

# A Generalized Maximum Entropy Approach to Bregman Co-clustering and Matrix Approximation

Arindam Banerjee Inderjit Dhillon  
Joydeep Ghosh Srujana Merugu  
University of Texas  
Austin, TX, USA

Dharmendra S. Modha  
IBM Almaden Research Center  
San Jose, CA, USA

## ABSTRACT

Co-clustering is a powerful data mining technique with varied applications such as text clustering, microarray analysis and recommender systems. Recently, an information-theoretic co-clustering approach applicable to empirical joint probability distributions was proposed. In many situations, co-clustering of more general matrices is desired. In this paper, we present a substantially generalized co-clustering framework wherein any Bregman divergence can be used in the objective function, and various conditional expectation based constraints can be considered based on the statistics that need to be preserved. Analysis of the co-clustering problem leads to the minimum Bregman information principle, which generalizes the maximum entropy principle, and yields an elegant meta algorithm that is guaranteed to achieve local optimality. Our methodology yields new algorithms and also encompasses several previously known clustering and co-clustering algorithms based on alternate minimization.

**Categories and Subject Descriptors:** I.2.6 [Artificial Intelligence]: Learning

**General Terms:** Algorithms

**Keywords:** Co-clustering, Matrix Approximation, Bregman divergences

## 1. INTRODUCTION

Co-clustering, or bi-clustering [9, 4], is the problem of simultaneously clustering rows and columns of a data matrix. The problem of co-clustering arises in diverse data mining applications, such as simultaneous clustering of genes and experimental conditions in bioinformatics [4, 5], documents and words in text mining [8], users and movies in recommender systems, etc. In order to design a co-clustering framework, we need to first characterize the “goodness” of a co-clustering. Existing co-clustering techniques [5, 4, 8] achieve this by quantifying the “goodness” of a co-clustering in terms of the approximation error between the original

data matrix and a reconstructed matrix based on co-clustering. Of these techniques, the most efficient and scalable algorithms are those based on alternate minimization schemes [5, 8], but these are restricted to only two distortion measures namely, KL-divergence and the squared Euclidean distance, and a few specific matrix reconstruction schemes. These two limitations restrict the applicability of these techniques to a small range of data matrices.

In this paper, we address the following two questions: (a) *what class of distortion functions admit efficient co-clustering algorithms based on alternate minimization?*, and (b) *what are the different possible matrix reconstruction schemes for these co-clustering algorithms?* We show that alternate minimization based co-clustering algorithms work for a large class of distortion measures called Bregman divergences [3], which include squared Euclidean distance, KL-divergence, Itakura-Saito distance, etc., as special cases. Further, we demonstrate that for a given co-clustering, a large variety of approximation models are possible based on the type of summary statistics that need to be preserved. Analysis of this general co-clustering problem leads to the *minimum Bregman information principle* that simultaneously generalizes the maximum entropy and the least squares principles. Based on this principle, and other related results, we develop an elegant meta-algorithm for the Bregman co-clustering problem with a number of desirable properties. Most previously known parametric clustering and co-clustering algorithms based on alternate minimization follow as special cases of our methodology.

### 1.1 Motivation

We start by reviewing information-theoretic co-clustering [8] and motivating the need for a more general co-clustering framework. Let  $[u]_1^m$  denote an index  $u$  running over  $\{1, \dots, m\}$  and let  $X$  and  $Y$  be discrete random variables that take values in the sets  $\{x_u\}$ ,  $[u]_1^m$ , and  $\{y_v\}$ ,  $[v]_1^n$ , respectively. Suppose we are in the idealized situation where the joint probability distribution  $p(X, Y)$  is known. In practice,  $p$  may be estimated from a contingency table or co-occurrence matrix. Suppose we want to co-cluster, or, simultaneously cluster  $X$  into  $k$  disjoint (row) clusters  $\{\hat{x}_g\}$ ,  $[g]_1^k$ , and  $Y$  into  $l$  disjoint (column) clusters,  $\{\hat{y}_h\}$ ,  $[h]_1^l$ . Let  $\hat{X}$  and  $\hat{Y}$  denote the corresponding clustered random variables that range over these sets. An information theoretic formulation of finding the optimal co-clustering is to solve the problem

$$\min_{\hat{X}, \hat{Y}} I(X; Y) - I(\hat{X}; \hat{Y}), \quad (1.1)$$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

KDD'04, August 22–25, 2004, Seattle, Washington, USA.  
Copyright 2004 ACM 1-58113-888-1/04/0008 ...\$5.00.

where  $I(X; Y)$  is the mutual information between  $X$  and  $Y$  [6]. In [8], it was shown that

$$I(X; Y) - I(\hat{X}, \hat{Y}) = D(p(X, Y) || q(X, Y)), \quad (1.2)$$

where  $q(X, Y)$  is a distribution of the form

$$q(X, Y) = p(\hat{X}, \hat{Y})p(X|\hat{X})p(Y|\hat{Y}), \quad (1.3)$$

and  $D(\cdot|\cdot)$  denotes the Kullback-Leibler(KL) divergence. Thus, the search for the optimal co-clustering may be conducted by searching for the nearest approximation  $q(X, Y)$  that has the form in (1.3). We note that  $q(X, Y)$  depends only on  $(m - k + n - l + kl - 1)$  independent parameters, which is much smaller than the  $(mn - 1)$  parameters that determine a general  $p(X, Y)$ . Hence, we call  $q(X, Y)$  a “low complexity” or low parameter matrix approximation.

