Latent Dirichlet Bayesian Co-Clustering

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. Co-clustering has emerged as an important technique for mining contingency data matrices. However, almost all existing co-clustering algorithms are hard partitioning, assigning each row and column of the data matrix to one cluster. Recently a Bayesian co-clustering approach has been proposed which allows a probability distribution membership in row and column clusters. The approach uses variational inference for parameter estimation. In this work, we modify the Bayesian co-clustering model, and use collapsed Gibbs sampling and collapsed variational inference for parameter estimation. Our empirical evaluation on real data sets shows that both collapsed Gibbs sampling and collapsed variational inference are able to find more accurate likelihood estimates than the standard variational Bayesian co-clustering approach.

Key words: Co-Clustering, Graph Learning, Dirichlet Distribution

1 Introduction

Co-clustering [2] has emerged as an important approach for mining dyadic and relational data. Often, data can be organized in a matrix, where rows and columns present a symmetrical relation. For example, documents can be represented as a matrix, where rows are indexed by the documents, and columns by words. Co-clustering allows documents and words to be grouped simultaneously: documents are clustered based on the contained words, and words are grouped based on the documents they appear in. The two clustering processes are inter-dependent.

Some researchers have proposed a hard-partition version [3], others a soft-partition version [1] of co-clustering. In the hard-partition case, each row (column) is assigned to exactly one row (column) cluster. In the soft-partition case, each row (column) has a probability of belonging to each row (column) cluster.

The Bayesian Co-Clustering (BCC) model proposed in [1] is a kind of generative model. BCC maintains separate Dirichlet priors for the distribution of row- and column-clusters given rows and columns. To generate each entry in the data matrix, the model first generates the row and column clusters of the current entry according to this Dirichlet distribution. The value of the current entry is then generated according to the corresponding row-cluster and column-cluster. The advantage of a generative model is that it can be used to predict unseen

data. Like the original *Latent Dirichlet Allocation* (LDA) [5] model, though, BCC assumes uniform priors for the entry value distributions given row- and column-clusters. The authors in [1] proposed a variational Bayesian algorithm to perform inference and estimate the BCC model. A lower bound of the likelihood function is learned and used to estimate model parameters.

In this work, we extend the BCC model and propose a collapsed Gibbs sampling and a collapsed variational Bayesian algorithm for it. Following [5], first we smooth the BCC model, by introducing priors for the entry value distributions given row- and column-clusters. Following [5], we call our approach *Latent Dirichlet Bayesian Co-Clustering* (LDCC), since it assumes Dirichlet priors for row- and column-clusters, which are unobserved in the data contingency matrix. The collapsed Gibbs sampling and collapsed variational Bayesian algorithms we propose can learn more accurate likelihood functions than the standard variational Bayesian algorithm [1]. This result is derived analytically for the collapsed variational Bayesian algorithm. More accurate likelihood estimates can lead to higher predictive performance, as corroborated by our experimental results.

The rest of the paper is organized as follows. In Section 2, we discuss related work. Section 3 introduces the LDCC model and the variational Bayesian algorithm. We then discuss the collapsed Gibbs sampling and the collapsed variational Bayesian algorithms. Section 4 demonstrates our empirical evaluation of the three methods. Finally, Section 5 summarizes the paper.

2 Related Work

Our work is closely related to [1], which we discuss in Section 3.1. Dhillon et al. proposed an information-theoretic co-clustering approach (hard-partition) in [3]. Shafiei et al. proposed a soft-partition co-clustering, called "Latent Dirichlet Co-clustering" in [4]. The proposed model, though, does not cluster rows and columns simultaneously. It first defines word-topics, i.e., groups of words, and then defines document-topics, i.e., groups of word-topics. Documents are modeled as mixtures of such document-topics. Thus, the resulting model is similar to a hierarchical extension of the "Latent Dirichlet Allocation" [5] model, since the defined document-topics are not groups of documents, but groups of word-topics. Our LDCC model and BCC [1] model assume independence between row-clusters and column-clusters, which is the same assumption as in [3].

Blei et al. proposed "Latent Dirichlet Allocation" (LDA) [5], which assumes that topics are mixtures of words, and documents are mixtures of topics. A standard variational Bayesian algorithm [5] is used to estimate the posterior distribution of model parameters given the model evidence. Griffiths et al. used a collapsed Gibbs sampling method to learn the posterior distribution of parameters for the LDA model [9]. Recently, Teh et al. proposed a collapsed variational Bayesian algorithm to perform model inference for LDA and "Hierarchical Dirichlet Processing"[7, 10].

3 Latent Dirichlet Co-Clustering

In this section, we first introduce the LDCC model. We then discuss three different learning methods: variational Bayesian, collapsed Gibbs sampling, and collapsed variational Bayesian. Table 1 gives a summary of the notation used.

Symbol	Description
X	data matrix
u	index for row
v	index for column
$[u, v]$	entry of the matrix at row u and column v
x_{uv}	value for matrix entry at row u and column v
$\mathcal{X}$	entry value set
i	index for row clusters
j	index for column clusters
N_1	number of rows
N_2	number of columns
$\boldsymbol{z}_1$	row clusters
$\boldsymbol{z}_2$	column clusters
K_1	number of row clusters
K_2	number of column clusters
α_1	Dirichlet prior hyperparameter for for rows
α_2	Dirichlet prior hyperparameter for columns
β	Dirichlet prior hyperparameter for the probabilities of each entry value given a row- and a column-clusters
$\boldsymbol{\pi}_1$	The probabilities of each row-cluster given each row
$\boldsymbol{\pi}_2$	The probabilities of each column-cluster given each column
$\theta_{ijx_{uv}}$	probability of entry value x_{uv} give $z_1 = i$ and $z_2 = j$
$n_{ijx_{uv}}$	number of entries with value x_{uv} assigned to row cluster i and column cluster j
n_{ui}	number of entries in row u assigned to row cluster i
n_{vj}	number of entries in column v assigned to column cluster j
n_{ij}	number of entries in matrix assigned to row cluster i and column cluster j
n_u	number of entries in row u
n_v	number of entries in column v

