Journal of Machine Learning Research 8 (2007) 1919-1986 Submitted 8/06; Published 8/07

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

Arindam Banerjee BANERJEE@CS.UMN.EDU
Department of Computer Science and Engineering
University of Minnesota, Twin Cities
Minneapolis, MN, USA

Inderjit Dhillon INDERJIT@CS.UTEXAS.EDU
Department of Computer Sciences
University of Texas at Austin
Austin, TX, USA

Joydeep Ghosh GHOSH@ECE.UTEXAS.EDU
Department of Electrical and Computer Engineering
University of Texas at Austin
Austin, TX, USA

Srujana Merugu SRUJANA@YAHOO-INC.COM
Yahoo! Research
Santa Clara, CA, USA

Dharmendra S. Modha DMODHA@US.IBM.COM
IBM Almaden Research Center
San Jose, CA, USA

Editor: John Lafferty

Abstract

Co-clustering, or simultaneous clustering of rows and columns of a two-dimensional data matrix, is rapidly becoming a powerful data analysis technique. Co-clustering has enjoyed wide success in varied application domains such as text clustering, gene-microarray analysis, natural language processing and image, speech and video analysis. In this paper, we introduce a partitional co-clustering formulation that is driven by the search for a good matrix approximation—every co-clustering is associated with an approximation of the original data matrix and the quality of co-clustering is determined by the approximation error. We allow the approximation error to be measured using a large class of loss functions called Bregman divergences that include squared Euclidean distance and KL-divergence as special cases. In addition, we permit multiple structurally different co-clustering schemes that preserve various linear statistics of the original data matrix. To accomplish the above tasks, we introduce a new *minimum Bregman information* (MBI) principle that simultaneously generalizes the *maximum entropy* and *standard least squares* principles, and leads to a matrix approximation that is optimal among all generalized additive models in a certain natural parameter space. Analysis based on this principle yields an elegant meta algorithm, special cases of which include most previously known alternate minimization based clustering algorithms such as kmeans and co-clustering algorithms such as information theoretic (Dhillon et al., 2003b) and minimum sum-squared residue co-clustering (Cho et al., 2004). To demonstrate the generality and flexibility of our co-clustering framework, we provide examples and empirical evidence on a vari-

ety of problem domains and also describe novel co-clustering applications such as missing value prediction and compression of categorical data matrices.

Keywords: co-clustering, matrix approximation, Bregman divergences, Bregman information, maximum entropy

1. Introduction

Data naturally arises in the form of matrices in a multitude of machine learning and data mining applications. Often, the data matrices that arise in real-world applications contain a large number of rows and columns, and may be very sparse. Understanding the natural structure of such matrices is a fundamental problem.

Clustering is an unsupervised learning technique that has been often used to discover the "latent structure" of data matrices that describe a set of objects (rows) by their feature values (columns). Typically, a clustering algorithm strives to group "similar" objects (or rows). A large number of clustering algorithms such as kmeans, agglomerative clustering, and their variants have been thoroughly studied (Jain and Dubes, 1988; Ghosh, 2003). Often, clustering is preceded by a dimensionality reduction phase, such as feature selection where only a subset of the columns is retained. As an alternative to feature selection, one can cluster the columns, and then represent each resulting group of features by a single derived feature (Dhillon et al., 2003a).

A recent paper (Dhillon and Modha, 2001) dealing with the spherical kmeans algorithm for clustering large, sparse document-term matrices arising in text mining graphically demonstrates (see Figures 13, 31, and 32 in the paper by Dhillon and Modha, 2001) that document clustering naturally brings together similar words. Intuitively, documents are similar because they use similar words. A natural question is whether it is possible to mathematically capture this relationship between rows and columns. Furthermore, is it possible to exploit this relationship to a practical advantage? This paper shows that both these questions can be answered in the affirmative in the context of clustering.

Co-clustering, also called bi-clustering (Hartigan, 1972; Cheng and Church, 2000), is the problem of simultaneously clustering rows and columns of a data matrix. Unlike clustering which seeks similar rows or columns, co-clustering seeks "blocks" (or "co-clusters") of rows and columns that are inter-related. Co-clustering has recently received a lot of attention in several practical applications such as simultaneous clustering of documents and words in text mining (Dhillon et al., 2003b; Gao et al., 2005; Takamura and Matsumoto, 2003), genes and experimental conditions in bioinformatics (Cheng and Church, 2000; Cho et al., 2004; Kluger et al., 2003), tokens and contexts in natural language processing (Freitag, 2004; Rohwer and Freitag, 2004; Li and Abe, 1998), users and movies in recommender systems (George and Merugu, 2005), etc.

Co-clustering is desirable over traditional "single-sided" clustering from a number of perspectives:

1. Simultaneous grouping of row and column clusters is more informative and digestible. Co-clustering provides compressed representations that are easily interpretable while preserving most of the information contained in the original data, which makes it valuable to a large class of statistical data analysis applications.

2. A row (or column) clustering can be thought of as dimensionality reduction along the rows (or columns). Simultaneous clustering along rows and columns reduces dimensionality along both axes, thus leading to a statistical problem with dramatically smaller number of param-

eters and hence, a much more compact representation for subsequent analysis. Since co-clustering incorporates row clustering information into column clustering and vice versa, one can think of it as a "statistical regularization" technique that can yield better quality clusters even if one is primarily interested in a single-sided clustering. The statistical regularization effect of co-clustering is extremely important when dealing with large, sparse data matrices, for example, those arising in text mining. A similar intuition can be drawn from subspace clustering methods (Parsons et al., 2004), which only use a part of the full potential of the co-clustering methodology.

3. As the size of data matrices increases, so does the need for scalable clustering algorithms. Single-sided, geometric clustering algorithms such as kmeans and its variants have computation time proportional to mnk per iteration, where m is the number of rows, n is the number of columns and k is the number of row clusters. Co-clustering algorithms based on a similar iterative process, on the other hand, involve optimizing over a smaller number of parameters, and can relax this dependence to $O(mkl + nkl)$ where m, n and k are defined as before and l is the number of column clusters. Since the number of row and column clusters is usually much smaller than the original number of rows and columns, co-clustering can lead to substantial reduction in the running time (see, for example, Dhillon et al. 2003b and Rohwer and Freitag 2004).

In summary, co-clustering is an exciting paradigm for unsupervised data analysis in that it is more informative, has less parameters, is scalable, and is able to effectively intertwine row and column information.

In this paper, we concentrate on partitional co-clustering (also called checkerboard bi-clustering by Kluger et al., 2003) where all the rows and columns are partitioned into disjoint row and column clusters respectively. We provide a general framework for addressing this problem that considerably expands the scope and applicability of the co-clustering methodology. To appreciate this generalization, it is helpful to view partitional co-clustering as a lossy data compression problem where, given a specified number of rows and column clusters, one attempts to retain as much information as possible about the original data matrix in terms of statistics based on the co-clustering (Dhillon et al., 2003b). The main idea is that a reconstruction based on co-clustering should result in the same set of user-specified statistics as the original matrix. There are two key components in formulating a co-clustering problem: (i) choosing a set of critical co-clustering-based statistics of the original data matrix that need to be preserved, and (ii) selecting an appropriate measure to quantify the information loss or the discrepancy between the original data matrix and the compressed representation provided by the co-clustering. For example, in the work of Cheng and Church (2000), the row and column averages of each co-cluster are preserved and the discrepancy between the original and the compressed representation is measured in terms of the sum of element-wise squared deviation. In contrast, information-theoretic co-clustering (ITCC) (Dhillon et al., 2003b), which is applicable to data matrices representing joint probability distributions, preserves a different set of summary statistics, that is, the row and column averages and the co-cluster averages. Further, the quality of the compressed representation is measured in terms of the sum of element-wise I-divergence. In the next subsection, we take a closer look at ITCC to provide a concrete motivating example.

1.1 ITCC: A Motivating Example

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, where $[u]_1^m$ denotes an index u running over $\{1,\cdots,m\}$. Information-theoretic co-clustering provides a principled approach for simultaneously clustering the rows and columns of the joint probability distribution $p(X,Y)$. In practice, the entries of this matrix may not be known and are, instead, estimated from a contingency table or co-occurrence matrix. Let the row clusters be denoted by $\{\hat{x}_g\},\ [g]_1^k$ and the column clusters by $\{\hat{y}_h\},\ [h]_1^l$. Let $\hat{X}$ and $\hat{Y}$ denote the clustered random variables induced by X and Y that range over the set of row and column clusters respectively. A natural goal is to choose a co-clustering that preserves the maximum amount of "information" in the original data. In particular, since the data corresponds to the joint probability distribution of random variables X and Y, it is natural to preserve the mutual information between X and Y, or, in other words, minimize the loss in mutual information due to the compression that results from co-clustering. Thus, a suitable formulation is to solve the problem:

$$\min_{\hat{X},\hat{Y}} \quad (I(X;Y) - I(\hat{X};\hat{Y}))\ , \tag{1}$$

where $I(X;Y)$ is the mutual information between X and Y (Cover and Thomas, 1991). Dhillon et al. (2003b) showed that

$$I(X;Y) - I(\hat{X},\hat{Y}) = KL(p(X,Y)||q(X,Y))\ , \tag{2}$$

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

$$q(X,Y) = p(\hat{X},\hat{Y})p(X|\hat{X})p(Y|\hat{Y})\ , \tag{3}$$

and $KL(\cdot||\cdot)$ denotes the Kullback-Leibler(KL) divergence, also known as relative entropy. Thus, the search for the optimal co-clustering may be conducted by searching for the nearest approximation $q(X,Y)$ that has the above form. Since $p(X)$, $p(Y)$ and $p(\hat{X},\hat{Y})$ are determined by $m-1$, $n-1$ and $kl-1$ parameters respectively, with $k+l$ dependencies due to $p(\hat{X})$ and $p(\hat{Y})$, for a given co-clustering the distribution $q(X,Y)$ depends only on $(kl+m+n-k-l-3)$ independent parameters, which is much smaller than the $mn-1$ parameters that determine a general joint distribution. Hence, $q(X,Y)$ is a "low-complexity" or low-parameter matrix approximation of $p(X,Y)$.

The above viewpoint was developed by Dhillon et al. (2003b). We now present an alternate viewpoint that will enable us to generalize our approach to arbitrary data matrices and general distortion measures. The following lemma highlights a key maximum entropy property that makes $q(X,Y)$ a "low-complexity" or low-parameter approximation.

Lemma 1 *Given a fixed co-clustering, consider the set of joint distributions p' that preserve the row, column and co-cluster marginals of the input distribution p:*

$$\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} p'(x,y) = p(\hat{x},\hat{y}) = \sum_{x\in\hat{x}}\sum_{y\in\hat{y}} p(x,y), \quad \forall \hat{x},\hat{y}, \tag{4}$$

$$p'(x) = p(x), \quad p'(y) = p(y), \quad \forall x,y. \tag{5}$$

Among all such distributions p', the distribution q given in (3) has the maximum entropy, that is, $H(q(X,Y)) \geq H(p'(X,Y))$.

A proof of the above lemma is presented in Appendix A. What is the significance of the above lemma? In the absence of any constraints, the uniform distribution, $p_0(X,Y) = \{\frac{1}{mn}\}$, has the maximum entropy. If only row and column marginals are to be preserved, that is, (5) holds, then the

product distribution $p(X)p(Y)$ has maximum entropy (see Cover and Thomas, 1991, Problem 5, Chapter 11). The above lemma states that among all distributions that preserve row, column, and co-cluster marginals, that is, (4) and (5) hold, the maximum entropy distribution has the form in (3). The maximum entropy characterization ensures that $q(X,Y)$ has a number of desirable properties. For instance, given the row, column and co-cluster marginals, it is the unique distribution that satisfies certain consistency criteria (Csiszár, 1991; Shore and Johnson, 1980). In Section 4, we also demonstrate that it is the optimal approximation to the original distribution p in terms of KL-divergence among all multiplicative combinations of the preserved marginals. It is important to note that the *maximum entropy characterization also implies that q is a low-complexity matrix approximation.*[1] In contrast, note that the input $p(X,Y)$ obviously satisfies the constraints in (4) and (5), but in general, is determined by $(mn-1)$ parameters and has lower entropy than q. Every co-clustering yields a unique maximum entropy distribution. Thus, by (2) and Lemma 1, the co-clustering problem (1) is equivalent to the problem of finding the nearest (in KL-divergence) maximum entropy distribution that preserves the row, column and co-cluster marginals of the original distribution. The maximum entropy property in Lemma 1 may be re-stated as $KL(q||p_0) \leq KL(p'||p_0)$, where p_0 is the uniform distribution. Thus, the maximum entropy principle is identical to the minimum relative entropy principle where the relative entropy is measured with respect to p_0.

The above formulation is applicable when the data matrix corresponds to an empirical joint distribution. However, there are important situations when the data matrix cannot be interpreted in this matter, for example the matrix may contain negative entries and/or a distortion measure other than KL-divergence, such as the squared Euclidean distance might be more appropriate.

1.2 Key Contributions

The contributions of this paper can be summarized as follows:

- We introduce a partitional co-clustering formulation driven by a matrix approximation viewpoint where the quality of co-clustering is characterized by the accuracy of an induced co-clustering-based matrix approximation, measured in terms of a suitable distortion measure. This formulation serves the dual purpose of (i) obtaining row and column clusterings that optimize a well-defined global objective function, and (ii) providing a new class of desirable matrix approximations.

- Our formulation is applicable to all Bregman divergences (Azoury and Warmuth, 2001; Banerjee et al., 2005b; Bregman, 1967; Censor and Zenios, 1998), which constitute a large class of distortion measures including the most commonly used ones such as squared Euclidean distance, KL-divergence, Itakura-Saito distance, etc. The generalization to Bregman divergences is useful due to a bijection between regular exponential families and a sub-class of Bregman divergences called regular Bregman divergences (Banerjee et al., 2005b). This bijection result enables us to choose the appropriate Bregman divergence based on the underlying data generation process or noise model. This, in turn, allows us to perform co-clustering on a wide variety of data matrices.

- Our formulation allows multiple co-clustering schemes wherein the reconstruction of the original matrix is based on different sets of linear summary statistics that one may be interested

1. The complexity here refers to the number of parameters required to construct a good approximation to the given matrix. It does not refer to the expected communication complexity, as is usual in the context of Shannon entropy.

in preserving. In particular, we focus on summary statistics that correspond to conditional expectations over partitions that result from the rows, columns and co-clusterings. We establish that there are exactly six non-trivial co-clustering schemes. Each of these schemes corresponds to a unique co-clustering basis, that is, combination of conditional expectations over various partitions. Using a formal abstraction, we explicitly enumerate and analyze the co-clustering problem for all the six bases. Existing partitional co-clustering algorithms (Cho et al., 2004; Dhillon et al., 2003b) can then be seen as special cases of the abstraction, employing one of the six co-clustering bases. Three of the six bases we discuss have not been used in the literature till date.

- Previous work on co-clustering assume that all the elements of the data matrix are equally important, that is, have uniform measure. In contrast, we associate a probability measure with the elements of the specified matrix and pose the co-clustering problem in terms of the random variable that takes values among the matrix elements following this measure. Our formulation based on random variables provides a natural mechanism for handling values with differing levels of uncertainty and in particular, missing values, while retaining both the analytical and algorithmic simplicity of the corresponding uniform-measure formulation.

- En route to formulating the Bregman co-clustering problem, we introduce the *minimum Bregman information* (MBI) principle that generalizes the well-known *maximum entropy* and *standard least-squares* principles to all Bregman loss functions. The co-clustering process is guided by the search for the matrix approximation that has the minimum Bregman information while preserving the specified co-clustering statistics.

- We provide an interpretation of the Bregman co-clustering problem in terms of minimizing the loss in Bregman information due to co-clustering, which enables us to generalize the viewpoint presented in information-theoretic co-clustering (Dhillon et al., 2003b).

- We develop an efficient meta co-clustering algorithm based on alternate minimization that is guaranteed to achieve (local) optimality for all Bregman divergences. Many previously known parametric clustering and co-clustering algorithms such as minimum sum-squared residue co-clustering (Cho et al., 2004) and information-theoretic co-clustering (Dhillon et al., 2003b) follow as special cases of our methodology.

- Lastly, we describe some novel applications of co-clustering such as predicting missing values and compression of categorical data matrices, and also provide empirical results comparing different co-clustering schemes for various application domains.

In summary, our results provide a sound theoretical framework for the analysis and design of efficient co-clustering algorithms for data approximation and compression, and considerably expand applicability of the co-clustering methodology.

1.3 Outline of the Paper and Notation

The rest of paper is organized as follows: We begin by reviewing preliminary definitions and describe the Bregman co-clustering problem at a conceptual level in Section 2. To present our co-clustering framework, we proceed as follows. First, we describe and analyze *block-average co-clustering* in Section 3, which is an important special case of our general formulation, in order

to provide intuition about the main results. Then, in Section 4, we enumerate various possible *co-clustering bases* corresponding to the summary statistics chosen to be preserved, and present a general formulation that is applicable to all these bases. In Section 5, we analyze the general Bregman co-clustering problem and propose a meta-algorithm that is applicable to all Bregman divergences and all co-clustering bases. In Appendix E, we describe how the Bregman co-clustering algorithm can be instantiated for various choices of Bregman divergence and co-clustering basis by providing the exact update steps. Readers interested in a purely computational recipe can jump to Apendix E. Empirical evidence on the benefits of co-clustering and preliminary experiments on novel co-clustering applications are presented in Section 6. We discuss related work in Section 7 and conclude in Section 8.

A brief word about the notation: Sets such as $\{x_1, \cdots, x_n\}$ are enumerated as $\{x_i\}_{i=1}^n$ and an index i running over the set $\{1, \cdots, n\}$ is denoted by $[i]_1^n$. Random variables are denoted using upper case letters, for example, Z. Matrices are denoted using upper case bold letters, for example, $\mathbf{Z}$, whereas the corresponding lower case letters z_{uv} denote the matrix elements. Transpose of a matrix $\mathbf{Z}$ is denoted by $\mathbf{Z}^T$. The effective domain of a function f is denoted by $\mathrm{dom}(f)$ and the inverse of a function f, when well defined, is denoted by $f^{(-1)}$. The relative interior and boundary of a set $\mathcal{S}$ are denoted by $\mathrm{ri}(\mathcal{S})$ and $\mathrm{bd}(\mathcal{S})$ respectively. Tables 15, 16 and 17 list the notation used in the paper.

2. Preliminaries

In this section, we discuss some important properties of Bregman divergences and also describe the basic setup of our co-clustering framework.

2.1 Bregman Divergences and Bregman Information

We start by defining Bregman divergences (Bregman, 1967; Censor and Zenios, 1998), which form a large class of well-behaved loss functions with a number of desirable properties.

Definition 1 Let ϕ be a real-valued convex function of Legendre type[2] (Rockafellar, 1970; Banerjee et al., 2005b) defined on the convex set $\mathcal{S} \equiv \mathrm{dom}(\phi)$ ($\subseteq \mathbb{R}^d$). The *Bregman divergence* $d_\phi : \mathcal{S} \times \mathrm{ri}(\mathcal{S}) \mapsto \mathbb{R}_+$ 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 ϕ.

Example 1.A (I-Divergence) Given $z \in \mathbb{R}_+$, let $\phi(z) = z\log z - z$. 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 $z_1, z_2 \in \mathbb{R}$, $d_\phi(z_1, z_2) = (z_1 - z_2)^2$.

Example 3.A (Itakura-Saito Distance) Given $z \in \mathbb{R}_+$, let $\phi(z) = -\log z$. For $z_1, z_2 \in \mathbb{R}_+$, $d_\phi(z_1, z_2) = \frac{z_1}{z_2} - \log\left(\frac{z_1}{z_2}\right) - 1$.

2. A proper, closed convex function ϕ is said to be of Legendre type if (i) $\mathrm{int}(\mathrm{dom}(\phi))$ is non-empty, (ii) ϕ is strictly convex and differentiable on $\mathrm{int}(\mathrm{dom}(\phi))$, and (iii) $\forall z_b \in \mathrm{bd}(\mathrm{dom}(\phi))$, $\lim_{z \to z_b} ||\nabla\phi(z)|| \to \infty$, where $z \in \mathrm{dom}(\phi)$.

Given a Bregman divergence and a random variable, the uncertainty in the random variable can be captured in terms of a useful concept called Bregman information (Banerjee et al., 2005b) defined below.

Definition 2 For any Bregman divergence $d_\phi : \mathcal{S} \times \text{int}(\mathcal{S}) \mapsto \mathbb{R}_+$ and any random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq \mathcal{S}$, the *Bregman information* of Z is defined as the expected Bregman divergence to the expectation, that is,

$$I_\phi(Z) = E[d_\phi(Z, E[Z])] \ .$$

Intuitively, this quantity is a measure of the "spread" or the "information" in the random variable.

Example 1.B (I-Divergence) Given a random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq \mathbb{R}_+$, the Bregman information corresponding to I-divergence is given by

$$I_\phi(Z) = E[Z \log(Z/E[Z]) - Z + E[Z]] = E[Z \log(Z/E[Z])] \ .$$

When w is the uniform measure and the support of Z (say $\mathcal{Z}$) consists of joint probability values of two other random variables X and Y, that is, $\mathcal{Z} = \{p(x_u, y_v),\ [u]_1^m, [v]_1^n\}$, then $E[Z] = \frac{1}{mn}$, that is, probability value corresponding to the uniform distribution $p_0(X,Y)$. The Bregman information in this case is given by

$$I_\phi(Z) = \frac{1}{mn} \sum_{u=1}^{m} \sum_{v=1}^{n} p(x_u, y_v) \log \left(\frac{p(x_u, y_v)}{p_0(x_u, y_v)} \right) = \frac{1}{mn} KL(p||p_0) = -\frac{1}{mn} H(p) + \text{constant},$$

where $H(\cdot)$ is the Shannon entropy.

Example 2.B (Squared Euclidean Distance) Given $Z \sim w(z),\ z \in \mathcal{Z} \subseteq \mathbb{R}$, the Bregman information corresponding to squared Euclidean distance is given by

$$I_\phi(Z) = E[Z - E[Z]]^2 \ ,$$

which is the variance of Z. When w is uniform and the support of Z, that is, $\mathcal{Z}$ consists of elements in a matrix $\mathbf{Z} \in \mathbb{R}^{m \times n}$, that is, $\mathcal{Z} = \{z_{uv},\ [u]_1^m, [v]_1^n\}$, then $E[Z] = \frac{1}{mn} \sum_{u=1}^{m} \sum_{v=1}^{n} z_{uv} \equiv \overline{z}$. The Bregman information in this case is given by

$$I_\phi(Z) = \frac{1}{mn} \sum_{u=1}^{m} \sum_{v=1}^{n} (z_{uv} - \overline{z})^2 = \frac{1}{mn} \sum_{u=1}^{m} \sum_{v=1}^{n} z_{uv}^2 - \overline{z}^2 = \frac{1}{mn} \|\mathbf{Z}\|_F^2 + \text{constant},$$

that is, a linear function of the squared Frobenius norm of $\mathbf{Z}$.

We note a useful property of Bregman information that will be extensively used in subsequent sections. The property, formally stated below, shows that the Bregman information exactly equals the difference between the two sides of Jensen's inequality (Cover and Thomas, 1991).

Lemma 2 *(Banerjee et al., 2005b) For any Bregman divergence $d_\phi : \mathcal{S} \times \text{ri}(\mathcal{S}) \mapsto \mathbb{R}_+$ and random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq \mathcal{S}$, the Bregman information $I_\phi(Z) = E[d_\phi(Z, E[Z])] = E[\phi(Z)] - \phi(E[Z])$.*

Clearly, Bregman information is always non-negative. For a detailed list of other properties and examples of Bregman divergences and Bregman information, the reader is referred to Banerjee et al. (2005b) and Appendix B.

2.2 Data Matrix

We focus on the problem of co-clustering a specified $m \times n$ data matrix $\mathbf{Z}$. Let each entry of $\mathbf{Z} = [z_{uv}]$ take values in the convex set[3] $\mathcal{S} = \text{dom}(\phi)$, for example, $\mathcal{S} = \mathbb{R}$ for $\phi(z) = z^2$ and $\mathcal{S} = \mathbb{R}_+$ for $\phi(z) = z\log z - z$. Hence, $\mathbf{Z} \in \mathcal{S}^{m\times n}$. Observe that we are now admitting a much larger class of matrices than those used in the co-clustering formulations of Cho et al. (2004) and Dhillon et al. (2003b).

Given the data matrix $\mathbf{Z}$, we consider a random variable Z, that takes values in $\mathbf{Z}$ following a probability measure as described below. Let U be a random variable that takes values in $\{1,\cdots,m\}$, the set of row indices, and let V be a random variable that takes values in $\{1,\cdots,n\}$, the set of column indices. Let (U,V) be distributed according to a probability measure $w = \{w_{uv} : [u]_1^m, [v]_1^n\}$, which is either pre-specified or set to be the uniform distribution.[4] Let Z be a (U,V)-measurable random variable that takes values in $\mathbf{Z}$ following w, that is, $p(Z(u,v) = z_{uv}) = w_{uv}$. Clearly, for a given matrix $\mathbf{Z}$, the random variable Z is a deterministic function of the random variable (U,V). Throughout the paper, we assume the matrix $\mathbf{Z}$ and the measure w to be fixed so that taking conditional expectations of the random variable Z is well defined. In pure numeric terms, such conditional expectations are simply weighted row/column/block averages of the matrix $\mathbf{Z}$ according to the weights w. The stochastic formalization enables a succinct way to analyze such weighted averages.

Example 1.C (I-Divergence) Let $(X,Y) \sim p(X,Y)$ be jointly distributed random variables with X and 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 $\mathbf{Z} = [z_{uv}]$, $[u]_1^m, [v]_1^n$, where $z_{uv} = p(x_u, y_v)$ is a deterministic function of u and v. This example with a uniform measure w corresponds to the setting described in Section 2, Example 1.B (originally in the work of Dhillon et al., 2003b).

Example 2.C (Squared Euclidean Distance) Let $\mathbf{Z} \in \mathbb{R}^{m\times n}$ denote a data matrix whose elements may assume positive, negative, or zero values and let w be a uniform measure. This example corresponds to the co-clustering setting described by Cheng and Church (2000) and Cho et al. (2004).

2.3 Bregman Co-clustering

We define a $k \times l$ partitional co-clustering as a pair of functions:

$$\rho : \{1,\cdots,m\} \mapsto \{1,\cdots,k\} ,$$
$$\gamma : \{1,\cdots,n\} \mapsto \{1,\cdots,l\} .$$

Let $\hat{U}$ and $\hat{V}$ be random variables that take values in $\{1,\cdots,k\}$ and $\{1,\cdots,l\}$ such that $\hat{U} = \rho(U)$ and $\hat{V} = \gamma(V)$. Let $\hat{\mathbf{Z}} = [\hat{z}_{uv}] \in \mathcal{S}^{m\times n}$ be an approximation for the data matrix $\mathbf{Z}$ such that $\hat{\mathbf{Z}}$ *depends only upon* a given co-clustering (ρ,γ) and certain summary statistics derived from the co-clustering. Let $\hat{Z}$ be a (U,V)-measurable random variable that takes values in this approximate matrix $\hat{\mathbf{Z}}$ following

3. $\mathcal{S}$ need not necessarily be a subset of $\mathbb{R}$. It is convenient to assume this for ease of exposition. In general, the elements of the matrix $\mathbf{Z}$ can take values over any convex domain with a well-defined Bregman divergence. We give examples of such settings in Section 6.
4. Associating a measure with the elements of a matrix is not common, but this construct allows us to deal with a wider variety of situations including the modeling of matrices with missing values. Further, several quantities of interest, such as row/column/block averages, can now be succinctly described in terms of conditional expectations.

w, that is, $p(\hat{Z}(U,V) = \hat{z}_{uv}) = w_{uv}$. Then the goodness of the underlying co-clustering can be measured in terms of the expected distortion between Z and $\hat{Z}$, that is,

$$E[d_\phi(Z,\hat{Z})] = \sum_{u=1}^{m}\sum_{v=1}^{n} w_{uv} d_\phi(z_{uv},\hat{z}_{uv}) = d_{\Phi_w}(\mathbf{Z},\hat{\mathbf{Z}}), \tag{6}$$

where $\Phi_w : \mathcal{S}^{m\times n} \mapsto \mathbb{R}$ is a separable convex function induced on the matrices such that the Bregman divergence between any pair of matrices is the weighted sum of the element-wise Bregman divergences corresponding to the convex function ϕ. From the matrix approximation viewpoint, the above quantity is simply the weighted element-wise distortion between the given matrix $\mathbf{Z}$ and the approximation $\hat{\mathbf{Z}}$. The co-clustering problem is then to find (ρ,γ) such that (6) is minimized. To carry out this plan, we need to make precise the connection between (ρ,γ) and $\hat{Z}$.

Example 1.D (I-Divergence) The Bregman co-clustering objective function (6) in this case is given by $E[d_\phi(Z,\hat{Z})] = E[Z\log(Z/\hat{Z}) - Z + \hat{Z}]$.

Example 2.D (Squared Euclidean Distance) The Bregman co-clustering objective function (6) in this case is given by $E[d_\phi(Z,\hat{Z})] = E[(Z-\hat{Z})^2]$.

The goodness of a co-clustering (ρ,γ) is determined by how well $\hat{Z}$ (or the matrix $\hat{\mathbf{Z}}$) approximates Z (or the matrix $\mathbf{Z}$). The crucial thing to note is that the construction of the approximation $\hat{Z}$ is based on the co-clustering (ρ,γ) and certain summary statistics of the original random variable Z that one wants to preserve in the approximation. The summary statistics may be properties of the co-clusters themselves, such as co-cluster marginals as in (4), and/or some other important statistics of the data, such as row and column marginals as in (5). Note that Z is not accessible while constructing $\hat{Z}$, since otherwise one could just set $\hat{Z} = Z$ and get perfect reconstruction. The special case when $\hat{Z}$ is constructed only using the co-clustering (ρ,γ) and the co-cluster marginals is important and easy to understand. Moreover, it is a straightforward generalization of one-sided clustering schemes such as kmeans. Hence, we first investigate this special case in detail in the next section. The general case, where additional summary information such as row/column marginals of the original matrix are available, will be analyzed in Sections 4 and 5.

