Journal of Machine Learning Research ? (20??) ??-?? Submitted 3/10; Revised 9/10; Published ??/??

# PAC-Bayesian Analysis of Co-clustering and Beyond

**Yevgeny Seldin** SELDIN@TUEBINGEN.MPG.DE
*Max Planck Institute for Biological Cybernetics*
*Tübingen, Germany*
*School of Computer Science and Engineering*
*The Hebrew University of Jerusalem, Israel*

**Naftali Tishby** TISHBY@CS.HUJI.AC.IL
*School of Computer Science and Engineering*
*Interdisciplinary Center for Neural Computation*
*The Hebrew University of Jerusalem, Israel*

**Editor:** Inderjit S. Dhillon

## Abstract

We derive PAC-Bayesian generalization bounds for supervised and unsupervised learning models based on clustering, such as co-clustering, matrix tri-factorization, graphical models, graph clustering, and pairwise clustering.[1] We begin with the analysis of co-clustering, which is a widely used approach to the analysis of data matrices. We distinguish among two tasks in matrix data analysis: discriminative prediction of the missing entries in data matrices and estimation of the joint probability distribution of row and column variables in co-occurrence matrices. We derive PAC-Bayesian generalization bounds for the expected out-of-sample performance of co-clustering-based solutions for these two tasks. The analysis yields regularization terms that were absent in the previous formulations of co-clustering. The bounds suggest that the expected performance of co-clustering is governed by a trade-off between its empirical performance and the mutual information preserved by the cluster variables on row and column IDs. We derive an iterative projection algorithm for finding a local optimum of this trade-off for discriminative prediction tasks. This algorithm achieved state-of-the-art performance in the MovieLens collaborative filtering task. Our co-clustering model can also be seen as matrix tri-factorization and the results provide generalization bounds, regularization terms, and new algorithms for this form of matrix factorization.

The analysis of co-clustering is extended to tree-shaped graphical models, which can be used to analyze high dimensional tensors. According to the bounds, the generalization abilities of tree-shaped graphical models depend on a trade-off between their empirical data fit and the mutual information that is propagated up the tree levels.

We also formulate weighted graph clustering as a prediction problem: given a subset of edge weights we analyze the ability of graph clustering to predict the remaining edge weights. The analysis of co-clustering easily extends to this problem and suggests that graph clustering should optimize the trade-off between empirical data fit and the mutual information that clusters preserve on graph nodes.

**Keywords:** matrix tri-factorization, graphical models, graph clustering, pairwise clustering, combinatorial priors, density estimation

1. This paper is based on (Seldin and Tishby, 2008, 2009; Seldin, 2009, 2010).

## 1. Introduction

Structure learning and, in particular, clustering is an important and long-standing problem in science. In many situations it has to be performed based on a limited data sample and with no or limited supervision. A natural question that arises in this context is to what extent the inferred structure is a reflection of a "true" structure underlying the data or a mere artifact of the learning model and/or statistical fluctuation of the finite sample. But is there a "true structure" in the first place? Consider the following example: assume we have a bag of blocks which we can cluster by multiple parameters, such as shape, color, material they are made of, and so on. All these possibilities are equally plausible and asking whether a clustering of blocks by shape is better or worse than a clustering of blocks by color makes no sense. But the absence of an objective comparison criterion for outputs of two structure learning algorithms poses a serious obstacle for their evaluation and the advancement of unsupervised learning in general.

We argue that one does not learn structure for its own sake, but rather to facilitate solving some higher level task. By evaluating the contribution of structure learning to the solution of the higher level task it is possible to derive an objective comparison of the utility of different structures in the context of this specific task. Returning to the bag of blocks example, if we know that after clustering the blocks we will have to pack them into a box, then the clustering of blocks by shape is much more useful than the clustering of blocks by color, since packing is indifferent to color. We can further measure the amount of time that different clusterings saved us in the packing task and thereby obtain an objective numerical evaluation of the utility of clustering of blocks by different parameters in this context. Moreover, by repeating the experiment several times (or by some more intelligent analysis) it is possible to provide generalization guarantees on how much time this or other clustering algorithm is expected to save us in the packing task in the future.

Since in any non-trivial data many structures coexist simultaneously, "blind" unsupervised learning without specification of its potential application is doomed to failure in the general case. This is because the potential application (or range of applications) can make any property or element of the structure either decisive or completely irrelevant for the task, and hence render it useful or useless for identification by unsupervised learning. The need to consider unsupervised learning within the context of its subsequent application has been pointed out by many researchers, especially those concerned with practical applications of these methods (Guyon et al., 2009). In the present paper we reformulate traditional unsupervised learning problems as prediction problems and then adapt well-developed tools from supervised learning to provide generalization bounds on their expected out-of-sample performance. We start with the problem of co-clustering, but then show that our approach to problem formulation is applicable to and can be analyzed in a much broader range of applications.

Co-clustering is a widely used method for analysis of data in matrix form by simultaneous clustering of rows and columns of the matrix (Banerjee et al., 2007). Here we focus solely on co-clustering solutions that result in a grid form partition of the data matrix. This form of co-clustering is also known as partitional co-clustering (Banerjee et al., 2007), checkerboard bi-clustering (Cheng and Church, 2000; Kluger et al., 2003), grid clustering (Devroye et al., 1996; Seldin and Tishby, 2008, 2009), and box clustering. Note that some authors use the terms co-clustering and bi-clustering to refer to a simultaneous grouping of rows and columns that does not result in a grid-form partition of the whole data matrix (Hartigan, 1972; Madeira and Oliveira, 2004), but these forms of partitions are not discussed in this work. Note as well that this paper considers soft assignments of

rows and columns to their clusters, while using two-level generative models such as those discussed in (Dhillon et al., 2003; Banerjee et al., 2007) for hard assignments. Recently, Bayesian approaches to co-clustering have been suggested, for example (Shan and Banerjee, 2008; Salakhutdinov and Mnih, 2008; Shafiei and Milios, 2006; Wang et al., 2009; Lashkari and Golland, 2009), which consider mixed memberships by introducing an additional level to the generative process. However, three-level generative models require approximate inference methods such as variational inference or Markov Chain Monte Carlo, whereas the two-level model for discriminative prediction discussed here can be learned by iterative projections. The analysis presented here is not limited to two-dimensional data matrices, but holds for higher dimensional tensors as well. A three-level Bayesian approach to clustered tensor factorization was recently presented by Sutskever et al. (2009).