Fig. 1. Notation used in this paper

Given an $N_1 \times N_2$ data matrix X, the values x_{uv} of each entry $[u, v]$, $u = 1, \ldots, N_1, v = 1, \ldots, N_2$ are defined in a value set, $x_{uv} \in \mathcal{X}$. For co-clustering, we assume there are K_1 row clusters z_1, and K_2 column clusters z_2. LDCC assumes two Dirichlet priors[3] $Dir(\alpha_1)$ and $Dir(\alpha_2)$ for rows and columns respectively, $\alpha_1 =< \alpha_{1_u} | u = 1, \ldots, N_1 >$, $\alpha_2 =< \alpha_{2_v} | v = 1, \cdots, N_2 >$, from which the probabilities of each row-cluster z_1 and column-cluster z_2 given each row u and each column v are generated, denoted as π_{1_u} and π_{2_v} respectively. Row clusters for entries in row u and column clusters for entries in column v are sampled from multinomial distributions $p(z_1|\pi_{1_u})$ and $p(z_2|\pi_{2_v})$ respectively. We denote $\boldsymbol{\pi}_1 =< \pi_{1_u} | u = 1, \ldots, N_1 >$, $\boldsymbol{\pi}_2 =< \pi_{2_v} | v = 1, \ldots, N_2 >$, $\boldsymbol{z}_1 =< z_{1_{uv}} | u = 1, \ldots, N_1, v = 1, \ldots, N_2 >$ and $\boldsymbol{z}_2 =< z_{2_{uv}} | u = 1, \ldots, N_1, v = 1, \ldots, N_2 >$, where $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ are row- and column-cluster assignment for all entries in the data matrix X. A row cluster $z_1 = i$ and a column cluster $z_2 = j$ together decide a co-cluster $(z_1, z_2) = (i, j)$, and entries in the matrix are also sampled from a multinomial distribution $p(x|\theta_{z_1,z_2})$ given a co-cluster $(z_1, z_2) = (i, j)$. We denote $\boldsymbol{\theta} =< \theta_{z_1=i,z_2=j} | i = 1, \ldots, K_1, j = 1, \ldots, K_2 >$. Here, $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ are

[3] In the rest of the paper, we assume symmetric Dirichlet priors, which means α_1 and α_2 do not depend on u or v.

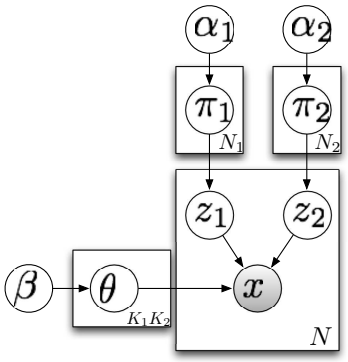

Fig. 2. Latent Dirichlet Bayesian Co-clustering Model

latent variables, while $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$ and $\boldsymbol{\theta}$ are unknown parameters. The generative process for the whole data matrix is as follows (see Figure 2):

- For each row u, choose $\pi_{1_u} \sim Dir(\alpha_1)$
- For each column v, choose $\pi_{2_v} \sim Dir(\alpha_2)$
- To generate the entry of row u and column v:
  - choose $z_{1_{uv}} \sim p(z_1|\pi_{1_u})$, $z_{2_{uv}} \sim p(z_2|\pi_{2_v})$
  - choose $\theta_{z_{1_{uv}} z_{2_{uv}}} \sim Dir(\beta)$
  - choose $x_{uv} \sim p(x|z_{1_{uv}}, z_{2_{uv}}, \theta_{z_{1_{uv}}, z_{2_{uv}}})$.

The LDCC model proposed here departs from the BCC model [1] by introducing a prior β for $\theta_{z_1 z_2}$. Thus, LDCC can assign a probability to an unseen entry value according to $p(\theta_{z_1,z_2}|\beta)$.

The marginal probability of an entry x in the data matrix X is given by:

$$p(x|\alpha_1, \alpha_2, \beta) = \int_{\pi_1}\int_{\pi_2}\int_{\theta} p(\pi_1|\alpha_1)p(\pi_2|\alpha_2)p(\theta_{z_1 z_2}|\beta) \cdot \sum_{z_1}\sum_{z_2} p(z_1|\pi_1)p(z_2|\pi_2)p(x|\theta_{z_1 z_2})\, d\pi_1 d\pi_2 d\theta_{z_1 z_2} \quad (1)$$

Note that the entries in the same row/column are generated from the same π_{1_u} or π_{2_v}, so the entries in the same row/column are related. Therefore, the model introduces a coupling between observations in the same row/column [1].

The overall joint distribution over X, $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$, $\boldsymbol{z}_1$, $\boldsymbol{z}_2$ and $\boldsymbol{\theta}$ is given by:

$$p(X, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\theta}|\alpha_1, \alpha_2, \beta) = \prod_u p(\pi_{1_u}|\alpha_1) \prod_v p(\pi_{2_v}|\alpha_2) \cdot \prod_{K_1}\prod_{K_2} p(\theta_{z_1,z_2}|\beta) \prod_{u,v} p(z_{1_{uv}}|\pi_{1_u})p(z_{2_{uv}}|\pi_{2_v})p(x_{uv}|\theta_{z_1,z_2}, z_{1_{uv}}, z_{2_{uv}})^{\delta_{uv}} \quad (2)$$

where δ_{uv} is an indicator function which takes value 0 when x_{uv} is empty, and 1 otherwise (only the non-missing entries are considered); $z_{1_{uv}} \in \{1, \ldots, K_1\}$ is

the latent row cluster, and $z_{2_{uv}} \in \{1, \ldots, K_2\}$ is the latent column cluster for observation x_{uv}.

Marginalizing out all unknown parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$ and $\boldsymbol{\theta}$, the marginal likelihood of observed and latent variables is:

$$p(X, \boldsymbol{z}_1, \boldsymbol{z}_2 | \alpha_1, \alpha_2, \beta) = p(X | \boldsymbol{z}_1, \boldsymbol{z}_2, \beta) p(\boldsymbol{z}_1 | \alpha_1) p(\boldsymbol{z}_2 | \alpha_2) = \tag{3}$$
$$\int p(X | \boldsymbol{\theta}, \boldsymbol{z}_1, \boldsymbol{z}_2) p(\boldsymbol{\theta} | \beta) \, d\boldsymbol{\theta} \int p(\boldsymbol{z}_1 | \boldsymbol{\pi}_1) p(\boldsymbol{\pi}_1 | \alpha_1) \, d\boldsymbol{\pi}_1 \int p(\boldsymbol{z}_2 | \boldsymbol{\pi}_2) p(\boldsymbol{\pi}_2 | \alpha_2) \, d\boldsymbol{\pi}_2$$

Summing over all possible latent variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, the probability of observing the entire matrix X is:

$$p(X | \alpha_1, \alpha_2, \beta) = \int_{\boldsymbol{\pi}_1} \int_{\boldsymbol{\pi}_2} \int_{\boldsymbol{\theta}} \left(\prod_u p(\pi_{1_u} | \alpha_1) \right) \left(\prod_v p(\pi_{2_v} | \alpha_2) \right) \left(\prod_{z_1, z_2} p(\theta_{z_1, z_2} | \beta) \right)$$
$$\cdot \left(\prod_{u,v} \sum_{z_{1_{uv}}} \sum_{z_{2_{uv}}} p(z_{1_{uv}} | \pi_{1_u}) p(z_{2_{uv}} | \pi_{2_v}) p(x_{uv} | \theta_{z_{1_{uv}}, z_{2_{uv}}})^{\delta_{uv}} \right) d\boldsymbol{\pi}_1 d\boldsymbol{\pi}_2 d\boldsymbol{\theta} \tag{4}$$

3.1 Variational Bayesian Algorithm for BCC

In this section, we briefly describe the variational Bayesian algorithm for the original BCC model (see Appendix). The BCC model assumes uniform priors for $\boldsymbol{\theta}$ and assumes that $\boldsymbol{\theta}$ has a Gaussian distribution. The authors in [1] derived a variational algorithm for their model. In this paper, we assume that the values for each entry in the data matrix are discrete[4]. We do so for mathematical convenience of the derivation of the collapsed Gibbs sampling and the collapsed variational Bayesian algorithms. Thus, unlike [1], we do not assume that $\boldsymbol{\theta}$ has a Gaussian distribution.

