h)\right]$$
$$\mathcal{L}^{[\beta]}_{VB} = \sum_{m=1}^{M}\Bigg(K\log\Gamma(J_m\beta) - KJ_m\log\Gamma(\beta) +$$
$$(\beta-1)\sum_{k=1}^{K}\sum_{j_m=1}^{J_m}\left[\Psi(\rho_{km j_m}) - \Psi(\sum_{h=1}^{J_m}\rho_{kmh})\right]\Bigg)$$

Estimates for α_0 and β are then obtained by maximization of $\mathcal{L}^{[\alpha_0]}_{VB}$ and $\mathcal{L}^{[\beta]}_{VB}$ using standard methods such as Newton-Raphson [6].

5.3 Collapsed Variational Bayesian Inference

Inspired by collapsed Gibbs sampling, collapsed variational Bayesian (CVB) inference for NBCE optimizes a lower bound $\mathcal{L}_{CVB}$ for the log-likelihood in the collapsed space, in which the model parameters $\boldsymbol{\theta}^*$ are marginalized out.

CVB assumes the following variational distribution:

$$q(\boldsymbol{Z}|\boldsymbol{\gamma}) = \prod_{n=1}^{N} q(z_n|\gamma_n) \tag{8}$$

where $\boldsymbol{\gamma} = \langle \gamma_n | n \in \{1, \cdots, N\} \rangle$ are variational parameters. Here, γ_n parameterizes an N-dimensional multinomial distribution for the indicator z_n of x_n. As shown in Figure 2, marginalizing out $\boldsymbol{\theta}^*$ removes the need to specify variational parameters for $\boldsymbol{\theta}^*$. Thus, CVB searches for an optimum in a less restricted space than VB, which may lead to a better posterior approximation than VB.

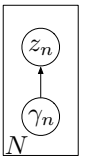

Fig. 2. Graphical model representation of the collapsed variational distribution used to approximate the posterior in NBCE

The lower bound $\mathcal{L}_{CVB}$ for the log-likelihood is:

$$\begin{aligned} &\log p(\boldsymbol{Y}|\alpha_0, G_0,) \geq \\ &\quad E_{q(\boldsymbol{Z}|\boldsymbol{\gamma})}[\log p(\boldsymbol{Y}, \boldsymbol{Z}|\alpha_0, G_0,)] - E_{q(\boldsymbol{Z}|\boldsymbol{\gamma})}[\log q(\boldsymbol{Z}|\boldsymbol{\gamma})] \end{aligned} \tag{9}$$

By taking the derivatives of $\mathcal{L}_{CVB}$ with respect to $q(z_n = k|\gamma_n)$, we have:

$$\begin{aligned} q(z_n = k|\gamma_n) \propto \exp\Bigg\{ E_{q(\boldsymbol{Z}_{\neg n},|\boldsymbol{\gamma})} \Bigg[\log p(z_n = k|\boldsymbol{Z}_{\neg n}) + \\ \left(\sum_{m=1}^{M} \sum_{j_m=1}^{J_m} \log(\beta + \mathcal{N}^{\neg n}_{z_\cdot = k, y_{\cdot m} = j_m}) \right) - \left(\sum_{m=1}^{M} \log(J_m \beta + \mathcal{N}^{\neg n}_{z_\cdot = k}) \right) \Bigg] \Bigg\} \end{aligned} \tag{10}$$

where for the two DP approximations, $\log p(z_n = k|\boldsymbol{Z}_{\neg n})$ is different:

$$\begin{aligned} \log p_{TSB}(z_n = k|\boldsymbol{Z}_{\neg n}) = &\log(1 + \mathcal{N}^{\neg n}_{z_\cdot = k}) - \log(1 + \alpha_0 + \mathcal{N}^{\neg n}_{z_\cdot \geq k}) + \\ &\sum_{h<k} \left[\log(\alpha_0 + \mathcal{N}^{\neg n}_{z_\cdot > h}) - \log(1 + \alpha_0 + \mathcal{N}^{\neg n}_{z_\cdot \geq h}) \right] \\ \log p_{FSD}(z_n = k|\boldsymbol{Z}_{\neg n}) = &\log(\frac{\alpha_0}{K} + \mathcal{N}^{\neg n}_{z_\cdot = k}) - \log(\alpha_0 + N - 1) \end{aligned}$$