In the past decade co-clustering has successfully been applied in multiple domains, including clustering of documents and words in text mining (Slonim and Tishby, 2000; El-Yaniv and Souroujon, 2001; Dhillon et al., 2003; Takamura and Matsumoto, 2003), genes and experimental conditions in bioinformatics (Cheng and Church, 2000; Cho et al., 2004; Kluger et al., 2003; Cho and Dhillon, 2008), tokens and contexts in natural language processing (Freitag, 2004; Rohwer and Freitag, 2004; Li and Abe, 1998), viewers and movies in recommender systems (George and Merugu, 2005; Seldin et al., 2007; Salakhutdinov and Mnih, 2008; Seldin, 2009), etc. In (Seldin et al., 2007; Seldin and Tishby, 2009) it was pointed out that there are actually two different classes of problems that are solved with co-clustering that correspond to two different high-level tasks and should be analyzed separately. The first class of problems are discriminative prediction tasks, one typical representative of which is collaborative filtering (Herlocker et al., 2004). In collaborative filtering, the analyst is given a matrix of viewers by movies with ratings, e.g. on a five-star scale, attributed by the viewers to the movies. The matrix is usually sparse, as most viewers have not seen all the movies. In this problem the task is usually to predict the missing entries. We assume that there is some unknown probability distribution $p(x_1,x_2,y)$ over the triplets of viewer $x_1$, movie $x_2$, and rating $y$. The goal is to build a discriminative predictor $q(y|x_1,x_2)$ that given a pair of viewer $x_1$ and movie $x_2$ will predict the expected rating $y$. A natural form of evaluation of such predictors, no matter whether they are based on co-clustering or not, is to evaluate the expected loss $\mathbb{E}_{p(x_1,x_2,y)}\mathbb{E}_{q(y'|x_1,x_2)}l(Y,Y')$, where $l(y,y')$ is an externally provided loss function for predicting $y'$ instead of $y$. In Section 3 we provide this analysis for co-clustering-based predictors. The analysis can be used not only to construct co-clustering solutions to this problem, but also to conduct a theoretical comparison of the co-clustering-based approach to this problem with other possible approaches.

The second class of problems, which are solved using co-clustering, are problems of estimation of a joint probability distribution in co-occurrence data analysis. A typical example of this kind of problem is the analysis of word-document co-occurrence matrices in text mining (Slonim and Tishby, 2000; El-Yaniv and Souroujon, 2001; Dhillon et al., 2003). Word-document co-occurrence matrices are matrices of words by documents where the number of times each word occurred in each document is counted in the corresponding entries. If normalized, such a matrix can be regarded as an empirical joint probability distribution of words and documents. To illustrate the difference between co-occurrence data and the data considered in discriminative prediction tasks, we point out that the ratings in the collaborative filtering example are functions of viewer and movie ID pairs and they do not depend on other viewers or movies. By contrast, in co-occurrence data the joint probability (or the number of co-occurrence events) is normalized by the size of the corpus and thus depends on the whole subset of words and documents considered (or the size of the corpus if no normalization is applied).

Although many researchers have analyzed co-occurrence data by clustering similar words and similar documents (Slonim and Tishby, 2000; El-Yaniv and Souroujon, 2001; Dhillon et al., 2003; Takamura and Matsumoto, 2003), or by using topic models (Steyvers and Griffiths, 2006; Blei and Lafferty, 2009) and other approaches, no clear learning task in this problem has been defined and it remains difficult to compare different approaches or perform model order selection. In (Seldin and Tishby, 2009) one possible way of defining a high-level task for this problem was suggested. It was assumed that the observed co-occurrence matrix was drawn from an unknown joint probability distribution $p(x_1,x_2)$ of words $x_1$ and documents $x_2$. The suggested task was an estimation of this joint probability distribution based on the observed sample. In such a formulation, the quality of an estimator $q(x_1,x_2)$ for $p(x_1,x_2)$ can be measured by $-\mathbb{E}_{p(x_1,x_2)} \ln q(X_1,X_2)$, where the choice of the logarithmic loss is natural in the context of density estimation. In particular, it corresponds to the expected code length of an encoder that uses $q(x_1,x_2)$ to encode samples generated by $p(x_1,x_2)$ (Cover and Thomas, 1991). In Section 3 we provide an analysis of this quantity for co-clustering-based density estimators. Similar to the case with co-clustering-based discriminative predictors, the analysis serves to perform model order selection in this problem. It further enables a theoretical comparison of the co-clustering-based approach to this problem with other possible approaches.

For the purpose of analysis and derivation of generalization bounds for the above two problems we found it convenient to apply the PAC-Bayesian framework (McAllester, 1998, 1999), which is reviewed in Section 2. Similar to the Probably Approximately Correct (PAC) learning model (Valiant, 1984), PAC-Bayesian bounds pose no assumptions or restrictions on the distribution that generates the data (apart from the usual assumption that the data are independent and identically distributed (i.i.d.) and that the train and test distributions are the same). However, unlike the usual PAC bounds, where the whole hypothesis space is characterized by its Vapnik-Chervonenkis (VC) dimension (Vapnik and Chervonenkis, 1968, 1971), PAC-Bayesian bounds apply a non-uniform treatment of the hypotheses by introducing a prior distribution over the hypothesis space. For example, within the class of decision trees a preference for shallow trees can be given by assigning a higher prior. If a good prior over the hypothesis space can be designed, the tightness of the bounds can be improved considerably. As shown in the literature, PAC-Bayesian analysis is able to provide practically useful bounds that in some cases are only 10%-20% away from the test error (Langford, 2005; Seldin and Tishby, 2008; Seldin, 2009; Germain et al., 2009).