The variational Bayesian algorithm introduces $q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$ as an approximation of the actual distribution $p(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | X, \alpha_1, \alpha_2, \boldsymbol{\theta})$, where $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ are called variational variables, $\boldsymbol{\gamma}_1 =< \gamma_{1_u} | u = 1, \cdots, N_1 >$, $\boldsymbol{\gamma}_2 =< \gamma_{2_v} | v = 1, \cdots, N_2 >$, $\boldsymbol{\phi}_1 =< \phi_{1_u} | u = 1, \cdots, N_1 >$, $\boldsymbol{\phi}_2 =< \phi_{2_v} | v = 1, \cdots, N_2 >$, γ_{1_u} and γ_{2_v} are variational Dirichlet distribution parameters with K_1 and K_2 dimensions respectively for rows and columns, ϕ_{1_u} and ϕ_{2_v} are multinomial parameters with K_1 and K_2 dimensions for rows and columns. It is assumed that $q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$ can be fully factorized as:

$$q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2) =$$
$$\left(\prod_{u=1}^{N_1} q(\pi_{1_u} | \gamma_{1_u}) \right) \left(\prod_{v=1}^{N_2} q(\pi_{2_v} | \gamma_{2_v}) \right) \left(\prod_{u=1}^{N_1} \prod_{v=1}^{N_2} q(z_{1_{uv}} | \phi_{1_u}) q(z_{2_{uv}} | \phi_{2_v}) \right) \tag{5}$$

[4] Technically, our theory applies to any exponential family distribution for data matrix.

The factorization assumption of $q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2)$ means that parameters and latent variables are independent, and the assignment of $z_{1_{uv}}$ and $z_{2_{uv}}$ for the current entry $[u, v]$ is independent of the assignments for other entries.

The variational Bayesian algorithm can find a lower bound of the true log-likelihood:

$$\log p(X|\alpha_1, \alpha_2, \boldsymbol{\theta}) \geq \tag{6}$$
$$E_q[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \alpha_1, \alpha_2, \boldsymbol{\theta})] - E_q[\log q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2)]$$

and we denote the lower bound as $L(\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \alpha_1, \alpha_2, \boldsymbol{\theta})$.

The variational Bayesian algorithm is an EM-style method: the E-step estimates the values for $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ that maximize the lower bound of the log-likelihood based on α_1, α_2 and $\boldsymbol{\theta}$; the M-step estimates α_1, α_2 and $\boldsymbol{\theta}$ according to the log-likelihood lower bound based on $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ learned during the previous E-step. Thus, in the E-step, in order to maximize $L(\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \alpha_1, \alpha_2, \boldsymbol{\theta})$, one takes the derivative of L w.r.t $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ respectively, and sets it to zero. We get:

$$\phi_{1_{ui}} \propto exp\left(\Psi(\gamma_{1_{ui}}) + \frac{\sum_{u=1}^{N_1}\sum_{i=1}^{K_1}\delta_{uv}\phi_{2_{vj}}\log\theta_{ijx_{uv}}}{n_u}\right) \tag{7}$$

$$\phi_{2_{vj}} \propto exp\left(\Psi(\gamma_{2_{vj}}) + \frac{\sum_{v=1}^{N_2}\sum_{j=1}^{K_2}\delta_{uv}\phi_{1_{vi}}\log\theta_{ijx_{uv}}}{n_v}\right) \tag{8}$$

$$\gamma_{1_{ui}} \propto \alpha_{1_i} + n_u\phi_{1_{u_i}} \tag{9}$$

$$\gamma_{2_{vj}} \propto \alpha_{2_j} + n_v\phi_{2_{v_j}} \tag{10}$$

where n_u and n_v are the number of entries in row u and column v respectively, and $\Psi(\cdot)$ is the digamma function, the first derivative of $\log\Gamma(\cdot)$, the log Gamma function. In the M-step, to estimate the Dirichlet parameters α_1 and α_2, one can use Newton method, as shown in [5] for LDA, to estimate $\boldsymbol{\theta}$, one takes the derivative of L w.r.t $\boldsymbol{\theta}$ and setting it to zero. We get:

$$\theta_{ijx_{uv}} \propto \sum_{u'=1}^{N_1}\sum_{v'=1}^{N_2}\delta_{u'v'}(x_{uv})\phi_{1_{u'i}}\phi_{2_{v'j}} \tag{11}$$

where $\delta_{u'v'}(x_{uv})$ is an indicator function, which equals 1 if the value of the entry at row u' and column v' equals to x_{uv}, 0 otherwise. The variational Bayesian method iterates through the E-step and the M-step until convergence.

Although efficient and easy to implement, the variational Bayesian algorithm can potentially lead to inaccurate results. The latent variables $\boldsymbol{z}_1$, $\boldsymbol{z}_2$ and the parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$, $\boldsymbol{\theta}$ can have a strong inter-dependence in the true posterior $p(X, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2 | \alpha_1, \alpha_2, \boldsymbol{\theta})$. This dependence is ignored in the variational Bayesian algorithm which assumes independence between the latent variables and the parameters. As a result, the lower bound learned for the log marginal likelihood can be very loose, leading to inaccurate estimates of the posterior.