3. Block Average Co-clustering: A Special Case

In this section, we discuss the important special case of Bregman co-clustering where the summary statistics are derived by aggregating along the co-clusters, that is, the summary statistics preserved are just the co-cluster means. Hence, in this case, for a given co-clustering (ρ,γ), $\hat{Z}$ has to be reconstructed based only on the co-cluster means, or equivalently, the conditional expectation random variable $E[Z|\hat{U},\hat{V}]$ where expectation is taken with respect to the measure w.[5] The quality of the co-clustering (ρ,γ) is determined by the approximation error between Z and $\hat{Z}$.

3.1 Minimum Bregman Information (MBI) Principle

In order to analyze the block co-clustering problem, we first focus on characterizing the approximation random variable $\hat{Z}$ given a fixed co-clustering (ρ,γ) and the resulting co-cluster means

5. Unless otherwise mentioned, the expectations in the rest of the paper are with respect to the probability measure w.

$\{E[Z|\hat{u},\hat{v}]\}$. While there can be many different ways to get an approximation $\hat{Z}$ from the available information, we consider a principled characterization based on the Bregman information of the reconstruction $\hat{Z}$. In particular, we propose and use the *minimum Bregman information principle* that can be shown to be a direct generalization of the maximum entropy as well as the least squares principles.

In order to get the "best" approximation, we consider a special class of approximating random variables Z' based on the given co-clustering and the available information $E[Z|\hat{U},\hat{V}]$. Let $\mathcal{S}_A$ be defined as

$$\mathcal{S}_A = \{Z' | E[Z'|\hat{u},\hat{v}] = E[Z|\hat{u},\hat{v}],\ \forall [\hat{u}]_1^k, [\hat{v}]_1^l\ \}\ . \tag{7}$$

It is reasonable to search for the best approximation in $\mathcal{S}_A$ since any random variable Z' in this class has the same co-cluster statistics as the original random variable Z. In other words, the corresponding reconstructed matrices preserve the co-cluster statistics of the original matrix, which is desirable. Then, with respect to the set $\mathcal{S}_A$, we ask: What is the "best" random variable to select from this set? We propose a new *minimum Bregman information principle* that recommends selecting a random variable that has the minimum Bregman information subject to the linear constraints (7):

$$\hat{Z}_A\ \equiv\ \hat{Z}_A(\rho,\gamma) = \underset{Z' \in \mathcal{S}_A}{\operatorname{argmin}}\ I_\phi(Z'). \tag{8}$$

The basic philosophy behind the minimum Bregman information principle is that the "best" approximation given certain information is one that does not make any extra assumptions over the available information. Mathematically, the notion of *no extra assumptions* or maximal uncertainty translates to *minimum Bregman information* while the available information is provided by the linear constraints that preserve the specified statistics.

As the following examples show, the widely used *maximum entropy principle* (Jaynes, 1957; Cover and Thomas, 1991) and *standard least squares principles* (Csiszár, 1991) can be obtained as special cases of the MBI principle.

Example 1.E From Example 1.B, we observe that the Bregman information of a random variable Z following a uniform distribution over the joint probability values of two other random variables X and Y is given by $-\frac{1}{mn}H(p(X,Y))$ upto an additive constant, that is, it is negatively related to entropy of the joint distribution of X and Y. Hence, minimizing the Bregman information is equivalent to maximizing the entropy demonstrating that the *maximum entropy principle* is a special case of the MBI principle corresponding to I-divergence.

Example 2.E From Example 2.B, we observe that the Bregman information of a random variable Z following a uniform distribution over the elements of a matrix $\mathbf{Z}$ is given by $\frac{1}{mn}\|\mathbf{Z}\|_F^2$ upto an additive constant. Hence, minimizing the Bregman information in this case is equivalent to minimizing the Frobenius norm of the matrix (L_2 norm for a vector), which in turn implies that the *standard least squares principle* is a special case of the MBI principle corresponding to squared error.

Now, we focus on getting a closed form solution of the minimum Bregman information problem. In the absence of any constraints, the minimum Bregman information solution corresponds to a constant random variable. For the current situation, where we are constrained to preserve the co-cluster means $\{E[Z|\hat{u},\hat{v}]\}$, the following theorem shows that the best approximation $\hat{Z}_A$ simply equals $E[Z|\hat{U},\hat{V}]$.

Theorem 1 *The solution to (8) is unique and is given by*

$$\hat{Z}_A = E[Z|\hat{U},\hat{V}].$$

Proof Let Z' be any random variable in $\mathcal{S}_A$ and let $\hat{Z}_A$ denote $E[Z|\hat{U},\hat{V}]$. By definition,

$$\begin{aligned}
I_\phi(Z') &\overset{(a)}{=} E[\phi(Z')] - \phi(E[Z']) \\
&= E[\phi(Z')] - E_{\hat{U},\hat{V}}[\phi(E[Z'|\hat{U},\hat{V}])] + E_{\hat{U},\hat{V}}[\phi(E[Z'|\hat{U},\hat{V}])] - \phi(E[Z']) \\
&\overset{(b)}{=} E[\phi(Z')] - E_{\hat{U},\hat{V}}[\phi(E[Z'|\hat{U},\hat{V}])] + E_{\hat{U},\hat{V}}[\phi(\hat{Z}_A)] - \phi(E_{\hat{U},\hat{V}}[\hat{Z}_A]) \\
&\overset{(c)}{=} E_{\hat{U},\hat{V}}\left[E_{Z'|\hat{U},\hat{V}}[\phi(Z')] - \phi(E[Z'|\hat{U},\hat{V}])\right] + I_\phi(\hat{Z}_A) \\
&\overset{(d)}{\geq} I_\phi(\hat{Z}_A),
\end{aligned}$$

where (a) and (c) follow from Lemma 2; (b) follows from the fact that $E[Z'|\hat{U},\hat{V}] = E[Z|\hat{U},\hat{V}] = \hat{Z}_A$ and $E_{\hat{U},\hat{V}}[E[Z|\hat{U},\hat{V}]] = E_{\hat{U},\hat{V}}[\hat{Z}_A] = E[Z] = E[Z']$; and (d) follows from conditional Jensen's inequality. In particular, since ϕ is convex, we have $E_{Z'|\hat{U},\hat{V}}[\phi(Z')] \geq \phi(E[Z'|\hat{U},\hat{V}])$.

Hence, $\hat{Z}_A$ has lower Bregman information than any random variable in $\mathcal{S}_A$. Further, $\hat{Z}_A \in \mathcal{S}_A$, that is, $E[\hat{Z}_A|\hat{U},\hat{V}] = \hat{Z}_A = E[Z|\hat{U},\hat{V}]$. Along with the strict convexity of ϕ, this ensures that $\hat{Z}_A = E[Z|\hat{U},\hat{V}]$ is the unique solution to (8). ∎

For an alternative constructive proof of Theorem 1, please see Appendix C.

Besides being the MBI solution, $\hat{Z}_A$ has an additional important property that makes it the "best" reconstruction. Although we focused on the set $\mathcal{S}_A$ that contains all Z' that preserve the known co-cluster statistics, an alternative could have been to investigate the set $\mathcal{S}_B$ that contains all deterministic functions of the available information $E[Z|\hat{U},\hat{V}]$, that is,

$$\mathcal{S}_B = \{Z''|Z'' = f(E[Z|\hat{U},\hat{V}])\}\ , \tag{9}$$

where f is an arbitrary $(\hat{U},\hat{V})$-measurable function. In $\mathcal{S}_B$, the optimal approximation $\hat{Z}_B$ is the one that is closest to the true Z:

$$\hat{Z}_B \equiv \operatorname*{argmin}_{Z'' \in \mathcal{S}_B} E[d_\phi(Z,Z'')]\ . \tag{10}$$

In order to show a relationship between $\hat{Z}_A$ and $\hat{Z}_B$, we start with the following lemma (Lemma 3), which establishes the fact that the MBI solution $\hat{Z}_A$ allows a Pythagorean decomposition of the expected divergence between any $Z' \in \mathcal{S}_A$ and any $Z'' \in \mathcal{S}_B$.[6] Recall that $\mathcal{S}_A$ consists of all random variables that have the same co-cluster statistics as Z and $\mathcal{S}_B$ consists of all measurable functions of $E[Z|\hat{U},\hat{V}]$.

Lemma 3 *For any $Z' \in \mathcal{S}_A$ as in (7), any $Z'' \in \mathcal{S}_B$ as in (9), and $\hat{Z}_A$ as in (8),*

$$E[d_\phi(Z',Z'')] = E[d_\phi(Z',\hat{Z}_A)] + E[d_\phi(\hat{Z}_A,Z'')].$$

6. The analysis using Pythagorean decomposition of Bregman divergences can be viewed as a special case of Bregman duality analysis of Della Pietra et al. (2001). The advantage of our special case analysis is that it has rich semantics relevant to the co-clustering setting, and the proofs are simpler than the general case proofs in Della Pietra et al. (2001). See Section 4.4 for more details.

A proof of the lemma is presented in Appendix C. Now, since $\hat{Z}_A = E[Z|\hat{U},\hat{V}]$, and is hence a function of $E[Z|\hat{U},\hat{V}]$, we have $\hat{Z}_A \in \mathcal{S}_B$. As a result, from Lemma 3, we get the following projection theorem, which states that the MBI solution $\hat{Z}_A$ is the "forward" Bregman projection of any element of $\mathcal{S}_A$ onto the set $\mathcal{S}_B$ as well as the "backward" Bregman projection of any element of $\mathcal{S}_B$ onto the set $\mathcal{S}_A$.

Theorem 2 (Projection Theorem) *For any $Z' \in \mathcal{S}_A$ as in (7), any $Z'' \in \mathcal{S}_B$ as in (9), and $\hat{Z}_A$ as in (8), we have,*

(a) $\hat{Z}_A = \underset{Z' \in \mathcal{S}_A}{\operatorname{argmin}} E[d_\phi(Z', Z'')]$,

(b) $\hat{Z}_A = \underset{Z'' \in \mathcal{S}_B}{\operatorname{argmin}} E[d_\phi(Z', Z'')]$.

A proof of the theorem is presented in Appendix C. Since the original $Z \in \mathcal{S}_A$, we observe that $\hat{Z}_A$ is the best approximation (by a backward Bregman projection) to Z in $\mathcal{S}_B$, implying $\hat{Z}_B = \hat{Z}_A$ as formally stated below.

Corollary 1 *For $\hat{Z}_A$ and $\hat{Z}_B$ given by (8) and (10) respectively, we have*

$$\hat{Z} \equiv \hat{Z}_A = \hat{Z}_B. \tag{11}$$

The equivalence result is a precise mathematical quantification of the optimal approximation property of the MBI solution for the special case where only $E[Z|\hat{U},\hat{V}]$ is available during reconstruction. It shows that the best approximation in terms of expected Bregman divergence given the co-cluster statistics is indeed the MBI solution that preserves those statistics.

3.2 Co-clustering Problem Formulation

Now that we have associated an approximation $\hat{Z}$ with a given co-clustering (ρ,γ), we return to the original Bregman co-clustering problem in (6). The goal is to obtain a co-clustering (ρ,γ) such that the expected Bregman divergence between Z and the approximation $\hat{Z}$ is minimized. So far, we know that the best reconstruction $\hat{Z}$ is the MBI solution and is expressed in closed form by Theorem 1. The following lemma presents an alternative characterization of the co-clustering objective function (6). It shows that the expected Bregman divergence to the approximation $\hat{Z}$ is exactly equal to the loss in Bregman information due to co-clustering.

Lemma 4 *For any random variable Z and $\hat{Z}$ as in (11),*

$$E[d_\phi(Z,\hat{Z})] = I_\phi(Z) - I_\phi(\hat{Z}) \ .$$

Proof By definition,

$$\begin{aligned} E[d_\phi(Z,\hat{Z})] &= E[\phi(Z)-\phi(\hat{Z})-\langle Z-\hat{Z},\nabla\phi(\hat{Z})\rangle] \\ &\overset{(a)}{=} E[\phi(Z)]-E[\phi(\hat{Z})]-E_{\hat{U},\hat{V}}[\langle E[Z|\hat{U},\hat{V}]-E[\hat{Z}|\hat{U},\hat{V}],\nabla\phi(\hat{Z})\rangle] \\ &\overset{(b)}{=} E[\phi(Z)]-E[\phi(\hat{Z})] \\ &\overset{(c)}{=} E[\phi(Z)]-\phi(E[Z])-E[\phi(\hat{Z})]+\phi(E[\hat{Z}]) \\ &\overset{(d)}{=} I_\phi(Z)-I_\phi(\hat{Z}), \end{aligned}$$

where (a) follows from the fact that $\hat{Z}$ and hence, $\nabla\phi(\hat{Z})$ is constant for fixed $(\hat{U},\hat{V})$, (b) follows since $\hat{Z}\in\mathcal{S}_A$, (c) follows since $E[Z]=E[\hat{Z}]$ and (d) follows from Lemma 2. ■

Using Lemma 4, the original Bregman clustering problem in (6) can be posed as one of finding the optimal co-clustering (ρ^*,γ^*) defined as follows:

$$(\rho^*,\gamma^*) = \operatorname*{argmin}_{(\rho,\gamma)} E[d_\phi(Z,\hat{Z})] = \operatorname*{argmin}_{(\rho,\gamma)} [I_\phi(Z)-I_\phi(\hat{Z})] = \operatorname*{argmax}_{(\rho,\gamma)} I_\phi(\hat{Z}) , \tag{12}$$

since $I_\phi(Z)$ is a constant. Further, using the fact that $\hat{Z}$ is the solution to the MBI problem, we have

$$(\rho^*,\gamma^*) = \operatorname*{argmax}_{(\rho,\gamma)} \min_{Z'\in\mathcal{S}_A} I_\phi(Z') . \tag{13}$$

Hence, the best co-clustering (ρ^*,γ^*) is the one that results in the matrix reconstruction corresponding to the minimum approximation error, or equivalently, the one that solves the max-min problem in (13).

3.3 Block Average Co-clustering Algorithm

In this section, we present an algorithm for block average co-clustering based on a useful decomposition of the objective function (12), which gives a better insight on how to update either the row clustering ρ or the column clustering γ.

3.3.1 A USEFUL DECOMPOSITION

From Theorem 1, it follows that for a given co-clustering (ρ,γ), the approximation $\hat{Z}$ that achieves the minimum Bregman information is given by $\hat{z}_{uv}=E[Z|\hat{u},\hat{v}]$, where $\hat{u}=\rho(u),\hat{v}=\gamma(v)$. We denote the co-cluster means corresponding to (ρ,γ) as $\mu_{\hat{u}\hat{v}}$, that is, $\mu_{\hat{u}\hat{v}}=E[Z|\hat{u},\hat{v}]$. Hence, the optimal approximation $\hat{Z}$ corresponding to (ρ,γ) is given by

$$\hat{z}_{uv}=\mu_{\hat{u}\hat{v}}=\mu_{\rho(u)\gamma(v)}.$$

With this closed form for $\hat{Z}$, we have

$$\begin{aligned} E[d_\phi(Z,\hat{Z})] &= \sum_{u,v} w_{uv} d_\phi(z_{uv},\mu_{\rho(u)\gamma(v)}) \\ &= \sum_{g=1}^{k}\sum_{h=1}^{l}\sum_{u:\rho(u)=g}\sum_{v:\gamma(v)=h} w_{uv} d_\phi(z_{uv},\mu_{gh}) . \end{aligned} \tag{14}$$

Note that (14) decomposes the objective function in terms of the row cluster assignment $\rho(u)$ of each row u and column cluster assignment $\gamma(v)$ of each column v.

3.3.2 UPDATING ROW AND COLUMN CLUSTERS

Since the decomposition (14) is additive over all the rows (or columns), we can update the current cluster assignment of each row (or column) in order to decrease the objective function. For any particular row u, the contribution to the overall objective function is determined by its current assignment $\rho(u)$. Assuming $\rho(u) = g$, we can express the objective function (14) as the sum of row contributions of the form

$$J_u(g) = \sum_{h=1}^{l} \sum_{v:\gamma(v)=h} w_{uv} d_\phi(z_{uv}, \mu_{gh}) \ . \tag{15}$$

Note that the co-cluster means μ_{gh} remain unchanged during the update of the row (or column) clustering.

The choice of row cluster assignment g exactly determines what set of l co-cluster means μ_{gh} occur in (15). Hence, the best possible choice for the new row cluster assignment $\rho^{new}(u)$ is to pick the value of g that has the minimum cost, that is,

$$\rho^{new}(u) = \operatorname*{argmin}_g J_u(g) = \operatorname*{argmin}_g \sum_{h=1}^{l} \sum_{v:\gamma(v)=h} w_{uv} d_\phi(z_{uv}, \mu_{gh}) \ . \tag{16}$$

Since the terms corresponding to each row are additive in (14), the row assignment update in (16) can be applied simultaneously to all rows to get the new row assignments $\rho^{new}(u), [u]_1^n$. The new row assignments effectively change the current approximation matrix $\hat{Z}$ to a new matrix $\tilde{Z}_{\rho_1\gamma_0}$, which is just a row-permuted version of $\hat{Z}$ that achieves a lower cost, that is,

$$E[d_\phi(Z, \tilde{Z}_{\rho_1\gamma_0})] \leq E[d_\phi(Z, \hat{Z})] \ .$$

The decrease in the objective function value is due to the optimal greedy update in the row cluster assignments. A similar approach can be applied to update the column cluster assignments in order to obtain an even better approximation $\tilde{Z}_{\rho_1\gamma_1}$. Note that the current approximation can possibly be further improved by another round of row clustering updates to get an approximation $\tilde{Z}_{\rho_2\gamma_1}$, where the subscript in ρ (or γ) denotes the number of times the row (column) cluster assignment has been updated. The same process can be repeated multiple times. For simplicity, we denote the final assignments by $(\rho^{new}, \gamma^{new})$ and the approximation obtained from such reassignments as $\tilde{Z}$.

Once all row and column assignments have been updated, the new approximation matrix $\tilde{Z}$ need not be the minimum Bregman information solution for the new co-clustering $(\rho^{new}, \gamma^{new})$. Hence, one needs to recompute the new minimum Bregman solution $\hat{Z}^{new}$ corresponding to $(\rho^{new}, \gamma^{new})$. The following lemma, proved in Appendix C, establishes that the updated $\hat{Z}^{new}$ is guaranteed to either decrease the objective, or keep it unchanged. In fact, $\hat{Z}^{new}$ is the best approximation possible based on the co-clustering $(\rho^{new}, \gamma^{new})$.

Lemma 5 *Let $\hat{Z}^{new}$ be the minimum Bregman information solution corresponding to $(\rho^{new}, \gamma^{new})$. Then,*

$$E[d_\phi(Z, \hat{Z}^{new})] \leq E[d_\phi(Z, \tilde{Z})] \ .$$

Algorithm 1 Bregman Block Average Co-clustering (BBAC) Algorithm

Input: Matrix $\mathbf{Z} \subseteq \mathcal{S}^{m\times n}$, probability measure w, Bregman divergence $d_\phi : \mathcal{S} \times \mathrm{ri}(\mathcal{S}) \mapsto \mathbb{R}_+$, num. of row clusters l, num. of column clusters k.
Output: Block Co-clustering (ρ^*, γ^*) that (locally) optimizes the objective function in (12).
Method:

```
{Initialize ρ, γ}
Start with an arbitrary co-clustering (ρ, γ)
repeat
  {Step A: Update Co-cluster Means}
  for g = 1 to k do
    for h = 1 to l do
      μ_gh = (Σ_{u:ρ(u)=g} Σ_{v:γ(v)=h} w_uv z_uv) / (Σ_{u:ρ(u)=g} Σ_{v:γ(v)=h} w_uv)
    end for
  end for
  {Step B: Update Row Clusters (ρ)}
  for u = 1 to m do
    ρ(u) = argmin_{g∈{1,...,k}} Σ_{h=1}^{l} Σ_{v:γ(v)=h} w_uv d_φ(z_uv, μ_gh)
  end for
  {Step C: Update Column Clusters (γ)}
  for v = 1 to n do
    γ(v) = argmin_{h∈{1,...,l}} Σ_{g=1}^{k} Σ_{u:ρ(u)=g} w_uv d_φ(z_uv, μ_gh)
  end for
until convergence
return (ρ, γ)
```

3.3.3 THE ALGORITHM

The above analysis leads to a simple iterative algorithm for Bregman block average co-clustering (BBAC in Algorithm 1). The algorithm starts with an arbitrary choice of co-clustering (ρ, γ). At every iteration, either the row clustering ρ or the column clustering γ is updated in order to decrease the objective function value in (12). In practice, one could run multiple iterations of such updates. After the assignments have been updated for all rows and columns, the co-clustering means are updated, which further decreases the objective. The process is repeated till convergence. Since the objective decreases at every iteration, and the objective is lower bounded, the algorithm is guaranteed to converge to a (local) minimum of the objective.

3.4 Block Average Co-clustering as Matrix Factorization

Since the MBI solution is always the co-cluster means, and the BBAC algorithm essentially alternates between updating the row and column assignments, and updating the co-cluster means, the BBAC algorithm is a direct generalization of the Bregman clustering algorithm (Banerjee et al., 2005b). As we show below, the BBAC algorithm can also be viewed as solving a matrix factorization problem.

Let $\mathbf{Z}$ be the $m \times n$ matrix corresponding to the random variable Z and $\mathbf{W} \in \mathbb{R}_+^{m\times n}$ denote the matrix corresponding to a probability measure over the matrix elements. Let $\mathbf{R} \in \{0,1\}^{m\times k}$ and

$\mathbf{C} \in \{0,1\}^{n\times l}$ denote the row and column cluster membership matrices, that is,

$$r_{ug} = \begin{cases} 1 & g = \rho(u), \\ 0 & \text{otherwise}, \end{cases} \qquad c_{vh} = \begin{cases} 1 & h = \gamma(v), \\ 0 & \text{otherwise}. \end{cases}$$

Further, let $\mathbf{M}$ be a $k \times l$ matrix corresponding to the co-cluster means, that is, expectations or weighted averages of the matrix values over the co-clusters. Since the minimum Bregman information solution for the block co-clustering case are the co-cluster averages, the reconstructed matrix $\hat{\mathbf{Z}}$ can be expressed as the product $\mathbf{RMC}^T$. Therefore, the co-clustering problem is essentially reduces to finding row assignment matrix $\mathbf{R}$, column assignment matrix $\mathbf{C}$ such that the approximation error $d_{\Phi_w}(\mathbf{Z}, \hat{\mathbf{Z}})$ is minimized where $\hat{\mathbf{Z}} = \mathbf{RMC}^T$. The BBAC algorithm returns matrices $\mathbf{R}, \mathbf{M}$ and $\mathbf{C}$ that achieves a local minimum of the above objective function. When $l = n$, the BBAC algorithm reduces to the Bregman clustering algorithm (Banerjee et al., 2005b) applied to rows of $\mathbf{Z}$. In particular, when the Bregman divergence is the squared Euclidean distance, we obtain the classical kmeans algorithm.

3.5 General Formulation and Analysis: Warm Up

So far, we have studied in detail the important special case of block average co-clustering. In the next section, we will formulate and analyze a more general class of co-clustering problems.

The differences between the various formulations will stem from the different summary statistics used in the approximation $\hat{Z}$. For the block co-clustering case, $\hat{Z}$ depended only on the co-cluster means $\{E[Z|\hat{u},\hat{v}]\}$. In Section 4, we shall consider the exhaustive list of summary statistics based on which $\hat{Z}$ can be reconstructed, and go on to propose a general case meta-algorithm with provable properties in Section 5. The BBAC algorithm can then be seen as a special case of this meta-algorithm obtained for a particular choice of summary statistics.

Before going into the formulation and analysis of the general case, we want to highlight the results that are specific to block average co-clustering as well as the results that continue to hold in the general case for any choice of summary statistics. We start with the results that hold only for block average co-clustering and do not carry over to the general case.

1. For block average co-clustering, the MBI solution is the *same* for all Bregman divergences (Theorem 1). However, in the general case, the solution generally depends on the choice of the Bregman divergence. In fact, block average co-clustering is the only case when the solution is independent of this choice.

2. In the general case, it is not possible to get a closed form MBI solution. In general, a convex optimization problem has to be solved to find the solution; see Section 5.5 for some iterative approaches for computing the MBI solution. We also provide exact solutions for the important special cases where closed form solutions do exist.

3. For block co-clustering, the reconstruction from the minimum Bregman information solution is also the best approximation of the original Z among all functions of the co-cluster means (Corollary 1). This result holds only when the reconstruction is based on one set of summary statistics, which was the co-cluster means in the block co-clustering case. More formally, the result holds when the random variable is approximated based on a single sub-σ-algebra (see Section 4.1). In general, multiple sets of summary statistics may need to be preserved and the reconstruction will be based on multiple sub-σ-algebras.

4. The matrix approximation obtained in the general case need not be expressible as a matrix factorization in terms of the cluster membership matrices **R** and **C**. In fact, block average co-clustering is the only formulation where such an interpretation is possible for all Bregman divergences.

Finally, we focus on the results that continue to hold in the general case for arbitrary choices of summary statistics:

1. Although there need not be a closed form solution to the minimum Bregman information problem and the solution may depend on the choice of the Bregman divergence, some important properties of the solution remain unchanged in the general case. In particular, the form of the solution in terms of the Lagrange multipliers (see the constructive proof of Theorem 1 in Appendix C) remains unchanged.

2. The Pythagorean decomposition (Lemma 3) and the projection theorem (Theorem 2) associated with the sets $\mathcal{S}_A$ and $\mathcal{S}_B$ continues to hold for the general case, with $\mathcal{S}_B$ defined as the set of all generalized additive models of the various summary statistics in a transformed space (see Section 4.4). For the block average co-clustering case, since we only preserve the co-cluster means, the set $\mathcal{S}_B$ turns out to be set of all functions of the co-cluster means. Further, the MBI solution can be shown to be the best approximation to the original Z among this special class of functions of the summary statistics $\mathcal{S}_B$ generalizing the equivalence in Corollary 1. The general result that we discuss in Section 4.4 provides an axiomatic justification of the minimum Bregman information principle (Csiszár, 1991).

3. The loss in Bregman information result (Lemma 4) continues to hold.

4. Similar to Algorithm 1, we obtain an iterative algorithm for the general case where we alternately optimize over the row cluster assignments, column cluster assignments and the MBI solution. As in the block-average case, the co-clustering objective function allows an additive decomposition over the rows and columns and the resulting meta-algorithm (Algorithm 2) guarantees monotonic decrease of the objective function at every iteration.

4. Bregman Co-clustering: Formulation and Analysis

In this section, we formulate a general version of the Bregman co-clustering problem by abstracting out the commonalities between various possible co-clustering schemes that arise due to constraints that preserve different choices of summary statistics. To achieve this, we first define the notion of a co-clustering basis in terms of the conditional expectation-based statistics that one might want to preserve, and then enumerate all the possible co-clustering bases that may be of interest.

4.1 Co-clustering Bases

Let us fix a co-clustering (ρ, γ). Given the co-clustering, there are essentially four random variables of interest: U, V, $\hat{U}$, and $\hat{V}$. To these, we add two random variables $U_\emptyset$ and $V_\emptyset$ corresponding to the constant random variables over the rows and columns respectively, for easy enumeration. Let Γ_1 denote the set $\{U_\emptyset, V_\emptyset, \hat{U}, \hat{V}, U, V\}$. Our goal is to approximate the random variable Z using (possibly multiple) conditional expectations of Z where the conditioning is done on one or more of

the random variables in Γ_1. Observe that choosing one or more random variables to condition on is equivalent to choosing a sub-σ-algebra[7] $\mathcal{G}$ of Z. We focus on approximating Z using conditional expectations $E[Z|\mathcal{G}]$ since the conditional expectation $E[Z|\mathcal{G}]$ is the optimal approximation of the true Z with respect to any Bregman divergence among all $\mathcal{G}$-measurable functions (Banerjee et al., 2005a).

Since duplication of information in the preserved conditional expectations does not lead to a different approximation, we only focus on combinations of random variables from Γ_1 that will lead to a unique set of summary statistics. First, we observe that some of the random variables in Γ_1 are measurable with respect to some others. In other words, some random variables are just "high resolution" versions of some others so that conditioning on certain sets of members of Γ_1 is equivalent to conditioning on the subset with respect to which the rest are measurable. For example, $E[Z|U,\hat{U},V_\emptyset,\hat{V}] = E[Z|U,\hat{V}]$, since $\hat{U}$ is U-measurable, and $V_\emptyset$ is $\hat{V}$-measurable. In fact, due to the natural ordering of the random variables $\{U_\emptyset,\hat{U},U\}$ and $\{V_\emptyset,\hat{V},V\}$ in terms of measurability, only the row and column random variables of the highest granularity matter. Hence, there are only 9 unique sub-σ-algebras of Z based on which conditional expectations may be taken. We denote this set by Γ_2:

$$\Gamma_2 = \{\{U_\emptyset,V_\emptyset\},\{U_\emptyset,\hat{V}\},\{U_\emptyset,V\},\{\hat{U},V_\emptyset\},\{\hat{U},\hat{V}\},\{\hat{U},V\},\{U,V_\emptyset\},\{U,\hat{V}\},\{U,V\}\}.$$

Γ_2 determines the set of all summary statistics that one maybe interested in preserving. A particular choice of an element of Γ_2, such as $\{\hat{U},\hat{V}\}$, leads to an approximation scheme where the reconstruction matrix preserves the corresponding summary statistics. For the choice of $\{\hat{U},\hat{V}\}$, we get the block average co-clustering discussed in Section 3 where the matrix approximation preserves all co-cluster means.

Now, we focus on a much more general scenario where one may want to preserve possibly more than one summary statistic. In fact, one could consider all possible subsets of Γ_2. Of these, some combinations of summary statistics are effectively equivalent, for example, $\{\{\hat{U},V_\emptyset\},\{U_\emptyset,\hat{V}\},\{\hat{U},\hat{V}\}\}$ and $\{\{\hat{U},\hat{V}\}\}$, whereas some others are trivial and even independent of the co-clustering, for example, $\{\{U_\emptyset,V_\emptyset\}\}$ and $\{\{U_\emptyset,V\},\{U,V_\emptyset\}\}$. In this paper, we focus only on unique and non-trivial combinations of elements of Γ_2, that we call *co-clustering bases* and define them as follows:

Definition 3 [8] A *co-clustering basis* $\mathcal{C}$ is a set of elements of Γ_2, that is, an element of the power set 2^{Γ_2}, which satisfies the following two conditions:

(a) There exist $\mathcal{G}_1,\mathcal{G}_2 \in \mathcal{C}$ (with $\mathcal{G}_1$ possibly the same as $\mathcal{G}_2$) such that $\hat{U} \in \mathcal{G}_1$ and $\hat{V} \in \mathcal{G}_2$.