Originally, PAC-Bayesian bounds were derived for classification tasks. They have been applied in the analysis of decision trees (Mansour and McAllester, 2000), Support Vector Machines (SVMs) (Langford and Shawe-Taylor, 2002; McAllester, 2003; Langford, 2005; Ambroladze et al., 2007; Crammer et al., 2009; Germain et al., 2009), transductive learning (Derbeko et al., 2004), structured prediction (Bartlett et al., 2005; McAllester, 2007), and other supervised learning models. In (Seldin and Tishby, 2009) we introduced PAC-Bayesian analysis to discrete density estimation. Recently, Higgs and Shawe-Taylor (2010) applied PAC-Bayesian analysis to continuous density estimation. In Section 3 we present the PAC-Bayesian analysis of discriminative prediction and density estimation with co-clustering. According to the derived bounds, the generalization performance of co-clustering-based models depends on a trade-off between their empirical performance and the mutual information that the clusters preserve on the observed parameters (row and column IDs). The mutual information term introduces model regularization that was absent in the previous formulations of co-clustering (Dhillon et al., 2003; Banerjee et al., 2007). We further suggest algorithms for optimization of the trade-off in Section 4. In Section 5, we achieve state-of-the-art

performance in the prediction of missing ratings in the MovieLens collaborative filtering dataset by optimization of the trade-off.

The co-clustering models analyzed here are tightly related to matrix tri-factorization - a 3-factor decomposition of a matrix of the form $A \approx Q_1^T F Q_2$ (Banerjee et al., 2007; Ding et al., 2006; Yoo and Choi, 2009a,b) and to Tucker decomposition of higher dimensional tensors (Kim and Choi, 2007). This relation is discussed in Section 6. We point out that similar to co-clustering itself there are at least two different forms of matrix tri-factorization; one that corresponds to discriminative prediction tasks, such as collaborative filtering, and the other which is more appropriate for co-occurrence data analysis. In the first case $A$ can be arbitrary, whereas in the second case the input matrix $A$ is a joint probability distribution matrix (its entries are non-negative and sum up to one). At the technical level, in discriminative prediction tasks $Q_1$ and $Q_2$ are right stochastic matrices (their rows sum up to one) and $F$ is arbitrary, whereas in co-occurrence data analysis $Q_1$ and $Q_2$ are left stochastic matrices (their columns sum up to one) and $F$ is a joint probability distribution matrix (in the cluster product space). Our analysis provides generalization bounds, regularization terms, and new algorithms for both forms of matrix tri-factorization.

Co-clustering can also be regarded as a simple graphical model. In Section 7 we suggest how to extend our analysis to more general tree-shaped graphical models. Such graphical models can be useful to treat the curse of dimensionality in the analysis of high dimensional tensors. This also provides a new perspective on learning graphical models: instead of learning a graphical model that fits the training data, the approach suggests optimizing the model's ability to predict new observations. In Sections 3 and 7 it is demonstrated that PAC-Bayesian bounds are able to take advantage of the factor form of graphical models and provide bounds that depend on the sizes of the cliques of graphical models and the amount of mutual information that is propagated up the tree levels.

In Section 8 we extend our approach to the formulation of unsupervised learning problems as prediction problems to graph clustering and pairwise clustering (the latter is equivalent to clustering of a weighted graph, where edge weights correspond to pairwise distances). We formulate weighted graph clustering as a prediction problem: given a sample of edge weights we analyze the ability of graph clustering to predict the remaining edge weights. We adapt the PAC-Bayesian analysis of co-clustering to derive a PAC-Bayesian generalization bound for graph clustering. The bound shows that graph clustering should optimize a trade-off between empirical data fit and the mutual information that clusters preserve on the graph nodes. A similar trade-off derived from information-theoretic considerations has been shown to produce state-of-the-art results in practice (Slonim et al., 2005; Yom-Tov and Slonim, 2009). This paper supports the empirical evidence by providing a better theoretical foundation, suggesting formal generalization guarantees, and offering a more accurate way to deal with finite sample issues.

## 2. PAC-Bayesian Generalization Bounds

This section is devoted to PAC-Bayesian generalization bounds, which are the main tool used for the analysis of our learning models in the subsequent sections. We review the well-known PAC-Bayesian bound for classification and present a slight variation of a less well-known PAC-Bayesian bound for discrete density estimation. The PAC-Bayesian generalization bounds pioneered by McAllester (1998, 1999) provide guarantees on generalization abilities of randomized predictors (formally defined below in Section 2.1) within the classical PAC learning model (Valiant, 1984) and build upon preceding works on PAC analysis of Bayesian learning models (Shawe-Taylor et al.,

1998; Shawe-Taylor and Williamson, 1997). The classical PAC learning model evaluates learning algorithms by their ability to predict new events generated by the same probability distribution as the one that was used to train the algorithm. No restrictions on the data generating probability distribution are imposed except the assumption that the samples are i.i.d. The PAC-Bayesian framework should be distinguished from Bayesian learning, which assumes that the data were generated by hypotheses from the hypothesis class and applies Bayes' rule for inference. Bayesian learning does not provide guarantees on the expected error of the Bayes' inference rule and in some situations can lead to overfitting (Kearns et al., 1997).

The classical PAC bounds are derived by covering the error space of a hypothesis class. For example, the most familiar PAC bounds are based on the VC-dimension of a hypothesis class, which is a logarithm of the maximal number of points that can be jointly classified in any possible way by functions from the hypothesis class (Vapnik and Chervonenkis, 1968, 1971; Vapnik, 1998; Devroye et al., 1996). More recent bounds involve Rademacher and Gaussian complexities (Koltchinskii, 2001; Bartlett et al., 2001; Bartlett and Mendelson, 2001; Boucheron et al., 2005). However, in all the above approaches the whole hypothesis class is characterized by a single number: its VC-dimension or Rademacher complexity, which means that all the incorporating hypotheses are treated identically and there is no way to differentiate them and give preference to "simpler" ones. For example, if the hypothesis class consists of straight lines and parabolas, its VC-dimension is equal to the VC-dimension of a hypothesis class consisting of parabolas only and there is no direct way to give preference to straight lines within the combined hypothesis class. PAC-Bayesian bounds are derived by covering the hypothesis space and they enable non-uniform treatment of the hypotheses. In the PAC-Bayesian approach each hypothesis is characterized by its own complexity defined by its prior. This refined approach provides several important benefits, which include: (1) the ability to give explicit preference to certain hypotheses (e.g, in the example above we can assign a higher prior to straight lines); (2) a gradient within the hypothesis space, which can be used in algorithms for bound minimization; (3) considerably tighter bounds, which are meaningful in a practical sense: in some applications the discrepancy between the bound value and the test error is only 10%-20% (Langford, 2005; Seldin and Tishby, 2008; Seldin, 2009; Germain et al., 2009). There is one more distinction between the usual PAC analysis and the PAC-Bayesian bounds that extend the scope of applicability of the latter. Classical PAC analysis aims at bounding the discrepancy between the expected performance of the hypothesis with the best empirical performance and the best hypothesis within the hypothesis class. Such types of bounds require a uniform bound on the discrepancies between empirical and expected performances for all the hypotheses within a hypothesis class. The uniform bound exists if and only if the hypothesis class has a finite VC-dimension. PAC-Bayesian bounds bound the expected performance of a given hypothesis, but do not attempt to bound its gap to the performance of the best hypothesis within the hypothesis class. This fact makes it possible to apply PAC-Bayesian bounds even in situations where the VC-dimension of a hypothesis class is infinite, for example, decision trees of unlimited depth or separating hyperplanes in infinite-dimensional spaces. This does not disprove the fundamental theorem of PAC learning theory, which states that learning is possible if and only if the VC-dimension of a hypothesis class is finite, but rather extends the notion of learnability. Instead of a regret-based definition of learnability, by which the ability to learn is the ability to achieve, up to a small epsilon, the best possible solution within a hypothesis class, the PAC-Bayesian approach defines learnability as the ability to bound the expected performance of the obtained solution. Then it is a question for the user to decide whether the guaranteed expected performance is sufficient for his or her needs.