3.2 Collapsed Gibbs Sampling for LDCC

Standard Gibbs sampling [8], which iteratively samples the latent variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, and the parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$ and $\boldsymbol{\theta}$, may converge very slowly due to the strong dependencies between the latent variables and the parameters. Collapsed Gibbs sampling improves upon Gibbs sampling by marginalizing out the parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$ and $\boldsymbol{\theta}$, and then sampling the latent variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ only, over the so called collapsed space. Consider a model in which each matrix element can have a discrete value from a value set $\mathcal{X}$, with $|\mathcal{X}| = N$. Using a symmetric Dirichlet prior, the marginal likelihood over X, $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, (Equation (3)), can be rewritten as:

$$p(X, \boldsymbol{z}_1, \boldsymbol{z}_2 | \alpha_1, \alpha_2, \beta) = \prod_{u=1}^{N_1} \left(\frac{\Gamma(K_1\alpha_1)}{\Gamma(K_1\alpha_1 + n_u)} \prod_{i=1}^{K_1} \frac{\Gamma(\alpha_1 + n_{ui})}{\Gamma(\alpha_1)} \right) \tag{12}$$
$$\prod_{v=1}^{N_2} \left(\frac{\Gamma(K_2\alpha_2)}{\Gamma(K_2\alpha_2 + n_v)} \prod_{j=1}^{K_2} \frac{\Gamma(\alpha_2 + n_{vj})}{\Gamma(\alpha_2)} \right) \prod_{i=1}^{K_1} \prod_{j=1}^{K_2} \left(\frac{\Gamma(N\beta)}{\Gamma(N\beta + n_{ij})} \prod_{x=1}^{N} \frac{\Gamma(\beta + n_{ijx})}{\Gamma(\beta)} \right)$$

Given all the latent variables but the ones for entry $[u, v]$, the conditional probability of $z_{1_{uv}} = i$ and $z_{2_{uv}} = j$ is:

$$p(z_{1_{uv}} = i, z_{2_{uv}} = j | X, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}, \alpha_1, \alpha_2, \beta) = \frac{(\alpha_1 + n_{ui}^{\neg uv})(\alpha_2 + n_{vj}^{\neg uv})(\beta + n_{ijx_{uv}}^{\neg uv})}{(K_1\alpha_1 + n_u^{\neg uv})(K_2\alpha_2 + n_v^{\neg uv})(N\beta + n_{ij}^{\neg uv})} \tag{13}$$

where $\neg uv$ denotes the corresponding count with x_{uv}, $z_{1_{uv}}$ and $z_{2_{uv}}$ excluded. The derivation can be found in the Appendix. The conditional probability can be rewritten as:

$$p(z_{1_{uv}} = i, z_{2_{uv}} = j | X, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}, \alpha_1, \alpha_2, \beta) = \frac{(\alpha_1 + n_{ui}^{\neg uv})(\alpha_2 + n_{vj}^{\neg uv})(\beta + n_{ijx_{uv}}^{\neg uv})(N\beta + n_{ij}^{\neg uv})^{-1}}{\sum_{i'=1}^{K_1} \sum_{j'}^{K_2} (\alpha_1 + n_{ui'}^{\neg uv})(\alpha_2 + n_{vj'}^{\neg uv})(\beta + n_{i'j'x_{uv}}^{\neg uv})(N\beta + n_{i'j'}^{\neg uv})^{-1}} \tag{14}$$

where the numerator covers the factors specific to $z_{1_{uv}} = i$ and $z_{2_{uv}} = j$, and the denominator serves as a normalization factor by summing over all combination of z_1 and z_2 for the current entry $[u, v]$.

Note that since collapsed Gibbs sampling marginalizes out the parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$ and $\boldsymbol{\theta}$, it induces new dependencies between the latent variables $z_{1_{uv}}, z_{2_{uv}}$ (which are conditionally independent given the parameters) [7]. Equation (14) shows that $z_{1_{uv}}$ and $z_{2_{uv}}$ depend on $\boldsymbol{z}_1^{\neg uv}$, $\boldsymbol{z}_2^{\neg uv}$ only through the counts $n_{ui'}^{\neg uv}$, $n_{vj'}^{\neg uv}$ and $n_{i'j'}^{\neg uv}$, which is to say that the dependence of $z_{1_{uv}} = i$ and $z_{2_{uv}} = j$ on any other variable $z_{1_{uv}} = i'$, $z_{2_{uv}} = j'$ is very small, especially for large datasets. This is precisely the right setting for a mean field (i.e., fully factorized variational) approximation: a particular variable interacts with the remaining variables only through a summary statistics called the field, and the impact of any single variable on the field is very small [7]. On the contrary, this is not

true in the joint space of parameters and latent variables because fluctuations in parameters can have a signicant impact on latent variables. As a consequence, the mean field assumption fits better the collapsed space of latent variables than the joint space of latent variables and parameters.

Given Equation (13) or (14), Gibbs sampling can generate row- and column-cluster probabilities for the current entry conditioned on row- and column-clusters for the other entries. One can calculate the following stationary distributions:

$$p(x_{uv}|z1_{uv} = i, z2_{uv} = j) = \frac{n_{ijx_{uv}} + \beta}{n_{ij} + N\beta} \tag{15}$$

$$p(z1_{uv} = i|u) = \frac{n_{ui} + \alpha_1}{n_u + K_1\alpha_1} \tag{16}$$

$$p(z2_{uv} = j|v) = \frac{n_{vj} + \alpha_2}{n_v + K_2\alpha_2} \tag{17}$$

which correspond to $\theta_{z_1=i,z_2=j}$, π_{1_u} and π_{2_v}.

Although Gibbs sampling leads to unbiased estimators, it also has some drawbacks: one needs to assess convergence of the Markov chain and to have some idea of mixing times to estimate the number of samples to collect, and to identify coherent topics across multiple samples. In practice, one often ignores these issues and collects as many samples as is computationally feasible, while the question of topic identication is often sidestepped by using just one sample. Hence, there still is a need for more efficient, accurate and deterministic inference procedures.

3.3 Collapsed Variational Bayesian Algorithm for LDCC

The collapsed variational Bayesian algorithm for LDCC is similar to the standard variational Bayesian one, except for the optimization of the lower bound of the log-likelihood in the collapsed space, which is inspired by collapsed Gibbs sampling. There are two ways to derive the collapsed variational Bayesian algorithm for LDCC, either in the collapsed space or in the original joint space of latent variables and parameters.

We start from the collapsed space with parameters marginalized out. We introduce $q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})$ to approximate $p(\boldsymbol{z}_1, \boldsymbol{z}_2|X, \alpha_1, \alpha_2, \beta)$, where $\boldsymbol{\gamma} =< \boldsymbol{\gamma}_{uv}|u = 1, \cdots, N_1, v = 1, \cdots, N_2 >$, and $\boldsymbol{\gamma}_{uv} =< \gamma_{uvij}|i = 1, \cdots, K_1, j = 1, \cdots, K_2 >$. Assume that $q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})$ can be factorized as:

$$q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma}) = \prod_{u=1}^{N_1}\prod_{v=1}^{N_2} q(z1_{uv}, z2_{uv}|\boldsymbol{\gamma}_{uv}) \tag{18}$$

where $q(z1_{uv}, z2_{uv}|\boldsymbol{\gamma}_{uv})$ is a multinomial with parameters $\boldsymbol{\gamma}_{uv}$.