The above is the viewpoint presented in [8]. We now present an alternate viewpoint that highlights the key maximum entropy property that makes  $q(X, Y)$  a “low complexity” or low parameter approximation.<sup>1</sup>

**Lemma 1** *Given a fixed co-clustering  $\hat{X}, \hat{Y}$ , consider the set of joint distributions  $p'$  that preserve the following statistics of the input distribution  $p$ :*

$$\begin{aligned} \sum_{x \in \hat{x}} \sum_{y \in \hat{y}} p'(x, y) &= p(\hat{x}, \hat{y}) = \sum_{x \in \hat{x}} \sum_{y \in \hat{y}} p(x, y), \quad \forall \hat{x}, \hat{y}, \\ p'(x) &= p(x), \quad p'(y) = p(y), \quad \forall x, y. \end{aligned}$$

*Among all such distributions  $p'$ , the distribution  $q$  in (1.3) has the maximum Shannon entropy, i.e.,  $H(q) \geq H(p')$ .*

Thus, among all distributions that preserve marginals and co-cluster statistics, the maximum entropy distribution has the form in (1.3). Thus, by (1.2) and Lemma 1, the co-clustering problem (1.1) is equivalent to the problem of finding the nearest (in KL-divergence) maximum entropy distribution that preserves the marginals, and the co-cluster statistics of the original data matrix.

The above formulation is applicable when the data matrix directly corresponds to an empirical joint distribution. However, there are important situations when the data matrix is more general, for example, the matrix may contain negative entries and/or a distortion measure other than KL-divergence, such as the squared Euclidean distance, or the Itakura-Saito distance might be more appropriate.

This paper addresses the general situation by extending information-theoretic co-clustering along three directions. First, “nearness” is now measured by any Bregman divergence. Second, we allow specification of a larger class of constraints that preserve various statistics of the data. Lastly, to accomplish the above, we generalize the maximum entropy approach: we guide our co-clustering generalization by appealing to the *minimum Bregman information principle* that we shall introduce shortly. The optimal co-clustering is guided by the search for the nearest (in Bregman divergence) matrix approximation that has minimum Bregman information while satisfying the desired constraints.

## 2. FORMULATION AND ANALYSIS

In this section, we formulate the Bregman co-clustering problem in terms of the Bregman divergence between a given matrix and an approximation based on the co-clustering.

<sup>1</sup>Proofs omitted due to lack of space, see [1] for details.

We start by defining Bregman divergences [3, 2]. Let  $\phi$  be a real-valued strictly convex function defined on the convex set  $S = \text{dom}(\phi) \subseteq \mathbb{R}$ , the domain of  $\phi$ , such that  $\phi$  is differentiable on  $\text{int}(S)$ , the interior of  $S$ . The **Bregman divergence**  $d_\phi : S \times \text{int}(S) \mapsto [0, \infty)$  is defined as  $d_\phi(z_1, z_2) = \phi(z_1) - \phi(z_2) - \langle z_1 - z_2, \nabla \phi(z_2) \rangle$ , where  $\nabla \phi$  is the gradient of  $\phi$ .

**Example 1.A (I-Divergence)** Given  $z \in \mathbb{R}_+$ , let  $\phi(z) = z \log z$ . For  $z_1, z_2 \in \mathbb{R}_+$ ,  $d_\phi(z_1, z_2) = z_1 \log(z_1/z_2) - (z_1 - z_2)$ .

**Example 2.A (Squared Euclidean Distance)** Given  $z \in \mathbb{R}$ , let  $\phi(z) = z^2$ . For  $z_1, z_2 \in \mathbb{R}$ ,  $d_\phi(z_1, z_2) = (z_1 - z_2)^2$ .

Based on Bregman divergences, it is possible to define a useful concept called **Bregman information**, which captures the “spread” or the “information” in a random variable. More precisely, for any random variable  $Z$  taking values in  $S = \text{dom}(\phi)$ , its **Bregman information** is defined as the expected Bregman divergence to the expectation, i.e.,  $I_\phi(Z) = E[d_\phi(Z, E[Z])]$ .

**Example 1.B (I-Divergence)** Given a real non-negative random variable  $Z$ , the Bregman information corresponding to I-divergence is given by  $I_\phi(Z) = E[Z \log \frac{Z}{E[Z]}]$ . When  $Z$  is uniformly distributed over the set  $\{p(x_u, y_v)\} [u]_1^m [v]_1^n$ , i.e.,  $\Pr(Z = p(x_u, y_v)) = \frac{1}{mn}, \forall u, v$ , then  $E[Z] = \frac{1}{mn}$  and the Bregman information of  $Z$  is proportional to  $D(p||p_0) = -H(p) + \text{constant}$ , where  $D(\cdot|\cdot)$  is KL-divergence,  $p_0$  is the uniform distribution and  $H(\cdot)$  is the Shannon entropy.