Since the strength of PAC-Bayesian analysis lies in its ability to provide a non-uniform treatment of the hypotheses within a hypothesis class, its advantage over traditional PAC analysis is best seen in the analysis of heterogeneous hypothesis classes (or, in other words, when the hypotheses constituting a hypothesis class are not symmetric). Some hypothesis classes exhibit a "natural heterogeneity"; for example, we can partition the class of decision trees into subclasses according to tree depth. A higher prior can then be assigned to shallow trees to provide them with preference over deep trees. For example, a prior $\mathcal{P}(t) = 2^{-(d(t)+1)}2^{-2^{d(t)}}$, where $d(t)$ is the depth of a tree $t$ would be a legal prior over the space of full binary decision trees of unlimited depth ($2^{-(d(t)+1)}$ is a prior over tree depth and $2^{-2^{d(t)}}$ is a prior over trees of a given depth). Note that it is possible to assign a higher prior to *all* shallow trees simultaneously because there are fewer shallow trees than deep trees. Hence, the above prior exploits knowledge about the structure of the hypothesis space, but makes no assumptions about the data. As we will see below, the bounds depend on $-\ln \mathcal{P}(t) = \ln(2)[(d(t)+1) + 2^{d(t)}]$; thus the value of the logarithm of the first part of the prior $(d(t)+1)$, which accounts for the tree depth, is negligible compared to the value of the logarithm of the second half of the prior $(2^{d(t)})$, which counts the number of symmetric trees given the depth. Given a strong prior knowledge on the problem domain, which breaks the symmetry between trees of a given depth, it is possible to give preference (a higher prior) to certain deep trees; however it is impossible to give a higher prior to all deep trees simultaneously, because there are too many of them. Thus, a choice of a different prior over tree depth and even precise prior knowledge of the tree depth can only negligibly improve the bound.

For some hypothesis spaces which seem homogeneous at a first glance it may still be possible to identify non-trivial asymmetries and define a corresponding structural prior. The best example comes from the analysis of SVMs, where the class of all possible separating hyperplanes in $\mathbb{R}^d$ is partitioned into subclasses according to the size of the margin and a higher prior is given to the hyperplanes with large margins (Langford and Shawe-Taylor, 2002; McAllester, 2003; Langford, 2005). In structure learning the hypothesis class usually exhibits a natural heterogeneity since the hypotheses (structures) can be differentiated by their complexity. Hence, PAC-Bayesian analysis has great potential in the analysis of structure learning which is only partially explored in this work. PAC-Bayesian bounds are further distinguished by their explicit dependence on model parameters, which makes their optimization easy.

Following the pioneering work of McAllester (1998, 1999), the PAC-Bayesian bounds were tightened and simplified by Seeger (2002, 2003). Some further improvements were suggested in (Maurer, 2004; Audibert and Bousquet, 2007; Blanchard and Fleuret, 2007). This section draws on the easier-to-read expositions by Maurer (2004) and Banerjee (2006). In order to present the bounds for classification and density estimation, we need to define the notion of randomized predictors, which is done next. We then present the PAC-Bayesian theorems and their proofs.

### 2.1 Randomized Predictors

Let $\mathcal{H}$ be a hypothesis class and let $Q(h)$ be a distribution over $\mathcal{H}$ (if $\mathcal{H}$ is infinite then $Q(h)$ is a probability density). A *randomized predictor* associated with $Q$, and with a small abuse of notation denoted by $Q$, is defined in the following way: For each sample $x$ a hypothesis $h \in \mathcal{H}$ is drawn according to $Q(h)$, and then applied to make a prediction on $x$. In the classification context, $Q$ is termed a *randomized classifier* (Langford, 2005). However, since this work extends the PAC-Bayesian framework beyond the classification scenario by using the same randomization technique,

we use the term "randomized predictor". In this more general context $h(x)$ is a general function of $x$, not necessarily a classifier.

In the context of classification, let $S = \{(x_1,y_1),..,(x_N,y_N)\}$ be an i.i.d. sample of size $N$ of instances and their labels drawn according to unknown distribution $p(x,y)$ and let $l(y,y')$ be a given loss function for predicting $y'$ instead of $y$. Let $h(x)$ be the label of $x$ predicted by hypothesis $h$. For each $h \in \mathcal{H}$ we denote by $\hat{L}(h) = \frac{1}{N}\sum_i l(y_i,h(x_i))$ the empirical loss of the hypothesis $h$ on $S$ and by $L(h) = \mathbb{E}_{p(x,y)} l(Y,h(X))$ the expected loss of $h$ with respect to the true, unknown distribution that generates the data. We further extend the definitions of the empirical and expected losses for randomized predictors in the following way:

$$\hat{L}(Q) = \mathbb{E}_{Q(h)}\hat{L}(h) \qquad \text{and} \qquad L(Q) = \mathbb{E}_{Q(h)}L(h).$$

For two distributions $q$ and $p$ over domain $\mathcal{X}$ we define

$$KL(q\|p) = \mathbb{E}_q \ln \frac{q(x)}{p(x)}$$

to be the Kullback-Leibler (KL) divergence between $q$ and $p$ (Cover and Thomas, 1991). As well, we define

$$kl(\hat{L}(Q)\|L(Q)) = \hat{L}(Q)\ln\frac{\hat{L}(Q)}{L(Q)} + (1-\hat{L}(Q))\ln\frac{1-\hat{L}(Q)}{1-L(Q)}$$

as the KL-divergence between two Bernoulli distributions with biases $\hat{L}(Q)$ and $L(Q)$. Now we are ready to state the PAC-Bayesian theorems.