The lower bound of the log-likelihood is:

$$\begin{aligned}&\log p(X|\alpha_1, \alpha_2, \beta) \geq \\ &\quad E_{q(\boldsymbol{z}_1,\boldsymbol{z}_2|\boldsymbol{\gamma})}[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)] - E_{q(\boldsymbol{z}_1,\boldsymbol{z}_2|\boldsymbol{\gamma})}[\log q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})]\end{aligned} \tag{19}$$

denoted as $L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta)$.

When using the original joint latent variables and parameters space, we introduce $q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{\gamma})$ to approximate $p(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|X, \alpha_1, \alpha_2, \beta)$, where we assume a factorization different from Equation (5):

$$q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{\gamma}) = q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2) \prod_{u=1}^{N_1} \prod_{v=1}^{N_2} q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv}) \tag{20}$$

where we model the conditional distribution of parameters $\boldsymbol{\pi}_1$, $\boldsymbol{\pi}_2$, and $\boldsymbol{\theta}$ given latent variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ without any assumptions on their form. By doing so, we drop the assumption made in Equation (5) that the parameters and the latent variables are independent. Furthermore, from Equations (18) and (20), we can see that we make the same assumption on $\boldsymbol{z}_1$, $\boldsymbol{z}_2$, that is the assignment of $z_{1_{uv}}$ and $z_{2_{uv}}$ to the current entry $[u, v]$ is independent w.r.t the assignments of the other entries.

The lower bound of the log-likelihood is:

$$\begin{aligned}
&\log p(X|\alpha_1, \alpha_2, \beta) \geq \\
&E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{\gamma})}[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\alpha_1, \alpha_2, \beta)] - \\
&E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{\gamma})}[\log q(\boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{\gamma})] = \\
&E_{q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2) q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\alpha_1, \alpha_2, \beta)] - \\
&E_{q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2) q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[\log q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2) q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})] = \\
&E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[E_{q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2)}[\log p(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|X, \boldsymbol{z}_1, \boldsymbol{z}_2) + \log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)]] - \\
&E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[E_{q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2)}[\log q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2)]] - E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[\log q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})]
\end{aligned} \tag{21}$$

Since we do not assume any specific form for $q(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|\boldsymbol{z}_1, \boldsymbol{z}_2)$, the lower bound will reach at the true posterior $p(\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\theta}|X, \boldsymbol{z}_1, \boldsymbol{z}_2)$. Therefore, the lower bound can be rewritten as:

$$\begin{aligned}
&\log p(X|\alpha_1, \alpha_2, \beta) \geq \\
&\quad E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)] - E_{q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})}[\log q(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})]
\end{aligned} \tag{22}$$

which is the same as $L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta)$. Thus, both approaches derive the same lower bound of the log-likelihood.

Since the collapsed variational Bayesian algorithm makes a strictly weaker assumption on the variational posterior than the standard variational Bayesian algorithm, the collapsed approach can find a tighter lower bound, i.e. $L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta) \leq L(\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \alpha_1, \alpha_2, \boldsymbol{\theta})$.

Maximizing Equation (22) w.r.t γ_{uvij} and setting it to zero, we obtain:

$$\gamma_{uvij} = q(z_{1_{uv}} = i, z_{1_{uv}} = j|\boldsymbol{\gamma}_{uv}) = \frac{\exp(E_{q(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv})}[\log p(X, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}, z_{1_{uv}} = i, z_{1_{uv}} = j|\alpha_1, \alpha_2, \beta)])}{\sum_{i'=1}^{K_1} \sum_{j'=1}^{K_2} \exp(E_{q(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv})}[\log p(X, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}, z_{1_{uv}} = i', z_{1_{uv}} = j'|\alpha_1, \alpha_2, \beta)])} \tag{23}$$

According to Equation (**??**), and setting

$$f(u,v,i,j) = \exp(E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[\log(\alpha_1 + n_{ui}^{\neg uv}) + \log(\alpha_2 + n_{vj}^{\neg uv}) + \log(\beta + n_{ijx_{uv}}^{\neg uv}) - \log(N\beta + n_{ij}^{\neg uv})])$$

we have:

$$\gamma_{uvij} = q(z_{1_{uv}} = i, z_{1_{uv}} = j|\boldsymbol{\gamma}_{uv}) = \frac{f(u,v,i,j)}{\sum_{i'=1}^{K_1}\sum_{j'=1}^{K_2} f(u,v,i',j')} \tag{24}$$

The derivation of Equations (23) and (24) can be found in the Appendix.

Following [7], we also apply a Gaussian approximation to Equation (24). Here we just illustrate how to calculate $E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[\log(\alpha_1 + n_{ui}^{\neg uv})]$. The calculation of the other three expectations is similar. Suppose $n_u \gg 0$, and note that $n_{ui}^{\neg uv} = \sum_{v'=1,v'\neq v}^{N_1}\sum_{j'=1,j'\neq j}^{K_2}\mathbf{1}(z_{1_{uv'}} = i, z_{1_{uv'}} = j')$ is a sum of a large number of independent Bernoulli variables $\mathbf{1}(z_{1_{uv'}} = i, z_{1_{uv'}} = j')$, each with mean parameter $\gamma_{uv'ij'}$; thus, it can be accurately approximated by a Gaussian. The mean and variance are given by the sum of the means and the variances of the individual Bernoulli variables:

$$E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}] = \sum_{v'=1,v'\neq v}^{N_1}\sum_{j'=1,j'\neq j}^{K_2}\gamma_{uv'ij'} \tag{25}$$

$$Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}] = \sum_{v'=1,v'\neq v}^{N_1}\sum_{j'=1,j'\neq j}^{K_2}\gamma_{uv'ij'}(1-\gamma_{uv'ij'}) \tag{26}$$

We further approximate $\log(\alpha_1 + n_{ui}^{\neg uv})$ using a second-order Taylor expansion, and evaluate its expectation under the Gaussian approximation:

$$E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[\log(\alpha_1 + n_{ui}^{\neg uv})] \approx \tag{27}$$

$$\log(\alpha_1 + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}]) - \frac{Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}]}{2(\alpha_1 + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}])^2} \tag{28}$$