**Example 2.B (Squared Euclidean Distance)** Given any real random variable  $Z$ , the Bregman information corresponding to squared Euclidean distance is given by  $I_\phi(Z) = E[(Z - E[Z])^2]$ , and when  $Z$  is uniformly distributed over the elements of a matrix, it is proportional to the squared Frobenius norm of the matrix + constant.

We focus on the problem of co-clustering a given  $m \times n$  data matrix  $Z$  whose entries take values in a convex set  $S = \text{dom}(\phi)$ , i.e.,  $Z \in S^{m \times n}$ . With a slight abuse of notation, we can consider the matrix  $Z$  as a random variable in  $S$  that is a known deterministic function of two underlying random variables  $U$  and  $V$ , which take values over the set of row indices  $\{1, \dots, m\}$  and the set of column indices  $\{1, \dots, n\}$  respectively. Further, let  $\nu = \{\nu_{uv} : [u]_1^m, [v]_1^n\}$  denote the joint probability measure of the pair  $(U, V)$ , which is either pre-specified or set to be the uniform distribution. Throughout the paper, all expectations are with respect to  $\nu$ .

**Example 1.C (I-Divergence)** Let  $(X, Y) \sim p(X, Y)$  be jointly distributed random variables with  $X, Y$  taking values in  $\{x_u\}, [u]_1^m$  and  $\{y_v\}, [v]_1^n$  respectively. Then,  $p(X, Y)$  can be written in the form of the matrix  $Z = [z_{uv}], [u]_1^m, [v]_1^n$ , where  $z_{uv} = p(x_u, y_v)$  is a deterministic function of  $u$  and  $v$ . This example with a uniform measure  $\nu$  corresponds to the setting described in Section 1.1 (originally in [8])<sup>2</sup>.

**Example 2.C (Squared Euclidean Distance)** Let  $Z \in \mathbb{R}^{m \times n}$  denote a data matrix whose elements may assume

<sup>2</sup>Note that in [8] KL-divergence was used, which is a special case of I-divergence applicable to probability distributions.

any real values. This example with a uniform measure  $\nu$  corresponds to the setting described in [5, 4].

A  $k \times l$  co-clustering of a given data matrix  $Z$  is a pair of maps:  $\rho : \{1, \dots, m\} \mapsto \{1, \dots, k\}$ ,  $\gamma : \{1, \dots, n\} \mapsto \{1, \dots, l\}$ . A natural way to quantify the “goodness” of a co-clustering is in terms of the accuracy of an approximation  $\hat{Z} = [\hat{z}_{uv}]$  obtained from the co-clustering, i.e., the quality of the co-clustering can be defined as

$$E[d_\phi(Z, \hat{Z})] = \sum_{u=1}^m \sum_{v=1}^n \nu_{uv} d_\phi(z_{uv}, \hat{z}_{uv}). \quad (2.4)$$

where  $\hat{Z}$  is uniquely determined by the co-clustering  $(\rho, \gamma)$ .

Given a co-clustering  $(\rho, \gamma)$ , there can be a number of different matrix approximations  $\hat{Z}$  based on the information we choose to retain. Let  $\hat{U}$  and  $\hat{V}$  be random variables for row and column clusterings respectively, taking values in  $\{1, \dots, k\}$  and  $\{1, \dots, l\}$  such that  $\hat{U} = \rho(U)$  and  $\hat{V} = \gamma(V)$ . Then, the co-clustering  $(\rho, \gamma)$  involves four underlying random variables  $U, V, \hat{U}$ , and  $\hat{V}$  corresponding to the various partitionings of the matrix  $Z$ . We can now obtain different matrix approximations based solely on the statistics of  $Z$  corresponding to the non-trivial combinations of  $\{U, V, \hat{U}, \hat{V}\}$  given by

$$\Gamma = \{\{U, \hat{V}\}, \{\hat{U}, V\}, \{\hat{U}, \hat{V}\}, \{U\}, \{V\}, \{\hat{U}\}, \{\hat{V}\}\}.$$

If  $\pi(\Gamma)$  denotes the power set of  $\Gamma$ , then every element of  $\pi(\Gamma)$  is a set of constraints that leads to a (possibly) different matrix approximation. Hence,  $\pi(\Gamma)$  can be considered as the *class of matrix approximation schemes* based on a given co-clustering  $(\rho, \gamma)$ . For the sake of illustration, we consider four examples corresponding to the non-trivial constraint sets in  $\pi(\Gamma)$  that are symmetric in  $U, \hat{U}$  and  $V, \hat{V}$ :

$$\begin{aligned} \mathcal{C}_1 &= \{\{\hat{U}\}, \{\hat{V}\}\}, & \mathcal{C}_2 &= \{\{\hat{U}, \hat{V}\}\}, \\ \mathcal{C}_3 &= \{\{\hat{U}, \hat{V}\}, \{U\}, \{V\}\}, & \mathcal{C}_4 &= \{\{U, \hat{V}\}, \{\hat{U}, V\}\}. \end{aligned}$$

Now, for a specified constraint set  $\mathcal{C} \in \pi(\Gamma)$  and a co-clustering  $(\rho, \gamma)$ , the set of possible approximations  $\mathcal{M}_A(\rho, \gamma, \mathcal{C})$  consists of all  $Z' \in S^{m \times n}$  that *depend only* on the relevant statistics of  $Z$ , i.e.,  $E[Z|\mathcal{C}]$ ,  $\mathcal{C} \in \mathcal{C}$ , or more precisely, satisfy the following *conditional independence* condition:  $Z \rightarrow \{E[Z|\mathcal{C}] : \mathcal{C} \in \mathcal{C}\} \rightarrow Z'$ . Hence, the approximations  $Z'$  can be a function of only  $\{E[Z|\mathcal{C}] : \mathcal{C} \in \mathcal{C}\}$ .