### 2.2 PAC-Bayesian Theorems

**Theorem 1 (PAC-Bayesian bound for classification)** *For a hypothesis class $\mathcal{H}$, a prior distribution $\mathcal{P}$ over $\mathcal{H}$ and a zero-one loss function $l$, with probability greater than $1-\delta$ over drawing a sample of size $N$, for all randomized classifiers $Q$ simultaneously*:

$$kl(\hat{L}(Q)\|L(Q)) \leq \frac{KL(Q\|\mathcal{P}) + \ln(N+1) - \ln\delta}{N}. \tag{1}$$

**Theorem 2 (PAC-Bayesian bound for discrete density estimation)** *Let $\mathcal{X}$ be the sample space (possibly infinite) and let $p(x)$ be an unknown distribution over $x \in \mathcal{X}$. Let $\mathcal{H}$ be a hypothesis class, such that each member $h \in \mathcal{H}$ is a function from $\mathcal{X}$ to a finite set $\mathcal{Z}$ with cardinality $|\mathcal{Z}|$. Let $p_h(z) = P_{X\sim p(x)}\{h(X) = z\}$ be the distribution over $\mathcal{Z}$ induced by $p(x)$ and $h$. Let $\mathcal{P}$ be a prior distribution over $\mathcal{H}$. Let $Q$ be an arbitrary distribution over $\mathcal{H}$ and $p_Q(z) = \mathbb{E}_{Q(h)}p_h(z)$ a distribution over $\mathcal{Z}$ induced by $p(x)$ and $Q$. Let $S$ be an i.i.d. sample of size $N$ generated according to $p(x)$ and let $\hat{p}(x)$ be the empirical distribution over $\mathcal{X}$ corresponding to $S$. Let $\hat{p}_h(z) = P_{X\sim\hat{p}(x)}\{h(X) = z\}$ be the empirical distribution over $\mathcal{Z}$ corresponding to $h$ and $S$. Let $\hat{p}_Q(z) = \mathbb{E}_{Q(h)}\hat{p}_h(z)$. Then with probability greater than $1-\delta$ for all possible $Q$ simultaneously*:

$$KL(\hat{p}_Q(z)\|p_Q(z)) \leq \frac{KL(Q\|\mathcal{P}) + (|\mathcal{Z}|-1)\ln(N+1) - \ln\delta}{N}. \tag{2}$$

**Remarks:**

1. This form of Theorem 1 first appeared in (Seeger, 2002). A slightly different version of Theorem 2 first appeared in (Seeger, 2003) and independently in (Seldin and Tishby, 2009), where it found the first non-trivial application.

2. The PAC-Bayesian bound for classification (1) is a direct consequence of the PAC-Bayesian bound for density estimation (2). To see this, let $Z$ be the error variable. Then each hypothesis $h \in \mathcal{H}$ is a function from the sample space (in this case the samples are pairs $\langle X,Y \rangle$) to the error variable $Z$ and $|\mathcal{Z}| = 2$. Furthermore, $\hat{L}(h) = \hat{p}_h(Z=1)$ and $L(h) = p_h(Z=1)$, hence $kl(\hat{L}(Q)\|L(Q)) = KL(\hat{p}_Q(z)\|p_Q(z))$. Substituting this into (2) yields (1).

3. Maurer (2004) showed that due to convexity of the KL-divergence, inequality (1) is valid for all loss functions bounded in the [0,1] interval, and not only for the zero-one loss. He also proved that due to tighter concentration of empirical means of binary variables, for $N \geq 8$ bound (1) can be further tightened:

$$kl(\hat{L}(Q)\|L(Q)) \leq \frac{KL(Q\|\mathcal{P}) + \frac{1}{2}\ln(4N) - \ln\delta}{N} \tag{3}$$

   and that this is the tightest result that can be proved using the techniques which are also used in this paper.

4. Although there is no analytical expression for the inverse of the $kl$-divergence, $L(Q)$ can be bounded numerically:

$$\begin{aligned} L(Q) &\leq kl^{-1}\left(\hat{L}(Q), \frac{KL(Q\|\mathcal{P}) + \frac{1}{2}\ln(4N) - \ln\delta}{N}\right) \\ &= \max\left\{v : kl(\hat{L}(Q)\|v) \leq \frac{KL(Q\|\mathcal{P}) + \frac{1}{2}\ln(4N) - \ln\delta}{N}\right\}. \end{aligned} \tag{4}$$

   Since $kl(\hat{L}(Q)\|v)$ is convex in $v$, (4) is easy to compute.

5. The proof of Theorem 2 presented below reveals a close relation between the PAC-Bayesian theorems and the method of types in information theory (Cover and Thomas, 1991). The trade-off between $\hat{L}(Q)$ and $KL(Q\|\mathcal{P})$ in the PAC-Bayesian bounds also has a tight relation to the maximum entropy principle in learning and statistical mechanics (Jaynes, 1957; Dudík et al., 2007; Catoni, 2007; Shawe-Taylor and Hardoon, 2009). The relations between the PAC-Bayesian bounds, information theory, and statistical mechanics are further discussed in (Catoni, 2007).

The proof of Theorem 2 presented below is based on two auxiliary results which have value in their own right and therefore are presented in dedicated subsections. The first auxiliary result applies the method of types to bound the expectation of the exponent of the divergence between empirical and expected distributions over $\mathcal{Z}$ for a single hypothesis: $\mathbb{E}_S e^{N \cdot KL(\hat{p}_h(z)\|p_h(z))}$. The second auxiliary result relates the divergence $\mathbb{E}_{Q(h)} KL(\hat{p}_h(z)\|p_h(z))$ for all $Q$ to a single (prior) reference measure $\mathcal{P}$. This relation is actually the cornerstone of the PAC-Bayesian analysis. Finally, in Section 2.5 a quantity depending on the prior measure $\mathcal{P}$ is treated using the first auxiliary result to obtain the final bound in equation (2).