Following [26], we apply the first-order latent-space variational Bayesian approximation to Eq (10). Applying the second-order latent-space variational Bayesian inference [27] will lead to a better approximation, but is more expensive. We plan to use it in our future work. Here we just illustrate how to calculate

$E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}[\log(\beta + \mathcal{N}^{\neg n}_{z.=k,y.m=j_m})]$ and $E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}[\log(J_m\beta + \mathcal{N}^{\neg n}_{z.=k})]$. The calculation of other expectations is similar.

According to [26], we have:

$$E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\log(\beta + \mathcal{N}^{\neg n}_{z.=k,y.m=j_m})\right] \approx \log\left(\beta + E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\mathcal{N}^{\neg n}_{z.=k,y.m=j_m}\right]\right)$$
$$E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\log(J_m\beta + \mathcal{N}^{\neg n}_{z.=k})\right] \approx \log\left(J_m\beta + E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\mathcal{N}^{\neg n}_{z.=k}\right]\right)$$

Denote $\gamma_{nk} = q(z_n = k|\gamma_n)$, then we get:

$$E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\mathcal{N}^{\neg n}_{z.=k,y.m=j_m}\right] = \sum_{n'=1,n'\neq n}^{N} \gamma_{n'k}\delta(y_{n'm} = j_m)$$
$$E_{q(\boldsymbol{Z}_{\neg n}|\boldsymbol{\gamma})}\left[\mathcal{N}^{\neg n}_{z.=k}\right] = \sum_{n'=1,n'\neq n}^{N} \gamma_{n'k} \tag{11}$$

Calculating all the expectations and plugging them back into Eq (10) yields approximations to $\gamma_{nk} = q(z_n = k|\gamma_n)$. Repeating this process gives an EM-style iterative algorithm for estimating the γ_{nk}. The algorithm terminates when the change in γ_{nk} drops below a threshold.

6 Empirical Evaluation

We compared several ensemble methods. We first used k-means with different initializations to obtain a set of base clusterings. Then we generated a consensus clustering using various clustering ensemble algorithms, including Bayesian clustering ensembles (BCE) [31], mixture model (MM) [28], CSPA, HGPA, and MCLA [25]. All of these are parametric methods. We also compared two different DP approximations, TSB and FSD, and the performance of NBCE estimated with collapsed Gibbs sampling, collapsed and standard variational approximation.

Datasets. We evaluated NBCE on both synthetic and real datasets. We generated a set of synthetic data with two clusters and some outliers to test the robustness of NBCE. The synthetic data are plotted in Figure 3. To generate the base clusterings on the synthetic data, following [31], we randomly added noise into the ground-truth labeling, e.g., we randomly modified the true labels of 5%, 10%, 15% and 20% of the data points. In each case, we generated 10 base noisy clusterings.

We also used five benchmark datasets from the UCI Machine Learning Repository[2]: *Glass*, *Ecoli*, *ImageSegmentation*, *ISOLET*, and *LetterRecognition*. *Glass* contains glass instances described by their chemical components. *Ecoli* contains