We can now define the “best” approximation  $\hat{Z}$  corresponding to a given co-clustering  $(\rho, \gamma)$  and the constraint set  $\mathcal{C}$  as the one in the class  $\mathcal{M}_A(\rho, \gamma, \mathcal{C})$  that minimizes the approximation error, i.e.,

$$\hat{Z} = \operatorname{argmin}_{Z' \in \mathcal{M}_A(\rho, \gamma, \mathcal{C})} E[d_\phi(Z, Z')]. \quad (2.5)$$

## 2.1 Minimum Bregman Information

Interestingly, it can be shown [1] that the “best” matrix approximation  $\hat{Z}$  turns out to be the minimum Bregman information matrix among the class of random variables  $\mathcal{M}_B(\rho, \gamma, \mathcal{C})$  consisting of all  $Z' \in S^{m \times n}$  that *preserve* the relevant statistics of  $Z$  or more precisely, satisfy the *linear constraints*:  $\forall \mathcal{C} \in \mathcal{C}, E[Z|\mathcal{C}] = E[Z'|\mathcal{C}]$ . Hence, the best approximation  $\hat{Z}$  of the original matrix  $Z$  for a specified co-clustering  $(\rho, \gamma)$  and constraint set  $\mathcal{C}$  is given by

$$\hat{Z} = \operatorname{argmin}_{Z' \in \mathcal{M}_A(\rho, \gamma, \mathcal{C})} E[d_\phi(Z, Z')] = \operatorname{argmin}_{Z' \in \mathcal{M}_B(\rho, \gamma, \mathcal{C})} I_\phi(Z'). \quad (2.6)$$

This leads to a new **minimum Bregman information principle**: the best estimate given certain statistics is one that has the minimum Bregman information subject to the linear constraints for preserving those statistics. It is easy to see that the widely used *maximum entropy principle* [6] is a special case of the proposed principle for I-divergence since the entropy of a joint distribution is negatively related to the Bregman information (Example 1.B). In fact, even the least squares principle [7] can be obtained as a special case when the Bregman divergence is squared Euclidean distance.

The following theorem characterizes the solution to the minimum Bregman information problem (2.6). For a proof, see [1].

**Theorem 1** *For a Bregman divergence  $d_\phi$ , any random variable  $Z \in S^{m \times n}$ , a specified co-clustering  $(\rho, \gamma)$  and a specified constraint set  $\mathcal{C}$ , the solution  $\hat{Z}$  to (2.6) is given by*

$$\nabla \phi(\hat{Z}) = - \sum_r \Lambda_r^*, \quad (2.7)$$

where  $\Lambda^* \equiv \{\Lambda_r^*\}$  are the optimal Lagrange multipliers corresponding to the set of linear constraints:  $E[Z'|C_r] = E[Z|C_r]$ ,  $\forall C_r \in \mathcal{C}$ . Furthermore,  $\hat{Z}$  always exists, and is unique.

## 2.2 Bregman Co-clustering Problem

We can now quantify the goodness of a co-clustering in terms of the expected Bregman divergence between the original matrix  $Z$  and the minimum Bregman information solution  $\hat{Z}$ . Thus, the Bregman co-clustering problem can be concretely defined as follows:

**Definition 1** Given  $k, l$ , a Bregman divergence  $d_\phi$ , a data matrix  $Z \in S^{m \times n}$ , a set of constraints  $\mathcal{C} \in \pi(\Gamma)$ , and an underlying probability measure  $\nu$ , we wish to find a co-clustering  $(\rho^*, \gamma^*)$  that minimizes:

$$(\rho^*, \gamma^*) = \operatorname{argmin}_{(\rho, \gamma)} E[d_\phi(Z, \hat{Z})] \quad (2.8)$$

where  $\hat{Z} = \operatorname{argmin}_{Z' \in \mathcal{M}_B(\rho, \gamma, \mathcal{C})} I_\phi(Z')$ .

The problem is NP-complete by a reduction from the **kmeans** problem. Hence, it is difficult to obtain a globally optimal solution. However, in Section 3, we analyze the problem in detail, and prove that it is possible to come up with an iterative update scheme that provides a locally optimal solution.