### 2.3 The Law of Large Numbers

In this subsection we analyze the rate of convergence of empirical distributions over finite domains to their true values. The following result is based on the method of types in information theory (Cover and Thomas, 1991).

**Theorem 3** *Let $S = \{X_1, .., X_N\}$ be i.i.d. distributed by $p(x)$. Denote by $\hat{p}(x)$ the empirical distribution over $\mathcal{X}$ corresponding to S and by $|\mathcal{X}|$ the cardinality of $\mathcal{X}$. Then*:

$$\mathbb{E}_S e^{N \cdot KL(\hat{p}(x) \| p(x))} \leq (N+1)^{|\mathcal{X}|-1}. \tag{5}$$

**Proof** Enumerate the possible values of $\mathcal{X}$ by $1, .., |\mathcal{X}|$ and let $n_i$ count the number of occurrences of value $i$. Let $p_i$ denote the probability of value $i$ and $\hat{p}_i = \frac{n_i}{N}$ be its empirical counterpart. Let $H(\hat{p}) = -\sum_i \hat{p}_i \ln \hat{p}_i$ be the empirical entropy. Then:

$$\begin{aligned}
\mathbb{E}_S e^{NKL(\hat{p}\|p)} &= \sum_{\substack{n_1,..,n_{|\mathcal{X}|}: \\ \sum_i n_i = N}} \binom{N}{n_1, .., n_{|\mathcal{X}|}} \cdot \prod_{i=1}^{|\mathcal{X}|} p_i^{N\hat{p}_i} \cdot e^{NKL(\hat{p}\|p)} \\
&\leq \sum_{\substack{n_1,..,n_{|\mathcal{X}|}: \\ \sum_i n_i = N}} e^{NH(\hat{p})} \cdot e^{N\sum_i \hat{p}_i \ln p_i} \cdot e^{NKL(\hat{p}\|p)} \qquad (6) \\
&= \sum_{\substack{n_1,..,n_{|\mathcal{X}|}: \\ \sum_i n_i = N}} 1 = \binom{N+|\mathcal{X}|-1}{|\mathcal{X}|-1} \leq (N+1)^{|\mathcal{X}|-1}. \qquad (7)
\end{aligned}$$

In (6) we used the $\binom{N}{n_1,..,n_{|\mathcal{X}|}} \leq e^{NH(\hat{p})}$ bound on the multinomial coefficient, which counts the number of sequences with a fixed cardinality profile (unnormalized type) $n_1, .., n_{|\mathcal{X}|}$ (Cover and Thomas, 1991). In the second equality in (7) the number of ways to choose $n_i$-s equals the number of ways we can place $|\mathcal{X}|-1$ ones in a sequence of $N+|\mathcal{X}|-1$ ones and zeros, where ones symbolize a partition of zeros ("balls") into $|\mathcal{X}|$ bins. ■

### 2.4 Change of Measure Inequality

The simultaneous treatment of all possible distributions (measures) $Q$ over $\mathcal{H}$ is done by relating them all to a single reference (prior) measure $\mathcal{P}$. We call this relation a *change of measure inequality*. This inequality was formulated as a standalone result in (Banerjee, 2006), although it originates much earlier. Banerjee (2006) terms it a *compression lemma*; however we find the term "change of measure inequality" more appropriate to its nature and usage. The inequality is a simple consequence of Jensen's inequality.

**Lemma 4 (Change of Measure Inequality)** *For any measurable function $\phi(h)$ on $\mathcal{H}$ and any distributions $\mathcal{P}$ and $Q$ on $\mathcal{H}$, we have:*

$$\mathbb{E}_{Q(h)} \phi(h) \leq KL(Q\|\mathcal{P}) + \ln \mathbb{E}_{\mathcal{P}(h)} e^{\phi(h)}. \tag{8}$$

**Proof** For any measurable function $\phi(h)$, we have:

$$\begin{aligned}\mathbb{E}_{Q(h)}\phi(h) &= \mathbb{E}_{Q(h)} \ln\left(\frac{Q(h)}{\mathcal{P}(h)}\cdot e^{\phi(h)}\cdot\frac{\mathcal{P}(h)}{Q(h)}\right)\\ &= KL(Q\|\mathcal{P}) + \mathbb{E}_{Q(h)}\ln\left(e^{\phi(h)}\cdot\frac{\mathcal{P}(h)}{Q(h)}\right)\\ &\leq KL(Q\|\mathcal{P}) + \ln\mathbb{E}_{Q(h)}\left(e^{\phi(h)}\cdot\frac{\mathcal{P}(h)}{Q(h)}\right) \qquad (9)\\ &= KL(Q\|\mathcal{P}) + \ln\mathbb{E}_{\mathcal{P}(h)}e^{\phi(h)},\end{aligned}$$

where (9) is by Jensen's inequality. ■

### 2.5 Proof of the PAC-Bayesian Generalization Bound for Density Estimation

We apply the results of the previous two subsections to prove the PAC-Bayesian generalization bound for density estimation.

**Proof of Theorem 2** Let $\phi(h,S,p) = NKL(\hat{p}_h(z)\|p_h(z))$. Then:

$$\begin{aligned}NKL(\hat{p}_Q(z)\|p_Q(z)) &= NKL(\mathbb{E}_{Q(h)}\hat{p}_h(z)\|\mathbb{E}_{Q(h)}p_h(z))\\ &\leq \mathbb{E}_{Q(h)}NKL(\hat{p}_h(z)\|p_h(z)) \qquad (10)\\ &\leq KL(Q\|\mathcal{P}) + \ln\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))}, \qquad (11)\end{aligned}$$

where (10) is by the convexity of the KL-divergence (Cover and Thomas, 1991) and (11) is by the change of measure inequality. To obtain (2) it is left to bound $\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))}$. This is a random quantity depending on the sample $S$ since $\hat{p}_h(z)$ for each $h$ depends on the sample. By Markov's inequality we know that with probability at least $1-\delta$ over the sample $\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))} \leq \frac{1}{\delta}\mathbb{E}_S\left[\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))}\right]$. In order to obtain a bound on $\mathbb{E}_S\left[\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))}\right]$ we note that it is possible to exchange $\mathbb{E}_S$ with $\mathbb{E}_{\mathcal{P}(h)}$ since $S$ and $h$ are independent:

$$\mathbb{E}_S\left[\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))}\right] = \mathbb{E}_{\mathcal{P}(h)}\left[\mathbb{E}_S e^{NKL(\hat{p}_h(z)\|p_h(z))}\right] \leq (N+1)^{|\mathcal{Z}|-1}. \qquad (12)$$