As discussed in [7], the Gaussian approximation will be accurate. Finally, plugging Equation (27) into (24), we have:

$$\begin{aligned}\gamma_{uvij} \propto & \\ & \left(\alpha_1 + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}]\right)\left(\alpha_2 + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{vj}^{\neg uv}]\right) \\ & \left(\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ijx_{uv}}^{\neg uv}]\right)\left(N\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ij}^{\neg uv}]\right)^{-1} \\ & \exp\Bigg(-\frac{Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}]}{2(\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ui}^{\neg uv}])^2} - \frac{Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{vj}^{\neg uv}]}{2(\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{vj}^{\neg uv}])^2} - \\ & \frac{Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ijx_{uv}}^{\neg uv}]}{2(\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ijx_{uv}}^{\neg uv}])^2} + \frac{Var_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ij}^{\neg uv}]}{2(N\beta + E_{q(\boldsymbol{z}_1^{\neg uv},\boldsymbol{z}_2^{\neg uv})}[n_{ij}^{\neg uv}])^2}\Bigg)\end{aligned} \tag{29}$$

An EM-style iterative algorithm can be applied to estimate the γ_{uvij}'s by defining Equation (29) as the recursion equation, we can compute every γ_{uvij} for $u \in$

$1, \cdots, N_1$, $v \in 1, \cdots, N_2$, $i \in 1, \cdots, K_1$, $j \in 1, \cdots, K_2$, until the change of γ_{uvij} between two consecutive iterations is less than a certain threshold, which we consider as converged.

4 Experiments

4.1 Datasets

Two real datasets are used in our experiments: (a) MovieLens[5]: MovieLens is a movie recommendation dataset created by the Grouplens Research Project. It contains 100,000 ratings in a sparse data matrix for 1682 movies rated by 943 users. The ratings are ranged from 1 to 5, with 5 being the highest score. We use 5-fold cross-validation for training and testing. (b) Jester[6]: Jester is a joke rating dataset. The original dataset contains 4.1 million continuous ratings of 100 jokes from 73,421 users. The ratings are ranged from -10 to 10, with 10 being the highest. Following [1], we pick 1000 users who rate all 100 jokes and use this dense data matrix in our experiment, and binarize the dataset such that the non-negative entries become 1 and the negative entries become 0. We held out 1/4 data to do prediction.

4.2 Methodology

We train the LDCC model using the three methods discussed in Section 3, and make prediction on the test data using the learned model parameters. For prediction, we report the perplexity [1], which is defined as:

$$perp(X) = \exp\left(\frac{-\log p(X)}{N}\right)$$

where N is the number of non-missing entries in X. Perplexity monotonically decreases as the log-likelihood increases. Thus, a lower perplexity value is an indication of a better model. In fact, a higher log-likelihood on the training set means that the model fits the data better, and a higher log-likelihood on the test set implies that the model can explain the data better.

The variational Bayesian algorithm can find local optima of α_1, α_2, and $\boldsymbol{\theta}$ for the training data, given a random initialization of these parameters. If the change in log-likelihood between two consecutive iterations is less than 1.0e-6, we stop the process. For collapsed Gibbs sampling and collapsed variational Bayesian algorithms, we use uniform priors to initialize the model parameters α_1, α_2, and β. We set to 5000 the maximum number of iterations for Gibbs sampling, the first 2000 as burn-in, and 500 sample lag. Again, if the maximum change between the model parameters $\boldsymbol{\gamma}$ of two consecutive iterations is less than 1.0e-6, we assume that the algorithm has converged, and stop the process.

[5] http://www.grouplens.org/node/73
[6] http://goldberg.berkeley.edu/jester-data/

Table 1. Perplexity Values

	Gibbs	CVB	VB
MovieLens	3.247	4.553	5.849
Binarized Jester	2.954	3.216	4.023

4.3 Experimental Results

In this section, we present two experimental results: perplexity comparison among the three methods, and the likelihood v.s. number of iterations comparison among the three methods.

Following [1], for the MovieLens dataset, we set $K_1 = 20$ and $K_2 = 19$, which are the numbers of user-clusters and movie-clusters; for the Jester dataset, we set $K_1 = 20$ and $K_2 = 5$, which are the numbers of user-clusters and joke-clusters; the matrices of both datasets roughly have 100,000 entries. Table 1 shows the perplexity values of the three methods on the test data. For the MovieLens dataset, we report the average perplexity of five-fold cross-validation for all the three methods. Doing prediction for the MovieLens dataset is harder than for the binarized Jester dataset. In fact, the binarized Jester data have only two rating states, while the MovieLens has 5. For this reason the perplexity values for the MovieLens are smaller than that for the binarized Jester data. From the table, we can see that collapsed Gibbs sampling achieves the best perplexity on both datasets, followed by collapsed variational Bayesian (CVB). The worst performer is the standard variational Bayesian (VB) approach. These results corroborate our theoretical analysis: collapsed Gibbs sampling and collapsed variational Bayesian can learn more accurate likelihood functions than the standard variational Bayesian algorithm, thus leading to higher predicting performance.

Figure 3 shows the log-likelihood as a function of the number of iterations for the three methods on the binarized Jester dataset. As expected, the collapsed Gibbs sampling algorithm provides higher log-likelihood values, but needs a larger number of iterations, 5000 in our case. Collapsed variational Bayesian provides better log-likelihood values than the standard variational Bayesian, but worse than collapsed Gibbs sampling. Collapsed and standard variational Bayesian algorithms have similar numbers of iterations at convergence (100).

Although collapsed Gibbs sampling is an unbiased estimator and can find the true likelihood function, it takes a long time to achieve the stationary distribution. Standard variational Bayesian suffers from the strong assumption of independence between model parameters and latent variables. As a consequence it finds a loose lower bound of the true likelihood function. Collapsed variational Bayesian, however, can find a tighter lower bound of the likelihood function than standard variational Bayesian, and at the same time it's much faster than collapsed Gibbs sampling.

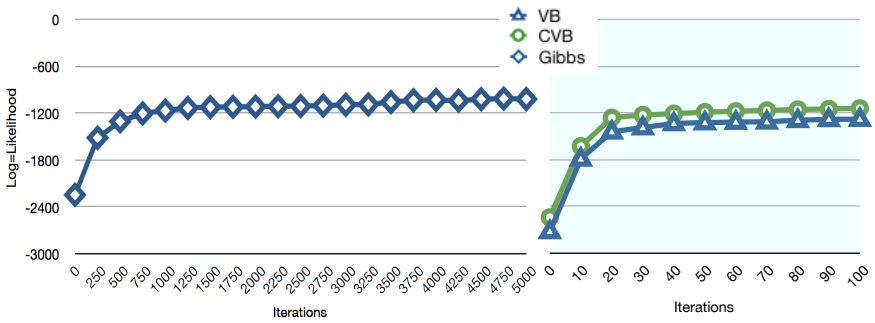

Fig. 3. Log-likelihood v.s. Number of Iterations

5 Conclusions

In this work, we extended the Bayesian co-clustering model, and proposed a collapsed Gibbs sampling and a collapsed variational Bayesian algorithm to perform estimation and inference. The empirical evaluation proved that collapsed Gibbs sampling and collapsed variational Bayesian algorithms can learn more accurate likelihood functions than the standard variational Bayesian algorithm, thus leading to higher predicting performance in general.