**Example 1.D (I-Divergence)** The Bregman co-clustering objective function is given by  $E[Z \log(\frac{Z}{\hat{Z}}) - Z + \hat{Z}] = E[Z \log(\frac{Z}{\hat{Z}})]$  where  $\hat{Z}$  is the minimum Bregman information solution (Table 1). Note that for the constraint set  $\mathcal{C}_3$  and  $Z$  based on a joint distribution  $p(X, Y)$ , the objective function reduces to  $D(p||q)$  where  $q$  is of the form (1.3) that was used in [8]

**Example 2.D (Squared Euclidean Distance)** The Bregman co-clustering objective function is  $E[(Z - \hat{Z})^2]$  where  $\hat{Z}$  is the minimum Bregman information solution (Table 2). Note that for the constraint set  $\mathcal{C}_4$ , this reduces to  $E[(Z - E[Z|U, \hat{V}] - E[Z|\hat{U}, V] + E[Z|\hat{U}, \hat{V}])^2]$ , which is identical to the objective function in [5, 4].

### 3. A META ALGORITHM

In this section, we shall develop an alternating minimization scheme for the general Bregman co-clustering problem (2.8). Our scheme shall serve as a *meta algorithm* from which a number of special cases (both new and previously known) can be derived. Throughout this section, let us suppose that the underlying measure  $\nu$ , the Bregman divergence  $d_\phi$ , the data matrix  $Z \in S^{m \times n}$ , number of row clusters  $k$ , number of column clusters  $l$ , and the constraint set  $\mathcal{C}$  are specified. We first outline the essence of our scheme.

**Step 1:** Start with an arbitrary row and column clustering, say,  $(\rho^0, \gamma^0)$ . Set  $t = 0$ . With respect to this clustering, compute the matrix approximation  $\hat{Z}^t$  by solving the minimum Bregman information problem (2.6).

**Step 2:** Repeat one of the two steps below till convergence:

**Step 2A:** Hold the column clustering  $\gamma^t$  fixed, and find a new row co-clustering, say,  $\rho^{t+1}$ . Set  $\gamma^{t+1} = \gamma^t$ . With respect to the co-clustering  $(\rho^{t+1}, \gamma^{t+1})$ , compute the matrix approximation  $\hat{Z}^{t+1}$  by solving the minimum Bregman information problem (2.6). Set  $t = t + 1$ .

**Step 2B:** Hold the row clustering  $\rho^t$  fixed, and find a new column co-clustering, say,  $\gamma^{t+1}$ . Set  $\rho^{t+1} = \rho^t$ . With respect to the co-clustering  $(\rho^{t+1}, \gamma^{t+1})$ , compute the matrix approximation  $\hat{Z}^{t+1}$  by solving the minimum Bregman information problem (2.6). Set  $t = t + 1$ .

Note that at any time in Step 2, the algorithm may choose to perform either Step 2A or 2B.

#### 3.1 Updating Row and Column Clusters

From the above outline, it is clear that the key steps in our algorithm involve finding a solution of the minimum Bregman information problem (2.6) and appropriately updating the row and column clusters. First, we focus on the latter task. Consider matrix approximations based on the functional form for the minimum Bregman solution  $\hat{Z}$  given in (2.7). For a given  $(\rho, \gamma, \mathcal{C})$ , there exist a unique set of optimal Lagrange multipliers  $\Lambda^*$  so that (2.7) uniquely specifies the minimum Bregman information solution  $\hat{Z}$ . In general, (2.7) provides a unique approximation, say  $\tilde{Z}$ , for any set of Lagrange multipliers  $\Lambda$  (not necessarily optimal) and  $(\rho, \gamma, \mathcal{C})$  since  $\nabla\phi(\cdot)$  is a monotonic function [3, 2]. To underscore the dependence of  $\tilde{Z}$  on the Lagrange multipliers, we shall use the notation  $\tilde{Z} = \zeta(\rho, \gamma, \Lambda) = (\nabla\phi)^{-1}(-\sum_{r=1}^s \Lambda_r)$ . The basic idea in considering approximations of the form  $\zeta(\rho, \gamma, \Lambda)$  is that alternately optimizing the co-clustering and the Lagrange multipliers leads to an efficient update scheme that does not require solving the minimum Bregman information problem anew for each possible co-clustering.

Further, the matrix approximations of the form  $\zeta(\rho, \gamma, \Lambda)$  have a nice separability property [1] that enables us to decompose the matrix approximation error in terms of either the rows and columns:

$$\begin{aligned} E[d_\phi(Z, \tilde{Z})] &= E_U[E_{V|U}[\xi(U, \rho(U), V, \gamma(V))]] \\ &= E_V[E_{U|V}[\xi(U, \rho(U), V, \gamma(V))]], \end{aligned}$$

where  $\tilde{Z} = \zeta(\rho, \gamma, \Lambda)$  and  $\xi(U, \rho(U), V, \gamma(V)) = d_\phi(Z, \tilde{Z})$ . Using the above separability property, we can efficiently ob-

Table 1: Minimum Bregman information solution for I-Divergence leads to multiplicative models.

Constraints $\mathcal{C}$	Approximation $\hat{Z}$
$\mathcal{C}_1$	$\frac{E[Z \tilde{U}] \times E[Z \tilde{V}]}{E[Z]}$
$\mathcal{C}_2$	$E[Z \tilde{U}, \tilde{V}]$
$\mathcal{C}_3$	$\frac{E[Z U] \times E[Z V] \times E[Z \tilde{U}, \tilde{V}]}{E[Z \tilde{U}] \times E[Z \tilde{V}]}$
$\mathcal{C}_4$	$\frac{E[Z U, \tilde{V}] \times E[Z \tilde{U}, V]}{E[Z \tilde{U}, \tilde{V}]}$

Table 2: Minimum Bregman information solution for squared Euclidean distance leads to additive models.