The last inequality in (12) is justified by the fact that $\mathbb{E}_S e^{NKL(\hat{p}_h(z)\|p_h(z))} \leq (N+1)^{|\mathcal{Z}|-1}$ for each $h$ individually according to (5). By Markov's inequality we conclude that with probability of at least $1-\delta$ over $S$:

$$\mathbb{E}_{\mathcal{P}(h)}e^{NKL(\hat{p}_h(z)\|p_h(z))} \leq \frac{(N+1)^{|\mathcal{Z}|-1}}{\delta}.$$

Substituting this into (11) and normalizing by $N$ yields (2). ■

### 2.6 Addendum to the Law of Large Numbers

Note in passing that it is straightforward to recover theorem 12.2.1 in (Cover and Thomas, 1991) from Theorem 3 (even with a slight improvement). This theorem is used later in the estimation of marginal distributions.

**Theorem 5 (12.2.1 in Cover and Thomas, 1991)** *Under the notations of Theorem 3 with probability greater than* $1-\delta$:

$$KL(\hat{p}(x)\|p(x)) \leq \frac{(|\mathcal{X}|-1)\ln(N+1)-\ln\delta}{N}. \tag{13}$$

**Proof** Immediate from application of Markov's inequality to (5). ■

### 2.7 Construction of a Density Estimator

Although we have bounded $KL(\hat{p}_Q(z)\|p_Q(z))$ in Theorem 2, $\hat{p}_Q(z)$ still cannot be used as a density estimator for $p_Q(z)$, because it is not bounded from zero. In order to bound the logarithmic loss $-\mathbb{E}_{p_Q(z)}\ln\hat{p}_Q(z)$, which corresponds, e.g., to the expected code length of encoder $\hat{p}_Q$ when samples are generated by $p_Q$, we have to smooth $\hat{p}_Q$. We denote a smoothed version of $\hat{p}_Q$ by $\tilde{p}_Q$ and define it as:

$$\begin{aligned}\tilde{p}_h(z) &= \frac{\hat{p}_h(z)+\gamma}{1+\gamma|\mathcal{Z}|}, \\ \tilde{p}_Q(z) &= \mathbb{E}_{Q(h)}\tilde{p}_h(z) = \frac{\hat{p}_Q(z)+\gamma}{1+\gamma|\mathcal{Z}|}.\end{aligned} \tag{14}$$

In the following theorem we show that if $KL(\hat{p}_Q(z)\|p_Q(z)) \leq \varepsilon(Q)$ and $\gamma(Q) = \frac{\sqrt{\varepsilon(Q)/2}}{|\mathcal{Z}|}$, then $-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z)$ is roughly within $\pm\sqrt{\varepsilon(Q)/2}\ln|\mathcal{Z}|$ range around $H(\hat{p}_Q(z))$. The bound on $KL(\hat{p}_Q(z)\|p_Q(z))$ is naturally obtained by Theorem 2. Thus, the performance of the density estimator $\tilde{p}_Q$ is optimized by distribution $Q$ that minimizes the trade-off between $H(\hat{p}_Q(z))$ and $\frac{1}{N}KL(Q\|\mathcal{P})$.

Note that for a uniform distribution $u(z) = \frac{1}{|\mathcal{Z}|}$ the value of $-\mathbb{E}_{p(z)}\ln u(z) = \ln|\mathcal{Z}|$. Thus, the theorem below is interesting when $\sqrt{\varepsilon(Q)/2}$ is significantly smaller than 1. For technical reasons in the proofs of the following section, the upper bound in the next theorem is stated for $-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z)$ and for $-\mathbb{E}_{Q(h)}\mathbb{E}_{p_h(z)}\ln\tilde{p}_h(z)$. We also denote $\varepsilon = \varepsilon(Q)$ for brevity.

**Theorem 6** *Let $Z$ be a random variable distributed according to $p_Q(z)$ and assume that $KL(\hat{p}_Q(z)\|p_Q(z)) \leq \varepsilon$. Then $-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z)$ is minimized by $\gamma = \frac{\sqrt{\varepsilon/2}}{|\mathcal{Z}|}$. For this value of $\gamma$ the following inequalities hold:*

$$-\mathbb{E}_{Q(h)}\mathbb{E}_{p_h(z)}\ln\tilde{p}_h(z) \leq H(\hat{p}_Q(z)) + \sqrt{\varepsilon/2}\ln|\mathcal{Z}| + \phi(\varepsilon), \tag{15}$$

$$-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z) \leq H(\hat{p}_Q(z)) + \sqrt{\varepsilon/2}\ln|\mathcal{Z}| + \phi(\varepsilon), \tag{16}$$

$$-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z) \geq H(\hat{p}_Q(z)) - \sqrt{\varepsilon/2}\ln|\mathcal{Z}| - \psi(\varepsilon), \tag{17}$$

*where:*

$$\psi(\varepsilon) = \sqrt{\varepsilon/2}\ln\frac{1+\sqrt{\varepsilon/2}}{\sqrt{\varepsilon/2}} \;\; \textit{and} \;\; \phi(\varepsilon) = \psi(\varepsilon) + \ln(1+\sqrt{\varepsilon/2}).$$

The proof is provided in appendix A. Note that both $\phi(\varepsilon)$ and $\psi(\varepsilon)$ converge to zero approximately as $-\sqrt{\varepsilon/2}\ln\sqrt{\varepsilon/2}$ and for $\sqrt{\varepsilon/2} < \frac{1}{|\mathcal{Z}|}$ they are in fact dominant over the $\sqrt{\varepsilon/2}\ln|\mathcal{Z}|$ term. Nevertheless, the main message is still that in order to minimize $-\mathbb{E}_{p_Q(z)}\ln\tilde{p}_Q(z)$ the trade-off between $H(\hat{p}_Q(z))$ and $KL(\hat{p}_Q(z)\|p_Q(z))$ should be minimized. This message is explored in more details in Section 3.4.