[2] `http://archive.ics.uci.edu/ml/`

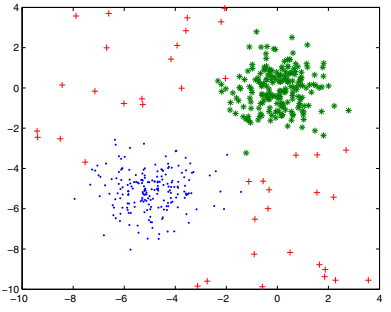

Fig. 3. Synthetic Data: Two Clusters with Outliers

data on E. Coli bacteria. *ImageSegmentation* contains data from images that were hand-segmented classifying each pixel. *ISOLET* contains data representing spoken letters of the alphabet; we selected the letters A, B, C, D, E, and G. *LetterRecognition* contains character images corresponding to the capital letters in the English alphabet; we selected 700 samples of the letters A to J. We also used two time-series datasets from different application domains, namely *Tracedata* and *ControlChart*[3]. *Tracedata* simulates signals representing instrumentation failures. *ControlChart* contains synthetically generated control charts that are classified into one of the following: normal, cyclic, increasing trend, decreasing trend, upward shift, and downward shift.

To generate an ensemble on real data, we varied the number of output clusters of the base clustering algorithms. We computed clustering solutions obtained from multiple runs of k-means with different random initializations. The output clustering solutions were composed of a number of clusters equal to 50%, 75%, 100%, 150%, and 200% of the number of ideal classes of the specific dataset. We used 10 base clusterings for each dataset.

Setting of Clustering Ensemble Methods. For each parametric method and dataset, we set the number of output clusters equal to the actual number of classes, according to the ground truth. For the graph-partitioning-based methods (i.e., CSPA, HGPA, and MCLA), we set the METIS parameters as suggested in [15]. For NBCE, we set the truncation level $K = 100$. When comparing NBCE with other ensemble methods, we use Gibbs sampling for the inference of NBCE.

Evaluation Criteria. Since k-means, CSPA, HGPA, and MCLA are non-generative approaches, to compare the quality of their consensus partitions with NBCE, we evaluated their clustering accuracy using the $F1$-measure. The objective is to evaluate how well a consensus clustering fits the ground-truth partition. The $F1$-measure is defined as the harmonic average of precision and recall. Given a set $D = \{x_1, \cdots, x_n\}$ of n data objects, and $A = \{A_1, \cdots, A_h\}$ and

[3] For a description see: `http://www.cs.ucr.edu/~eamonn/time_series_data/`

$B = \{B_1, \cdots, B_k\}$ being two clustering solutions defined over D, the precision (P) and recall (R) are defined as:

$$P(A_i, B_j) = \frac{|A_i \cap B_j|}{|A_i|} \qquad R(A_i, B_j) = \frac{|A_i \cap B_j|}{|B_j|}$$

$$P(A, B) = \frac{1}{h} \sum_{i=1}^{k} \max_{j \in \{1,\cdots,k\}} P(A_i, B_j)$$

$$R(A, B) = \frac{1}{h} \sum_{i=1}^{k} \max_{j \in \{1,\cdots,k\}} R(A_i, B_j)$$

The F1-measure is defined as: $F1 = \frac{2P(A,B)R(A,B)}{P(A,B)+R(A,B)}$.

Since MM, BCE and NBCE are generative models, we used perplexity to compare them. The perplexity of the observed base clusterings $\boldsymbol{Y}$ is defined as [6]:

$$perp(\boldsymbol{Y}) = \exp\left(-\frac{\log p(\boldsymbol{Y})}{NM}\right) \tag{12}$$

Clearly, the perplexity monotonically decreases with the log-likelihood. Thus, a lower perplexity value on the training data means that the model fits the data better, and a lower value on the test data means that the model can better explain unseen data.

6.1 Results

Evaluation of Clustering Ensemble Methods. We held out 1/4 of the data to evaluate the predictive performance of MM, BCE and NBCE. Table 2 compares the clustering ensemble results for k-means, CSPA, HGPA, MCLA and NBCE in terms of the $F1$-measure on the real datasets excluding the hold-out set. We can see clearly that all ensemble methods outperform the baseline k-means algorithm, and NBCE gives the highest accuracy for each dataset. A paired t-test of NBCE against the next best accuracy is significant at the 0.002 level. Thus the comparison results of NBCE versus all competitors are statistically significant.