Constraints $\mathcal{C}$	Approximation $\hat{Z}$
$\mathcal{C}_1$	$E[Z \tilde{U}] + E[Z \tilde{V}] - E[Z]$
$\mathcal{C}_2$	$E[Z \tilde{U}, \tilde{V}]$
$\mathcal{C}_3$	$E[Z U] + E[Z V] + E[Z \tilde{U}, \tilde{V}] - E[Z \tilde{U}] - E[Z \tilde{V}]$
$\mathcal{C}_4$	$E[Z U, \tilde{V}] + E[Z \tilde{U}, V] - E[Z \tilde{U}, \tilde{V}]$

tain the best row clustering by optimizing over the individual row assignments while keeping the column clustering fixed and vice versa. In particular, optimizing the contribution of each row to the overall approximation error leads to the row cluster update step,

$$\rho^{t+1}(u) = \operatorname{argmin}_{g: [g]_1^k} E_{V|u}[\xi(u, g, V, \gamma^t(V))], \quad [u]_1^m.$$

Similarly, we obtain the column cluster update step,

$$\gamma^{t+1}(v) = \operatorname{argmin}_{h: [h]_1^l} E_{U|v}[\xi(U, \rho^t(U), v, h)], \quad [v]_1^n.$$

So far, we have only considered updating the row (or column clustering) keeping the Lagrange multipliers fixed. After row (or column) updates, the approximation  $\hat{Z}^t = \zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda^{*t})$  is closer to the original matrix  $Z$  than the earlier minimum Bregman information solution  $\hat{Z}^t$ , but it is not necessarily the best approximation to  $Z$  of the form  $\zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda)$ . Hence, we need to now optimize over the Lagrange multipliers keeping the co-clustering fixed. It turns out [1] that the Lagrange multipliers that result in the best approximation to  $Z$  are same as the optimal Lagrange multipliers of the minimum Bregman information problem based on the new co-clustering  $(\rho^{t+1}, \gamma^{t+1})$ . Based on this observation, we set  $\hat{Z}^{t+1}$  to be the minimum Bregman information solution in steps 2A and 2B.

#### 3.2 The Algorithm

Finally, we state the meta algorithm for generalized Bregman co-clustering (see Algorithm 1) that is a concrete “implementation” of our outline at the beginning of Section 3. Further, since the row/column cluster update steps and the minimum Bregman solution steps all progressively decrease the matrix approximation error, i.e., the Bregman co-clustering objective function, the alternate minimization scheme shown in Algorithm 1 is guaranteed to achieve local optimality.

**Theorem 2** *The general Bregman co-clustering algorithm (Algorithm 1) converges to a solution that is locally optimal for the Bregman co-clustering problem (2.8), i.e., the objective function cannot be improved by changing either the row clustering, or the column clustering.*

**Table 3: Row and column cluster updates for I-divergence.**

$\mathcal{C}$	$\xi(u, g, V, \gamma(V))$	$\xi(U, \rho(U), v, h)$
$\mathcal{C}_1$	$E_{V u}[Z \log \frac{Z}{E[Z g]}]$	$E_{U v}[Z \log \frac{Z}{E[Z h]}]$
$\mathcal{C}_2$	$E_{V u}[Z \log \frac{Z}{E[Z g, \hat{V}]}]$	$E_{U v}[Z \log \frac{Z}{E[Z \hat{U}, h]}]$
$\mathcal{C}_3$	$E_{V u}[Z \log \frac{Z \times E[Z g]}{E[Z g, \hat{V}]}]$	$E_{U v}[Z \log \frac{Z \times E[Z h]}{E[Z \hat{U}, h]}]$
$\mathcal{C}_4$	$E_{V u}[Z \log \frac{Z \times E[Z g, V]}{E[Z g, \hat{V}]}]$	$E_{U v}[Z \log \frac{Z \times E[Z \hat{U}, h]}{E[Z \hat{U}, h]}]$

**Table 4: Row and column cluster updates for squared Euclidean distance.**

$\mathcal{C}$	$\xi(u, g, V, \gamma(V))$	$\xi(U, \rho(U), v, h)$
$\mathcal{C}_1$	$E_{V u}[(Z - E[Z g])^2]$	$E_{U v}[(Z - E[Z h])^2]$
$\mathcal{C}_2$	$E_{V u}[(Z - E[Z g, \hat{V}])^2]$	$E_{U v}[(Z - E[Z \hat{U}, h])^2]$
$\mathcal{C}_3$	$E_{V u}[(Z - E[Z g, \hat{V}] + E[Z g])^2]$	$E_{U v}[(Z - E[Z \hat{U}, h] + E[Z h])^2]$
$\mathcal{C}_4$	$E_{V u}[(Z - E[Z g, \hat{V}] + E[Z g, V])^2]$	$E_{U v}[(Z - E[Z \hat{U}, h] + E[Z \hat{U}, h])^2]$

When the Bregman divergence is I-divergence or squared Euclidean distance, the minimum Bregman information problem has a closed form analytic solution as shown in Tables 1 and 2. Hence, it is straightforward to obtain the row and column cluster update steps (Tables 3 and 4) and implement the Bregman co-clustering algorithm (Algorithm 1). The resulting algorithms involve a computational effort that is linear in the size of the data and are hence, scalable. In general, the minimum Bregman information problem need not have a closed form solution and the update steps need to be determined using numerical computation. However, since the Lagrange dual  $L(\Lambda)$  in the minimum Bregman information problem (2.6) is convex in the Lagrange multipliers  $\Lambda$ , it is possible to obtain the optimal Lagrange multipliers using convex optimization techniques [3]. The minimum Bregman information solution and the row and column cluster update steps can then be obtained from the optimal Lagrange multipliers.