(b) There do not exist $\mathcal{G}_1,\mathcal{G}_2 \in \mathcal{C},\ \mathcal{G}_1 \neq \mathcal{G}_2$ such that $\mathcal{G}_2$ is a sub-σ-algebra of $\mathcal{G}_1$.

In the above definition, condition (a) ensures that the approximation depends on the co-clustering while condition (b) ensures that for any pair $\mathcal{G}_1,\mathcal{G}_2$, the conditional expectation $E[Z|\mathcal{G}_2]$ cannot be obtained from $E[Z|\mathcal{G}_1]$. The latter ensures that the approximation obtained using the basis $\mathcal{C}$ is not identical to that obtained using $\mathcal{C} \setminus \mathcal{G}_2$.

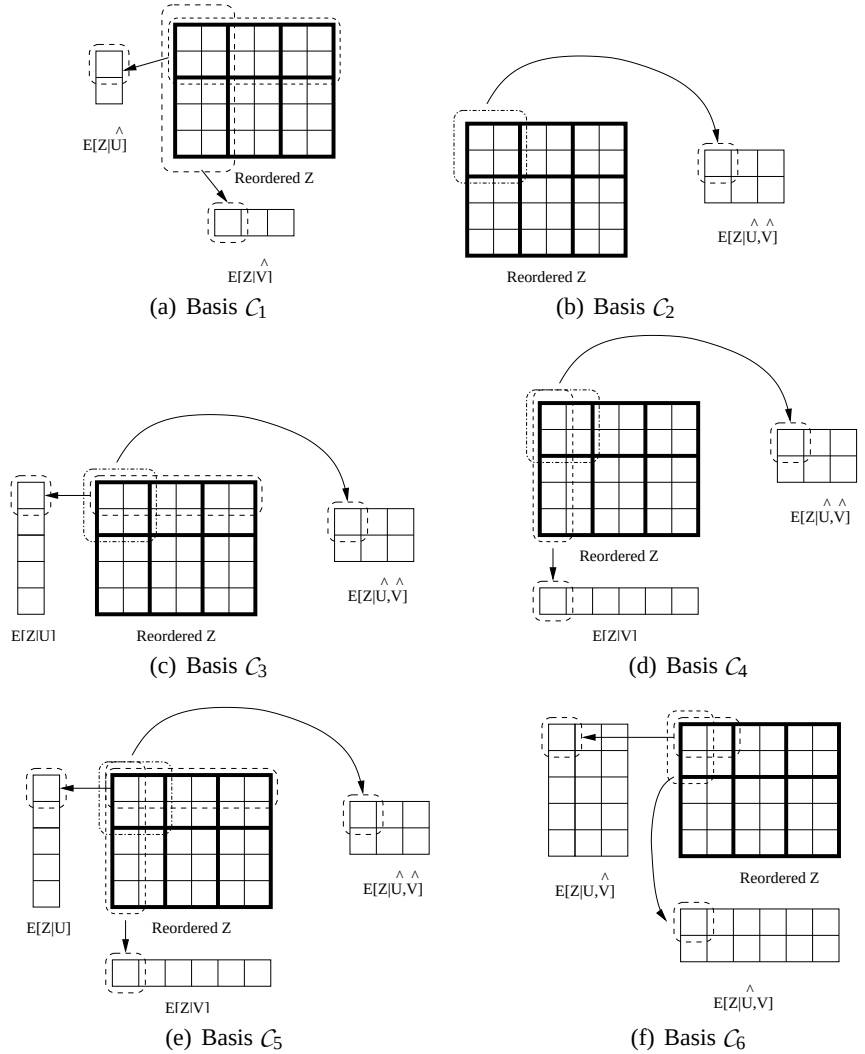
7. A σ-algebra is a collection of sets that includes the empty-set and is closed w.r.t. complements, countable unions and intersections. Further, $\mathcal{G}_1$ is a sub-σ-algebra of a σ-algebra $\mathcal{G}$ (or a $\mathcal{G}$-measurable random variable) if $\mathcal{G}_1$ is itself a σ-algebra and $\mathcal{G}_1 \subseteq \mathcal{G}$.

8. Note that each element of Γ_2 corresponds to a unique sub-σ-algebra of Z, and hence, we use identical notation for the elements of the co-clustering bases and the corresponding sub-σ-algebras.

The following theorem shows that there are only six possible co-clustering bases, each of which leads to a distinct matrix approximation scheme.

Theorem 3 *Given the random variable Z, there are only six distinct co-clustering bases that approximate Z using conditional expectations of Z given combinations of the row and column random variables $\{U,V,\hat{U},\hat{V}\}$. The six bases correspond to the sets*

$$\begin{aligned} \mathcal{C}_1 &= \{\{\hat{U}\},\{\hat{V}\}\}, & \mathcal{C}_2 &= \{\{\hat{U},\hat{V}\}\}, \\ \mathcal{C}_3 &= \{\{\hat{U},\hat{V}\},\{U\}\}, & \mathcal{C}_4 &= \{\{\hat{U},\hat{V}\},\{V\}\}, \\ \mathcal{C}_5 &= \{\{\hat{U},\hat{V}\},\{U\},\{V\}\}, & \mathcal{C}_6 &= \{\{U,\hat{V}\},\{\hat{U},V\}\}. \end{aligned}$$



(a) Basis $\mathcal{C}_1$ (b) Basis $\mathcal{C}_2$

(c) Basis $\mathcal{C}_3$ (d) Basis $\mathcal{C}_4$

(e) Basis $\mathcal{C}_5$ (f) Basis $\mathcal{C}_6$

Figure 1: Schematic diagram of the six co-clustering bases. In each case, the summary statistics used for reconstruction (e.g., $E[Z|\hat{U}]$ and $E[Z|\hat{V}]$) are expectations taken over the corresponding dotted regions (e.g., over all the columns and all the rows in the row cluster determined by $\hat{U}$ in case of $E[Z|\hat{U}]$).

Figure 1 contains a graphical representation of the various co-clustering bases. For example, in Figure 1(a), the expectations along the row clusters ($E[Z|\hat{U}]$) and the column clusters ($E[Z|\hat{V}]$) are the statistics used for reconstructing the original Z. From the table, we can see that the various conditional expectations correspond to matrices of different sizes. We make use of this observation later in Appendix E to obtain a computational recipe for the Bregman co-clustering problem. The sets $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_5$ and $\mathcal{C}_6$ are symmetric in the row and column random variables whereas $\mathcal{C}_3$ and $\mathcal{C}_4$ are not. Further, if we have access to $\{E[Z|\mathcal{G}] : \mathcal{G} \in \mathcal{C}_i\}$, for some $1 \leq i \leq 6$, then we can compute $\{E[Z|\mathcal{G}] : \mathcal{G} \in \mathcal{C}_j\}$ for all $1 \leq j \leq i,\ i \neq 4, j \neq 3$. In this sense, we say that the constraint set $\mathcal{C}_i$ is more complex than $\mathcal{C}_j$ for all $j < i \leq 6, i \neq 4, j \neq 3$ as illustrated in Figure 2. From a practical perspective, a more complex set of constraints allows us to retain more information about Z, but obviously requires an increased number of parameters.

Our abstraction allows us to handle all the above schemes in a systematic way. Now, consider a co-clustering basis $\mathcal{C} \in \{\mathcal{C}_i\}_{i=1}^{6}$ as the pertinent one. Given the choice of a particular basis, we need to decide on the "best" reconstruction $\hat{Z}$ for a given co-clustering (ρ, γ). Then the general co-clustering problem will effectively reduce to one of finding an optimal co-clustering (ρ^*, γ^*) whose reconstruction has the lowest approximation error with respect to the original Z.

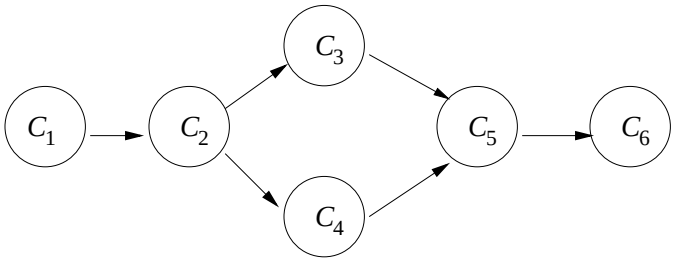


Figure 2: Relative complexity of the 6 co-clustering bases.

4.2 Minimum Bregman Information (MBI) Approximation

As in Section 3.1, for a given co-clustering (ρ, γ) and a given co-clustering basis $\mathcal{C}$, we use the MBI principle to obtain the "best" approximation $\hat{Z}$. Recall that for block average co-clustering, the search for the MBI solution was restricted to all Z' that preserved the co-cluster means. For a general co-clustering basis $\mathcal{C}$, the search space has to be appropriately generalized (or restricted) such that Z' preserves all the summary statistics relevant to $\mathcal{C}$. Let $\mathcal{S}_A$ denote a class of random variables such that every Z' in the class satisfies the following *linear constraints*, that is,

$$\mathcal{S}_A = \{Z' | E[Z|\mathcal{G}] = E[Z'|\mathcal{G}],\ \forall \mathcal{G} \in \mathcal{C}\}. \tag{17}$$

The reader may wish to compare the above definition (17) to the more specific definition (7) that is applicable in the case of block co-clustering. It can be readily seen that (7) follows by assuming that the co-clustering basis $\mathcal{C} = \{\{\hat{U}, \hat{V}\}\}$.

We now select the random variable $\hat{Z}_A \in \mathcal{S}_A$ that has the minimum Bregman information as the "best" approximation, that is,

$$\hat{Z}_A \equiv \operatorname*{argmin}_{Z' \in \mathcal{S}_A} I_\phi(Z'). \tag{18}$$

Coclustering basis $\mathcal{C}$	**Lagrange multipliers**	**Approximation $\hat{Z}_A$**
$\mathcal{C}_1$	$\Lambda^*_{\hat{U}} = -w_{\hat{U}} \log\left(\frac{E[Z\|\hat{U}]}{E[Z]}\right)$, $\Lambda^*_{\hat{V}} = -w_{\hat{V}} \log\left(\frac{E[Z\|\hat{V}]}{E[Z]}\right)$	$\frac{E[Z\|\hat{U}]\times E[Z\|\hat{V}]}{E[Z]}$
$\mathcal{C}_2$	$\Lambda^*_{\hat{U},\hat{V}} = -w_{\hat{U},\hat{V}} \log\left(\frac{E[Z\|\hat{U},\hat{V}]}{E[Z]}\right)$	$E[Z\|\hat{U},\hat{V}]$
$\mathcal{C}_3$	$\Lambda^*_{\hat{U},\hat{V}} = -w_{\hat{U},\hat{V}} \log\left(\frac{E[Z\|\hat{U},\hat{V}]}{E[Z]}\right)$, $\Lambda^*_{U} = -w_{U} \log\left(\frac{E[Z\|U]}{E[Z\|\hat{U}]}\right)$	$\frac{E[Z\|\hat{U},\hat{V}]\times E[Z\|U]}{E[Z\|\hat{U}]}$
$\mathcal{C}_4$	$\Lambda^*_{\hat{U},\hat{V}} = -w_{\hat{U},\hat{V}} \log\left(\frac{E[Z\|\hat{U},\hat{V}]}{E[Z]}\right)$, $\Lambda^*_{V} = -w_{V} \log\left(\frac{E[Z\|V]}{E[Z\|\hat{V}]}\right)$	$\frac{E[Z\|\hat{U},\hat{V}]\times E[Z\|V]}{E[Z\|\hat{V}]}$
$\mathcal{C}_5$	$\Lambda^*_{\hat{U},\hat{V}} = -w_{\hat{U},\hat{V}} \log\left(\frac{E[Z\|\hat{U},\hat{V}]}{E[Z]}\right)$ $\Lambda^*_{U} = -w_{U} \log\left(\frac{E[Z\|U]}{E[Z\|\hat{U}]}\right)$, $\Lambda^*_{V} = -w_{V} \log\left(\frac{E[Z\|V]}{E[Z\|\hat{V}]}\right)$	$\frac{E[Z\|\hat{U},\hat{V}]\times E[Z\|U]\times E[Z\|V]}{E[Z\|\hat{U}]\times E[Z\|\hat{V}]}$
$\mathcal{C}_6$	$\Lambda^*_{\hat{U},V} = -w_{\hat{U},V} \log\left(\frac{E[Z\|\hat{U},V]}{(E[Z]E[Z\|\hat{U},\hat{V}])^{1/2}}\right)$ $\Lambda^*_{U,\hat{V}} = -w_{U,\hat{V}} \log\left(\frac{E[Z\|U,\hat{V}]}{(E[Z]E[Z\|\hat{U},\hat{V}])^{1/2}}\right)$	$\frac{E[Z\|U,\hat{V}]\times E[Z\|\hat{U},V]}{E[Z\|\hat{U},\hat{V}]}$

Table 1: MBI solution and optimal Lagrange multipliers for I-Divergence.

The following theorem characterizes the solution to the MBI problem (18).

Theorem 4 *For any random variable Z and a specified co-clustering basis $\mathcal{C} = \{\mathcal{G}_r\}_{r=1}^s$, the solution $\hat{Z}_A$ to (18) is given by*

$$\nabla\phi(\hat{Z}_A) = \nabla\phi(E[Z]) - \sum_{r=1}^{s} \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}, \tag{19}$$

where $w_{\mathcal{G}_r}$ is the measure corresponding to $\mathcal{G}_r$ and $\{\Lambda^*_{\mathcal{G}_r}\}_{r=1}^s$ are the optimal Lagrange multipliers corresponding to the set of linear constraints:

$$E[Z'|\mathcal{G}_r] = E[Z|\mathcal{G}_r], \;\; [r]_1^s.$$

In the above theorem, note that every instantiation of the random variables $\{\mathcal{G}_r\}_{r=1}^s$ determines a single linear constraint and corresponds to uniquely determined scalar values for the optimal Lagrange multipliers $\{\Lambda^*_{\mathcal{G}_r}\}_{r=1}^s$, that is, $\Lambda^*_{\mathcal{G}_r}$ is a deterministic function of $\mathcal{G}_r$. Similarly, for each instantiation of $\mathcal{G}_r$, $w_{\mathcal{G}_r}$ equals the total measure associated with that particular instantiation, for example, $w_{\hat{u},\hat{v}} = \sum_{u:\rho(u)=\hat{u},v:\gamma(v)=\hat{v}} w_{uv}$. Further, the fact that ϕ is a strictly convex function ensures that $\nabla\phi$ is a one-to-one function so that (19) uniquely determines the approximation $\hat{Z}_A$. A proof of the above theorem is given in Appendix D.

For easy reference, in Tables 1-2, we present the optimal Lagrange multipliers[9] and the MBI solutions for I-divergence and squared Euclidean distance for each of the six co-clustering bases. Note that the approximation $\hat{Z}_A$ is itself a (U,V) measurable random variable and the elements of the corresponding matrix approximation $\hat{\mathbf{Z}}_A$ can be obtained by instantiating $\hat{Z}_A$ for specific choices of U and V. From Table 1, we observe that in case of I-divergence and original Z taking values over the probabilities of a joint distribution $p(X,Y)$, the approximation $\hat{Z}_A$ for the co-clustering basis $\mathcal{C}_5$ is given by $\frac{E[Z|\hat{U},\hat{V}]E[Z|U]E[Z|V]}{E[Z|\hat{U}]E[Z|\hat{V}]}$ which reduces to $q(X,Y) = \frac{p(X)p(Y)p(\hat{X},\hat{Y})}{p(\hat{X})p(\hat{Y})}$ (same as (3)) since the marginal over the various row, column and co-cluster partitions are directly proportional to the

9. The Lagrange dual $L(\Lambda)$ of Bregman information is concave in Λ for all bases, but strictly concave only for $\mathcal{C}_2$. Hence, the multipliers shown in Tables 1 and 2 are only one of the possible maximizers of $L(\Lambda)$ (for all the cases except $\mathcal{C}_2$).

Coclustering basis $\mathcal{C}$	**Lagrange multipliers**	**Approximation $\hat{Z}_A$**
$\mathcal{C}_1$	$\Lambda^*_{\hat{U}} = -2w_{\hat{U}}(E[Z\|\hat{U}] - E[Z])$, $\Lambda^*_{\hat{V}} = -2w_{\hat{U}}(E[Z\|\hat{V}] - E[Z])$	$E[Z\|\hat{U}] + E[Z\|\hat{V}] - E[Z]$
$\mathcal{C}_2$	$\Lambda^*_{\hat{U},\hat{V}} = -2w_{\hat{U},\hat{V}}(E[Z\|\hat{U},\hat{V}] - E[Z])$	$E[Z\|\hat{U},\hat{V}]$
$\mathcal{C}_3$	$\Lambda^*_{\hat{U},\hat{V}} = -2w_{\hat{U},\hat{V}}(E[Z\|\hat{U},\hat{V}] - E[Z])$, $\Lambda^*_U = -2w_U(E[Z\|U] - E[Z\|\hat{U}])$	$E[Z\|\hat{U},\hat{V}] + E[Z\|U] - E[Z\|\hat{U}]$
$\mathcal{C}_4$	$\Lambda^*_{\hat{U},\hat{V}} = -2w_{\hat{U},\hat{V}}(E[Z\|\hat{U},\hat{V}] - E[Z])$, $\Lambda^*_V = -2w_V(E[Z\|V] - E[Z\|\hat{V}])$	$E[Z\|\hat{U},\hat{V}] + E[Z\|V] - E[Z\|\hat{V}]$
$\mathcal{C}_5$	$\Lambda^*_{\hat{U},\hat{V}} = -2w_{\hat{U},\hat{V}}(E[Z\|\hat{U},\hat{V}] - E[Z])$ $\Lambda^*_U = -2w_U(E[Z\|U] - E[Z\|\hat{U}])$ $\Lambda^*_V = -2w_V(E[Z\|V] - E[Z\|\hat{V}])$	$E[Z\|\hat{U},\hat{V}] + E[Z\|U] + E[Z\|V]$ $-E[Z\|\hat{U}] - E[Z\|\hat{V}]$
$\mathcal{C}_6$	$\Lambda^*_{\hat{U},V} = -2w_{\hat{U},V}(E[Z\|\hat{U},V] - \frac{E[Z]}{2} - \frac{E[Z\|\hat{U},\hat{V}]}{2})$ $\Lambda^*_{U,\hat{V}} = -2w_{U,\hat{V}}(E[Z\|U,\hat{V}] - \frac{E[Z]}{2} - \frac{E[Z\|\hat{U},\hat{V}]}{2})$	$E[Z\|U,\hat{V}] + E[Z\|\hat{U},V] - E[Z\|\hat{U},\hat{V}]$

Table 2: MBI solution and optimal Lagrange multipliers for Squared Euclidean distance.

corresponding conditional expectations of Z. Further, the fact that $q(X,Y)$ is the minimum Bregman information solution for KL-divergence under certain constraints is equivalent to Lemma 1, which shows that it is the maximum entropy distribution under those constraints.

4.3 Co-clustering Problem Formulation

The expected Bregman divergence between the given random variable Z and the minimum Bregman information solution $\hat{Z}$ provides us with an elegant way to quantify the goodness of a co-clustering. This expected Bregman divergence is also exactly equal to the loss in Bregman information due to co-clustering as the following lemma shows. This equivalence provides another nice interpretation for the Bregman co-clustering formulation while generalizing the viewpoint presented in the information-theoretic co-clustering formulation (1) (originally Lemma 2.1 of Dhillon et al., 2003b).

Lemma 6 *For any random variable Z,*

$$E[d_\phi(Z,\hat{Z})] = I_\phi(Z) - I_\phi(\hat{Z}),$$

where $\hat{Z} = \hat{Z}_A$ defined in (18).

We are now ready to define the generalized co-clustering problem.

Definition 4 Given k, l, a Bregman divergence d_ϕ, a random variable Z following a non-negative measure w over the data matrix $\mathbf{Z} \in \mathcal{S}^{m\times n}$, and a co-clustering basis $\mathcal{C}$, we wish to find a co-clustering $(\rho^\star,\gamma^\star)$ that minimizes:

$$(\rho^\star,\gamma^\star) = \operatorname*{argmin}_{(\rho,\gamma)} E[d_\phi(Z,\hat{Z})] = \operatorname*{argmin}_{(\rho,\gamma)} (I_\phi(Z) - I_\phi(\hat{Z})) = \operatorname*{argmax}_{(\rho,\gamma)} I_\phi(\hat{Z}) \ , \tag{20}$$

where $\hat{Z} = \operatorname*{argmin}_{Z'\in\mathcal{S}_A} I_\phi(Z')$ as defined in (18).

The general problem is NP-hard by a reduction from the kmeans problem. Hence, it is difficult to obtain a globally optimal solution efficiently. However, in Section 5, we prove that it is possible to come up with an iterative update scheme that (a) monotonically decreases the objective function, and (b) converges to a local minimum of the problem.

Example 1.F (I-Divergence) Continuing from Example 1.C, the Bregman co-clustering objective function is given by $E[Z\log(Z/\hat{Z}) - Z + \hat{Z}] = E[Z\log(Z/\hat{Z})]$ since $E[Z] = E[\hat{Z}]$ where $\hat{Z}$ is the minimum Bregman information solution from Table 1. Note that for the co-clustering basis $\mathcal{C}_5$ and Z based on a joint distribution $p(X,Y)$, this reduces to $KL(p||q)$ where q is the joint distribution corresponding to the minimum Bregman solution indicating that (1) follows as a special case of (20).

Example 2.F (Squared Euclidean Distance) Continuing from Example 2.C, the Bregman co-clustering objective function is $E[(Z-\hat{Z})^2]$ where $\hat{Z}$ is the minimum Bregman information solution from Table 2. Note that for the co-clustering basis $\mathcal{C}_6$, this reduces to $E[(Z - E[Z|U,\hat{V}] - E[Z|\hat{U},V] + E[Z|\hat{U},\hat{V}])^2]$, which is equivalent to the squared residue objective function used in Cho et al. (2004) and Cheng and Church (2000).

4.4 Optimality of the MBI Solution

We now present an analysis of the optimality of the MBI solution as the "best" reconstruction of the original matrix given the row and column clustering and the summary statistics corresponding to any of the co-clustering bases. In Section 3, we showed that the minimum Bregman information solution is the best reconstruction among all measurable functions of the preserved summary statistics, that is, conditional expectations with respect to the co-clusters (Theorem 2). In this section, we present a generalization of that result, applicable to all the co-clustering bases discussed above.

Ideally, we would like to demonstrate that the MBI solution minimizes the approximation error with respect to the original matrix among all reconstructions that correspond to measurable functions of the available summary statistics. However, this property is not true for a general co-clustering basis since the optimal reconstruction depends on the structure of the original matrix, which is not available during the reconstruction process. For example, if the original matrix admits a perfect additive decomposition with respect to some coclustering basis, for example, $Z = E[Z|\hat{U}] + E[Z|\hat{V}] - E[Z]$ for basis $\mathcal{C}_1$, then the "best" reconstruction among all measurable functions of the conditional expectation statistics is given by this additive decomposition itself irrespective of the choice of the Bregman divergence. From Table 1, one can readily see that this solution is different from the MBI solution for I-divergence and basis $\mathcal{C}_1$ and in fact, it is different from the MBI solution for all Bregman divergences other than squared Euclidean distance. Therefore, instead of seeking the optimal reconstruction from the class of all measurable functions of the available summary statistics, we focus on a special class of approximations that correspond to additive models over the summary statistics.

Let $\mathcal{S}_B$ denote the set of all matrices Z'' whose inverse image under $\nabla\phi$ can be written as an additive model over the summary statistics, that is,

$$\mathcal{S}_B = \left\{ Z'' \,\middle|\, Z'' = (\nabla\phi)^{-1}\left(\sum_{r=1}^{s} g_r(E[Z|\mathcal{G}_r])\right)\right\}, \tag{21}$$

where $\{g_r\}_{r=1}^s$ are arbitrary functions measurable with respect to $\{\mathcal{G}_r\}_{r=1}^s$. Note that unlike $\mathcal{S}_A$, the set $\mathcal{S}_B$ explicitly depends on the choice of the convex function ϕ. The reader may wish to compare (9) for block average co-clustering with (21). Since $\mathcal{S}_B$ is defined in terms of arbitrary measurable functions, when there is only one conditional expectation to be preserved as in the case of block average co-clustering, $\mathcal{S}_B$ turns out to be the set of all possible measurable functions of that conditional expectation.

Interestingly, S_B can be alternatively understood from the perspective of Lagrange duality for Bregman divergences. Following Della Pietra et al. (2001), consider the Bregman projection problem of minimizing $d_\phi(p, q_0)$ over $p \in \mathbb{R}^d$ such that p lies in a linear subspace determined by π and a set of features $F = \{f_j, [j]_1^J\}$, that is, $\langle p, f_j \rangle = \langle \pi, f_j \rangle, [j]_1^J$. The dual of the problem turns out to be one of minimizing $d_\phi(\pi, q)$ over q, where q belongs to the dual space determined by q_0, feature set F, and Lagrange multipliers λ. Della Pietra et al. (2001) give a complete characterization of the dual space as a Legendre-Bregman projection family $Q(q_0, F)$ of approximations q. By generalizing their analysis, one can show that S_B is the Legendre-Bregman projection family corresponding to the set of linear constraints determined by S_A. Therefore, the Bregman duality and projection results of Della Pietra et al. (2001) also apply to our setting. Related analyses have appeared in the literature in the context of incremental learning of generalized entropy functionals (Lafferty, 1999), convergence analysis of boosting algorithms (Collins et al., 2000), and game theoretic interpretation of Bayesian decision theory (Grünwald and Dawid, 2004). However, we present our analysis using co-clustering semantics for ease of exposition. Further, our analysis leads to simpler proofs compared to the general setting of Della Pietra et al. (2001).

Example 1.G (I-Divergence) When $\phi(z) = z \log z - z$, the Legendre transformation or the gradient mapping turns out to be log-transformation, that is, $\nabla\phi(z) = \log z$ so that addition in the natural parameter space corresponds to multiplication in the original expectation parameter space and generalized additive models in the natural parameter space lead to generalized multiplicative models. The set $\mathcal{S}_B$ in this case can, therefore, be characterized as the set of all reconstructions corresponding to generalized multiplicative models, or in other words, products of arbitrary functions of the conditional expectations, that is,

$$\mathcal{S}_B = \left\{ Z'' \,\middle|\, Z'' = \prod_{r=1}^{s} g_r(E[Z|\mathcal{G}_r]) \right\},$$

where $\{g_r(\cdot)\}_{r=1}^s$ are arbitrary functions measurable with respect to $\{\mathcal{G}_r\}_{r=1}^s$.

Example 2.G (Squared Euclidean Distance) When $\phi(z) = z^2$, the Legendre transformation or the gradient mapping is the identity transformation, that is, $\nabla\phi(z) = z$ so that natural parameter space is identical to the original space. Therefore, $\mathcal{S}_B$ is just the set of all reconstructions corresponding to generalized additive models, or in other words, additive combinations of arbitrary functions of the conditional expectations, that is,

$$\mathcal{S}_B = \left\{ Z'' \,\middle|\, Z'' = \sum_{r=1}^{s} g_r(E[Z|\mathcal{G}_r]) \right\},$$

where $\{g_r(\cdot)\}_{r=1}^s$ are arbitrary functions measurable with respect to $\{\mathcal{G}_r\}_{r=1}^s$.

Among this class of reconstructions $\mathcal{S}_B$, let $\hat{Z}_B$ be the best approximation to Z in terms of Bregman divergence, that is,

$$\hat{Z}_B = \operatorname*{argmin}_{Z'' \in \mathcal{S}_B} E[d_\phi(Z, Z'')] \ . \tag{22}$$

As Corollary 2 below shows, the best reconstruction $\hat{Z}_B$ among all elements of $\mathcal{S}_B$ is exactly identical to $\hat{Z}_A$, that is, the MBI solution among all elements of $\mathcal{S}_A$, which preserve the relevant conditional expectations. In order to arrive at this result, we make use of a projection theorem (Theorem 5) that characterizes the backward and forward Bregman projections of elements of $\mathcal{S}_B$ onto the set $\mathcal{S}_A$ and vice versa. This projection theorem, in turn, readily follows from the observation (Lemma 7) that the expected Bregman divergence between any $Z' \in \mathcal{S}_A$ and any $Z'' \in \mathcal{S}_B$ follows a Pythagorean decomposition involving the MBI solution $\hat{Z}_A$, that is, it can be expressed as the sum of expected Bregman divergences between the pairs $(Z', \hat{Z}_A)$, and $(\hat{Z}_A, Z'')$.

Lemma 7 *For any $Z' \in \mathcal{S}_A$ as in (17) and any $Z'' \in \mathcal{S}_B$ as in (21) and $\hat{Z}_A$ as in (18)*

$$E[d_\phi(Z', Z'')] = E[d_\phi(Z', \hat{Z}_A)] + E[d_\phi(\hat{Z}_A, Z'')].$$

A proof of the above lemma is given in Appendix D.[10] Using Lemma 7, we can now obtain the following projection theorem, which states that the MBI solution is the forward Bregman projection of any element of $\mathcal{S}_A$ onto the set $\mathcal{S}_B$ and the backward Bregman projection of any element of $\mathcal{S}_B$ onto the set $\mathcal{S}_A$.

Theorem 5 (Projection Theorem) *For any $Z' \in \mathcal{S}_A$ as in (17) and any $Z'' \in \mathcal{S}_B$ as in (21) and $\hat{Z}_A$ as in (18), the following two statements hold true:*

(a) $\hat{Z}_A = \operatorname*{argmin}_{Z' \in \mathcal{S}_A} E[d_\phi(Z', Z'')], \ \ \forall Z'' \in \mathcal{S}_B,$

(b) $\hat{Z}_A = \operatorname*{argmin}_{Z'' \in \mathcal{S}_B} E[d_\phi(Z', Z'')], \ \ \forall Z' \in \mathcal{S}_A.$

Since the original Z is also an element of $\mathcal{S}_A$, we observe that $\hat{Z}_A$ is the forward Bregman projection of Z onto $\mathcal{S}_B$, which leads to the equivalence between $\hat{Z}_A$ and $\hat{Z}_B$, which is the best reconstruction in $\mathcal{S}_B$.