Table 2. $F1$-measure Results

	Base k-means		CSPA	HGPA	MCLA	NBCE
	max	avg				
Glass	0.57	0.51	0.66	0.59	0.61	0.69
Ecoli	0.61	0.56	0.67	0.65	0.68	0.72
ImageSegmentation	0.52	0.42	0.53	0.44	0.59	0.65
ISOLET	0.53	0.41	0.59	0.50	0.65	0.66
LetterRecognition	0.48	0.40	0.49	0.50	0.53	0.62
Tracedata	0.49	0.44	0.51	0.62	0.61	0.66
ControlChart	0.62	0.56	0.73	0.70	0.67	0.77

Table 3. Perplexity Results on the Synthetic Dataset

	5%	10%	15%	20%
MM	10.04	13.27	17.36	21.20
BCE	7.92	9.76	14.22	18.98
NBCE	5.63	8.31	11.16	15.87

Table 4 compares MM, BCE and NBCE in terms of the perplexity on the synthetic datasets. It's clear that NBCE fits the data better than BCE and MM. BCE and MM are parametric models, and thus fail to handle outliers. In contrast, NBCE is robust to outliers because it can find the number of clusters that fits the data best.

Tables 4 and 5 compare MM, BCE and NBCE in terms of the perplexity on training and test (i.e., hold-out) data for the real datasets. NBCE fits the data better than BCE, and BCE is better than MM.

Table 4. Perplexity Results on Training data for Real Datasets

	Glass	Ecoli	ImageSegmentation	ISOLET	LetterRecognition	Tracedata	ControlChart
MM	1.02	1.33	1.40	1.63	2.21	2.97	4.34
BCE	0.99	1.10	1.23	1.34	1.98	2.53	4.01
NBCE	0.77	0.92	1.03	1.24	1.76	2.38	3.63

Table 5. Perplexity Results on Test Data for Real Datasets

	Glass	Ecoli	ImageSegmentation	ISOLET	LetterRecognition	Tracedata	ControlChart
MM	1.15	1.51	1.49	1.72	2.51	3.22	5.56
BCE	1.07	1.39	1.37	1.60	2.33	2.94	4.88
NBCE	0.98	1.18	1.16	1.47	1.96	2.62	4.58

Comparison of TSB and FSD. In principle, TSB tends to produce larger clusters then FSD. The experimental results confirm this fact by showing that NBCE with a TSB prior gives a smaller number of singleton clusters than NBCE with FSD. Table 6 shows the percentage of outliers in singleton clusters for the five UCI datasets, when using collapsed Gibbs sampling with the two different priors.

Comparison of CVB, VB and Gibbs. Table 7 illustrates the perplexity of the three inference methods of NBCE on the UCI datasets excluding the hold-out set. Collapsed Gibbs sampling is asymptotically unbiased, so it gives lower perplexity than CVB and VB; CVB has less restricted assumption than VB, and CVB has lower perplexity than VB. The perplexity is calculated at convergence.

Table 6. Outlier Percentage

	TSB	FSD
Glass	3.2%	5.4%
Ecoli	4.3%	5.1%
ImageSegmentation	3.2%	3.5%
ISOLET	2.9%	3.1%
LetterRecognition	3.3%	3.6%

Table 7. Perplexity of Gibbs, CVB and VB

	Gibbs	CVB	VB
Glass	0.77	0.85	0.91
Ecoli	0.92	0.96	1.02
ImageSegmentation	1.03	1.06	1.11
ISOLET	1.24	1.28	1.30
LetterRecognition	1.76	1.80	1.88

7 Conclusion

A nonparametric Bayesian clustering ensemble model was proposed and three inference methods were considered: collapsed Gibbs sampling, variational Bayesian inference, and collapsed variational Bayesian inference. The versatility, and superior stability and accuracy of NBCE were demonstrated through empirical evaluation.

Acknowledgements

This work is in part supported by NSF CAREER Award IIS-0447814.