## 4. EXPERIMENTS

There are a number of experimental results in existing literature [4, 5, 8, 10] that illustrate the usefulness of particular instances of our Bregman co-clustering framework. In fact, a large class of parametric partitioning clustering algorithms [2] including `kmeans` can be shown to be special cases of the proposed framework wherein only rows or only columns are being clustered.

In recent years, co-clustering has been successfully applied to various application domains such as text mining [8] and analysis of microarray gene-expression data. Hence, here we do not experimentally re-evaluate the Bregman co-clustering algorithms against other methods. Instead, we present brief case studies to demonstrate two salient features of the proposed co-clustering algorithms: (a) dimensionality reduction, and (b) missing value prediction.

### 4.1 Dimensionality Reduction

Dimensionality reduction techniques are widely used for text clustering to handle sparsity and high-dimensionality of text data. Typically, the dimensionality reduction step comes before the clustering step, and the two steps are almost independent. In practice, it is not clear which dimen-

---

### Algorithm 1 Bregman Co-clustering Algorithm

---

**Input:** Matrix  $Z \subseteq S^{m \times n}$ , probability measure  $\nu$ , Bregman divergence  $d_\phi : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ , number of row clusters  $l$ , number of column clusters  $k$ , constraint set  $\mathcal{C}$ .

**Output:** Co-clustering  $(\rho^*, \gamma^*)$  that (locally) optimizes the objective function in (2.8).

**Method:**

{**Initialize**  $\rho, \gamma$  }

$\hat{U} \leftarrow \rho(U), \hat{V} \leftarrow \gamma(V)$

**repeat**

{**Step A: Update Row Clusters**  $(\rho)$ }

**for**  $u = 1$  to  $m$  **do**

$\rho(u) \leftarrow \operatorname{argmin}_{g: [g]_k^+} E_{V|u}[\xi(u, g, V, \gamma(V))]$

where  $\xi(U, \rho(U), V, \gamma(V)) = d_\phi(Z, \tilde{Z})$ ,  $\tilde{Z} = \zeta(\rho, \gamma, \Lambda)$  and  $\Lambda$  are optimal Lagrange multipliers before updates.

**end for**

$\hat{U} \leftarrow \rho(U)$

{**Step B: Update Column Clusters**  $(\gamma)$ }

**for**  $v = 1$  to  $n$  **do**

$\gamma(v) \leftarrow \operatorname{argmin}_{h: [h]_k^+} E_{U|v}[\xi(U, \rho(U), v, h)]$

where  $\xi(U, \rho(U), V, \gamma(V)) = d_\phi(Z, \tilde{Z})$ ,  $\tilde{Z} = \zeta(\rho, \gamma, \Lambda)$  and  $\Lambda$  are optimal Lagrange multipliers before updates.

**end for**

$\hat{V} \leftarrow \gamma(V)$

**until** convergence

---

**Table 5: Effect of Implicit Dimensionality Reduction by Co-clustering on Classic3. Each subtable is for a fixed number of (document,word) co-clusters.**

	(3,20)		(3,500)			(3,2500)		
1389	1	2	1364	3	18	920	49	292
9	1455	33	5	1446	21	31	1239	404
0	4	998	29	11	994	447	172	337

sionality reduction technique to use in order to get a good clustering. Co-clustering has the interesting capability of *interleaving* dimensionality reduction and clustering. This implicit dimensionality reduction often results in superior results than regular clustering techniques [8].

Using the bag-of-words model for text, let each column of the input matrix represent a document, and let each row represent a word. Keeping the number of document clusters fixed, we present results by varying the number of word clusters. We ran the experiments on the `Classic3` dataset, a document collection from the SMART project at Cornell University with 3 classes. Co-clustering was performed without looking at the class labels. We present confusion matrices between the cluster labels assigned by co-clustering and the true class labels, over various numbers of word clusters. The number of document clusters were fixed at 3 for all experiments reported. As we can clearly see from Table 5 (for `Classic3`), implicit dimensionality reduction by co-clustering actually gives better document clusters, in the sense that the cluster labels agree more with the true class labels with fewer word clusters.