Corollary 2 *For $\hat{Z}_A$ and $\hat{Z}_B$ given by (18) and (22), we have*

$$\hat{Z}_A = \hat{Z}_B \equiv \hat{Z} \ .$$

Proof Follows from the definition of $\hat{Z}_B$ and the projection theorem (Theorem 5). ∎

Corollary 2 gives a concrete justification for the use of the minimum Bregman information solution as the best matrix reconstruction for a given co-clustering since it is the optimum approximation among a large class of possible reconstructions obtained from the summary statistics. It is

10. The result can be derived by an application of Della Pietra et al. (2001, Proposition 3.2). We give a different proof, appropriate for the co-clustering setting.

straightforward to see that the corresponding result for block average co-clustering (Corollary 1) is a special case of this result. This equivalence result is also closely related to Csiszar's axiomatic justification (Csiszár, 1991) of the least squares and maximum entropy principles for linear inverse problems based on *sum consistency* and *product consistency* respectively. More specifically, the sum and product consistency conditions along with certain regularity, locality and fixed point assumptions[11] restrict the best reconstruction $\hat{Z}$ to generalized additive and multiplicative combinations of the observed linear functionals (i.e., conditional expectations in our case) respectively. Hence, the best approximation $\hat{Z} \in \mathcal{S}_B$ where $\mathcal{S}_B$ is defined as in Examples 1.G and 2.G. On the other hand, the constraint of preserving the observed linear functionals (i.e., conditional expectations) ensures that $\hat{Z} \in \mathcal{S}_A$ as well. Since $\mathcal{S}_A \bigcap \mathcal{S}_B = \{\hat{Z}\}$, it follows that $\hat{Z}$ is the MBI solution itself. In particular, the best reconstruction satisfying sum consistency is the least squares solution while the one satisfying product consistency is the maximum entropy solution.

Example 1.H (I-divergence) From Example 1.E, we observe that when $\phi(z) = z\log z - z$, the MBI solution $\hat{Z}_A$ is identical to the maximum entropy solution that preserves the conditional expectations. Further from Example 1.G, we note that the set $\mathcal{S}_B$ consists of generalized multiplicative combinations of the conditional expectations. Hence, from the projection theorem, it follows that the maximum entropy solution is the only generalized multiplicative solution that preserves the relevant conditional expectations. It is also the best reconstruction of Z (or any other $Z' \in \mathcal{S}_A$) among all multiplicative combinations of arbitrary functions of the conditional expectations.

Example 2.H (Squared Euclidean Distance) From Example 2.E, we observe that when $\phi(z) = z^2$, the MBI solution $\hat{Z}_A$ is identical to the standard least squares solution that preserves the conditional expectations. Further from Example 2.G, we note that the set $\mathcal{S}_B$ consists of generalized additive combinations of the conditional expectations. Hence, from the projection theorem, it follows that the least squares solution is the only generalized additive solution that preserves the relevant conditional expectations. It is also the best reconstruction of Z (or any other $Z' \in \mathcal{S}_A$) among all additive combinations of arbitrary functions of the conditional expectations.

5. A Meta Algorithm

In this section, we shall develop an alternating minimization scheme for the general Bregman co-clustering problem (20). Our scheme shall serve as a *meta algorithm* from which a number of special cases (both previously known and unknown) can be derived.

Throughout this section, let us suppose that the underlying measure w, the Bregman divergence d_ϕ, the data matrix $\mathbf{Z}$, number of row clusters k, number of column clusters l, and the co-clustering basis $\mathcal{C}$ are specified and fixed.

5.1 Intuition and Plan of Attack

We first outline the essence of our meta algorithm.

Step 1: Start with an arbitrary row and column clustering, say, (ρ^0, γ^0). Set $t = 0$.

Step 2: Repeat either of the following steps till convergence:

11. Please refer to Csiszár (1991) for details.

Step 2A: With respect to co-clustering (ρ^t, γ^t), compute the matrix approximation $\hat{Z}^t$ by solving the MBI problem (18).

Step 2B: Hold the column clustering γ^t fixed, and find a better row co-clustering, say, ρ^{t+1}. Set $\gamma^{t+1} = \gamma^t$. Set $t = t+1$.

Step 2C: Hold the row clustering ρ^{t+1} fixed, and find a better column co-clustering, say, γ^{t+1}. Set $\rho^{t+1} = \rho^t$. Set $t = t+1$.

We shall prove that this meta algorithm converges in a finite number of steps to a local minima.[12] As is clear from the outline above, a key step in our algorithm will involve finding a solution of the MBI problem (18). Further, since the number of possible row (or column) clusterings is exponential in the number of rows (or columns), it is also essential to have an efficient means for determining the best row (or column) clustering for a fixed choice of the column (or row) clustering and the MBI solution. Fortunately for the co-clustering problem, the expected distortion measure that quantifies the quality of a row (or column) clustering admits a separability property that allows independent optimal updates of the cluster assignments of every row (or column). We discuss this property in more detail below.

5.2 A Separability Property

We begin by considering the quality of a candidate row (or column) clustering ρ in Step 2B (or step 2C) for a fixed choice of column (or row) clustering and MBI solution parameters. Since our objective is to obtain an accurate reconstruction of the original matrix, a natural choice is to consider the expected Bregman distortion between the original Z and a reconstruction $\tilde{Z}$ based on the row (or column) clustering ρ while keeping everything else fixed. To characterize this reconstruction, we employ the functional form for the MBI solution $\hat{Z}$ given in Theorem 4. In general, the formula in Theorem 4 provides a unique reconstruction $\tilde{Z}$ for any set of Lagrange multipliers Λ (not necessarily optimal) and (ρ, γ), since $\nabla\phi(\cdot)$ is a monotonic, hence invertible, function (Azoury and Warmuth, 2001; Banerjee et al., 2005b). To underscore the dependence of $\tilde{Z}$ on the Lagrange multipliers, we shall use the notation $\tilde{Z} = \zeta(\rho, \gamma, \Lambda) = (\nabla\phi)^{-1}(\nabla\phi(E[Z]) - \sum_{r=1}^{s} \Lambda_{\mathcal{G}_r}/w_{\mathcal{G}_r})$. The quality of a candidate row (or column) clustering can now be quantified in terms of the accuracy of the corresponding $\tilde{Z}$ where the other two arguments, that is, the column (or row) clustering and Lagrange multipliers are fixed. In particular, $\hat{Z} = \zeta(\rho, \gamma, \Lambda^*)$ is the approximation corresponding to the optimal Lagrange multipliers Λ^*.

Given a set of (not necessarily optimal) Lagrange multipliers Λ, we now consider updating the current co-cluster assignments (ρ, γ) in order to improve the current approximation $\tilde{Z} = \zeta(\rho, \gamma, \Lambda)$. Although $\tilde{Z}$ looks complex, the fact that $\nabla\phi$ is a one-one invertible function ensures that each element $\tilde{z}_{uv}$ in the matrix $\tilde{\mathbf{Z}}$ corresponding to $\tilde{Z}$ depends only on $(u, \rho(u), v, \gamma(v))$ for a given Λ. Hence, for any given Λ, there exists a function ξ such that the point-wise distortion $d_\phi(z_{uv}, \tilde{z}_{uv})$ can be expressed as $\xi(u, \rho(u), v, \gamma(v))$, that is, it depends only on the corresponding row/column and cluster assignments. Since the expected distortion $E[d_\phi(Z, \tilde{Z})]$ is weighted sum of the point wise distortions, it satisfies a nice separability property that allows the current row (or column) assignments to be efficiently updated. In particular, for any given Λ, the expected distortion $E[d_\phi(Z, \tilde{Z})]$ can be expressed

12. In fact, any ordering of Steps 2B and 2C gives the same guarantees. Alternatively, one can run Steps 2A and 2C for some iterations followed by Steps 2A and 2B. We will establish that each step can only improve the quality of the current approximation. Hence, any ordering is sufficient to reach a local minimum.

as the sum of contributions from the rows (or columns) where each row (or column) contribution only depends on the row and its current cluster assignment. Note that this separability property is similar to that of the kmeans objective function, which can be also be expressed as the sum of terms corresponding to each point and its cluster assignment. As in the case of kmeans, the separability property allows independent updates of the cluster assignments of every row (or column). Further, for a fixed Λ and γ, since the total approximation error is the sum over the approximation errors due to each row (or column) and its cluster assignment, greedy cluster assignments of the individual rows result in a globally optimal row clustering ρ for the given Λ and γ. An equivalent statement is true for column assignments for a given Λ and ρ. The following lemma formally states this separability property. The proof simply follows from definitions, and is hence omitted.

Lemma 8 *For a fixed co-clustering* (ρ,γ) *and a fixed set of (not necessarily optimal) Lagrange multipliers* Λ*, and* $\tilde{Z} = \zeta(\rho,\gamma,\Lambda)$*, we can write:*

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

where $\xi(U,\rho(U),V,\gamma(V)) = d_\phi(Z,\tilde{Z})$.

5.3 Updating Row and Column Clusters

We will now present the details of our plan in Section 5.1. First, we will demonstrate how to update row clustering (or column clustering) with respect to a fixed column clustering (or row clustering) and a fixed set of Lagrange multipliers. Then, we will find the optimal Lagrange multipliers corresponding to the minimum Bregman solution of the updated co-clustering.

Suppose we are in Step 2A outlined in Section 5.1. Updating the row clustering keeping the column clustering and the Lagrange multipliers fixed leads to a new value for the Bregman co-clustering objective function. Now making use of the separability property in Lemma 8, we can efficiently optimize the contribution of each row assignment to the overall objective function to obtain the following row cluster update step.

Lemma 9 *Let* ρ^{t+1} *be defined as*

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

and let $\tilde{Z}^t = \zeta(\rho^{t+1},\gamma^t,\Lambda^{*t})$*. Then,*

$$E[d_\phi(Z,\tilde{Z}^t]) \leq E[d_\phi(Z,\hat{Z}^t)].$$

where $\hat{Z}^t = \zeta(\rho^t,\gamma^t,\Lambda^{*t})$.

A similar argument applies to step 2B where we seek to update the column clustering keeping the row clustering fixed.

Lemma 10 *Let* γ^{t+1} *be defined as*

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

and let $\tilde{Z}^t = \zeta(\rho^t, \gamma^{t+1}, \Lambda^{*t})$. *Then,*

$$E[d_\phi(Z, \tilde{Z}^t)] \leq E[d_\phi(Z, \hat{Z}^t)].$$

where $\hat{Z}^t = \zeta(\rho^t, \gamma^t, \Lambda^{*t})$.

We now consider step 2C. So far, we have only considered updating the row (or column) assignments keeping the Lagrange multipliers fixed. After such updates, the approximation $\tilde{Z}^t = \zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda^{*t})$ is closer to the original matrix Z than the earlier minimum Bregman information solution $\hat{Z}^t$, but the Lagrange multipliers Λ^{*t} are not optimal, in general. In other words, the approximation $\tilde{Z}^t$ is not a minimum Bregman information solution. For the given co-clustering $(\rho^{t+1}, \gamma^{t+1})$, let Λ^{*t+1} be the optimal Lagrange multipliers for the corresponding minimum Bregman information problem in (18). The corresponding matrix approximation $\hat{Z}^{t+1} = \zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda^{*t+1})$ is a minimum Bregman information solution. As the following lemma shows, this approximation is better than the current approximation $\tilde{Z}^t$.

Lemma 11 *Let* $\hat{Z}^{t+1} = \zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda^{*t+1})$ *be the minimum Bregman information solution corresponding to* $(\rho^{t+1}, \gamma^{t+1})$ *with* Λ^{*t+1} *being the optimal Lagrange multipliers for (18). Then,*

$$E[d_\phi(Z, \hat{Z}^{t+1}) \leq E[d_\phi(Z, \tilde{Z}^t)],$$

where $\tilde{Z}^t = \zeta(\rho^{t+1}, \gamma^{t+1}, \Lambda^{*t})$.

Proof By definition,

$$\begin{aligned}
E[d_\phi(Z, \hat{Z}^{t+1})] &= E[\phi(Z) - \phi(\hat{Z}^{t+1}) - \langle Z - \hat{Z}^{t+1}, \nabla\phi(\hat{Z}^{t+1})\rangle] \\
&\overset{(a)}{=} E[\phi(Z) - \phi(\hat{Z}^{t+1})] \\
&= E[d_\phi(Z, \tilde{Z}^t)] - E[d_\phi(\hat{Z}^{t+1}, \tilde{Z}^t)] - E[\langle Z - \hat{Z}^{t+1}, \nabla\phi(\tilde{Z}^t)\rangle] \\
&\overset{(b)}{=} E[d_\phi(Z, \tilde{Z}^t)] - E[d_\phi(\hat{Z}^{t+1}, \tilde{Z}^t)] \\
&\leq E[d_\phi(Z, \tilde{Z}^t)] \,,
\end{aligned}$$

where (a) follows since $\hat{Z}^{t+1}$ belongs to both Γ_A and Γ_B so that taking conditional expectations over $E[Z|\mathcal{G}], \mathcal{G} \in \mathcal{C}$ makes the last term zero and (b) follows since $\nabla\phi(\tilde{Z}^t)$ is a summation of terms involving $E[Z]$ and $\Lambda_{\mathcal{G}_r}, [r]_1^s$, and $E[\hat{Z}^{t+1}|\mathcal{G}_r] = E[Z|\mathcal{G}_r]$, thus making the last term vanish. ■

5.4 The Algorithm

The meta algorithm for generalized Bregman co-clustering (see Algorithm 2) is a concrete "implementation" of our plan in Section 5.1. Comparing this algorithm with the solution for block average co-clustering (Algorithm 1), one can readily see that both the algorithms are based on an identical alternate minimization strategy and Algorithm 1 is in fact a special case of Algorithm 2 when the MBI solution corresponds to the co-cluster means. We now establish that our algorithm is guaranteed to achieve local optimality.

Algorithm 2 Bregman Co-clustering Algorithm

Input: Matrix $\mathbf{Z} \subseteq S^{m\times n}$, probability measure w, Bregman divergence $d_\phi : S \times \text{int}(S) \mapsto \mathbb{R}_+$, num. of row clusters l, num. of column clusters k, co-clustering basis $\mathcal{C}$.
Output: Co-clustering (ρ^*, γ^*) that (locally) optimize the objective function in (20).
Method:

{**Initialize** ρ, γ }
Start with an arbitrary co-clustering (ρ, γ)
repeat
{**Step A: Update Minimum Bregman Information Solution** (Λ^*)}
$\Lambda^* \leftarrow \underset{\Lambda}{\text{argmax}}\ L(\Lambda)$ where $L(\cdot)$ is Lagrange dual of the MBI problem (18).
{**Step B: Update Row Clusters** (ρ)}
for $u = 1$ to m **do**
$\rho(u) \leftarrow \underset{g:[g]_1^k}{\text{argmin}}\ E_{V|u}[\xi(u, g, V, \gamma(V))]$
where $\xi(U, \rho'(U), V, \gamma(V)) = d_\phi(Z, \tilde{Z}),\ \ \tilde{Z} = \zeta(\rho', \gamma, \Lambda^*)$ for any ρ'
end for
{**Step C: Update Column Clusters** (γ)}
for $v = 1$ to n **do**
$\gamma(v) \leftarrow \underset{h:[h]_1^l}{\text{argmin}}\ E_{U|v}[\xi(U, \rho(U), v, h)]$
where $\xi(U, \rho(U), V, \gamma'(V)) = d_\phi(Z, \tilde{Z}),\ \ \tilde{Z} = \zeta(\rho, \gamma', \Lambda^*)$ for any γ'
end for
until *convergence*
return (ρ, γ)

Theorem 6 *The general Bregman co-clustering algorithm (Algorithm 2) converges to a solution that is locally optimal for the Bregman co-clustering problem (20), that is, the objective function cannot be improved by changing either the row clustering, the column clustering or the Lagrange multipliers.*

Proof From Lemmas 9, 10, and 11, it follows that updating the row clustering ρ, the column clustering γ and the Lagrange multipliers Λ one at a time decreases the objective function of the Bregman co-clustering problem. Hence, the Bregman co-clustering algorithm (Algorithm 2) which proceeds by alternately updating $\rho \rightarrow \gamma \rightarrow \Lambda$ monotonically decreases the Bregman co-clustering objective function. Since the number of distinct co-clusterings is finite, the algorithm is guaranteed to converge to a locally optimal solution. ■

Note that updating Λ is the same as obtaining the MBI solution. When the Bregman divergence is I-divergence or squared deviation, the minimum Bregman information problem has an analytic closed form solution as shown in Tables 1 and 2. Hence, it is straightforward to obtain the row and column cluster update steps and implement these Bregman co-clustering algorithms. The resulting algorithms involve computational effort that is linear per iteration in the size of the data and are hence scalable. In general, the MBI problem has a unique solution since it involves a strictly convex objective function and linear constraints. However, the solution need not have a closed form and has to be obtained numerically using iterative projection algorithms, which in turn involves solving non-

linear systems of equations. In the general case, the Bregman co-clustering algorithm will include such iterative projection procedures as a sub-routine.

Details on several different instantiations of the meta-algorithm (using matrix notation) for certain specific choices of Bregman divergences and co-clustering bases are given in Appendix E. In particular, exact algorithms for (i) basis $\mathcal{C}_2$ and all Bregman divergences, (ii) Euclidean distance and all co-clustering bases, and (iii) I-divergence and all co-clustering bases have been worked out. The MBI problem has a closed form solution in all of the above three cases. Further, as a representative of the general case of arbitrary Bregman divergences and co-clustering bases, we show an instantiation of the meta-algorithm to Itakura-Saito distance, for which the MBI problem does not have a closed form solution.

5.5 Iterative Algorithms for the Minimum Bregman Information Problem

An important part of the Bregman co-clustering algorithm involves solving the MBI problem. While there are closed form solutions for some important choices of Bregman divergences and summary statistics, the general case leads to a convex programming problem and does not have a closed form solution. In this section, we discuss two simple iterative algorithms to solve the MBI problem. The first one is *Bregman's algorithm* (Bregman, 1967; Censor and Zenios, 1998) and the second is an *iterative scaling* method (Della Pietra et al., 2001).

Recall that the MBI solution $\hat{Z}$ for a co-clustering basis $\mathcal{C}$ is given by

$$\hat{Z} = \operatorname*{argmin}_{Z' | E[Z'|C]=E[Z|C],\ \forall C\in\mathcal{C}} E[d_\phi(Z', E[Z'])] \ .$$

For notational convenience, let $\mathbf{z}$, $\mathbf{z}'$ and $\bar{\mathbf{z}}$ denote vectorized versions of the original matrix $\mathbf{Z}$, the tentative solution matrix $\mathbf{Z}'$, and a constant matrix consisting of the expectation $E[Z]$ respectively. Then $\mathbf{z}, \mathbf{z}'$ and $\bar{\mathbf{z}}$ are all vectors of dimension mn. Let $\mathbf{A}$ denote the $c \times mn$ matrix corresponding to the linear constraints $E[Z'|\mathcal{G}] = E[Z|\mathcal{G}],\ \forall \mathcal{G} \in \mathcal{C}$, where c is the total number of constraints, so that the constraints can be written as $\mathbf{A}\mathbf{z}' = \mathbf{A}\mathbf{z}$. The vectorized version $\hat{\mathbf{z}}$ of the MBI solution can now be written as

$$\hat{z} = \operatorname*{argmin}_{\mathbf{z}' | \mathbf{A}\mathbf{z}'=\mathbf{A}\mathbf{z}} \sum_{\iota=1}^{mn} w_\iota d_\phi(z'_\iota, \bar{z}_\iota). \tag{23}$$

Since a convex combination of Bregman divergences is again a Bregman divergence, the objective function in (23) can be readily expressed as the Bregman divergence between the vectors $\mathbf{z}'$ and $\bar{\mathbf{z}}$ derived from the convex function $\phi_w(\mathbf{z}') = \sum_{\iota=1}^{mn} w_\iota \phi(z'_\iota)$, that is,

$$\hat{z} = \operatorname*{argmin}_{\mathbf{z}' | \mathbf{A}\mathbf{z}'=\mathbf{A}\mathbf{z}} d_{\phi_w}(\mathbf{z}', \bar{\mathbf{z}}).$$

Since ϕ_w is the convex function induced on the vectorized matrices by the original convex function ϕ, we ignore this distinction and use ϕ to denote ϕ_w as well when it is clear that the function is being applied to matrices.

5.5.1 BREGMAN'S ALGORITHM (BREGMAN, 1967)

Bregman's algorithm requires that the initial guess $\mathbf{z}'_0$ belong to the set $\{\mathbf{z}' | \mathbf{z}' \in \text{int}(\text{dom}(\phi)),\ \nabla\phi(\mathbf{z}') = \mathbf{A}^T\mathbf{x}, \mathbf{x} \in \mathbb{R}^c\}$. The unconstrained global optimum $\mathbf{z}_*$ belongs to this set since $\nabla\phi(\mathbf{z}_*) = \mathbf{0}$ which is

$\mathbf{A}^T\mathbf{x}$ for $\mathbf{x} = \mathbf{0} \in \mathbb{R}^c$. Hence, we use $\mathbf{z}_*$ as the initial guess, that is,

$$\mathbf{z}'_0 \;=\; \mathbf{z}_* \;. \tag{24}$$

Subsequent iterative updates are obtained by solving the following set of equations:

$$\nabla\phi(\mathbf{z}'^{t+1}) \;=\; \nabla\phi(\mathbf{z}'^{t}) + \lambda A_i^T \;, \tag{25}$$

$$A_i\,\mathbf{z}'^{t+1} \;=\; A_i\,\mathbf{z} \;, \tag{26}$$

where A_i is the i^{th} row of $\mathbf{A}$ and $\lambda \in \mathbb{R}$. The solution to the above set of equations can be considered as the Bregman projection of the current tentative solution $\mathbf{z}'^t$ onto the hyperplane $\{\mathbf{z}' | A_i\mathbf{z}' = A_i\mathbf{z}\}$. Due to the strict convexity of ϕ, the update equations, under proper regularity conditions (Bregman, 1967), uniquely determine $\mathbf{z}'^{t+1}$ and λ. However, the equations are non-linear and one needs to use appropriate numerical techniques to solve for $\mathbf{z}'^{t+1}$.

The update equations (25) and (26) are based on only one linear constraint. For convergence to the optimum, the updates must touch upon all the constraints following a schedule known as relaxation control (Bregman, 1967; Bauschke and Borowein, 1997). For simplicity, we consider updates based on a cyclic ordering of the constraints, where all constraints are considered one after the other. The cyclic ordering schedule is sufficient to guarantee convergence to the optimum solution, although more general schedules are admissible (Bauschke and Borowein, 1997).

5.5.2 ITERATIVE SCALING (DELLA PIETRA ET AL., 2001)

We now discuss an auxiliary function-based iterative scaling method to solve the problem. The method makes use of the Legendre-Bregman projection $\mathcal{L}_\phi(\mathbf{z}'^t, \mathbf{A}^T\lambda)$, which is the "backward" Bregman projection of $\mathbf{z}'^t$ onto the hyperplane determined by $\{\mathbf{z}' | \mathbf{z}'^T\mathbf{A}^T\lambda = \mathbf{z}^T\mathbf{A}^T\lambda\}$, so that

$$\begin{aligned} \mathbf{z}'^{t+1} \;&=\; \mathcal{L}_\phi(\mathbf{z}', \mathbf{A}^T\lambda) \;=\; (\nabla\phi)^{-1}(\nabla\phi(\mathbf{z}') + \mathbf{A}^T\lambda) \\ \Rightarrow \nabla\phi(\mathbf{z}'^{t+1}) \;&=\; \nabla\phi(\mathbf{z}') + \mathbf{A}^T\lambda \;. \end{aligned} \tag{27}$$

The similarity between the Legendre-Bregman projection as in (27) and the first update equation (25) is due to the fact that both are Bregman projections of a point onto a hyperplane. However, Bregman's algorithm considers one constraint at a time, whereas iterative scaling works with all the constraints simultaneously.

As before, we set the initial guess $\mathbf{z}'_0 = \mathbf{z}_*$. Using the constraint matrix $\mathbf{A}$, we select $N_j \geq \sum_{i=1}^{c} A_{ij}$ for $j = 1, \ldots, mn$. Then, the iterative update of the tentative solution is given by (27), where $\lambda \in \mathbb{R}^c$ and each component λ_i satisfies

$$\sum_{j=1}^{mn} A_{ij}\; \mathcal{L}_\phi(z'^t{}_j,\; s_{ij}N_j\lambda_i) \;=\; A_i\mathbf{z} \;, \tag{28}$$

where $s_{ij} = \text{sign}(A_{ij})$ and ϕ operates on the matrix elements.

As before, the system of equations (27) and (28) is non-linear and one needs to use proper numerical methods to obtain the updates. However, there is an important difference between the iterative scaling updates and the updates of Bregman's algorithm. Since (28) is in terms of each component of λ, one can obtain λ entirely from (28). This λ can then be used in (27) to get $\mathbf{z}'^{t+1}$. In other words, analogous to the EM algorithm, iterative scaling allows one to alternate updates to λ

and $\mathbf{z}'$ till convergence. This is not possible in case of Bregman's algorithm where both the equations (25) and (26) have to be solved simultaneously. Note that both the algorithms require regularity conditions to guarantee convergence. The reader is referred to the original papers (Bregman, 1967; Della Pietra et al., 2001) for details.

6. Experiments

In recent years, co-clustering has been successfully applied to a number of application domains such as text mining (Dhillon et al., 2003b; Gao et al., 2005; Takamura and Matsumoto, 2003), image and video analysis (Zhong et al., 2004; Qiu, 2004; Guan et al., 2005; Cai et al., 2005), natural language processing (Freitag, 2004; Rohwer and Freitag, 2004; Li and Abe, 1998), bio-informatics (Cheng and Church, 2000; Cho et al., 2004; Kluger et al., 2003) as well as other applications (Carrasco et al., 2003). In particular, there exist a number of empirical studies that illustrate the usefulness of particular instances of the Bregman co-clustering framework that we describe in this paper. Hence, instead of extensively evaluating our methodology on various application domains, we present a brief summary of existing experimental results. Further, we present a comparative empirical study of the different co-clustering bases as well as the divergences discussed in this paper. Finally, we highlight new applications such as missing value prediction and co-clustering of matrices with categorical elements.

6.1 Existing Applications and Results

In this section, we present a brief overview of some of the existing applications of co-clustering.

6.1.1 TEXT CLUSTERING

Text clustering is one of the first domains where a special case of the Bregman co-clustering algorithm, namely the information-theoretic co-clustering algorithm based on I-divergence and basis $\mathcal{C}_5$, has been successfully applied. The key task in text clustering is to identify document clusters. Since most of the information in a document can be captured using a *bag-of-words* model, a convenient vector-space representation is in the form of word-document co-occurrence matrices with documents corresponding to rows and words corresponding to columns. However, it is often difficult to obtain good document clusters by directly clustering the matrix rows due to the inherent sparsity and high dimensionality (i.e., large number of words). Co-clustering, on the other hand, performs an implicit dimensionality reduction by clustering the words and hence, is more effective and efficient for identifying document clusters. Since word-document co-occurrence matrices can be interpreted as estimates of unnormalized joint distribution, an appropriate choice for the loss function is the I-divergence cost used by Dhillon et al. (2003b) and Takamura and Matsumoto (2003). Previous empirical evaluations on some of the popular text data sets (*NG20* and *CLASSIC3*) (Dhillon et al., 2003b) reveal that this choice of co-clustering algorithm provides performance comparable to the best text-clustering algorithms while yielding superior results than single-sided information-theoretic clustering. In particular, there is a significant improvement in the micro-averaged precision values with respect to single-sided clustering; See Dhillon et al. (2003b) for more details.

6.1.2 NATURAL LANGUAGE PROCESSING

Natural language processing is yet another domain where co-clustering has been widely employed as a key intermediate technique for obtaining an informative partitioning of both the language tokens and contexts, which in turn facilitates improved performance on various tasks such as named-entity recognition (Freitag, 2004), automatic construction of lexicon (Rohwer and Freitag, 2004) and prepositional phrase attachment disambiguation (Li and Abe, 1998). In all these applications, the relevant structural information in an unlabeled text corpus can be effectively captured in terms of the distributional properties of appropriately defined language tokens with respect to the contexts in which they occur, for example, k-neighborhood of tokens on either side, verb preceding the token, etc. Hence, one could expect improved performance by leveraging the token-context co-occurrence matrices. However, for most natural language processing applications, the number of tokens and contexts is extremely large, making it infeasible to directly employ computationally intensive learning algorithms. Co-clustering alleviates this problem by producing a highly informative, but reduced cluster-based representation for both tokens and contexts, thus making it possible to incorporate additional information from unlabeled text. As in the case of text clustering, the normalized token-context co-occurrence matrices can be interpreted as a joint distribution and hence, most of the co-clustering methods employed in natural processing applications are based on the KL-divergence loss function, or equivalently, the loss in mutual information using co-clustering basis $\mathcal{C}_5$. Empirical studies (Freitag, 2004; Rohwer and Freitag, 2004; Li and Abe, 1998) demonstrate that the use of co-clustering as an intermediate step makes it straightforward to leverage the additional information in unlabeled repositories and leads to substantial performance improvement for a number of natural language processing applications with negligible manual supervision. In particular, Freitag (2004) shows that including additional features based on co-clustering resulted in better entity recognition accuracy (statistically significant for certain entity types) on the MUC 6 named entity data set, while Li and Abe (1998) demonstrate that predictive methods based on the conditional probabilities derived from co-clustering noun and verb phrases provide better accuracy than state-of-the-art rule-based methods on the prepositional phrase attachment task.

6.1.3 BIO-INFORMATICS

In recent years, co-clustering methods are being increasingly employed for analyzing biological data as well, in particular for studying microarray data consisting of gene expression matrices where rows corresponds to genes and columns correspond to experimental conditions. The fundamental problem in this setting is to identify groups of similar genes and similar conditions based on their expression levels. To address this problem, a number of co-clustering configurations (e.g., overlapping, partitional) and loss functions based on additive and multiplicative models have been proposed (Madeira and Oliveira, 2004). These methods have been shown to be quite effective for identifying highly correlated genes and conditions. In particular, a special case of the Bregman co-clustering (Cheng and Church, 2000; Cho et al., 2004) corresponding to squared loss function and basis $\mathcal{C}_6$ has been shown to provide high quality co-clusters on biological data sets involving a variety of human cancer data sets.

6.1.4 VIDEO/IMAGE/SPEECH CONTENT ANALYSIS

There have also been a number of interesting applications of co-clustering in areas such as video, image and speech content analysis for performing unsupervised categorization of video segments

(Zhong et al., 2004), images (Qiu, 2004; Guan et al., 2005) and auditory scenes (Cai et al., 2005). Each of these settings involves two large sets of entities related to each other through co-occurrence matrices—(i) auditory scenes and audio effects in case of speech content analysis, (ii) fixed length video segments and prototype images for video content recognition, and (iii) images and low level features in case of image recognition. Further, as in the case of text clustering, information-theoretic co-clustering methods based on preserving mutual information effectively handle the sparsity and high dimensionality problems to provide high quality categorization of the dual sets of entities. Empirical results on auditory scene and image categorization show improved classification accuracy as compared to single-sided clustering methods.