References

1. Shan, H. and Banerjee, A.: Bayesian co-clustering. IEEE International Conference on Data Mining (2008)
2. Hartigan, J. A.: Direct Clustering of a Data Matrix. Journal of the American Statistical Association, 337, 123–129 (1972)
3. Dhillon, I. S., Mallela, S., and Modha, D. S.: Information-Theoretic Co-Clustering. ACM SIGKDD international conference on Knowledge discovery and data mining, 89–98 (2003)
4. Shafiei, M. M. and Milios, E. E.: Latent Dirichlet Co-Clustering. International Conference on Data Mining, 542–551 (2006)
5. Blei, D., Ng, A., and Jordan, M.: Latent Dirichlet Allocation. Journal of Machine Learning Research, 3, 993–1022 (2003)
6. Beal, M. J.: Variational Algorithms for Approximate Bayesian Inference. PhD thesis, Gatsby Computational Neuroscience Unit,University College London (2003)
7. Teh, Y. W., Newman, D., and Welling, M.: A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation. Advances in Neural Information Processing Systems, vol. 19 (2007)
8. 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)
9. Griffiths, T. L. and Steyvers, M.: Finding Scientific Topics. National Academy of Science, 101, 5228-5235 (2004)
10. Teh, Y. W., Kurihara, K., and Welling, M.: Collapsed Variational Inference for HDP. Advances in Neural Information Processing Systems, vol. 20 (2008)

Appendix

BCC Model

For the BCC model, the marginal probability of an entry x in the data matrix X is:

$$p(x|\alpha_1,\alpha_2,\boldsymbol{\theta}) = \int_{\pi_1}\int_{\pi_2} p(\pi_1|\alpha_1)p(\pi_2|\alpha_2)\sum_{z_1}\sum_{z_2} p(z_1|\pi_1)p(z_2|\pi_2)p(x|\theta_{z_1 z_2})\, d\pi_1 d\pi_2$$

The overall joint distribution over all observable and latent variables is given by:

$$p(X,\boldsymbol{\pi}_1,\boldsymbol{\pi}_2,\boldsymbol{z}_1,\boldsymbol{z}_2|\alpha_1,\alpha_2,\boldsymbol{\theta}) = \prod_u p(\pi_{1_u}|\alpha_1)\prod_v p(\pi_{2_v}|\alpha_2)$$
$$\cdot\prod_{u,v} p(z_{1_{uv}}|\pi_{1_u})p(z_{2_{uv}}|\pi_{2_v})p(x_{uv}|\theta_{z_{1_{uv}},z_{2_{uv}}})^{\delta_{uv}}$$

The probability of observing the entire matrix X is:

$$p(X|\alpha_1,\alpha_2,\boldsymbol{\theta}) = \int_{\boldsymbol{\pi}_1}\int_{\boldsymbol{\pi}_2}\int_{\boldsymbol{\theta}}\left(\prod_u p(\pi_{1_u}|\alpha_1)\right)\left(\prod_v p(\pi_{2_v}|\alpha_2)\right)$$
$$\cdot\left(\prod_{u,v}\sum_{z_{1_{uv}}}\sum_{z_{2_{uv}}} p(z_{1_{uv}}|\pi_{1_u})p(z_{2_{uv}}|\pi_{2_v})p(x_{uv}|\theta_{z_{1_{uv}},z_{2_{uv}}})^{\delta_{uv}}\right) d\boldsymbol{\pi}_1 d\boldsymbol{\pi}_2$$

Derivation of Equation (13)

$$p(X, z_{1_{uv}}=i, z_{2_{uv}}=j, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\alpha_1,\alpha_2,\beta) = \tag{30}$$
$$\prod_{u'=1}^{N_1}\left(\frac{\Gamma(K_1\alpha_1)}{\Gamma(K_1\alpha_1+n_{u'}^{\neg uv}+\delta_{u'=u})}\prod_{i'=1}^{K_1}\frac{\Gamma(\alpha_1+n_{u'i'}^{\neg uv}+\delta_{u'=u}^{i'=i})}{\Gamma(\alpha_1)}\right)$$
$$\prod_{v'=1}^{N_2}\left(\frac{\Gamma(K_2\alpha_2)}{\Gamma(K_2\alpha_2+n_{v'}^{\neg uv}+\delta_{v'=v})}\prod_{j'=1}^{K_2}\frac{\Gamma(\alpha_2+n_{v'j'}^{\neg uv}+\delta_{v'=v}^{j'=j})}{\Gamma(\alpha_2)}\right)$$
$$\prod_{i'=1}^{K_1}\prod_{j'=1}^{K_2}\left(\frac{\Gamma(N\beta)}{\Gamma(N\beta+n_{i'j'}^{\neg uv}+\delta_{i'=i}^{j'=j})}\prod_{x'=1}^{N}\frac{\Gamma(\beta+n_{i'j'x'}^{\neg uv}+\delta_{i'=i,j'=j}^{x'=x_{uv}})}{\Gamma(\beta)}\right)$$

$$p(X, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\alpha_1,\alpha_2,\beta) = \tag{31}$$
$$\prod_{u'=1}^{N_1}\left(\frac{\Gamma(K_1\alpha_1)}{\Gamma(K_1\alpha_1+n_{u'}^{\neg uv}+\delta_{u'=u})}\prod_{i'=1}^{K_1}\frac{\Gamma(\alpha_1+n_{u'i'}^{\neg uv}+\delta_{u'=u}^{i'=i})}{\Gamma(\alpha_1)}\right)$$
$$\prod_{v'=1}^{N_2}\left(\frac{\Gamma(K_2\alpha_2)}{\Gamma(K_2\alpha_2+n_{v'}^{\neg uv}+\delta_{v'=v})}\prod_{j'=1}^{K_2}\frac{\Gamma(\alpha_2+n_{v'j'}^{\neg uv}+\delta_{v'=v}^{j'=j})}{\Gamma(\alpha_2)}\right)$$
$$\prod_{i'=1}^{K_1}\prod_{j'=1}^{K_2}\left(\frac{\Gamma(N\beta)}{\Gamma(N\beta+n_{i'j'}^{\neg uv}+\delta_{i'=i}^{j'=j})}\prod_{x'=1}^{N}\frac{\Gamma(\beta+n_{i'j'x'}^{\neg uv}+\delta_{i'=i,j'=j}^{x'=x_{uv}})}{\Gamma(\beta)}\right)$$

where $\delta_{(\cdot)}^{(\cdot)}$ is an indicator function: if all input equations are true, it takes value 1, else 0. Note that if $u' \neq u$, $n_{u'}^{\neg uv} = n_{u'}$. The same holds for the other counting variables. Thus, Equation (13) can be derived by taking the ratio of Equations (30) and (31).