References

1. Alexey, D.G., Tsymbal, A., Bolshakova, N., Cunningham, P.: Ensemble clustering in medical diagnostics. In: IEEE Symposium on Computer-Based Medical Systems, pp. 576–581. IEEE Computer Society, Los Alamitos (2004)
2. Antoniak, C.E.: Mixtures of dirichlet processes with applications to bayesian nonparametric problems. The Annals of Statistics 2(6), 1152–1174 (1974)
3. Ayad, H., Kamel, M.: Finding natural clusters using multi-clusterer combiner based on shared nearest neighbors. In: Windeatt, T., Roli, F. (eds.) MCS 2003. LNCS, vol. 2709, pp. 166–175. Springer, Heidelberg (2003)
4. Beal, M.J.: Variational Algorithms for Approximate Bayesian Inference. PhD thesis, Gatsby Computational Neuroscience Unit, University College London (2003)
5. Blackwell, D., Macqueen, J.B.: Ferguson distributions via pólya urn schemes. The Annals of Statistics 1, 353–355 (1973)
6. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research 3(4-5), 993–1022 (2003)
7. Dudoit, S., Fridlyand, J.: Bagging to improve the accuracy of a clustering procedure. Bioinformatics 19(9), 1090–1099 (2003)
8. Ferguson, T.S.: A bayesian analysis of some nonparametric problems. The Annals of Statistics 1(2), 209–230 (1973)
9. Fern, X.Z., Brodley, C.E.: Random projection for high-dimensional data clustering: A cluster ensemble approach. In: International Conference on Machine Learning, pp. 186–193 (2003)
10. Fred, A.L.N., Jain, A.K.: Data clustering using evidence accumulation. In: International Conference on Pattern Recognition, Washington, DC, USA, vol. 4, pp. 276–280. IEEE Computer Society, Los Alamitos (2002)
11. Gondek, D., Hofmann, T.: Non-redundant clustering with conditional ensembles. In: KDD '05: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, pp. 70–77. ACM, New York (2005)

12. Hu, X.: Integration of cluster ensemble and text summarization for gene expression analysis. In: Fourth IEEE Symposium on Bioinformatics and Bioengineering, pp. 251–258 (2004)
13. Ishwaran, H., James, L.: Gibbs sampling methods for stick breaking priors. Journal of the American Statistical Association 96(453), 161–173 (2001)
14. Ishwaran, H., Zarepour, M.: Exact and approximate sum-representations for the dirichlet process. The Canadian Journal of Statistics 30(2), 269–283 (2002)
15. Karypis, G., Kumar, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on Scientific Computing 20(1), 359–392 (1998)
16. Kuncheva, L.I.: Experimental comparison of cluster ensemble methods. In: International Conference on Information Fusion, pp. 1–7 (2006)
17. Kuncheva, L.I., Hadjitodorov, S.T.: Using diversity in cluster ensembles. In: International Conference on Systems, Man and Cybernetics, vol. 2, pp. 1214–1219 (2004)
18. Kuncheva, L.I., Vetrov, D.: Evaluation of stability of k-means cluster ensembles with respect to random initializationtent semantic analysis. PAMI 28(11), 1798–1808 (2006)
19. Kurihara, K., Welling, M., Teh, Y.W.: Collapsed variational dirichlet process mixture models. In: IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence, pp. 2796–2801. Morgan Kaufmann Publishers Inc., San Francisco (2007)
20. Minaei-bidgoli, B., Topchy, A., Punch, W.F.: A comparison of resampling methods for clustering ensembles. In: International Conference on Machine Learning: Models, Technologies and Applications, pp. 939–945 (2004)
21. Neal, R.M.: Probabilistic inference using markov chain monte carlo methods. Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto (1993)
22. Neal, R.M.: Markov chain sampling methods for dirichlet process mixture models. Journal of Computational and Graphical Statistics 9(2), 249–265 (2000)
23. Punera, K., Ghosh, J.: Soft cluster ensembles. In: de Oliveira, J.V., Pedrycz, W. (eds.) Advances in Fuzzy Clustering and its Applications, ch. 4, pp. 69–90. John Wiley & Sons, Ltd., Chichester (2007)
24. Sethuraman, J.: A constructive definition of dirichlet priors. Statistica Sinica 4, 639–650 (1994)
25. Strehl, A., Ghosh, J.: Cluster ensembles — a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3, 583–617 (2003)
26. Sung, J., Ghahramani, Z., Bang, S.-Y.: Latent-space variational bayes. IEEE Trans. Pattern Anal. Mach. Intell. 30(12), 2236–2242 (2008)
27. Sung, J., Ghahramani, Z., Bang, S.-Y.: Second-order latnet-space variational bayes for approximate bayesian inference. IEEE Signal Processing Letters 15, 918–921 (2008)
28. Topchy, A., Jain, A.K., Punch, W.: A mixture model for clustering ensembles. In: SIAM International Conference on Data Mining, pp. 379–390 (2004)
29. Topchy, A., Jain, A.K., Punch, W.: Clustering ensembles: models of consensus and weak partitions. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(12), 1866–1881 (2005)
30. Topchy, A., Topchy, E., Jain, A.K., Punch, W.: Combining multiple weak clusterings. In: International Conference on Data Mining, pp. 331–338 (2003)
31. Wang, H., Shan, H., Banerjee, A.: Bayesian clustering ensembles. In: SIAM Data Mining (2009)