6.2 Choice of Bregman Divergence and Co-clustering Basis

We now empirically study the appropriateness of the choice of the Bregman divergence and the co-clustering basis for specific tasks. When the choice of the Bregman divergence and the specified statistics capture the natural structure of the data, it is possible to obtain a more accurate low parameter representation of the original data. To illustrate this idea, we perform co-clustering on synthetic data matrices produced using certain generative models as well as on real-life matrices—(i) word-document matrices encountered in text analysis, and (ii) user-movie rating matrices for recommender systems.

6.2.1 SYNTHETIC DATA MATRICES

First, to study the dependence on the Bregman divergence, we generated multiple (10) sets of three classes of artificial 50×50 matrices $\mathbf{M}_{Euc}, \mathbf{M}_{Idiv}$, and $\mathbf{M}_{IS}$, using generative models corresponding to three different choices of Bregman divergences—squared Euclidean distance, I-divergence, and Itakura-Saito distance. It can be shown that the appropriate generative models in this case respectively correspond to mixtures of Gaussian, Poisson and exponential distributions centered at the co-cluster means.[13] In the generative model, we used 5 row clusters and 5 column clusters. The means of each of the co-clusters were chosen to be identical (all positive values) for all the three classes of matrices. Table 3 shows the results (averaged over 10 sets) of co-clustering these matrices using the Bregman co-clustering algorithms corresponding to the basis $\mathcal{C}_2$ and the three choices of Bregman divergence with $k = l = 5$. In each case, the co-clustering algorithms were run 10 times and the reported quality corresponds to the best run in terms of the objective function. Since the co-clustering objective functions based on the different divergences are not comparable and sometimes not even well-defined,[14] we measure the co-clustering quality in terms of the average of the normalized mutual information (Strehl and Ghosh, 2002) between the clustering and true class labels over both the rows and the columns. The standard-deviations reported in the table correspond to the deviations over multiple sets of matrices. From the table, it is clear that the co-clustering quality (i.e., row and column clustering), as indicated by the normalized mutual information with true labels, is better when the Bregman divergence used in the co-clustering algorithm matches that of the generative model.

13. The reader is referred to Banerjee et al. (2005b) for a connection between Bregman divergences and exponential family distributions. The data sets were generated based on extensions of the results obtained by Banerjee et al. (2005b).
14. For example, I-divergence and Itakura-Saito costs are not defined for approximation matrices with negative values.

	NMI for Co-clustering		
Matrix	**Squared Euclidean Distance**	**I-divergence**	**Itakura-Saito distance**
$\mathbf{M}_{Euc}$	$\mathbf{0.812 \pm 0.029}$	0.685 ± 0.041	0.637 ± 0.044
$\mathbf{M}_{Idiv}$	0.645 ± 0.037	$\mathbf{0.689 \pm 0.035}$	0.621 ± 0.042
$\mathbf{M}_{IS}$	0.586 ± 0.082	0.622 ± 0.047	$\mathbf{0.636 \pm 0.039}$

Table 3: Normalized mutual information (NMI) between the true labels and the clusters obtained using different Bregman divergences, basis $\mathcal{C}_2$ and $k = l = 5$. Results indicate better performance when the Bregman divergence matches the generative model.

Matrix	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$	$\mathcal{C}_5$	$\mathcal{C}_6$
$\mathbf{M}_1$	6.10 ± 0.13	6.02 ± 0.13	5.80 ± 0.15	5.69 ± 0.14	5.40 ± 0.12	4.89 ± 0.10
$\mathbf{M}_2$	22.62 ± 1.81	6.32 ± 0.94	6.15 ± 0.91	6.16 ± 0.95	5.99 ± 0.89	5.12 ± 0.23
$\mathbf{M}_3$	22.39 ± 1.87	12.84 ± 1.06	6.76 ± 1.24	8.82 ± 1.15	6.57 ± 1.03	5.04 ± 0.29
$\mathbf{M}_4$	23.28 ± 1.93	12.98 ± 1.11	8.87 ± 1.04	6.19 ± 0.98	6.42 ± 0.96	5.08 ± 0.31
$\mathbf{M}_5$	24.53 ± 2.08	14.19 ± 1.28	10.31 ± 1.22	11.96 ± 1.18	6.14 ± 0.99	5.29 ± 0.25
$\mathbf{M}_6$	44.41 ± 2.75	33.34 ± 1.79	29.18 ± 2.05	31.26 ± 1.99	25.74 ± 1.26	5.01 ± 0.33

Table 4: Approximation errors on synthetic matrices for different co-clustering bases using squared Euclidean distance and $k = l = 5$. The results indicate that the performance saturates when the complexity of the co-clustering basis matches that of the generative model.

In order to study how the approximation error depends on the choice of co-clustering basis, we created multiple (10) sets of six 50×50 data matrices, $\mathbf{M}_1, \mathbf{M}_2, \ldots,$ and $\mathbf{M}_6$ using generative models based on the Gaussian family, but with increasing levels of complexity corresponding to the various co-clustering bases. This was done by first obtaining the minimum Bregman information approximations of an arbitrary 50×50 matrix corresponding to the various co-clustering bases and then adding Gaussian noise to each of the approximations. We perform Bregman co-clustering on each of these matrices using squared Euclidean distance and $k = l = 5$. Table 4 presents the approximation error obtained for each of these matrices using the various co-clustering bases. From the table, it is clear that for relatively simple matrices such as $\mathbf{M}_1$ and $\mathbf{M}_2$, reasonably low parameter bases such as $\mathcal{C}_1$ or $\mathcal{C}_2$ suffice, whereas for more complex matrices such as $\mathbf{M}_6$, high parameter co-clustering bases such as $\mathcal{C}_6$ are necessary. Figures 3 and 4 show images of the original data matrix $\mathbf{M}_2$ and $\mathbf{M}_6$, and the reconstructions obtained using the different co-clustering bases. The figures reinforce the observation we make from the table. In particular, in Figure 3, one can visually infer that the reconstruction of the matrix $\mathbf{M}_2$ obtained using $\mathcal{C}_2$ is reasonably accurate and cannot be improved much using more complex co-clustering bases whereas, in Figure 4, the reconstruction of $\mathbf{M}_6$ obtained using $\mathcal{C}_6$ is significantly better than that obtained using the other co-clustering bases, thus clearly demonstrating that the choice of co-clustering basis should match the generative model in order to obtain an accurate approximation.

6.2.2 WORD-DOCUMENT MATRICES

As mentioned earlier, co-clustering has been successfully applied to text analysis (Dhillon et al., 2003b). Since several results comparing specific co-clustering schemes to alternative text clustering approaches have already been studied, we focus on the relative performance of the different co-clustering bases introduced in this paper. We use the *CLASSIC3* data set with 3891 documents represented in the bag-of-words model with 4666 words. We fix the number of document clusters to be three, which is the number of document classes in the data set. Figure 5 shows the relative performance (averaged over 10 runs) of all the six co-clustering schemes for a varying number

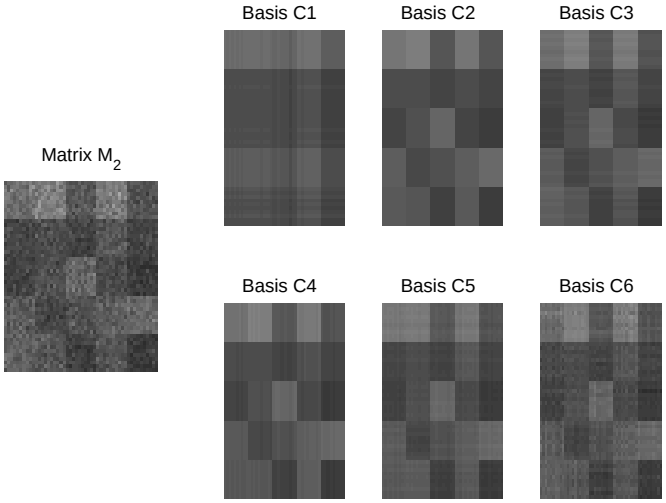


Figure 3: Co-clustering-based approximation of a simple 50×50 matrix $\mathbf{M}_2$ using various co-clustering bases, squared distortion and $k = l = 5$. While the matrix is too complex for $\mathcal{C}_1$, all bases from $\mathcal{C}_2$ onwards get an accurate approximation. Note that all matrices are shown with a consistent permutation (which the co-clustering finds) for easy visual comparison.

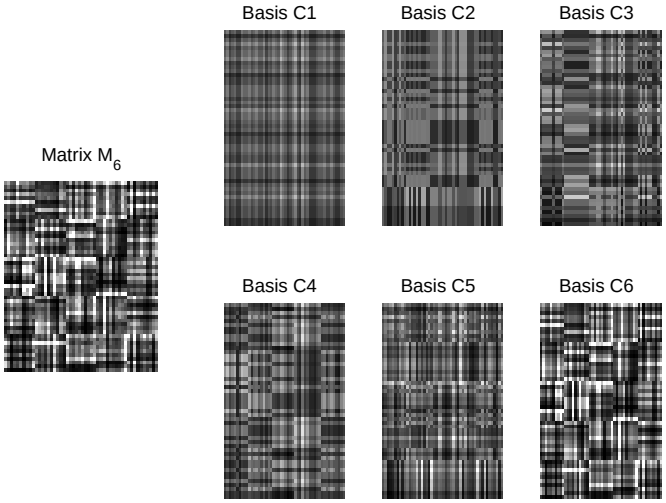


Figure 4: Co-clustering-based approximation of a 50×50 matrix $\mathbf{M}_6$ using various co-clustering bases, squared distortion and $k = l = 5$. Since the given matrix has a fairly complicated structure, only $\mathcal{C}_6$ gets an accurate approximation. All other schemes have more errors, with the simple bases ($\mathcal{C}_1$ and $\mathcal{C}_2$) having high errors. As before, the matrices are consistently permuted for visualization. The co-clustering algorithm also finds this permutation.

of word clusters and for two Bregman divergences—squared Euclidean distance and I-divergence. Performance is evaluated by the normalized mutual information of the document clusters with the true labels of the documents (Strehl and Ghosh, 2002). As in many of the other experiments, we note that co-clustering bases $\mathcal{C}_2$ and $\mathcal{C}_5$ are suitable for both divergences. In Figure 6, we compare the performances of $\mathcal{C}_2$ and $\mathcal{C}_5$ for both divergences, using the spherical k-means (SPKmeans) algorithm

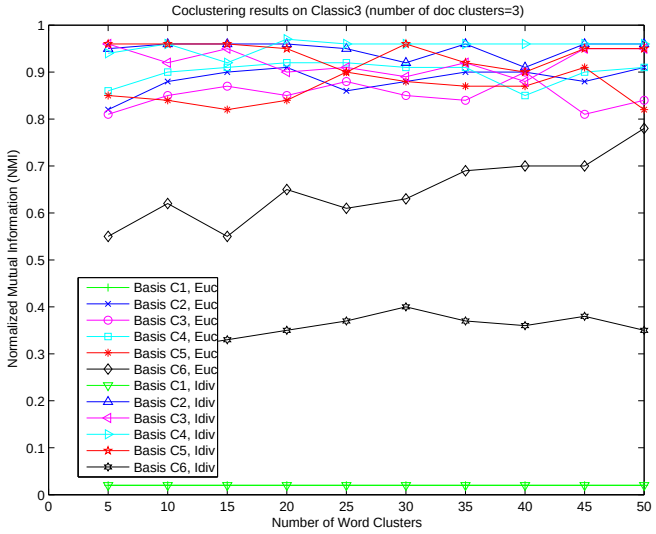


Figure 5: Co-clustering results from *CLASSIC3*—6 bases and 2 divergences. Bases $\mathcal{C}_2 - \mathcal{C}_5$ perform very well in getting back the hidden true labels. Basis $\mathcal{C}_1$ performs the worst as it has access to minimal amount of information. Interestingly, basis $\mathcal{C}_6$, in spite of having the maximal information, performs poorly according to NMI. Possibly $\mathcal{C}_6$ is overfitting, that is, finding some additional structure in the data that goes beyond what is needed to get the labels right. There is no significant difference between the two loss functions used.

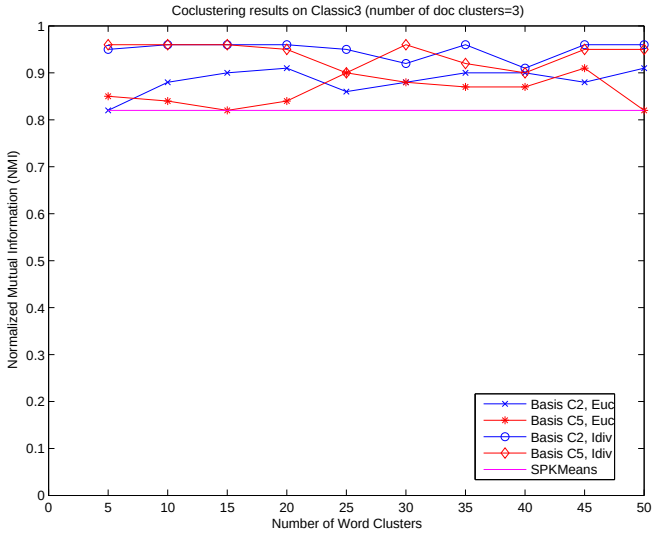


Figure 6: Co-clustering on *CLASSIC3*—Bases $\mathcal{C}_2$ and $\mathcal{C}_5$ using squared Euclidean distance and I-divergence compared with SPKmeans. The co-clustering results compare favorably to SPKmeans.

(Dhillon and Modha, 2001) as a benchmark. We note that the co-clustering algorithms, in particular the ones based on I-divergence, have very good performance for the entire range of word clusters. Our results are in agreement with similar results reported in the literature (Dhillon et al., 2003b).

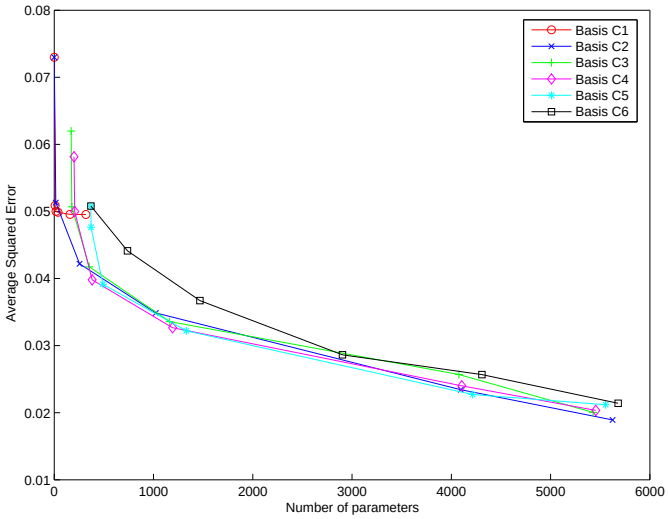


Figure 7: Approximation error (average squared error) on MovieLens data using squared Euclidean distance-based co-clustering. As expected, the error decreases with increasing number of parameters for all bases. For each basis, the number of parameters varies as a function of the number of row and column clusters that the co-clustering algorithm uses.

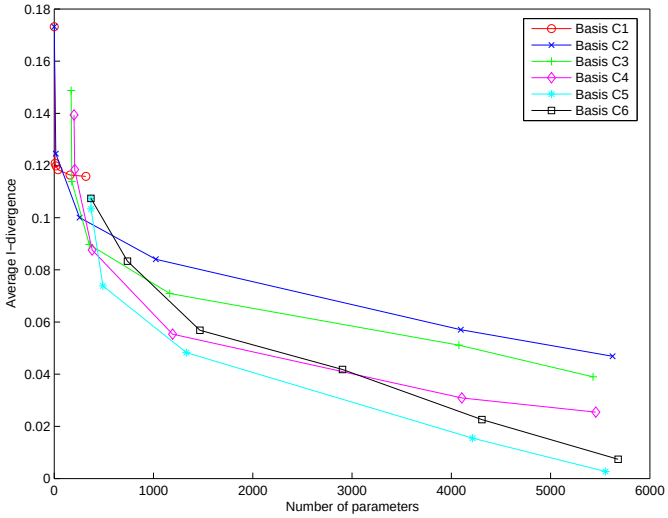


Figure 8: Approximation error (average I-divergence) on MovieLens data using I-divergence-based co-clustering. The error decreases with increasing number of parameters.

Bregman divergence	$k=l=1$	$k=l=2$	$k=l=12$	$k=l=32$	$k=l=64$	$k=l=75$
Squared Euclidean distance	0.7004	0.6816	0.6048	0.5547	0.4451	0.4052
I-divergence	0.7006	0.6824	0.6029	0.5573	0.4492	0.4080

Table 5: Mean absolute error (MAE) for reconstructing MovieLens data (all values) using co-clustering methods based on squared Euclidean distance and I-divergence and co-clustering basis $\mathcal{C}_5$.

6.2.3 USER-MOVIE RATING MATRICES

The other real-life data domain that we studied is that of movie recommender systems. The data matrices in this case consist of user ratings for various movies. For our experiments, we used the MovieLens data set (GroupLens) consisting of 100,000 ratings in the range 0-5 corresponding to 943 users and 1682 movies. To figure out the appropriate divergence and co-clustering basis for this data, we performed experiments using both squared Euclidean distance and I-divergence and various co-clustering bases with varying number of row and column clusters. For each case, the co-clustering was performed assuming uniform weights on the known ratings and zero weights for the unknown ones. The known ratings were then reconstructed using the MBI principle. Figures 7 and 8 show how the approximation error varies with the number of parameters for different co-clustering bases using squared Euclidean distance and I-divergence cost functions respectively. In the case of squared Euclidean distance-based co-clustering, we observe that $\mathcal{C}_2$ provides the best accuracy when an extremely low parameter approximation is required while $\mathcal{C}_2$-$\mathcal{C}_5$ are more suitable for moderately low parameter sizes. In the case of I-divergence-based co-clustering, $\mathcal{C}_5$ is better than the other bases over a wide range of parameter sizes. Further as Table 5 shows, both choices of Bregman divergence, that is, squared Euclidean distance and I-divergence, seem to provide similar performance in terms of the mean absolute error for $\mathcal{C}_5$.

6.3 Novel Applications of Bregman Co-clustering

We now briefly describe two novel applications of our Bregman co-clustering framework and illustrate these with specific real-life examples.

6.3.1 MISSING VALUE PREDICTION

Prediction of missing values is an important task encountered in a number of real-world domains such as recommender systems, bioinformatics, etc. For our experiments, we consider a collaborative filtering-based recommender system where the main task is to predict the preference of a given user for a given item using known preferences of the other users. One of the earliest and most popular approaches to solve this problem is by computing the Pearson correlation of each user with all other users based on the known preferences and predict the unknown rating by proportionately combining all the users' ratings. Based on the observation that the known ratings correspond to elements in a matrix and the missing ratings can be predicted using suitable low parameter approximations of the ratings matrix, a number of other collaborative filtering approaches based on matrix approximation methods such as SVD (Sarwar et al., 2000), and PLSI (Hofmann, 2004) have been proposed in recent years.

Following the same general intuition, we propose a mathematically well-motivated solution based on co-clustering. The main idea is to (i) assume that the ratings matrix has a low parameter structure involving properties of user and item clusters, (ii) deduce the relevant parameters using the

SVD	NNMF	CORR	$\mathcal{C}_2$	$\mathcal{C}_5$
0.7721 ± 0.0164	0.7636 ± 0.0186	0.8214 ± 0.0201	0.8733 ± 0.197	0.7608 ± 0.0211

Table 6: Mean absolute error on MovieLens data set for various collaborative filtering approaches. Number of row and column clusters for co-clustering (based on squared Euclidean distance and basis $\mathcal{C}_5$) and rank of SVD and NNMF is set to 5 and the number of neighbors in the correlation method was set to 50.

available ratings so that the desired loss function is minimized, and (iii) use a matrix reconstruction based on this structure for predicting the missing values. More specifically, in our co-clustering approach, we assume a low parameter structure by using the MBI principle so that the parameter learning can be readily performed using the Bregman co-clustering algorithm with a suitably weighted loss function (weight is uniform for known ratings, 0 otherwise). The missing values are then predicted using the reconstructed approximate matrix. Based on the results in Section 7.2.3, we consider low parameter structures corresponding to the bases $\mathcal{C}_2$ and $\mathcal{C}_5$. In case of $\mathcal{C}_2$, the use of the MBI principle implies that the user-item rating depends equals the average rating in the co-cluster whereas in $\mathcal{C}_5$, the user-item rating is a combination of the user-bias, item-bias and the average rating in the co-cluster.

For our experiments, we used the MovieLens data set (GroupLens) described earlier and the results reported are averaged over multiple runs of five-fold cross-validation with 80% of ratings as the training data and 20% of the ratings as the test data in each run.

Table 6 shows the mean absolute error (MAE) obtained using various existing collaborative filtering approaches (Sarwar et al., 2000; Hofmann, 2004; Resnick et al., 1994) as well as the co-clustering approach based on squared Euclidean distance. From the table, we note that the co-clustering method based on $\mathcal{C}_5$ provides accuracy comparable to that of the SVD and NNMF-based methods. The co-clustering approach also has significant benefits in terms of computational effort as the training time is linear in the number of known ratings and the missing value prediction is a constant time operation unlike in other approaches. The number of parameters in the compressed representation is also much lower in the case of co-clustering as compared to SVD, NNMF and correlation methods when the rank or neighborhood size is of the same order as the number of row and column clusters.

6.3.2 CO-CLUSTERING CATEGORICAL DATA MATRICES

The second data analysis task we consider involves co-clustering data matrices consisting of categorical values from a finite set. Examples of such data include (i) market-basket data matrices with users as rows and products as columns and the entries corresponding to preferred brands, and (ii) genomic data matrices with rows corresponding to patients and columns corresponding to various positions/loci of gene sequences (also referred to as single nucleotide polymorphisms) and matrix entries indicating the occupying allele (usually only 4 possible alleles for each location) (Lin and Altman, 2004). Though the matrix elements take a finite number of values, there is no natural ordering, which makes it impossible to directly map them to the set of reals $\mathbb{R}$ (except in the case of binary valued data) in order to perform co-clustering as in the case of co-occurrence matrices. However, it is straightforward to represent each of these categorical values using discrete distributions over the set of all possible values. For example, when the matrix elements take values in $\{A, B, C, D\}$, then A can be represented as the distribution $[1, 0, 0, 0]$ while B can be represented as $[0, 1, 0, 0]$ and so on. With this representation, each element of the data matrix is a member of the r-simplex where

Num. clusters	Co-clustering Code	Summary Statistics Code	Matrix Code (Co-clustering Cost)	Total
$k = l = 1$	0	32.4	4973.3 ± 30.8	5005.7 ± 30.8
$k = l = 5$	232.2	425.8 ± 4.7	2695.4 ± 47.6	3353.4 ± 52.3
$k = l = 50$	564.4	5000	0	5564.4

Table 7: Description length (in bits) for encoding matrix information. Summary statistic code is the number of bits for encoding the counts of the four possible values in each co-cluster given the co-clustering whereas the matrix code is description length of the actual matrix given the summary statistics and the co-clustering. Co-clustering was performed using relative entropy cost function and basis $\mathcal{C}_2$.

r denotes the number of possible categorical values. Defining the domain $\mathcal{S}$ of the matrix elements to be the r-simplex, we can now proceed to perform co-clustering on the categorical data matrix by choosing an appropriate Bregman divergence over $\mathcal{S}$ and a suitable co-clustering basis. Since elements of $\mathcal{S}$ correspond to probability distributions, a natural choice of distortion measure is the relative entropy (or KL-divergence) over the r-simplex. The co-clustering objective function in this case is given by

$$J(\rho, \gamma) = \sum_{u=1}^{m} \sum_{v=1}^{n} KL(\mathbf{z}_{uv} || \hat{\mathbf{z}}_{uv})$$

where $\mathbf{Z} = [\mathbf{z}_{uv}]$ is the original matrix, $\hat{\mathbf{Z}} = [\hat{\mathbf{z}}_{\mathbf{uv}}]$ is the MBI solution based on the co-clustering, and the elements $\mathbf{z}_{uv}$ and $\hat{\mathbf{z}}_{\mathbf{uv}}$ belong to the r-simplex. This co-clustering objective function is also exactly equal to the minimum achievable description length (in bits) required for a lossless encoding of the original matrix $\mathbf{Z}$ given the MBI solution $\hat{\mathbf{Z}}$. Hence, assuming that the cost of describing the co-clustering and the summary statistics depends only on the pre-specified number of row and column clusters, the Bregman co-clustering algorithm corresponding to the relative entropy-based cost function automatically seeks to find an optimal (minimum length) lossless code for the matrix. A recent paper (Chakrabarti et al., 2004) follows a similar co-clustering based approach using binary relative entropy and basis $\mathcal{C}_2$ for performing lossless coding of binary valued matrices.

To demonstrate the effectiveness of the co-clustering approach described above, we generated 10 artificial 50×50 matrices consisting of four categorical values $\{A, B, C, D\}$. For all the matrices, we assumed generative models corresponding to multinomial distributions over $\{A, B, C, D\}$ and co-clustering basis $\mathcal{C}_2$ with $k = l = 5$. The elements in each co-cluster were generated using a single multinomial distribution with a purity of about 0.8, that is, the most likely categorical value had a probability of 0.8 with the rest all being equally likely with probability 0.067. Each of these matrices was then co-clustered using the relative entropy-based cost function on a 4-simplex with $k = l = 5$. Table 7 shows a comparison of the description lengths for various choices of k and l using a three-step encoding protocol where we first encode the co-clustering, then the summary statistics, that is, counts of $\{A, B, C, D\}$ in each co-cluster, and finally the original matrix given the summary statistics and the co-clustering.

For encoding the co-clustering, we employ a naive scheme that involves specifying the row and column clusters for each row and column respectively. Since there are k row clusters and l column clusters, the total number of bits required is given by $m \log_2 k + n \log_2 l$, as shown in the second column of Table 7. Given this co-clustering, we then proceed to encode the summary statistics, that is, counts of $\{A, B, C, D\}$, corresponding to each co-cluster. First, we observe that for each co-cluster, the four counts have to be non-negative integers that sum up to the total size of the

particular co-cluster. Since the co-clustering already specifies the total size of all the co-clusters, it is sufficient to specify any three of the four counts. Further, information about the count of a particular categorical value reduces the number of possible choices for the rest of the counts. In particular, if $m_{\hat{u}}$ and $n_{\hat{v}}$ denote the number of rows and columns in row cluster $\hat{u}$ and column cluster $\hat{v}$ respectively, then the number of bits for encoding the first count (say that of A) is given by $\log_2(1+m_{\hat{u}}n_{\hat{v}})$ while the cost for the second count (say that of B) is given by $\log_2(1+m_{\hat{u}}n_{\hat{v}}-N_A)$ where N_A is the count of A. Similarly, the encoding cost for the third count is given by $\log_2(1+m_{\hat{u}}n_{\hat{v}}-N_A-N_B)$ where N_B denotes the count of B. Thus, the total number of bits for encoding the summary statistics in this case is given by[15]

$$\sum_{\hat{u}=1}^{k}\sum_{\hat{v}=1}^{l}\left(\log_2(1+m_{\hat{u}}n_{\hat{v}})+\log_2(1+m_{\hat{u}}n_{\hat{v}}-N_A)+\log_2(1+m_{\hat{u}}n_{\hat{v}}-N_A-N_B)\right).$$

The third column in Table 7 shows the above encoding cost for different choices of k and l. When $k=l=50$, the co-clusters are all singleton sets so that it is sufficient to specify the single categorical value in each co-cluster. Since there are 4 possible values and mn co-clusters, the encoding cost in this case equals $2mn=5000$ bits.

The final step is to specify the original matrix given the summary statistics and the co-clustering and as mentioned earlier, the description length in this case is identical to the co-clustering objective function, which is shown in the fourth column of Table 7. When $k=l=50$, the description length is zero since the summary statistics fully specify the original matrix. From the table, we observe that with an optimal choice of row and column clusters, one can obtain an efficient lossless compression of matrix consisting of finite categorical values. On examining the resulting co-clusters, we find that most of them are quite homogeneous as well.

7. Related Work

We have discussed several related methods that have appeared in the literature throughout the paper. We have also discussed existing as well as novel applications of co-clustering in Section 6. In this section, we briefly review further connections and contrast our work to the existing literature. Our current work is related to several active areas of research, namely co-clustering, matrix approximation, learning based on Bregman divergences and convex optimization. In particular, our formulation of a general co-clustering problem was motivated by earlier work on co-clustering and matrix approximation (Dhillon et al., 2003b).

Co-clustering has been a topic of much interest in the recent years because of its applications to problems such as microarray analysis (Cheng and Church, 2000; Cho et al., 2004), natural language processing (Li and Abe, 1998; Freitag, 2004; Rohwer and Freitag, 2004), recommender systems (Hofmann, 2004) and text, image and speech analysis (Dhillon et al., 2003b; Takamura and Matsumoto, 2003; Qiu, 2004; Cai et al., 2005). Currently, there exist many formulations of the co-clustering problem such as the hierarchical co-clustering model (Hartigan, 1972), the sequential bi-clustering model (Cheng and Church, 2000) that involves finding the best co-clusters one at a time, and the spectral co-clustering model (Dhillon, 2001; Kluger et al., 2003) that involves partitioning a bipartite graph with vertices corresponding to the rows and columns. The reader

15. It is possible to have a more efficient encoding scheme by choosing an ordering of the categorical values $\{A,B,C,D\}$ that is likely to lead to the lowest number of bits, but does not make a significant difference in the current experiment as all the categorical values have nearly equal counts over the entire matrix.

should refer to Madeira and Oliveira (2004) for an extensive survey on various co-clustering models proposed in literature and their applications. Recently, there have also been other clustering formulations (Bekkerman et al., 2005; Gao et al., 2005) that are closely connected to co-clustering, but involve simultaneous clustering of multiple sets of related entities. In our current work, we focus on the partitional co-clustering formulation, first introduced by Hartigan (1972), where the objective is to partition the data matrix into $k \times l$ non-overlapping co-clusters where the quality of co-clusters is determined in terms of an appropriate cost function. Recently, quite a few algorithms (Cho et al., 2004; Dhillon et al., 2003b; Li and Abe, 1998; Li, 2005) have been proposed to address the above partitional problem for various cost functions based on squared Euclidean distance and I-divergence. One of the objectives of the current work is to generalize these algorithms to a large set of loss functions based on Bregman divergences.