Derivation of Equations (23) and (24)

$$L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta) =$$
$$\int \prod_{u=1}^{N_1} \prod_{v=1}^{N_2} q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv}) \log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)\, dq(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma}) -$$
$$\int \prod_{u=1}^{N_1} \prod_{v=1}^{N_2} q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv}) \log \prod_{u=1}^{N_1} \prod_{v=1}^{N_2} q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv})\, dq(\boldsymbol{z}_1, \boldsymbol{z}_2|\boldsymbol{\gamma})$$

Taking the derivative of $L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta)$ w.r.t $q(z_{1_{uv}}, z_{1_{uv}}|\boldsymbol{\gamma}_{uv})$, we get:

$$\frac{\partial L(\boldsymbol{\gamma}, \alpha_1, \alpha_2, \beta)}{\partial q(z_{1_{uv}}, z_{1_{uv}}|\boldsymbol{\gamma}_{uv})} =$$
$$\int \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'}) \log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)\, dq(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma}) -$$
$$\int \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'}) \log \prod_{u'=1}^{N_1} \prod_{v'=1}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'})\, dq(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma}) -$$
$$\prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'}) =$$
$$\int \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'}) \log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)\, dq(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma}) -$$
$$\log q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv}) \int \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'})\, dq(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma}) -$$
$$\int \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'}) \log \prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'})\, dq(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma}) -$$
$$\prod_{u'=1, u' \neq u}^{N_1} \prod_{v'=1, v' \neq v}^{N_2} q(z_{1_{u'v'}}, z_{2_{u'v'}}|\boldsymbol{\gamma}_{u'v'})$$

Setting the derivative to zero, it's clear that:

$$q(z_{1_{uv}}, z_{2_{uv}}|\boldsymbol{\gamma}_{uv}) \propto \exp(E_{q(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\boldsymbol{\gamma})}[\log p(X, \boldsymbol{z}_1, \boldsymbol{z}_2|\alpha_1, \alpha_2, \beta)])$$

from which we derive Equation (23). From Equation (30), we can see that:

$$\log p(X, z_{1_{uv}} = i, z_{2_{uv}} = j, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv}|\alpha_1, \alpha_2, \beta) =$$
$$\sum_{u'=1}^{N_1} \left(\log \Gamma(K_1\alpha_1) - \log \Gamma(K_1\alpha_1 + n_{u'}^{\neg uv} + \delta_{u'=u}) + \sum_{i'=1}^{K_1} (\log \Gamma(\alpha_1 + n_{u'i'}^{\neg uv} + \delta_{u'=u}^{i'=i}) - \log \Gamma(\alpha_1)) \right) +$$
$$\sum_{v'=1}^{N_2} \left(\log \Gamma(K_2\alpha_2) - \log \Gamma(K_2\alpha_2 + n_{v'}^{\neg uv} + \delta_{v'=v}) + \sum_{j'=1}^{K_2} (\log \Gamma(\alpha_2 + n_{v'j'}^{\neg uv} + \delta_{v'=v}^{j'=j}) - \log \Gamma(\alpha_2)) \right) +$$
$$\sum_{i'=1}^{K_1} \sum_{j'=1}^{K_2} \left(\log \Gamma(N\beta) - \log \Gamma(N\beta + n_{i'j'}^{\neg uv} + \delta_{i'=i}^{j'=j}) + \sum_{x'=1}^{N} (\log \Gamma(\beta + n_{i'j'x'}^{\neg uv} + \delta_{i'=i,j'=j}^{x'=x_{uv}}) - \log \Gamma(\beta)) \right)$$

Note that if $u' = u$, $\Gamma(K_1\alpha_1 + n_{u'}^{\neg uv} + \delta_{u'=u}) = (K_1\alpha_1 + n_{u'}^{\neg uv})\Gamma(K_1\alpha_1 + n_{u'}^{\neg uv})$, as for other Gamma functions. Then, we have:

$$\begin{aligned}
&\log p(X, z_{1_{uv}} = i, z_{2_{uv}} = j, \boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv} | \alpha_1, \alpha_2, \beta) = -\log(K_1\alpha_1 + n_u^{\neg uv}) + \log(\alpha_1 + n_{ui}^{\neg uv}) - \\
&\log(K_2\alpha_2 + n_v^{\neg uv}) + \log(\alpha_2 + n_{v'j'}^{\neg uv}) - \log(N\beta + n_{i'j'}^{\neg uv}) + \log(\beta + n_{i'j'x'}^{\neg uv}) + \\
&\sum_{u'=1}^{N_1} \left(\log\Gamma(K_1\alpha_1) - \log\Gamma(K_1\alpha_1 + n_{u'}^{\neg uv}) + \sum_{i'=1}^{K_1} (\log\Gamma(\alpha_1 + n_{u'i'}^{\neg uv}) - \log\Gamma(\alpha_1)) \right) + \\
&\sum_{v'=1}^{N_2} \left(\log\Gamma(K_2\alpha_2) - \log\Gamma(K_2\alpha_2 + n_{v'}^{\neg uv}) + \sum_{j'=1}^{K_2} (\log\Gamma(\alpha_2 + n_{v'j'}^{\neg uv}) - \log\Gamma(\alpha_2)) \right) + \\
&\sum_{i'=1}^{K_1} \sum_{j'=1}^{K_2} \left(\log\Gamma(N\beta) - \log\Gamma(N\beta + n_{i'j'}^{\neg uv}) + \sum_{x'=1}^{N} (\log\Gamma(\beta + n_{i'j'x'}^{\neg uv}) - \log\Gamma(\beta)) \right) \quad (32)
\end{aligned}$$

where for a chosen entry $[u, v]$, no matter what $z_{1_{uv}}$ and $z_{2_{uv}}$ are, $\log(K_1\alpha_1 + n_u^{\neg uv})$, $\log(K_2\alpha_2 + n_v^{\neg uv})$, and the summations in Equation (32) are the same. So it's clear that:

$$\begin{aligned}
&q(z_{1_{uv}} = i, z_{1_{uv}} = j | \boldsymbol{\gamma}_{uv}) \propto \\
&\exp(E_{q(\boldsymbol{z}_1^{\neg uv}, \boldsymbol{z}_2^{\neg uv})}[\log(\alpha_1 + n_{ui}^{\neg uv}) + \log(\alpha_2 + n_{vj}^{\neg uv}) + \log(\beta + n_{ijx_{uv}}^{\neg uv}) - \log(N\beta + n_{ij}^{\neg uv})])
\end{aligned}$$

Thus, we derive Equation (24).