### 4.2 Missing Value Prediction

To illustrate missing value prediction, we consider a collaborative filtering based recommender system. The main problem in this setting is to predict the preference of a given user for a given item using the known preferences of all other users. A popular approach to handle this is by computing the Pearson correlation of each user with all other users

**Table 6: Mean Absolute Error for Movie Ratings**

Algo.	$\mathcal{C}_2, \text{SqE}$	$\mathcal{C}_3, \text{SqE}$	$\mathcal{C}_2, \text{IDiv}$	$\mathcal{C}_3, \text{IDiv}$	Pearson
Error	0.8398	0.7639	0.8397	0.7723	1.4211

based on the known preferences and predict the unknown rating by proportionately combining the other users' ratings. We adopt a co-clustering approach to address the same problem. The main idea is to simultaneously compute the user and item co-clusters by assigning zero measure to the missing values. As a result, the co-clustering algorithm tries to recover the original structure of the data while disregarding the missing values and the reconstructed approximate matrix can be used for prediction.

For our experimental results, we use a subset of the Each-Movie dataset<sup>3</sup> consisting of 500 users, 200 movies and containing 25809 ratings, each rating being an integer between 0 (bad) to 5 (excellent). Of these, we use 90% ratings for co-clustering, i.e., as the training data and 10% ratings as the test data for prediction. We applied four different co-clustering algorithms ( $k = 10$ ,  $l = 10$ ) corresponding to constraint sets  $\mathcal{C}_2$  and  $\mathcal{C}_3$  with squared Euclidean (SqE) distance and I-divergence (IDiv) to the training data and used the reconstructed matrix for predicting the test ratings. We also implemented a simple collaborative filtering scheme based on Pearson's correlation. Table 6 shows the mean absolute error between the predicted ratings and the actual ratings for the different methods. From the table, we observe that the co-clustering techniques achieve superior results. For constraint set  $\mathcal{C}_3$ , the individual biases of the users (row average) and the movies (column average) are accounted for, hence resulting in a better prediction. The co-clustering algorithms are computationally efficient since the processing time is linear in the number of the known ratings.

## 5. RELATED WORK

Our work is primarily related to three main areas: co-clustering, matrix approximation and learning based on Bregman divergences.

Co-clustering has been a topic of much interest in the recent years because of its applications to problems in microarray analysis [4, 5] and text mining [8]. In fact, there exist many formulations of the co-clustering problem such as the hierarchical co-clustering model [9], the bi-clustering model [4] that involves finding the best co-clusters one at a time, etc. In this paper, we have focussed on the partitional co-clustering formulation first introduced in [9].

Matrix approximation approaches based on singular value decomposition (SVD) have been widely studied and used. However, they are quite often inappropriate for data matrices such as co-occurrence and contingency tables, since SVD-based decompositions are difficult to interpret, which is necessary for data mining applications. Alternative techniques involving non-negativity constraints [11] using KL-divergence as the approximation loss function [10, 11] have been proposed. However, these approaches apply to special types of matrices. A general formulation that is both interpretable and applicable to various classes of matrices would be invaluable. The proposed Bregman co-clustering formulation attempts to address this requirement.

Recent research [2] has shown that several results involving the KL-divergence and the squared Euclidean distance

are in fact based on certain convexity properties and hence, generalize to all Bregman divergences. This intuition motivated us to consider co-clustering based on Bregman divergences. Further, the similarities between the maximum entropy and the least squares principles [7] prompted us to explore a more general minimum Bregman information principle for all Bregman divergences.

## 6. DISCUSSION

This paper makes three main contributions. First, we generalized parametric co-clustering to loss functions corresponding to all Bregman divergences. The generality of the formulation makes the technique applicable to practically all types of data matrices. Second, we showed that approximation models of various complexities are possible depending on the statistics that are to be preserved. Third, we proposed and extensively used the minimum Bregman information principle as a generalization of the maximum entropy principle. For the two Bregman divergences that we focussed on, viz., I-divergence and squared Euclidean distance, the proposed algorithm has linear time complexity and is hence scalable.

**Acknowledgements:** This research was supported by NSF grants IIS-0307792, IIS0325116, NSF CAREER Award ACI-0093404, and by an IBM PhD fellowship to Arindam Banerjee.

## 7. REFERENCES

- [1] A. Banerjee, I. Dhillon, J. Ghosh, S. Merugu, and D. Modha. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. Technical Report UTCS TR04-24, UT, Austin, 2004.
- [2] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh. Clustering with Bregman divergences. In *SDM*, 2004.
- [3] Y. Censor and S. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press, 1998.
- [4] Y. Cheng and G. M. Church. Biclustering of expression data. In *ICMB*, pages 93–103, 2000.
- [5] H. Cho, I. S. Dhillon, Y. Guan, and S. Sra. Minimum sum squared residue co-clustering of gene expression data. In *SDM*, 2004.
- [6] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.
- [7] I. Csiszar. Why least squares and maximum entropy? an axiomatic approach to inference for linear inverse problems. *Annals of Statistics*, 19:2032–2066, 1991.
- [8] I. Dhillon, S. Mallela, and D. Modha. Information-theoretic co-clustering. In *KDD*, pages 89–98, 2003.
- [9] J. A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.
- [10] T. Hofmann and J. Puzicha. Unsupervised learning from dyadic data. Technical Report TR-98-042, ICSI, Berkeley, 1998.
- [11] D. L. Lee and S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, 2001. 556-562.

<sup>3</sup><http://www.research.compaq.com/src/eachmovie/>