Partitional co-clustering can also be readily viewed as an efficient low parameter matrix approximation technique as each homogeneous co-cluster can be accurately approximated by a small number of parameters. In fact, the flexibility to approximate a given data matrix in terms of a wide range of loss functions subject to a large class of constraints makes the co-clustering methods more widely applicable than traditional matrix approximation methods based on singular value decomposition. In particular, classical singular value decomposition (SVD) (Papadimitriou et al., 1998) based approaches to matrix approximation are quite often inappropriate for certain data matrices such as co-occurrence and contingency tables as singular vectors can have negative entries and the contributions of the component vectors in the approximation matrix are not localized. Both these issues make the interpretation of SVD-based approximations difficult, which is necessary for data mining purposes. To address these and related issues, techniques involving non-negativity constraints (Lee and Seung, 2001) using KL-divergence as the approximation loss function (Hofmann and Puzicha, 1998; Lee and Seung, 2001) have been proposed. However, these approaches apply to special types of matrices such as doubly stochastic and fully non-negative matrices. A general formulation that is both interpretable and applicable to various classes of matrices is often necessary for a number of real-life applications and the proposed Bregman co-clustering formulation attempts to address this requirement.

Co-clustering involving constraints on conditional expectations gives rise to theoretically elegant models with wide range of practical applicability since key summary statistics can be naturally preserved. Several co-clustering algorithms (Dhillon et al., 2003b; Cho et al., 2004) that have been proposed in the recent years can be derived from conditional expectation-based constraints. Conditional expectation constrained co-clustering, along with its demonstrated connection to the widely used maximum entropy principle (Jaynes, 1957; Cover and Thomas, 1991) and conditional independence based models (Hofmann and Puzicha, 1998), provides a strong foundation for a unified analysis and design of unsupervised learning algorithms.

Recent research (Azoury and Warmuth, 2001; Banerjee et al., 2005b) 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 (Csiszár, 1991) prompted us to explore a more general minimum Bregman information principle for all Bregman divergences.

It is important to note that most clustering and co-clustering techniques based on the alternate minimization scheme can be obtained as special cases of the Bregman co-clustering algorithm. For example, information-theoretic co-clustering (Dhillon et al., 2003b) corresponds to the case where

the constraint set is $\mathcal{C}_5$ and the Bregman divergence is KL-divergence. Similarly, the minimum sum-squared residue co-clustering algorithms (Cho et al., 2004) correspond to the cases where the constraint sets are $\mathcal{C}_2$ and $\mathcal{C}_6$ respectively while the Bregman divergence is the squared Euclidean distance. The one-sided Bregman clustering algorithms (Banerjee et al., 2005b) are also a special case with $l = n$.

8. Discussion

In this paper, we have presented a general theory of partitional Bregman co-clustering. Our analysis leads to a unified treatment of several known co-clustering methods that are being successfully used in the literature. Further, the analysis gives rise to an entire class of new co-clustering algorithms based on particular choices of the Bregman divergence and the set of summary statistics to be preserved. We have provided a meta-algorithm for the general case, and have demonstrated how to instantiate the algorithm for specific choices of divergences and statistics. There are several potential benefits to our formulation and analysis:

- Since our co-clustering formulation allows loss functions corresponding to all Bregman divergences, the technique now becomes applicable to practically all types of data matrices. The particular choice of the divergence function can be determined by (i) the data type, for example, if the data corresponds to joint probability distributions, relative entropy is an appropriate choice as the divergence function; (ii) the appropriate noise model, for example, Euclidean distance is appropriate for Gaussian noise, Itakura-Saito is appropriate for Poisson noise, etc.; or (iii) domain knowledge/requirements, for example, sparsity of the original matrix can be preserved using I-divergence.

- Our formulation allows approximation models of various complexities depending on the statistics that are constrained to be preserved. There are two key advantages to this flexibility. First, preserving summary statistics of the data may be crucial for some applications as well as important for subsequent analysis. Since the statistics preserving property is intrinsic to our approach, it is readily applicable to domains where summary statistics are important. Second, the multiple sets of preserved statistics may enable discovery of different structural patterns in the data.

- We have proposed and extensively used the minimum Bregman information (MBI) principle as a generalization of the maximum entropy principle. Since the approximations obtained from the MBI principle extend some of the desirable properties of the maximum entropy models to settings where a Bregman divergence other than the relative entropy is more appropriate, we get a new class of statistical modeling techniques that are applicable to more general settings. The MBI principle has potential applications beyond the co-clustering approximations considered in this paper.

- While the central focus of this paper has been to obtain good co-clusterings using matrix approximation error to evaluate goodness, as a by-product, we have obtained a general class of fast matrix approximation techniques with several desirable properties. In particular, the approximation techniques can work with general divergence functions and preserve desirable statistical properties of the original data. The approximations are based on co-clustering, and are expected to have different behavior from the spectral methods typically employed for

matrix approximations. Further, since the methods are iterative and do not involve eigenvalue computations, they are significantly faster than existing methods and hence, more appropriate for large data matrices.

In this paper, our analysis of co-clustering has focused on data matrices that represent the relationship between two entities. Many emerging application domains collect data on relationships between multiple entities, which can be represented as a tensor. Our proposed co-clustering technique can be extended to this general setting involving tensors unlike other methods that are specific to matrices. It will be worthwhile to investigate how the extensions of co-clustering to tensor data perform compared to existing techniques. In particular, several practical problem domains have known statistical dependency relationships between the several entities of interest. One of the key challenges of an extension of co-clustering to such multi-entity relational domains is to come up with efficient algorithms that take advantage of the statistical relationships and maintain succinct representations of the entities and their relationships.

Acknowledgments

We would like to thank Hyuk Cho and the reviewers for their detailed comments and suggestions that significantly improved the paper. We would like to thank John Lafferty and an anonymous reviewer for pointing out the connection of our projection results to existing literature on Bregman duality. The research was partly supported by NSF grants IIS-0307792, CCF-0431257, III-0713142, NSF Career Award ACI-0093404, and NSF ITR award IIS-0325116.

Appendix A. Information Theoretic Co-clustering

Proof of Lemma 1 Let p' be any distribution that satisfies (4) and (5), and let q be as in (3). Consider

$$\begin{aligned}
KL(p'||q) &= \sum_{\hat{x}}\sum_{\hat{y}}\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} p'(x,y)\log\frac{p'(x,y)}{q(x,y)} \\
&= -H(p') - \sum_{\hat{x}}\sum_{\hat{y}}\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} p'(x,y)\left(\log p(\hat{x},\hat{y}) + \log p(x|\hat{x}) + \log p(y|\hat{y})\right) \\
&= -H(p') - \sum_{\hat{x}}\sum_{\hat{y}} p(\hat{x},\hat{y})\log p(\hat{x},\hat{y}) - \sum_{\hat{x}}\sum_{x\in\hat{x}} p(x)\log p(x|\hat{x}) - \sum_{\hat{y}}\sum_{y\in\hat{y}} p(y)\log p(y|\hat{y}) \\
&= -H(p') - \sum_{\hat{x}}\sum_{\hat{y}} p(\hat{x},\hat{y})\left(\sum_{x\in\hat{x}} p(x|\hat{x})\right)\left(\sum_{y\in\hat{y}} p(y|\hat{y})\right)\log p(\hat{x},\hat{y}) \\
&\qquad - \sum_{\hat{x}}\sum_{x\in\hat{x}} q(x)\log p(x|\hat{x}) - \sum_{\hat{y}}\sum_{y\in cy} q(y)\log p(y|\hat{y})
\end{aligned}$$

$$
\begin{aligned}
&= -H(p') - \sum_{\hat{x}}\sum_{\hat{y}}\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} p(x|\hat{x})\,p(\hat{x},\hat{y})\,p(y|\hat{y})\log p(\hat{x},\hat{y}) \\
&\qquad - \sum_{\hat{x}}\sum_{x\in\hat{x}}\left(\sum_{\hat{y}}\sum_{y\in\hat{y}} q(x,y)\right)\log p(x|\hat{x}) - \sum_{\hat{y}}\sum_{y\in\hat{y}}\left(\sum_{\hat{x}}\sum_{x\in\hat{x}} q(x,y)\right)\log p(y|\hat{y}) \\
&= -H(p') - \sum_{\hat{x}}\sum_{\hat{y}}\sum_{x\in\hat{x}}\sum_{y\in\hat{y}} q(x,y)\log\left(p(\hat{x},\hat{y})\,p(x|\hat{x})\,p(y|\hat{y})\right) \\
&= -H(p') + H(q).
\end{aligned}
$$

Since $KL(p'||q) \geq 0$, we have $H(q) \geq H(p')$. ∎

Appendix B. Some Properties of Bregman Divergences

We present some useful properties of Bregman divergences and Bregman information that we use in our analysis in the paper.

Lemma 12 (Bregman 1967; Censor and Zenios 1998) *For any Bregman divergence $d_\phi : S \times \mathrm{int}(S) \mapsto \mathbb{R}_+$ and $z_1 \in \mathcal{S}$ and $z_2, z_3 \in \mathrm{ri}(\mathcal{S})$, the following three-point property holds:*

$$d_\phi(z_1, z_3) = d_\phi(z_1, z_2) + d_\phi(z_2, z_3) + \langle z_1 - z_2, \nabla\phi(z_2) - \nabla\phi(z_3)\rangle \ .$$

Theorem 7 (Banerjee et al. 2005a) *For any Bregman divergence $d_\phi : S \times \mathrm{ri}(S) \mapsto \mathbb{R}_+$, random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq S$ and sub-σ algebra $\mathcal{G}$ for Z, the conditional expectation $E[Z|\mathcal{G}]$ is the optimal predictor of Z among all $\mathcal{G}$ measurable random variables in terms of Bregman divergence, that is,*

$$E[Z|\mathcal{G}] = \operatorname*{argmin}_{Z' \in \mathcal{G}} d_\phi(Z, Z') \ .$$

Lemma 13 (Banerjee et al. 2005b) *For any Bregman divergence $d_\phi : S \times \mathrm{ri}(S) \mapsto \mathbb{R}_+$, random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq S$ and any constant $c \in \mathrm{int}(S)$, the following decomposition holds:*

$$E[d_\phi(Z, c)] = E[d_\phi(Z, E[Z])] + d_\phi(E[Z], c) \ .$$

Lemma 14 (Banerjee et al. 2005b) *For any Bregman divergence $d_\phi : S \times \mathrm{ri}(S) \mapsto \mathbb{R}_+$ and random variable $Z \sim w(z),\ z \in \mathcal{Z} \subseteq S$, the optimal constant predictor of Z in terms of Bregman divergence is its expectation, that is,*

$$E[Z] = \operatorname*{argmin}_{c} E[d_\phi(Z, c)] \ .$$

Appendix C. Block Average Co-clustering

Proof of Theorem 1 Consider the Lagrangian $J(Z',\Lambda)$ of the MBI problem:

$$\begin{aligned} J(Z',\Lambda) &= I_\phi(Z') + \sum_{\hat{u},\hat{v}} \lambda_{\hat{u}\hat{v}} (E[Z'|\hat{u},\hat{v}] - E[Z|\hat{u},\hat{v}]) \\ &\overset{(a)}{=} E[\phi(Z')] - \phi(E[Z']) + \sum_{\hat{u},\hat{v}} \lambda_{\hat{u}\hat{v}} (E[Z'|\hat{u},\hat{v}] - E[Z|\hat{u},\hat{v}]) \\ &\overset{(b)}{=} E[\phi(Z')] - \phi(E[Z]) + \sum_{\hat{u},\hat{v}} \lambda_{\hat{u}\hat{v}} (E[Z'|\hat{u},\hat{v}] - E[Z|\hat{u},\hat{v}]) \ , \end{aligned}$$

where $\lambda_{\hat{u}\hat{v}}$ is the Lagrange multiplier corresponding to the constraint $E[Z'|\hat{u},\hat{v}] - E[Z|\hat{u},\hat{v}] = 0$ Further, (a) follows from Lemma 2 and (b) follows since $E[Z'] = E_{\hat{U},\hat{V}}[E[Z'|\hat{U},\hat{V}]] = E[Z]$.

Rewriting the Lagrangian in terms of matrix elements $\{\{z'_{uv}\}_{u=1}^m\}_{v=1}^n$ corresponding to Z', we obtain

$$J(Z',\Lambda) = \sum_{u=1}^{m}\sum_{v=1}^{n} w_{uv}(\phi(z'_{uv}) - \phi(\overline{z})) + \sum_{\hat{u},\hat{v}} \lambda_{\hat{u}\hat{v}} \frac{1}{w_{\hat{u}\hat{v}}} \sum_{\substack{u:\rho(u)=\hat{u} \\ \gamma(v)=\hat{v}}} w_{uv}(z'_{uv} - z_{uv}) \ , \tag{29}$$

where $w_{\hat{u}\hat{v}} = \sum_{u:\rho(u)=\hat{u},v:\gamma(v)=\hat{v}} w_{uv}$ and $\overline{z} = \sum_{u=1}^m \sum_{v=1}^n w_{uv} z'_{uv} = \sum_{u=1}^m \sum_{v=1}^n w_{uv} z_{uv}$.

To obtain the optimal solution $\hat{Z}_A$, we consider the first order necessary conditions, that is, set the partial derivatives with respect to the matrix elements and the Lagrange multipliers. Taking partial derivatives with respect to $\lambda_{\hat{u},\hat{v}}$, we obtain

$$\frac{1}{w_{\hat{u}\hat{v}}} \sum_{\substack{u:\rho(u)=\hat{u} \\ \gamma(v)=\hat{v}}} w_{uv}(z'_{uv} - z_{uv}) = 0 \qquad \forall \hat{u},\hat{v}, \tag{30}$$

that is, $E[Z|\hat{u},\hat{v}] = E[Z'|\hat{u},\hat{v}]$ for all $[\hat{u}]_1^k$ and $[\hat{v}]_1^l$.

Now, setting partial derivatives of (29) with respect to z'_{uv} equal to 0, we get

$$w_{uv}\nabla\phi(z'_{uv}) - w_{uv}\nabla\phi(\overline{z}) + \lambda_{\hat{u}\hat{v}} \frac{w_{uv}}{w_{\hat{u}\hat{v}}} = 0 \ ,$$

where $\hat{u} = \rho(u)$ and $\hat{v} = \gamma(v)$. Since $w_{uv} \in \mathbb{R}_+$ and $\overline{z} = E[Z] = E[Z']$, the optimal solution $\hat{Z} = \hat{Z}_A$ has the form

$$\hat{z}_{uv} = \nabla\phi^{(-1)}\left(\nabla\phi(E[Z]) - \frac{\lambda^*_{\hat{u}\hat{v}}}{w_{\hat{u}\hat{v}}}\right), \ \ \hat{u} = \rho(u), \hat{v} = \gamma(v), \tag{31}$$

where $\lambda^*_{\hat{u}\hat{v}}$ corresponds to the optimal Lagrange multiplier. Note that the right hand side is constant for a given $(\hat{u},\hat{v})$. Substituting (31) into (30) gives us

$$E[Z|\hat{u},\hat{v}] = \nabla\phi^{(-1)}\left(\nabla\phi(E[Z]) - \frac{\lambda^*_{\hat{u}\hat{v}}}{w_{\hat{u}\hat{v}}}\right).$$

Hence, the only solution satisfying the first order necessary conditions is $\hat{z}_{uv} = E[Z|\hat{u},\hat{v}], \ \forall u,v$, that is, $\hat{Z}_A = E[Z|\hat{U},\hat{V}]$. The existence and uniqueness of $\hat{Z}_A$ follow from the strict convexity of ϕ. ■

Proof of Lemma 3 Using the three point property (Lemma 12) and taking expectations, for any $Z' \in \mathcal{S}_A$ and $Z'' \in \mathcal{S}_B$, we have

$$E[d_\phi(Z', Z'')] = E[d_\phi(Z', \hat{Z}_A)] + E[d_\phi(\hat{Z}_A, Z'') + E[\langle Z' - \hat{Z}_A, \nabla\phi(\hat{Z}_A)\rangle] - E[\langle Z' - \hat{Z}_A, \nabla\phi(Z'')\rangle] \ .$$

We now argue that the last two terms in the above expression vanish to give the desired result. From Theorem 1, we note that $\hat{Z}_A = E[Z|\hat{U}, \hat{V}]$ so that

$$E[\langle Z' - \hat{Z}_A, \nabla\phi(\hat{Z}_A)] = E_{\hat{U},\hat{V}}[\langle E[Z'|\hat{U}, \hat{V}] - E[\hat{Z}_A|\hat{U}, \hat{V}], \nabla\phi(\hat{Z}_A)\rangle] = 0 \ ,$$

since $\hat{Z}_A$ is a constant given $(\hat{U}, \hat{V})$ and has the same co-cluster means as $Z' \in \mathcal{S}_A$.

To show that the last term $E[\langle Z' - \hat{Z}_A, \nabla\phi(Z'')\rangle]$ also vanishes, we note that for any $Z'' \in S_B$, $\nabla\phi(Z'') = g(E[Z|\hat{U}, \hat{V}])$ for some deterministic function g so that

$$\begin{aligned} E[\langle Z' - \hat{Z}_A, \nabla\phi(Z'')\rangle] &= E[\langle Z' - \hat{Z}_A, g(E[Z|\hat{U}, \hat{V}])\rangle] \\ &= E_{\hat{U},\hat{V}}[\langle (E[Z'|\hat{U}, \hat{V}] - E[\hat{Z}_A|\hat{U}, \hat{V}]), g(E[Z|\hat{U}, \hat{V}])\rangle] \ = \ 0 \ , \end{aligned}$$

since $\hat{Z}_A$ and Z' both belong to $\mathcal{S}_A$ and hence, have the same co-cluster means. ∎

Proof of Theorem 2 From Lemma 7, we observe that for any $Z' \in \mathcal{S}_A$ and $Z'' \in \mathcal{S}_B$,

$$E[d_\phi(Z', Z'')] = E[d_\phi(Z', \hat{Z}_A)] + E[d_\phi(\hat{Z}_A, Z'') \ .$$

Hence, for a given $Z'' \in \mathcal{S}_B$ and any $Z' \in \mathcal{S}_A$, $E[d_\phi(Z', Z'')] \geq E[d_\phi(\hat{Z}_A, Z'')]$, with equality only when $Z' = \hat{Z}_A$. Since $\hat{Z}_A \in \mathcal{S}_A$, this implies that

$$\hat{Z}_A = \operatorname*{argmin}_{Z' \in \mathcal{S}_A} E[d_\phi(Z', Z'')], \ \ \forall Z'' \in \mathcal{S}_B \ .$$

Similarly, for a given $Z' \in \mathcal{S}_A$ and any $Z'' \in \mathcal{S}_A$, $E[d_\phi(Z', Z'')] \geq E[d_\phi(Z', \hat{Z}_A)]$ with equality only when $Z'' = \hat{Z}_A$. Since $\hat{Z}_A \in \mathcal{S}_B$ as well, we obtain the second part of the result, that is,

$$\hat{Z}_A = \operatorname*{argmin}_{Z'' \in \mathcal{S}_B} E[d_\phi(Z', Z'')], \ \ \forall Z' \in \mathcal{S}_A \ . \ ∎$$

Proof of Lemma 5 By definition,

$$\begin{aligned} E[d_\phi(Z, \hat{Z}^{new})] &= E[\phi(Z) - \phi(\hat{Z}^{new}) - \langle Z - \hat{Z}^{new}, \nabla\phi(\hat{Z}^{new})\rangle] \\ &\overset{(a)}{=} E[\phi(Z) - \phi(\hat{Z}^{new})] \\ &= E[\phi(Z) - \phi(\tilde{Z}) - \langle Z - \tilde{Z}, \nabla\phi(\tilde{Z})\rangle] - E[\phi(\hat{Z}^{new}) - \phi(\tilde{Z}) - \langle Z - \tilde{Z}, \nabla\phi(\tilde{Z})\rangle] \\ &= E[d_\phi(Z, \tilde{Z})] - E[d_\phi(\hat{Z}^{new}, \tilde{Z})] + E[\langle Z - \hat{Z}^{new}, \nabla\phi(\tilde{Z})\rangle] \\ &\overset{(b)}{=} E[d_\phi(Z, \tilde{Z})] - E[d_\phi(\hat{Z}^{new}, \tilde{Z})] \\ &\leq E[d_\phi(Z, \tilde{Z})] \ , \end{aligned}$$

where (a) follows since $\hat{Z}^{new} \in S_A$ and $\hat{Z}^{new} \in S_B$ so that taking conditional expectations over $E[Z|\hat{U}, \hat{V}]$ makes the last term zero and (b) follows since $\nabla\phi(\tilde{Z})$ remains unchanged given $(\hat{U}, \hat{V})$ corresponding to $\hat{Z}^{new}$, and $E[\hat{Z}^{new}|\hat{U}, \hat{V}] = E[Z|\hat{U}, \hat{V}]$, thus making the last term vanish. ∎

Appendix D. Analysis of the General Case

Proof of Theorem 3 In order to identify the various matrix approximation schemes, we determine the subsets $\mathcal{C} \subseteq \Gamma_2$ that satisfy conditions (a) and (b). First, observe that $E[Z|U_\emptyset, V_\emptyset] = E[Z]$ and $E[Z|U,V] = Z$. Since $E[Z] = E[Z|U_\emptyset, V_\emptyset]$ can be obtained from every other conditional expectation $E[Z|C], C \in \Gamma_2$, and $Z = E[Z|U,V]$ determines every other conditional expectation, condition (b) implies that the pairs $\{U_\emptyset, V_\emptyset\}$ and $\{U,V\}$ cannot occur in combination with any other. As these pairs do not contain $\hat{U}$ or $\hat{V}$, we only need to consider combinations of the remaining members of Γ_2.

Further, we note that if there are two pairs $\mathcal{G}_1, \mathcal{G}_2 \in \mathcal{C},\ \mathcal{G}_1 \neq \mathcal{G}_2$ such that $\hat{U} \in \mathcal{G}_1$ and $\hat{U} \in \mathcal{G}_2$, then either $E[Z|\mathcal{G}_1]$ subsumes $E[Z|\mathcal{G}_2]$ or vice versa depending on the granularity of the column random variables in $\mathcal{G}_1$ and $\mathcal{G}_2$. A similar observation holds for $\hat{V}$. Hence, condition (b) implies that each non-trivial combination $\mathcal{C} \subseteq \Gamma_2$ should contain *exactly one* pair (possibly the same) that contains $\hat{U}$ and $\hat{V}$. Using the above observation, we enumerate the various possible cases as follows:

case 1: $\{\hat{U}, V_\emptyset\} \in \mathcal{C}$. $\mathcal{C}$ should also contain a pair containing $\hat{V}$, which can only be $\{U_\emptyset, \hat{V}\}$ since every other eligible pair $\in \Gamma_2$ uniquely determines $\{\hat{U}, V_\emptyset\}$ so that inclusion of any other pair leads to a violation of condition (b). Therefore, the only possible combination in this case is $\{\{\hat{U}, V_\emptyset\}, \{U_\emptyset, \hat{V}\}\}$.

case 2: $\{\hat{U}, \hat{V}\} \in \mathcal{C}$. Since condition (a) is already satisfied, we only need to identify the pairs in Γ_2 that can be included in $\mathcal{C}$ without violating condition (b), that is, pairs for which the row random variable is of higher granularity than $\hat{U}$ and the column random variable is of lower granularity than $\hat{V}$ or vice versa, which leads to two possibilities—$\{U, V_\emptyset\}$ and $\{U_\emptyset, V\}$. Hence, there are four combinations corresponding to the cases where we include neither of the pairs, exactly one of the pairs and both of them, that is,

$$\{\{\hat{U}, \hat{V}\}\}, \quad \{\{\hat{U}, \hat{V}\}, \{U\}, \{V\}\}, \quad \{\{\hat{U}, \hat{V}\}, \{U, V_\emptyset\}\}, \quad \text{and} \quad \{\{\hat{U}, \hat{V}\}, \{U_\emptyset, V\}\}.$$

case 3: $\{\hat{U}, V\} \in \mathcal{C}$. $\mathcal{C}$ should also contain a pair containing $\hat{V}$, which can only be $\{U, \hat{V}\}$ since every other eligible pair $\in \Gamma_2$ is subsumed by $\{\hat{U}, V\}$. Therefore, the only possible combination in this case is $\{\{\hat{U}, V\}, \{U, \hat{V}\}\}$.

Ignoring $U_\emptyset$ and $V_\emptyset$ since they are constant random variables and putting together all the different possible bases, we obtain the desired result. ∎

Proof of Theorem 4 Consider the Lagrangian $J(Z', \Lambda)$ of the MBI problem:

$$\begin{aligned} J(Z', \Lambda) &= I_\phi(Z') + \sum_{r=1}^{s} E_{\mathcal{G}_r}\left[\frac{\Lambda_{\mathcal{G}_r}}{w_{G_r}}(E[Z'|\mathcal{G}_r] - E[Z|\mathcal{G}_r])\right] \\ &= E[\phi(Z')] - \phi(E[Z']) + \sum_{r=1}^{s} E_{\mathcal{G}_r}\left[\frac{\Lambda_{\mathcal{G}_r}}{w_{G_r}}(E[Z'|\mathcal{G}_r] - E[Z|\mathcal{G}_r])\right], \end{aligned}$$

where $\Lambda_{\mathcal{G}_r}$ is a deterministic function of the random variable $\mathcal{G}_r$ and equals the appropriate Lagrange multiplier when $\mathcal{G}_r$ is specified. The Lagrange dual, $L(\Lambda) = \inf_{Z'} J(Z', \Lambda)$, is concave in Λ. By maximizing the Lagrange dual, we get the optimal Lagrange multipliers, that is, $\Lambda^* = \{\Lambda^*_{\mathcal{G}_r}\} =$

$\text{argmax}_\Lambda L(\Lambda)$. Substituting Λ^* into the first order necessary conditions corresponding to the minimizer $\hat{Z}_A$, we get

$$\begin{aligned} \nabla\left(E[\phi(\hat{Z}_A)] - \phi(E[\hat{Z}_A]) + \sum_{r=1}^{s} E_{G_r}\left[\frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}(E[\hat{Z}_A|\mathcal{G}_r] - E[Z|\mathcal{G}_r])\right]\right) = 0, \\ \implies \quad \nabla\phi(\hat{Z}_A) \;=\; \nabla\phi(E[Z]) - \sum_{r=1}^{s}\frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}, \end{aligned} \tag{32}$$

where $w_{\mathcal{G}_r}$ is the measure corresponding to $\mathcal{G}_r$ and $E[\hat{Z}_A] = E[Z]$. Rearranging terms proves the first part of the theorem. The existence and uniqueness of $\hat{Z}_A$ follow from the strict convexity of ϕ. ■

Proof of Lemma 6 From Theorem 4, we note that

$$\hat{Z} = (\nabla\phi)^{(-1)}\left(\nabla\phi(E[Z]) - \sum_{r=1}^{s}\frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}\right),$$

where $\mathcal{C} = \{\mathcal{G}_r\}_{r=1}^{s}$ and $\Lambda^*_{\mathcal{G}_r}$ are the optimal Lagrange multipliers corresponding to the constraints $E[Z|\mathcal{G}_r] = E[\hat{Z}|\mathcal{G}_r]$. Now, by definition,

$$\begin{aligned} E[d_\phi(Z,\hat{Z})] &= E[\phi(Z) - \phi(\hat{Z}) - \langle Z-\hat{Z}, \nabla\phi(\hat{Z})\rangle] \\ &= E[\phi(Z) - \phi(\hat{Z})] - E[\langle Z-\hat{Z}, (\nabla\phi(E[Z]) - \sum_{r=1}^{s}\frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}})\rangle] \\ &= E[\phi(Z) - \phi(\hat{Z})] - E[\langle Z-\hat{Z}, \nabla\phi(E[Z])\rangle] + \sum_{r=1}^{s} E[\langle Z-\hat{Z}, \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}\rangle] \\ &\overset{(a)}{=} E[\phi(Z) - \phi(\hat{Z})] + \sum_{r=1}^{s} E_{\mathcal{G}_r}[\langle E[Z|\mathcal{G}_r] - E[\hat{Z}|\mathcal{G}_r], \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}\rangle] \\ &\overset{(b)}{=} E[\phi(Z) - \phi(\hat{Z})] \\ &\overset{(c)}{=} E[\phi(Z) - \phi(E[Z])] - E[\phi(\hat{Z}) - \phi(E[\hat{Z}])] \\ &\overset{(d)}{=} E[\phi(Z) - \phi(E[Z]) - \langle Z - E[Z], \nabla\phi(E[Z])\rangle] \\ &\quad -E[\phi(\hat{Z}) - \phi(E[\hat{Z}]) - \langle \hat{Z} - E[\hat{Z}], \nabla\phi(E[\hat{Z}])\rangle] \\ &= I_\phi(Z) - I_\phi(\hat{Z}), \end{aligned}$$

where (a) and (c) follow since $E[Z] = E[\hat{Z}]$, (b) follows since $E[Z|\mathcal{G}_r] = E[\hat{Z}|\mathcal{G}_r]$, $\forall \mathcal{G}_r \in \mathcal{C}$, and (d) follows since $E[\langle Z - E[Z], \nabla\phi(E[Z])\rangle] = 0$ and $E[\langle \hat{Z} - E[\hat{Z}], \nabla\phi(E[\hat{Z}])] = 0$. ■

Proof of Lemma 7 Using the three point property (Lemma 12) and taking expectations, for any $Z' \in \mathcal{S}_A$ and $Z'' \in \mathcal{S}_B$, we have

$$E[d_\phi(Z',Z'')] = E[d_\phi(Z',\hat{Z}_A)] + E[d_\phi(\hat{Z}_A,Z'') + E[\langle Z'-\hat{Z}_A, \nabla\phi(\hat{Z}_A)\rangle] - E[\langle Z'-\hat{Z}_A, \nabla\phi(Z'')\rangle]\,.$$

We now argue that the last two terms in the expression vanish to give the desired result. Note that since $\hat{Z}_A$ and $Z' \in \mathcal{S}_A$, we have $E[Z|\mathcal{G}_r] = E[\hat{Z}_A|\mathcal{G}_r] = E[Z'|\mathcal{G}_r],\ \forall \mathcal{G}_r \in \mathcal{C}$. By (32),

$$\begin{aligned} E[\langle Z' - \hat{Z}_A, \nabla\phi(\hat{Z}_A)\rangle] &= E[\langle Z' - \hat{Z}_A, (\nabla\phi(E[Z]) - \sum_{r=1}^{s} \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}})\rangle] \\ &= \langle E[Z' - \hat{Z}_A], \nabla\phi(E[Z])\rangle - \sum_{r=1}^{s} E[\langle Z' - \hat{Z}_A, \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}\rangle] \\ &\overset{(a)}{=} -\sum_{r=1}^{s} E_{\mathcal{G}_r}[\langle E[Z'|\mathcal{G}_r] - E[\hat{Z}_A|\mathcal{G}_r], \frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}\rangle] \overset{(b)}{=} 0\ , \end{aligned}$$

where (a) follows since $E[Z] = E[Z_A] = E[Z']$, and (b) follows since both Z' and $\hat{Z}_A$ satisfy the constraints, $E[Z|\mathcal{G}_r] = E[\hat{Z}_A|\mathcal{G}_r],\ \forall \mathcal{G}_r \in \mathcal{C}$.