Appendix

$\mathcal{L}_{VB}$, Eq (7), has 7 terms. After the expansion, $\mathcal{L}_{VB}$ can be rewritten as follows, where each line corresponds to a term of Eq (7):

$$
\begin{aligned}
&\mathcal{L}_{VB} = \\
&\left(\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j_m=1}^{J_m}\gamma_{nk}\delta(y_{nm},j_m)\log\rho_{kmj_m}\right) + \\
&\left(\log\Gamma(\alpha_0) - K\log\Gamma(\frac{\alpha_0}{K}) + (\frac{\alpha_0}{K}-1)\sum_{k=1}^{K}\left[\Psi(\xi_k) - \Psi(\sum_{h=1}^{K}\xi_h)\right]\right) + \\
&\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}[\Psi(\xi_k) - \Psi(\sum_{h=1}^{K}\xi_h)] + \\
&\sum_{m=1}^{M}\left(K\log\Gamma(J_m\beta) - KJ_m\log\Gamma(\beta) + (\beta-1)\sum_{k=1}^{K}\sum_{j_m=1}^{J_m}\left[\Psi(\rho_{kmj_m}) - \Psi(\sum_{h=1}^{J_m}\rho_{kmh})\right]\right) - \\
&\left(\log\Gamma(\sum_{k=1}^{K}\xi_k) - \sum_{k=1}^{K}\log\Gamma(\xi_k) + \sum_{k=1}^{K}(\xi_k-1)[\Psi(\xi_k) - \Psi(\sum_{h=1}^{K}\xi_h)]\right) - \\
&\sum_{k=1}^{K}\sum_{m=1}^{M}\left(\log\Gamma(\sum_{j_m=1}^{J_m}\rho_{kmj_m}) - \sum_{j_m=1}^{J_m}\log\Gamma(\rho_{kmj_m}) + \right. \\
&\qquad\left.\sum_{j_m=1}^{J_m}(\rho_{kmj_m}-1)\left[\Psi(\rho_{kmj_m}) - \Psi(\sum_{h=1}^{J_m}\rho_{kmh})\right]\right) - \\
&\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma_{nk}\log\gamma_{nk}
\end{aligned}
$$

Here, $\delta(\cdot,\cdot)$ is the Kronecker delta function; $\Psi(\cdot)$ is the digamma function, the first derivative of the log Gamma function; $\gamma_{nk} = q(z_n = k|\gamma_n)$; $\rho_{kmj_m} = q(\theta^*_{kmj_m}|\rho_{km})$; and $\gamma_{nk} = q(z_n = k|\gamma_n)$.