To show that the last term $E[\langle Z' - \hat{Z}_A, \nabla\phi(Z'')]$ also vanishes, we use the fact that by definition $\nabla\phi(Z'') = \sum_{r=1}^{s} g_r(E[Z|\mathcal{G}_r])$. Hence,

$$\begin{aligned} E[\langle Z' - \hat{Z}_A, \nabla\phi(Z'')\rangle] &= E[\langle Z' - \hat{Z}_A, \sum_{r=1}^{s} g_r(E[Z|\mathcal{G}_r])\rangle] \\ &= \sum_{r=1}^{s} E[\langle Z' - \hat{Z}_A, g_r(E[Z|\mathcal{G}_r])\rangle] \\ &= \sum_{r=1}^{s} E_{\mathcal{G}_r}[\langle E[Z'|\mathcal{G}_r] - E[\hat{Z}_A|\mathcal{G}_r], g_r(E[Z|\mathcal{G}_r])\rangle] = 0\ , \end{aligned}$$

since $E[Z|\mathcal{G}_r] = E[\hat{Z}_A|\mathcal{G}_r],\ \forall \mathcal{G}_r \in \mathcal{C}$. That completes the proof. ∎

Proof of Theorem 5 From Lemma 7, we observe that for any $Z' \in \mathcal{S}_A$ and $Z'' \in \mathcal{S}_B$,

$$E[d_\phi(Z', Z'')] = E[d_\phi(Z', \hat{Z}_A)] + E[d_\phi(\hat{Z}_A, Z'')]\ .$$

that is, it is additive in functions of the conditional expectations, that is, $\frac{\Lambda^*_{\mathcal{G}_r}}{w_{\mathcal{G}_r}}$ in the natural parameter space, which implies that $\hat{Z}_A \in \mathcal{S}_B$ as well. For a given $Z'' \in \mathcal{S}_B$ and any $Z' \in \mathcal{S}_A$, $E[d_\phi(Z', Z'')] \geq E[d_\phi(\hat{Z}_A, Z'')]$, with equality only when $Z' = \hat{Z}_A$, due to strict convexity of ϕ . Since $\hat{Z}_A \in \mathcal{S}_A$, this implies that

$$\hat{Z}_A = \operatorname*{argmin}_{Z' \in \mathcal{S}_A} E[d_\phi(Z', Z'')],\ \ \forall Z'' \in \mathcal{S}_B\ .$$

Similarly, for a given $Z' \in \mathcal{S}_A$ and any $Z'' \in \mathcal{S}_A$, $E[d_\phi(Z', Z'')] \geq E[d_\phi(Z', \hat{Z}_A)]$ with equality only when $Z'' = \hat{Z}_A$. By (32), we observe that $\nabla\phi(\hat{Z}_A)$ is an additive function of the conditional expectations, which implies that $\hat{Z}_A \in \mathcal{S}_B$. Thus, we obtain the second part of the result, that is,

$$\hat{Z}_A = \operatorname*{argmin}_{Z'' \in \mathcal{S}_B} d_\phi(Z', Z''),\ \ \forall Z' \in \mathcal{S}_A\ .\ \blacksquare$$

Proof of Lemma 9 From Lemma 8, we have

$$\begin{aligned} E[d_\phi(Z, \tilde{Z}^t)] &= E_U[E_{V|U}[\xi(U, \rho^{t+1}(U), V, \gamma^t(V))]] \\ &= E_U[\min_{g:[g]_1^k} E_{V|U}[\xi(U, g, V, \gamma^t(V))]] \\ &\leq E_U[E_{V|U}[\xi(U, \rho^t(U), V, \gamma(V))]] \\ &= E[d_\phi(Z, \hat{Z}^t)]\ .\ \blacksquare \end{aligned}$$

Appendix E. A Recipe for Implementation

To instantiate the Bregman co-clustering meta-algorithm, two key ingredients need to be selected: (a) the Bregman divergence suitable for a given data matrix, and (b) a co-clustering basis. The goal of this section is to show how to translate the abstract meta algorithm in Section 5 into a concrete and operational co-clustering recipe that is customized for the selected ingredients. We discuss four such concrete recipes. The first three cases concern special cases that admit significant structural and computational simplifications in the meta-algorithm and the last case concerns an example that requires us to use the full power of the abstract framework.

The Bregman co-clustering algorithm (Algorithm 2) involves three main steps—(i) obtaining the MBI solution (Section 5.5) or the optimal Lagrange multipliers, (ii) row assignment, and (iii) column assignment. Of these three steps, the last two involve conceptually straightforward comparisons to determine the optimal row and column assignments at each stage whereas the first step usually involves non-linear optimization and can be computationally expensive. Nonetheless, it is possible to implement these steps in a computationally economical fashion. For certain special cases, the MBI problem has a closed form solution, which eliminates the need for the MBI routine and allows significant simplification of the overall co-clustering algorithm. In particular, there are three cases for which such closed form exists:

Case A: When the co-clustering basis $\mathcal{C}$ is $\mathcal{C}_2$ and d_ϕ is any Bregman divergence. Conceptually, this case was covered in complete detail in Section 3, but we present additional operational details in this section.

Case B: When d_ϕ is squared Euclidean distance and $\mathcal{C}$ is any co-clustering basis in the set $\{\mathcal{C}_i\}_{i=1}^{6}$,

Case C: When d_ϕ is I-divergence and $\mathcal{C}$ is any co-clustering basis in the set $\{\mathcal{C}_i\}_{i=1}^{6}$.

For these special cases, the cost functions that determine the row and column assignments in steps 2B and 2C of the co-clustering algorithm (Algorithm 2) can be directly expressed in terms of the co-clustering (ρ,γ) and the input matrix $\mathbf{Z}$ without any Lagrange multipliers and the computational effort required to evaluate the cost is linear in the size of $\mathbf{Z}$ (i.e., number of non-zeros). For the general case, the computation time per iteration for the co-clustering algorithm is still linear in the size of $\mathbf{Z}$, but the total time taken will depend on the number of iterations required to obtain the MBI solution.

In order to describe the Bregman co-clustering algorithm for the special cases mentioned above, we use a matrix notation that is more suitable for computation and exposition. From Theorem 1 and Tables 1 and 2, we observe that the MBI solution for the three special cases mentioned above can be expressed as a combination of conditional expectations of the random variable Z corresponding to the input matrix. Since the computation of the MBI solution is an important task in the co-clustering algorithm, we proceed by first expressing the various conditional expectations in matrix notation. We use the symbols $\otimes$ and $\oslash$ respectively to denote element-wise multiplication (i.e., the Hadamard product) and element-wise division between two matrices of the same size.

E.1 Matrix Representation of Conditional Expectations

Let $\mathbf{Z} \in S^{m\times n}$ denote the data matrix and $\mathbf{W} \in \mathbb{R}_+^{m\times n}$ denote the matrix corresponding to a probability measure over the matrix elements. Let $\mathbf{R} \in \{0,1\}^{m\times k}$ and $\mathbf{C} \in \{0,1\}^{n\times l}$ denote the row and column

cluster membership matrices, that is,

$$r_{ug} = \begin{cases} 1 & g = \rho(u), \\ 0 & \text{otherwise}, \end{cases} \qquad c_{vh} = \begin{cases} 1 & h = \gamma(v), \\ 0 & \text{otherwise}. \end{cases}$$

In other words, the entry $r_{ug} = 1$ iff row u belongs to row cluster g and the entry $c_{vh} = 1$ iff column v belongs to column cluster h. Further, let $\mathbf{E}_m$ and $\mathbf{E}_n$ denote $m \times 1$ and $n \times 1$ vectors consisting of all ones. It should now be straightforward to see that elements in different partitions (e.g., rows or row clusters) of the input matrix $\mathbf{Z}$ can be aggregated using the appropriate matrix multiplication operations, from the ones listed below:

(a) Left multiplication by $\mathbf{R}^T$—Aggregation of the rows into row clusters

(b) Right multiplication by $\mathbf{C}$—Aggregation of the columns into column clusters

(c) Left multiplication by $\mathbf{E}_m^T$—Aggregation of all the rows into a single group

(d) Right multiplication by $\mathbf{E}_n$—Aggregation of all the columns into a single group

To obtain the expected values along the various partitions instead of the sums, we need to perform an element-wise multiplication with the measure matrix $\mathbf{W}$ before aggregation and later follow it up with an appropriate element-wise division. It is important to note here that the size of matrix containing the expected values is equal to the number of corresponding partitions, which is usually smaller than that of the original $\mathbf{Z}$. Therefore, to create a $m \times n$ matrix such that the uv^{th} element reflects the expectation along the partition containing z_{uv}, we need to replicate the expected values for all members of the corresponding partitions, which can be achieved using the following matrix multiplications:

(a) Left multiplication by $\mathbf{R}$—Replication of the given (row) vectors corresponding to each row cluster along all the constituent rows.

(b) Right multiplication by $\mathbf{C}^T$—Replication of the given (column) vectors corresponding to each column cluster along all the constituent columns.

(c) Left multiplication by $\mathbf{E}_m$—Replication of a given (row) vector along all the rows.

(d) Right multiplication by $\mathbf{E}_n^T$—Replication of a given (column) vector along all the columns

For example, the conditional expectation $E[Z|\hat{U},\hat{V}]$ involves partitioning along $(\hat{U},\hat{V})$, that is, both row and column clusters. Since there are k row clusters and l column clusters, there are kl partitions (or co-clusters) and a conditional expectation value corresponding to each of these partitions. To obtain these expectation values, we need to aggregate the rows into the row clusters as well as the columns into column clusters. In particular, the conditional expectation values are given by

$$E[Z|\hat{u},\hat{v}] = \bar{z}_{\hat{u},\hat{v}} \text{ where } \bar{\mathbf{Z}}_{\hat{U},\hat{V}} = \left(\mathbf{R}^T(\mathbf{W} \otimes \mathbf{Z})\mathbf{C}\right) \oslash \left(\mathbf{R}^T\mathbf{W}\mathbf{C}\right) .$$

Though seemingly complicated, the above expression has a simple interpretation in terms of the aggregation and replication operators described earlier. Operation $\mathbf{W} \otimes \mathbf{Z}$ has the effect of attenuating each element z_{uv} by its corresponding weight w_{uv}. Left multiplication by $\mathbf{R}^T$ aggregates the matrix

$E[Z\|\mathcal{G}]$	$\bar{\mathbf{Z}}_{\mathcal{G}}$	size($\bar{\mathbf{Z}}_{\mathcal{G}}$)	$\mathbf{Z}^f_{\mathcal{G}}$ $(m \times n)$
$E[Z]$	$(\mathbf{E}_m^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{E}_n) \oslash (\mathbf{E}_m^T\mathbf{W}\mathbf{E}_n)$	1×1	$\mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T$
$E[Z\|U]$	$((\mathbf{W}\otimes\mathbf{Z})\mathbf{E}_n) \oslash (\mathbf{W}\mathbf{E}_n)$	$m\times 1$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T$
$E[Z\|V]$	$(\mathbf{E}_m^T(\mathbf{W}\otimes\mathbf{Z})) \oslash (\mathbf{E}_m^T\mathbf{W})$	$1\times n$	$\mathbf{E}_m\bar{\mathbf{Z}}_V$
$E[Z\|\hat{U}]$	$(\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{E}_n) \oslash (\mathbf{R}^T\mathbf{W}\mathbf{E}_n)$	$k\times 1$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$
$E[Z\|\hat{V}]$	$(\mathbf{E}_m^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{E}_m^T\mathbf{W}\mathbf{C})$	$1\times l$	$\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T$
$E[Z\|U,\hat{V}]$	$((\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{W}\mathbf{C})$	$m\times l$	$\bar{\mathbf{Z}}_{U,\hat{V}}\mathbf{C}^T$
$E[Z\|\hat{U},V]$	$(\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})) \oslash (\mathbf{R}^T\mathbf{W})$	$k\times n$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},V}$
$E[Z\|\hat{U},\hat{V}]$	$(\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{R}^T\mathbf{W}\mathbf{C})$	$k\times l$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$
$E[Z\|U,V]$	$(\mathbf{W}\otimes\mathbf{Z}) \oslash \mathbf{W}$	$m\times n$	$\bar{\mathbf{Z}}_{U,V}$

Table 8: Conditional expectations in matrix notation.

along rows in the same row cluster across each column, and then right multiplication by $\mathbf{C}$ aggregates this reduced matrix consisting of row cluster sums along columns in the same column cluster. Thus, each element of $\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{C}$ represents the sum along each co-cluster of the attenuated $\mathbf{Z}$. Similarly, the matrix $\mathbf{R}^T\mathbf{W}\mathbf{C}$ contains the probability mass assigned to the different co-cluster by $\mathbf{W}$ and the element-wise division results in $k\times l$ matrix whose $\hat{u}\hat{v}^{th}$ entry is the expected value in $\hat{u}\hat{v}^{th}$ co-cluster.

To obtain a $m\times n$ full matrix $\mathbf{Z}^f_{\hat{U},\hat{V}}$ such that $z^f_{uv} = E[Z|\rho(u),\gamma(v)]$, we need to replicate the co-cluster values along the rows and columns corresponding to the row and column clusters respectively. Hence, the reconstructed matrix

$$\mathbf{Z}^f_{\hat{U},\hat{V}} = \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T = \mathbf{R}\left(\left(\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{C}\right) \oslash \left(\mathbf{R}^T\mathbf{W}\mathbf{C}\right)\right)\mathbf{C}^T.$$

Table 8 shows the matrices corresponding to the various conditional expectations. Note that the number of independent parameters in $\mathbf{Z}^f_{\mathcal{G}}$ (in Table 8) is equal to that in $\bar{\mathbf{Z}}_{\mathcal{G}}$ in spite of the difference in the matrix sizes.

E.2 Bregman Co-clustering Algorithm for Special Cases

We will now consider the three special cases mentioned above and illustrate how the various steps in the Bregman co-clustering algorithm can be instantiated.

E.2.1 CASE A: BASIS $\mathcal{C}_2$ AND ANY BREGMAN DIVERGENCE

1. **Obtaining the MBI Solution.** From Theorem 1, we note that the MBI solution for case A is $\hat{Z} = E[Z|\hat{U},\hat{V}]$. From Table 8, the corresponding MBI matrix $\hat{\mathbf{Z}}$ is given by $\mathbf{Z}^f_{\hat{U},\hat{V}} = \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$ where $\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$ is computed as $(\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{R}^T\mathbf{W}\mathbf{C})$. Since $\hat{\mathbf{Z}}$ is completely determined by the smaller $k\times l$ matrix $\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$, we only compute and store the reduced matrix. Using the fact that $\mathbf{R}$ and $\mathbf{C}$ are binary matrices, this computation can be performed efficiently using $O(mn)$ operations.

2. **Row Cluster Assignment Step.** Given the parameters of the MBI solution and a fixed column clustering determined by $\mathbf{C}$, we want to find for each row, the row cluster assignment that leads to the best approximation to the original matrix. In other words, we are searching for a

row cluster membership matrix $\mathbf{R}'$ that results in the most accurate reconstruction $\tilde{\mathbf{Z}}$. For the current case, this reconstructed matrix $\tilde{\mathbf{Z}}$ takes the same functional form as the MBI solution and is given by $\mathbf{R}'\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$ where $\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$ is based on the previous row clustering. From step 2B of the Bregman co-clustering algorithm (Algorithm 2), the optimal row assignment for each row u is given by

$$\begin{aligned}
\rho^*(u) &= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} E_{V|u}[d_\phi(Z,\tilde{Z})] = \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv} d_\phi(z_{uv},\tilde{z}_{uv}),\ [u]_1^m, \\
\overset{(a)}{\Rightarrow}\quad \mathbf{R}^* &= \operatorname*{argmin}_{\mathbf{R}'} d_{\Phi_w}(\mathbf{Z},\tilde{\mathbf{Z}}) = \operatorname*{argmin}_{\mathbf{R}'} d_{\Phi_w}(\mathbf{Z},\mathbf{R}'\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T), \\
\overset{(b)}{\Rightarrow}\quad \mathbf{R}^* &= \operatorname*{argmin}_{\mathbf{R}'} d_{\Phi_w}(\mathbf{Z}^{rowRed},\mathbf{R}'\bar{\mathbf{Z}}_{\hat{U},\hat{V}}), \\
\overset{(c)}{\Rightarrow}\quad \rho^*(u) &= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{h=1}^{l} w_{uh} d_\phi(z_{uh}^{rowRed},\bar{z}_{gh}),\ [u]_1^m,
\end{aligned}$$

where $\mathbf{Z}^{rowRed} \equiv ((\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{W}\mathbf{C})$ and d_{Φ_w} is the induced Bregman divergence that applies to matrices in $\mathcal{S}^{k\times n}$.[16] In the above, (a) and (c) follow from the definition of the row cluster membership matrix, and (b) follows from the fact that minimizing the (weighted) average Bregman divergence from a set $\{\mathbf{x}_i\}_{i=1}^n$ to a fixed point $\mathbf{a}$ is equivalent to minimizing the Bregman divergence between the (weighted) average of the set and $\mathbf{a}$ (Banerjee et al., 2005b). Assuming the matrix $\mathbf{Z}^{rowRed}$ is computed apriori, the row clustering only requires $O(mkl)$ operations as opposed to $O(mkn)$ since for each row, we only compare the reduced rows (of size $1\times l$) in $\mathbf{Z}^{rowRed}$ with the k possible row cluster representatives.

3. **Column Cluster Assignment Step.** The column cluster assignment step is similar to that of the previous row cluster assignment step and involves finding a column cluster membership matrix $\mathbf{C}'$ that results in the most accurate reconstruction $\tilde{\mathbf{Z}} = \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}'^T$. From step 2C of the Bregman co-clustering algorithm (Algorithm 2), the optimal column assignment for each column v is given by

$$\begin{aligned}
\gamma^*(v) &= \operatorname*{argmin}_{h\in\{1,\cdots,l\}} E_{U|v}[d_\phi(Z,\tilde{Z})] = \operatorname*{argmin}_{h\in\{1,\cdots,l\}} \sum_{u=1}^{m} w_{uv} d_\phi(z_{uv},\tilde{z}_{uv}),\ [v]_1^n, \\
\overset{(a)}{\Rightarrow}\quad \mathbf{C}^* &= \operatorname*{argmin}_{\mathbf{C}'} d_{\Phi_w}(\mathbf{Z},\tilde{\mathbf{Z}}) = \operatorname*{argmin}_{\mathbf{C}'} d_{\Phi_w}(\mathbf{Z},\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}'^T), \\
\overset{(b)}{\Rightarrow}\quad \mathbf{C}^* &= \operatorname*{argmin}_{\mathbf{C}'} d_{\Phi_w}(\mathbf{Z}^{colRed},\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}'^T), \\
\overset{(c)}{\Rightarrow}\quad \gamma^*(v) &= \operatorname*{argmin}_{h\in\{1,\cdots,l\}} \sum_{g=1}^{k} w_{gv} d_\phi(z_{gv}^{colRed},\bar{z}_{gh}),\ [v]_1^n,
\end{aligned}$$

where $\mathbf{Z}^{colRed} \equiv (\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})) \oslash (\mathbf{R}^T\mathbf{W})$, (a) and (c) follow from the definition of the column cluster membership matrix, and (b) follows from the same reduction (Banerjee et al., 2005b)

16. Note that d_{Φ_w} has been overloaded to denote the separable Bregman divergences induced from the original d_ϕ and the measure w that apply to matrices in $\mathcal{S}^{m\times n}$, $\mathcal{S}^{k\times n}$ and $\mathcal{S}^{m\times l}$.

employed in the row cluster assignment step. As in the previous case, the column clustering involves a reduced number of distance computations and comparisons and in particular, requires $O(nkl)$ operations.

E.2.2 CASE B: SQUARED EUCLIDEAN DISTANCE

1. **Obtaining the MBI Solution.** For this case, the MBI solution $\hat{Z}$ has a closed form for all the six co-clustering bases in terms of the appropriate conditional expectations as shown in Table 2. Using Table 8, we can exactly compute each of the relevant conditional expectations, which requires $O(mn)$ operations. Though we do not explicitly compute it, the MBI matrix $\hat{\mathbf{Z}}$ (shown in Table 9) can be expressed in terms of the row clustering $\mathbf{R}$, column clustering $\mathbf{C}$ and these conditional expectations for any co-clustering basis.

2. **Row Cluster Assignment Step.** To obtain the row cluster assignment step, we observe that the reconstructed matrix $\tilde{\mathbf{Z}}$, which has the same form as $\hat{\mathbf{Z}}$ can be split into two additive terms of which only one depends on the candidate row clustering. In particular, for the row assignment step, the reconstructed matrix $\tilde{\mathbf{Z}}$ based on a candidate row clustering $\mathbf{R}'$ can be written as

$$\tilde{\mathbf{Z}} = \tilde{\mathbf{Z}}^{rowConst} + \mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}, \tag{33}$$

where $\tilde{\mathbf{Z}}^{rowConst}$ is an $m \times n$ matrix corresponding to the constant part of $\tilde{\mathbf{Z}}$ and $\tilde{\mathbf{Z}}^{rowVar}$ is a $k \times n$ matrix corresponding to the variable part of $\tilde{\mathbf{Z}}$. Table 10 provides the $\tilde{\mathbf{Z}}^{rowConst}$ and $\tilde{\mathbf{Z}}^{rowVar}$ for the different co-clustering bases. From step 2B of Algorithm 2 and (33), the row cluster update step for squared Euclidean distance is given by

$$\begin{aligned}
\rho^*(u) &= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} E_{V|u}[(Z-\tilde{Z})^2] = \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv}(z_{uv}-\tilde{z}_{uv})^2,\ [u]_1^m, \\
\Rightarrow \quad \mathbf{R}^* &= \operatorname*{argmin}_{\mathbf{R}'} ||\mathbf{Z}-\tilde{\mathbf{Z}}||_w^2 = \operatorname*{argmin}_{\mathbf{R}'} ||\mathbf{Z}-\tilde{\mathbf{Z}}^{rowConst}-\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}||_w^2, \\
\Rightarrow \quad \mathbf{R}^* &= \operatorname*{argmin}_{\mathbf{R}'} ||\mathbf{Z}^{row}-\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}||_w^2, \\
\Rightarrow \quad \rho^*(u) &= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv}(z_{uv}^{row}-\tilde{z}_{gv}^{rowVar})^2,\ [u]_1^m,
\end{aligned}$$

where $\mathbf{Z}^{row} = \mathbf{Z} - \tilde{\mathbf{Z}}^{rowConst}$ and $||\cdot||_w$ is the weighted squared Euclidean distance. The optimal row assignments can, therefore, be determined by finding the *nearest* row (among k possible ones) in $\tilde{\mathbf{Z}}^{rowVar}$ for each of the m rows in $\mathbf{Z}^{row}$. The above row assignment step can be readily instantiated for any specified co-clustering basis by choosing the appropriate matrices $\tilde{\mathbf{Z}}^{rowConst}$ and $\tilde{\mathbf{Z}}^{rowVar}$ from Table 10.

For co-clustering bases $\{C_i\}_{i=1}^5$, it is possible to further optimize the above update step using the same observation as in case A, that is, minimizing the row update cost function $||\mathbf{Z}^{row}-\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}||_w^2$ is equivalent to minimizing the distortion between reduced versions of these matrices, that is, $||\mathbf{Z}^{rowRed}-\mathbf{R}'\tilde{\mathbf{Z}}^{rowVRed}||_w^2$ where $\mathbf{Z}^{rowRed} \equiv ((\mathbf{W}\otimes\mathbf{Z}^{row})\mathbf{C})\oslash(\mathbf{WC})$ and $\mathbf{R}'\tilde{\mathbf{Z}}^{rowVRed} \equiv ((\mathbf{W}\otimes(\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}))\mathbf{C})\oslash(\mathbf{WC})$. Though the expression for $\tilde{\mathbf{Z}}^{rowVRed}$ looks complicated, it can be simplified using the fact that $\tilde{\mathbf{Z}}^{rowVar}$ can always be written as $\mathbf{AC}^T+\mathbf{BE}_n^T$ for some matrices $\mathbf{A}$ and $\mathbf{B}$, which ensures that $\tilde{\mathbf{Z}}^{rowVRed} = \mathbf{A}+\mathbf{BE}_l^T$, that is, independent of $\mathbf{R}'$. For all the five co-clustering bases, $\tilde{\mathbf{Z}}^{rowVRed}$ is determined by the relevant conditional

Co-clustering basis $\mathcal{C}$	$\hat{\mathbf{Z}}$ $(m \times n)$
$\mathcal{C}_1$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T + \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T - \mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T$
$\mathcal{C}_2$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$
$\mathcal{C}_3$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T + \bar{\mathbf{Z}}_U\mathbf{E}_n^T - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$
$\mathcal{C}_4$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T + \mathbf{E}_m\bar{\mathbf{Z}}_V - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T$
$\mathcal{C}_5$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T + \bar{\mathbf{Z}}_U\mathbf{E}_n^T - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T + \mathbf{E}_m\bar{\mathbf{Z}}_V - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T$
$\mathcal{C}_6$	$\bar{\mathbf{Z}}_{U,\hat{V}}\mathbf{C}^T + \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},V} - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$

Table 9: MBI matrix for squared Euclidean distance.

Co-clustering basis $\mathcal{C}$	$\tilde{\mathbf{Z}}^{rowConst}$ $(m \times n)$	$\tilde{\mathbf{Z}}^{rowVar}$ $(k \times n)$	$\tilde{\mathbf{Z}}^{rowVRed}$ $(k \times l)$
$\mathcal{C}_1$	$\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T - \mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T$
$\mathcal{C}_2$	$\mathbf{0}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_3$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T - \bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T$
$\mathcal{C}_4$	$\mathbf{E}_m\bar{\mathbf{Z}}_V - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_5$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T + \mathbf{E}_m\bar{\mathbf{Z}}_V - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T - \bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T$
$\mathcal{C}_6$	$\bar{\mathbf{Z}}_{U,\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},V} - \bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$	n/a

Table 10: Row assignment update matrices for squared Euclidean distance.

expectations and can be looked up from Table 10. As a result of this optimization, the row clustering step involves comparisons between smaller matrices and requires only $O(mkl)$ operations.

3. **Column Cluster Assignment Step.** The column assignment step employs a similar decomposition of $\tilde{\mathbf{Z}}$ in terms of the column clustering, that is, $\tilde{\mathbf{Z}} = \tilde{\mathbf{Z}}^{colConst} + \tilde{\mathbf{Z}}^{colVar}\mathbf{C}'^T$ and the optimal assignments are determined by

$$\gamma(v) = \operatorname*{argmin}_{h\in\{1,\cdots,l\}} \sum_{u=1}^{m} w_{uv}(z_{uv}^{col} - \tilde{z}_{uh}^{colVar})^2,\ [v]_1^n,$$

where $\mathbf{Z}^{col} \equiv \mathbf{Z} - \tilde{\mathbf{Z}}^{colConst}$ and the matrices $\tilde{\mathbf{Z}}^{colConst}$ and $\tilde{\mathbf{Z}}^{colVar}$ can be obtained from Table 11. As in the case of row clustering, it is possible to further optimize the above update step for co-clustering bases $\{C_i\}_{i=1}^5$ by computing $\mathbf{Z}^{colRed} \equiv (\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z}^{col}))\oslash(\mathbf{R}^T\mathbf{W})$ and comparing it with $\tilde{\mathbf{Z}}^{colVRed}\mathbf{C}^T \equiv (\mathbf{R}^T(\mathbf{W}\otimes(\tilde{\mathbf{Z}}^{colVar}\mathbf{C}^T)))\oslash(\mathbf{R}^T\mathbf{W})$, which can be obtained from Table 11. Further, as in the previous step, the column clustering step only requires $O(nkl)$ operations similar to that in case A.

E.2.3 CASE C: I-DIVERGENCE

1. **Obtaining the MBI Solution.** As in the previous case, the MBI solution for case C has a closed form for all the six co-clustering bases in terms of the appropriate conditional expectations as shown in Table 1. Using Table 8, one can exactly compute each of the relevant conditional expectations, which completely determine the MBI matrix $\hat{\mathbf{Z}}$ (shown in Table 12) for a given row clustering $\mathbf{R}$ and column clustering $\mathbf{C}$.

Co-clustering basis $\mathcal{C}$	$\tilde{\mathbf{Z}}^{colConst}$ $(m \times n)$	$\tilde{\mathbf{Z}}^{colVar}$ $(m \times l)$	$\tilde{\mathbf{Z}}^{colVRed}$ $(k \times l)$
$\mathcal{C}_1$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T - \mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T$	$\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}$	$\mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}}$
$\mathcal{C}_2$	$\mathbf{0}$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_3$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_4$	$\mathbf{E}_m\bar{\mathbf{Z}}_V$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}}$
$\mathcal{C}_5$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T + \mathbf{E}_m\bar{\mathbf{Z}}_V - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}} - \mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}}$
$\mathcal{C}_6$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},V}$	$\bar{\mathbf{Z}}_{U,\hat{V}} - \mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$	n/a

Table 11: Column assignment update matrices for squared Euclidean distance.

2. **Row Cluster Assignment Step.** To obtain the row assignment steps for I-divergence, we make use of the fact that the reconstructed matrix $\tilde{\mathbf{Z}}$, can be decomposed as the Hadamard product of two terms of which only one depends on the candidate row or column clustering. In particular, the reconstructed matrix $\tilde{\mathbf{Z}}$ can be expressed as

$$\tilde{\mathbf{Z}} = (\tilde{\mathbf{Z}}^{rowConst}) \otimes (\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}),$$

where $\tilde{\mathbf{Z}}^{rowConst}$ is the constant factor and $\tilde{\mathbf{Z}}^{rowVar}$ is the variable factor that depends on $\mathbf{R}'$, both of which can be looked up from Table 13.

From step 2B of Algorithm 2 and (33), the row cluster update step for I-divergence for $[u]_1^m$ is given by

$$\begin{aligned}
\rho^*(u) &= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} E_{V|u}\left[Z\log\left(\frac{Z}{\tilde{Z}}\right) - Z + \tilde{Z}\right], \\
&= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv}\left(z_{uv}\log\left(\frac{z_{uv}}{\tilde{z}_{uv}}\right) - z_{uv} + \tilde{z}_{uv}\right), \\
&= \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv}\left(z_{uv}\log\left(\frac{z_{uv}}{\tilde{z}_{uv}^{rowConst}}\right) - z_{uv}\right) \\
&\qquad + \sum_{v=1}^{n} w_{uv}(\tilde{z}_{uv}^{rowConst}\tilde{z}_{\rho'(u)v}^{rowVar} - z_{uv}\log(\tilde{z}_{\rho'(u)v}^{rowVar})), \\
&\overset{(a)}{=} \operatorname*{argmin}_{g\in\{1,\cdots,k\}} \sum_{v=1}^{n} w_{uv}\left(\tilde{z}_{uv}^{rowConst}\tilde{z}_{\rho'(u)v}^{rowVar} - z_{uv}\log(\tilde{z}_{\rho'(u)v}^{rowVar})\right),
\end{aligned}$$

where (a) follows since the first term in the cost function is independent of the row clustering.

As in case B, it is possible to optimize the row assignment step for the co-clustering bases $\{C_i\}_{i=1}^5$ by minimizing a simplified row update cost function $d_{\Phi_w}(\mathbf{Z}^{rowRed}, \tilde{\mathbf{Z}}^{rowCRed} \otimes \mathbf{R}'\tilde{\mathbf{Z}}^{rowVRed})$ based on equivalent reduced matrices instead of the original cost function $d_{\Phi_w}(\mathbf{Z}, \tilde{\mathbf{Z}}^{rowConst} \otimes \mathbf{R}'\tilde{\mathbf{Z}}^{rowVar})$ where $\mathbf{Z}^{rowRed} \equiv ((\mathbf{W}\otimes\mathbf{Z})\mathbf{C}) \oslash (\mathbf{W}\mathbf{C})$, $\mathbf{Z}^{rowCRed} \equiv ((\mathbf{W}\otimes\mathbf{Z}^{rowConst})\mathbf{C}) \oslash (\mathbf{W}\mathbf{C})$, and $\mathbf{R}'\tilde{\mathbf{Z}}^{rowVRed} \equiv ((\mathbf{W}\otimes(\mathbf{R}'\tilde{\mathbf{Z}}^{rowVar}))\mathbf{C}) \oslash (\mathbf{W}\mathbf{C})$. Further as in the previous case, $\tilde{\mathbf{Z}}^{rowVRed}$ can be simplified by noticing that $\tilde{\mathbf{Z}}^{rowVar}$ in this case can be written as $\mathbf{A}\mathbf{C}^T \otimes \mathbf{B}\mathbf{E}_n^T$ for some matrices $\mathbf{A}$ and $\mathbf{B}$, ensuring that $\tilde{\mathbf{Z}}^{rowVRed} = \mathbf{A} \otimes (\mathbf{B}\mathbf{E}_l^T)$, that is, independent of $\mathbf{R}'$. Table 13 shows the matrix $\tilde{\mathbf{Z}}^{rowVRed}$ for the different co-clustering bases.

Co-clustering basis $\mathcal{C}$	$\hat{\mathbf{Z}}$ $(m \times n)$
$\mathcal{C}_1$	$((\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T)) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T)$
$\mathcal{C}_2$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$
$\mathcal{C}_3$	$((\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T) \otimes (\bar{\mathbf{Z}}_U\mathbf{E}_n^T)) \oslash (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T)$
$\mathcal{C}_4$	$((\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_V)) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T)$
$\mathcal{C}_5$	$((\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T) \otimes (\bar{\mathbf{Z}}_U\mathbf{E}_n^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_V)) \oslash ((\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T))$
$\mathcal{C}_6$	$((\bar{\mathbf{Z}}_{U,\hat{V}}\mathbf{C}^T) \otimes (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},V})) \oslash (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T)$

Table 12: MBI matrix for I-divergence.

Co-clustering basis $\mathcal{C}$	$\tilde{\mathbf{Z}}^{rowConst}$ $(m \times n)$	$\tilde{\mathbf{Z}}^{rowVar}$ $(k \times n)$	$\tilde{\mathbf{Z}}^{rowVRed}$ $(k \times l)$
$\mathcal{C}_1$	$(\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T)$	$\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T$	$\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T$
$\mathcal{C}_2$	$\mathbf{E}_{mn}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_3$	$\bar{\mathbf{Z}}_U\mathbf{E}_n^T$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T) \oslash (\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T)$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T)$
$\mathcal{C}_4$	$(\mathbf{E}_m\bar{\mathbf{Z}}_V) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T)$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_5$	$((\bar{\mathbf{Z}}_U\mathbf{E}_n^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_V)) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}\mathbf{C}^T)$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T) \oslash (\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T)$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_l^T)$
$\mathcal{C}_6$	$\bar{\mathbf{Z}}_{U,\hat{V}}\mathbf{C}^T$	$(\bar{\mathbf{Z}}_{\hat{U},V}) \oslash (\bar{\mathbf{Z}}_{\hat{U},\hat{V}}\mathbf{C}^T)$	n/a

Table 13: Row assignment update matrices for I-divergence.

3. **Column Cluster Assignment Step.** The optimal column assignments can be obtained in similar fashion by computing the matrices $\tilde{\mathbf{Z}}^{colConst}$ and $\tilde{\mathbf{Z}}^{colVar}$ shown in Table 14 and optimizing the part of the column update cost function that depends on the column clustering, that is,

$$\gamma(v) = \underset{h\in\{1,\cdots,l\}}{\operatorname{argmin}} \sum_{u=1}^{n} w_{uv}\left(\tilde{z}_{uv}^{colConst}\tilde{z}_{uh}^{colVar} - z_{uv}\log(\tilde{z}_{uh}^{colVar})\right),\ [v]_1^n.$$

Further, as in the row clustering case, the column assignment step can be optimized further for co-clustering bases $\{C_i\}_{i=1}^5$ by computing $\mathbf{Z}^{colRed} \equiv (\mathbf{R}^T(\mathbf{W}\otimes\mathbf{Z})) \oslash (\mathbf{R}^T\mathbf{W})$, $\mathbf{Z}^{colCRed} \equiv (\mathbf{R}^T(\mathbf{W}\otimes\tilde{\mathbf{Z}}^{colConst})) \oslash (\mathbf{R}^T\mathbf{W})$ and $\tilde{\mathbf{Z}}^{colVRed}C^T \equiv (\mathbf{R}^T(\mathbf{W}\otimes(\tilde{\mathbf{Z}}^{colVar}C^T))) \oslash (\mathbf{R}^T\mathbf{W})$, using Table 14 and finding the column clustering $\mathbf{C}'$ that optimizes the cost $d_{\Phi_w}(\mathbf{Z}^{colRed}, \tilde{\mathbf{Z}}^{colCRed} \otimes \tilde{\mathbf{Z}}^{colVRed}\mathbf{C})$. The computational time for these update steps is same as in the cases A and B.

E.2.4 CASE D: ANY BREGMAN DIVERGENCE AND CO-CLUSTERING BASIS

The proposed meta-algorithm can be instantiated for any Bregman divergence and co-clustering basis. We now consider a particular example of the general case corresponding to Itakura-Saito distance, which is the Bregman divergence corresponding to the convex function $\phi(z) = -\log(z)$, a uniform measure and the co-clustering basis $\mathcal{C}_1$. The example is a representative of the general case, since no divergence/basis specific optimizations are possible in this case.

1. **Obtaining the MBI Solution.** For the general case involving a Bregman divergence other than squared Euclidean distance and I-divergence and a co-clustering basis different from $\mathcal{C}_2$, the MBI solution does not have a closed form, which makes it necessary to use a convex

Co-clustering basis $\mathcal{C}$	$\tilde{\mathbf{Z}}^{colConst}$ $(m \times n)$	$\tilde{\mathbf{Z}}^{colVar}$ $(m \times l)$	$\tilde{\mathbf{Z}}^{colVRed}$ $(k \times l)$
$\mathcal{C}_1$	$(\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}\mathbf{E}_n^T)$	$\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}}$	$\mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}}$
$\mathcal{C}_2$	$\mathbf{E}_{mn}$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_3$	$(\bar{\mathbf{Z}}_U\mathbf{E}_n^T) \oslash (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T)$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$	$\bar{\mathbf{Z}}_{\hat{U},\hat{V}}$
$\mathcal{C}_4$	$\mathbf{E}_m\bar{\mathbf{Z}}_V$	$(\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}})$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}})$
$\mathcal{C}_5$	$(\bar{\mathbf{Z}}_U\mathbf{E}_n^T) \otimes (\mathbf{E}_m\bar{\mathbf{Z}}_V) \oslash (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U}}\mathbf{E}_n^T)$	$(\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\mathbf{E}_m\bar{\mathbf{Z}}_{\hat{V}})$	$(\bar{\mathbf{Z}}_{\hat{U},\hat{V}}) \oslash (\mathbf{E}_k\bar{\mathbf{Z}}_{\hat{V}})$
$\mathcal{C}_6$	$\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},V}$	$(\bar{\mathbf{Z}}_{U,\hat{V}}) \oslash (\mathbf{R}\bar{\mathbf{Z}}_{\hat{U},\hat{V}})$	n/a

Table 14: Column assignment update matrices for I-divergence.

optimization algorithm (e.g., Bregman's algorithm or Iterative Scaling algorithm). Further, since the reconstructed $\tilde{Z}$ is defined in terms of the optimal Lagrange multipliers, we also need to compute these Lagrange parameters from the MBI solution. For the example under consideration, $\nabla\phi(z) = -\frac{1}{z}$. Hence, using the notation in Section 5.5, the matrix $\mathbf{A}$ for co-clustering basis $\mathcal{C}_1$ corresponds to a $(k+l) \times mn$ membership matrix where the rows correspond to the clusters (first k rows to row clusters and the next l rows to the column clusters) and the columns correspond to the elements of the matrix $\mathbf{Z}$ (or the corresponding $mn \times 1$ vector $\mathbf{z}$). Assuming $\mathbf{E}_{mn}$ is $mn \times 1$ vector consisting of all ones, the update steps in Bregman's algorithm (Section 5.5.1) are, therefore, given by

$$\begin{aligned} \mathbf{E}_{mn} \oslash \mathbf{z}'^{t+1} &= \mathbf{E}_{mn} \oslash \mathbf{z}'^{t} + \lambda_i A_i^T \\ A_i\mathbf{z}^{t+1} &= A_i\mathbf{z}, \end{aligned}$$

where A_i is the i^{th} row in $\mathbf{A}$ and $\lambda_i \in \mathbb{R}$. These updates are cyclically repeated over all the $k+l$ rows in $\mathbf{A}$. On convergence, we get the MBI solution $\hat{\mathbf{Z}}$ ($m \times n$ matrix) as well as the $k \times 1$ and $1 \times l$ matrices $\Lambda_{\hat{U}}, \Lambda_{\hat{V}}$ containing the optimal Lagrange multipliers.

2. **Row Cluster Assignment Step.** To obtain the row cluster assignment step, we first reconstruct $\tilde{\mathbf{Z}}$ for a candidate co-clustering $\mathbf{R}'$ using the Lagrange multipliers $\Lambda_{\hat{U}}$ and $\Lambda_{\hat{V}}$ computed in the previous step. More specifically, the reconstruction $\tilde{\mathbf{Z}}$ is given by

$$\tilde{\mathbf{Z}} = \mathbf{E}_{mn} \oslash (\bar{\mathbf{Z}} - \mathbf{R}'\Lambda_{\hat{U}}\mathbf{E}_n^T - \mathbf{E}_m\Lambda_{\hat{V}}\mathbf{C}^T), \tag{34}$$

that is, $\tilde{z}_{uv} = 1/(\bar{z} - \lambda_{\rho'(u)} - \lambda_{\gamma(v)})$.

Using (34) the row update cost function reduces to

$$\begin{aligned} & E_{V|u}[d_\phi(Z, \tilde{Z}] \\ &= E_{V|u}[Z/\tilde{Z} - \log(Z/\tilde{Z}) - 1] = \sum_{v=1}^{n} w_{uv}(z_{uv}/\tilde{z}_{uv} - \log(z_{uv}/\tilde{z}_{uv}) - 1) \\ &= \sum_{v=1}^{n} w_{uv}(z_{uv}(\bar{z} - \lambda_{\rho'(u)} - \lambda_{\gamma(v)}) - \log(z_{uv}) + \log(\bar{z} - \lambda_{\rho'(u)} - \lambda_{\gamma(v)}) - 1) \\ &= \sum_{v=1}^{n} w_{uv}(z_{uv}(\bar{z} - \lambda_{\gamma(v)}) - \log(z_{uv}) - 1) + \sum_{v=1}^{n} w_{uv}(-z_{uv}\lambda_{\rho'(u)} + \log(\bar{z} - \lambda_{\rho'(u)} - \lambda_{\gamma(v)})). \end{aligned}$$

Since the first term is independent of the row clustering, it is sufficient to optimize only the second term. Hence, the row assignment step is given by

$$\rho(u) = \underset{g\in\{1,\cdots,k\}}{\operatorname{argmin}} \sum_{v=1}^{n} w_{uv}(-z_{uv}\lambda_g + \log(\overline{z} - \lambda_g - \lambda_{\gamma(v)})),\ [u]_1^m.$$

3. **Column Cluster Assignment Step.** The column assignment step can be similarly obtained by substituting the appropriate reconstructed matrix $\tilde{\mathbf{Z}}$ into the column update cost function and optimizing the part that depends on the column clustering, that is,

$$\gamma(v) = \underset{h\in\{1,\cdots,l\}}{\operatorname{argmin}} \sum_{u=1}^{m} w_{uv}(-z_{uv}\lambda_h + \log(\overline{z} - \lambda_{\rho(u)} - \lambda_h)),\ [v]_1^n.$$

Appendix F. Notation

Notation	Usage	Introduced in
X, Y	Random variables over $\{x_1,\ldots,x_m\}$ and $\{y_1,\ldots,y_n\}$	Sec 1.1
m, n	Cardinality of support sets of X and Y	Sec 1.1
u, v	Indices over the sets $\{1,\cdots,m\}$ and $\{1,\cdots,n\}$	Sec 1.1
$\hat{X}, \hat{Y}$	Compressed/clustered versions of random variables X and Y	Sec 1.1
k, l	Number of row and column clusters	Sec 1.1
g, h	Indices over the sets $\{1,\cdots k\}$ and $\{1,\cdots,l\}$	Sec 1.1
$p(\cdot)$	Given joint (and induced) distributions over X, Y, $\hat{X}$ and $\hat{Y}$	Sec 1.1
$p'(\cdot)$	Candidate joint (and induced) distributions over X, Y, $\hat{X}$ and $\hat{Y}$	Sec 1.1
$q(\cdot)$	Max. entropy joint (and induced) distributions over X, Y, $\hat{X}$ and $\hat{Y}$	Sec 1.1
$p_0(\cdot)$	Uniform joint (and induced) distributions over X, Y, $\hat{X}$ and $\hat{Y}$	Sec 1.1
$\phi(\cdot)$	Strictly convex, differentiable function of Legendre type	Sec 2.1
$d_\phi(\cdot)$	Bregman divergence derived from ϕ	Sec 2.1
S	Effective domain of ϕ	Sec 2.1
z, z_i	Elements of S	Sec 2.1
Z	Random variable taking values in S	Sec 2.1
$\mathcal{Z}$	Support of Z	Sec 2.1
w	Probability measure associated with random variable Z	Sec 2.1
$\mathbf{Z}$	Matrix $\in S^{m\times n}$	Sec 2.1
U, V	Random variables over $\{1,\ldots,m\}$ and $\{1,\ldots,n\}$	Sec 2.2
ρ, γ	Row and column cluster mapping	Sec 2.3
$\hat{U}, \hat{V}$	Cluster random variables $\rho(U)$ and $\gamma(V)$	Sec 2.3
$\hat{\mathbf{Z}}$	Matrix approximation of $\mathbf{Z}$ (size $m\times n$)	Sec 2.3
$\hat{Z}$	Random variable approximating Z	Sec 2.3
Φ_w	Convex function induced on matrix by ϕ	Sec 2.3

Table 15: Notation used in the paper

Notation	Usage	Introduced in
$\hat{u}, \hat{v}$	Indices representing $\rho(u)$ and $\gamma(v)$	Sec 3.1
$\mathcal{S}_A$	Set of random variables preserving co-cluster means	Sec 3.1
$\hat{Z}_A$	Minimum Bregman information solution	Sec 3.1
Z'	Element of $\mathcal{S}_A$	Sec 3.1
$\mathcal{S}_B$	Set of random variables that are functions of co-cluster means	Sec 3.1
$\hat{Z}_B$	Best approximation to Z in $\mathcal{S}_B$	Sec 3.1
Z''	Element of $\mathcal{S}_B$	Sec 3.1
$\hat{Z}$	Same as $\hat{Z}_A$ and $\hat{Z}_B$	Sec 3.1
(ρ^*, γ^*)	Optimal row and column clustering	Sec 3.2
$\mu_{\hat{u},\hat{v}}$	co-cluster mean $E[Z\|\hat{u},\hat{v}]$	Sec 3.3
$J_u(\cdot)$	Contribution of u^{th} row to the objective function	Sec 3.3
ρ^t	Row clustering in the t^{th} iteration	Sec 3.3
γ^t	Column clustering in the t^{th} iteration	Sec 3.3
$\hat{Z}^t$	MBI solution corresponding to (ρ^t, γ^t)	Sec 3.3
$\tilde{Z}^t$	Row permuted version of $\hat{Z}^t$ according to ρ^t	Sec 3.3
$\mathbf{R}$	Row assignment matrix (size $m \times k$)	Sec 3.4
$\mathbf{C}$	Column assignment matrix (size $n \times l$)	Sec 3.4
$\mathbf{M}$	Co-cluster mean matrix (size $k \times l$)	Sec 3.4
$U_\emptyset$	Constant random variable over rows	Sec 4.1
$V_\emptyset$	Constant random variable over columns	Sec 4.1
Γ_1	Set of index random variables	Sec 4.1
Γ_2	Unique sub-σ-algebra of Z	Sec 4.1
$\mathcal{C}, \mathcal{C}_i$	Co-clustering basis	Sec 4.1
$\mathcal{G}, \mathcal{G}_i$	Sub-σ algebra corresponding to co-clustering basis	Sec 4.1
s	Total number of constraints in a co-clustering basis	Sec 4.2
r	Index over the set $\{1, \cdots s\}$	Sec 4.2
$\Lambda^*_{\mathcal{G}_r}, \Lambda_{\mathcal{G}_r}$	(Optimal) Lagrange multipliers associated with $\mathcal{G}_r$	Sec 4.2
$w_{\mathcal{G}_r}$	Induced measure on $\mathcal{G}_r$	Sec 4.2
$J(\cdot)$	Lagrangian for the minimum Bregman information problem	Sec 4.2
$L(\cdot)$	Lagrange dual of the Bregman information	Sec 4.2
$\mathcal{S}_A$	Set of random variables preserving summary statistics	Sec 4.2
$\hat{Z}_A$	MBI solution in $\mathcal{S}_A$	Sec 4.2
Z'	Element of $\mathcal{S}_A$	Sec 4.2
ψ	Legendre conjugate of ϕ	Sec 4.4
Θ	Domain of ψ	Sec 4.4
$\theta_{\mathcal{G}_r}$	Random variables corresponding to $E[Z\|\mathcal{G}_r]$ in Θ	Sec 4.4
Θ_B	Set of generalized additive models of $\theta_{\mathcal{G}_r}$ in Θ space	Sec 4.4
θ''	Element of Θ_B	Sec 4.4
$\mathcal{S}_B$	Set of generalized additive models of summary statistics in Θ space	Sec 4.4
$\hat{Z}_B$	Best approximation to Z in $\mathcal{S}_B$	Sec 4.4
Z''	Element of $\mathcal{S}_B$	Sec 4.4
$g_r(\cdot)$	Arbitrary function of $E[Z\|\mathcal{G}_r]$ and $\theta_{\mathcal{G}_r}$	Sec 4.4

Table 16: Notation used in the paper

Notation	Usage	Introduced in
$\zeta(\rho,\gamma,\Lambda)$	Functional form of the min. Bregman information solution for (ρ,γ) with Lagrange multipliers Λ possibly instead of optimal Λ^*	Sec 5.2
$\xi(U,\rho(U),V,\gamma(V))$	Objective function $E[d_\phi(Z,\tilde{Z})]$	Sec 5.2
$\mathbf{z},\hat{\mathbf{z}},\mathbf{z}'$	Vectorized versions of $\mathbf{Z}$, $\hat{\mathbf{Z}}$ and Z' respectively	Sec 5.5
$\overline{\mathbf{z}}$	$mn \times 1$ vector with all values $= E[Z]$	Sec 5.5
$\mathbf{A}$	Matrix corresponding to the linear conditional expectation constraints	Sec 5.5
c	Number of linear constraints (rows in $\mathbf{A}$)	Sec 5.5
$\mathcal{L}_\phi$	Legendre-Bregman projection derived from ϕ	Sec 5.5
λ_i,λ	Lagrange multipliers corresponding to A_i and $\mathbf{A}$ resp.	Sec 5.5
$\mathbf{z}'_0$	Initial choice of $\mathbf{z}'$	Sec 5.5
s_{ij}	Sign of A_{ij}	Sec 5.5
N_j	Upper bound on L_1 norm of j^{th} column of $\mathbf{A}$	Sec 5.5
$\mathbf{W}$	$m \times n$ matrix corresponding to the measure w	Sec E.1
$\mathbf{E}_m$ ($\mathbf{E}_n$)	constant $m \times 1$ ($n \times 1$) vector consisting of all ones	Sec E.1
$\mathbf{Z}_\mathcal{G}$	Matrix of conditional expectations over $\mathcal{G}$	Sec E.1
$\mathbf{Z}_\mathcal{G}^f$	$m \times n$ matrix expansion of $\overline{\mathbf{Z}}_\mathcal{G}$	Sec E.1
$\tilde{\mathbf{Z}}$	Matrix corresponding to $\tilde{Z}$	Sec E.2
ρ',γ'	Candidate row and column clustering	Sec E.2
$\mathbf{R}',\mathbf{C}'$	Candidate row and column membership matrices	Sec E.2
$\tilde{\mathbf{Z}}^{rowVar}$	Variable part of $\tilde{\mathbf{Z}}$ during row clustering (size $k \times n$)	Sec E.2
$\tilde{\mathbf{Z}}^{rowConst}$	Constant part of $\tilde{\mathbf{Z}}$ during row clustering (size $m \times n$)	Sec E.2
$\mathbf{Z}^{row}$	Constant matrix determining row-clustering (size $m \times n$)	Sec E.2
$\mathbf{Z}^{rowRed}$	Reduced representation of $\mathbf{Z}^{row}$ (size $m \times l$)	Sec E.2
$\tilde{\mathbf{Z}}^{rowVRed}$	Reduced representation of $\tilde{\mathbf{Z}}^{rowVar}$ (size $k \times l$)	Sec E.2
$\tilde{\mathbf{Z}}^{colVar}$	Variable part of $\tilde{\mathbf{Z}}$ during column clustering (size $m \times l$)	Sec E.2
$\tilde{\mathbf{Z}}^{colConst}$	Constant part of $\tilde{\mathbf{Z}}$ during column clustering (size $m \times n$)	Sec E.2
$\mathbf{Z}^{col}$	Constant matrix determining column clustering (size $m \times n$)	Sec E.2
$\mathbf{Z}^{colRed}$	Reduced representation of $\mathbf{Z}^{col}$ (size $k \times n$)	Sec E.2
$\tilde{\mathbf{Z}}^{colVRed}$	Reduced representation of $\tilde{\mathbf{Z}}^{colVar}$ (size $k \times l$)	Sec E.2
$\mathbf{E}_{m\times n}$	$m \times n$ matrix consisting of all ones	Sec E.2

Table 17: Notation used in the paper

References

K. S. Azoury and M. K. Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43(3):211–246, 2001.

A. Banerjee, X. Guo, and H. Wang. On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory*, 51(7):2664–2669, July 2005a.

A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh. Clustering with Bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005b.

H. H. Bauschke and J. M. Borowein. Legendre functions and the method of random Bregman projections. *Journal of Convex Analysis*, 4(1):27–67, 1997.

R. Bekkerman, R. El-Yaniv, and A. McCallum. Multi-way distributional clustering via pairwise interactions. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 41–48, 2005.

L. M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Physics*, 7:200–217, 1967.

R. Cai, L. Lu, and L. Cai. Unsupervised auditory scene categorization via key audio effects and information-theoretic co-clustering. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP05)*, pages 1073–1076, 2005.

J. J. M. Carrasco, D. Fain, K. Lang, and L. Zhukov. Clustering of bipartite advertiser-keyword graph. In *Proceedings of the Workshop on Large Scale Clustering, ICDM*, 2003.

Y. Censor and S. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press, 1998.

D. Chakrabarti, S. Papadimitriou, D. S. Modha, and C. Faloutsos. Fully automatic cross-associations. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 79–88, 2004.

Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 93–103, 2000.

H. Cho, I. S. Dhillon, Y. Guan, and S. Sra. Minimum sum-squared residue co-clustering of gene expression data. In *Proceedings of the 4th SIAM International Conference on Data Mining (SDM)*, pages 114–125, 2004.

M. Collins, R. E. Schapire, and Y. Singer. Logistic regression, adaboost and bregman distances. In *Proceedings of the 13th Annual Conference on Computational Learing Theory (COLT)*, pages 158–169, 2000.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

I. Csiszár. Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems. *The Annals of Statistics*, 19:2032–2066, 1991.

S. Della Pietra, V. Della Pietra, and J. Lafferty. Duality and auxiliary functions for Bregman distances. Technical Report CMU-CS-01-109, School of Computer Science, Carnegie Mellon University, 2001.

I. Dhillon, S. Mallela, and R. Kumar. A divisive information-theoretic feature clustering algorithm for text classification. *Journal of Machine Learning Research*, 3(4):1265–1287, 2003a.

I. Dhillon, S. Mallela, and D. Modha. Information-theoretic co-clustering. In *Proceedings of the 9th International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 89–98, 2003b.

I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the 7th International Conference on Knowledge Discovery and Data Mining*, pages 269–274, 2001.

I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1):143–175, January 2001.

D. Freitag. Trained named entity recognition using distributional clusters. In *EMNLP*, pages 262–269, 2004.

B. Gao, T. Liu, X. Zheng, Q. Cheng, and W. Ma. Consistent bipartite graph co-partitioning for star-structured high-order heterogeneous data co-clustering. In *Proceedings of the 11th International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 41–50, 2005.

T. George and S. Merugu. A scalable collaborative filtering framework based on co-clustering. In *Proceedings of the IEEE Conference on Data Mining*, pages 625–628, 2005.

J. Ghosh. Scalable clustering. In Nong Ye, editor, *The Handbook of Data Mining*, pages 247–277. Lawrence Erlbaum Assoc., 2003.

GroupLens. Movielens data set. http://www.cs.umn.edu/Research/GroupLens/data/ml-data.tar.gz.

P. D. Grünwald and A. Dawid. Game theory, maximum entropy, minimum discrepancy, and robust Bayesian decision theory. *Annals of Statistics*, 32(4), 2004.

J. Guan, G. Qiu, and X. Y. Xue. Spectral images and features co-clustering with application to content-based image retrieval. In *IEEE Workshop on Multimedia Signal Processing*, 2005.

J. A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

T. Hofmann. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, 22(1):89–115, 2004.

T. Hofmann and J. Puzicha. Unsupervised learning from dyadic data. Technical Report ICSI TR-98-042, International Computer Science Institute (ICSI), Berkeley, 1998.

A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, New Jersey, 1988.

E. T. Jaynes. Information theory and statistical mechanics. *Physical Reviews*, 106:620–630, 1957.

Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein. Spectral biclustering of microarray data: Coclustering genes and conditions. *Genome Research*, 13(4):703–716, 2003.

J. Lafferty. Additive models, boosting, and inference for generalized divergences. In *Proceedings of the 13th Annual Conference on Computational Learing Theory (COLT)*, 1999.

D. L. Lee and S. Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the 14th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 556–562, 2001.

H. Li and N. Abe. Word clustering and disambiguation based on co-occurence data. In *COLING-ACL*, pages 749–755, 1998.

T. Li. A general model for clustering binary data. In *Proceedings of the 11th International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 188–197, 2005.

Z. Lin and R.B. Altman. Finding haplotype tagging snps by use of principal components analysis. *The American Journal of Human Genetics*, 75:850–861, 2004.

S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: A survey. *IEEE Trans. Computational Biology and Bioinformatics*, 1(1):24–45, 2004.

C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: A probabilistic analysis. In *Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 159–168, 1998.

L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensinal data: A review. *ACM SIGKDD Explorations*, 6(1):90–105, 2004.

G. Qiu. Image and feature co-clustering. In *Proceedings of the International Conference on Pattern Recognition*, pages 991–994, 2004.

P. Resnick, N. Iacovou, M. Suchak, P. Bergstorm, and J. Riedl. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of the ACM Conference on CSCW*, pages 175–186, 1994.

R. T. Rockafellar. *Convex Analysis*. Princeton Landmarks in Mathematics. Princeton University Press, 1970.

R. Rohwer and D. Freitag. Towards full automation of lexicon construction. In Dan Moldovan and Roxana Girju, editors, *HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, pages 9–16, 2004.

B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Application of dimensionality reduction in recommender systems–a case study. In *WebKDD Workshop.*, 2000.

J. Shore and R. Johnson. Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross entropy. *IEEE Transactions on Information Theory*, 26(1):26–37, 1980.

A. Strehl and J. Ghosh. Cluster ensembles – a knowledge reuse framework for combining partitionings. *Journal of Machine Learning Research*, 3(3):583–617, 2002.

H. Takamura and Y. Matsumoto. Co-clustering for text categorization. *Information Processing Society of Japan Journal*, 2003.

H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 819–826, 2004.