**Remark:** As an aside we consider the case of direct density estimation. Assume we are given a set of $N$ i.i.d. observations $x_1,..,x_N$ generated according to an unknown distribution $p(x)$ over a finite domain $\mathcal{X}$. We want to construct an estimate $\tilde{p}(x)$ for $p(x)$ based on the empirical frequencies $\hat{p}(x)$, such that the expectation $-\mathbb{E}_{p(x)}\ln\tilde{p}(x)$ is minimized. This problem, known as "histogram smoothing", has received significant attention in statistics and information theory (Gilbert, 1971; Cover, 1972; Krichevskiy, 1998; Paninski, 2004). Uniform smoothing of the form

$$\tilde{p}(x) = \frac{\hat{p}(x)+\gamma}{1+\gamma|\mathcal{X}|},$$

such as the one applied in (14) is known as the Dirichlet-Bayes or "add-constant" estimator. Theorem 6 provides the optimal value of $\gamma$

$$\gamma = \frac{1}{|\mathcal{X}|}\sqrt{\varepsilon/2} = \frac{1}{|\mathcal{X}|}\sqrt{\frac{(|\mathcal{X}|-1)\ln(N+1)-\ln\delta}{2N}}$$

for which with probability greater than $1-\delta$ over the sample

$$H(\hat{p}(x)) - \sqrt{\varepsilon/2}\ln|\mathcal{X}| - \psi(\varepsilon) \le -\mathbb{E}_{p(x)}\ln\tilde{p}(x) \le H(\hat{p}(x)) + \sqrt{\varepsilon/2}\ln|\mathcal{X}| + \phi(\varepsilon),$$

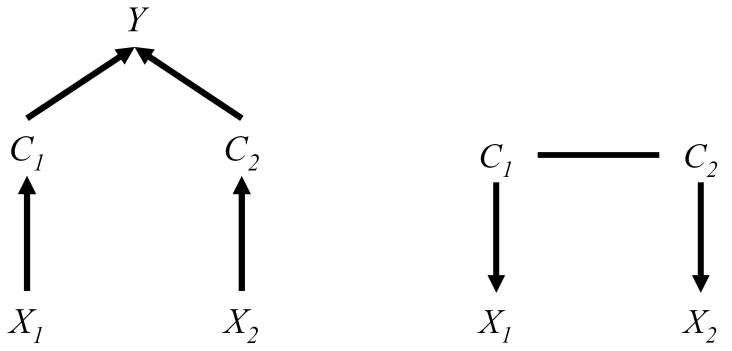
where $\varepsilon = \frac{(|\mathcal{X}|-1)\ln(N+1)-\ln\delta}{N}$ is obtained from Theorem 5. By Theorem 6 the optimal smoothing $\gamma$ decreases as the sample size $N$ increases. A more detailed comparison of this result with preceding work is beyond the scope of this paper and will be presented elsewhere. Note that in the more general case considered in Theorem 6, where the distribution $p_Q(z)$ depends on $Q$ the smoothing parameter $\gamma$ also depends on $Q$.

## 3. PAC-Bayesian Analysis of Co-clustering

In the introduction we defined two high-level goals, which can be solved via co-clustering. The first is discriminative prediction of the matrix entries, as in the collaborative filtering example. The second is estimation of the joint probability distribution in co-occurrence data analysis. We further defined the notion of generalization for each of the two problems. In this section we derive PAC-Bayesian generalization bounds for the two settings. We begin with the co-clustering approach to discriminative prediction, which is slightly easier in terms of presentation. Then we consider the discrete density estimation problem.

### 3.1 PAC-Bayesian Analysis of Discriminative Prediction with Grid Clustering

Let $\mathcal{X}_1 \times .. \times \mathcal{X}_d \times \mathcal{Y}$ be a $(d+1)$-dimensional product space. We assume that each $\mathcal{X}_i$ is categorical and its cardinality, denoted by $|\mathcal{X}_i| = n_i$, is fixed and known. We also assume that $\mathcal{Y}$ is finite with cardinality $|\mathcal{Y}|$ and that a bounded loss function $l(y,y')$ for predicting $y'$ instead of $y$ is given. As an example, consider collaborative filtering. In collaborative filtering $d = 2$, $\mathcal{X}_1$ is the space of the




(a) Graphical Model for Discriminative Prediction

(b) Graphical Model for Density Estimation

Figure 1: **Illustration of graphical models corresponding to discriminative prediction** (18) **and density estimation** (27)**.** The illustrations are for $d = 2$.

viewers, $n_1$ is the number of viewers, $\mathcal{X}_2$ is the space of the movies, $n_2$ is the number of movies, and $\mathcal{Y}$ is the space of the ratings (e.g., on a five-star scale). The loss $l(y,y')$ can be, for example, the absolute loss $l(y,y') = |y-y'|$ or the quadratic loss $l(y,y') = (y-y')^2$. There is no natural metric either on the space of viewers or on the space of movies; thus both $\mathcal{X}_1$ and $\mathcal{X}_2$ are categorical.

We assume an existence of an unknown probability distribution $p(x_1,..,x_d,y)$ over the $\mathcal{X}_1 \times .. \times \mathcal{X}_d \times \mathcal{Y}$ product space. We further assume that we are given an i.i.d. sample of size $N$ generated according to $p(x_1,..,x_d,y)$. We use $\hat{p}(x_1,..,x_d,y)$ to denote the empirical frequencies of $(d+1)$-tuples $\langle x_1,..,x_d,y\rangle$ in the sample. We consider the following form of discriminative predictors:

$$q(y|x_1,..,x_d) = \sum_{c_1,..,c_d} q(y|c_1,..,c_d) \prod_{i=1}^{d} q(c_i|x_i). \tag{18}$$

The hidden variables $C_1,..,C_d$ represent a clustering of the observed variables $X_1,..,X_d$. The hidden variable $C_i$ accepts values in $\{1,..,m_i\}$, where $m_i = |C_i|$ denotes the number of clusters used along dimension $i$. The conditional probability distribution $q(c_i|x_i)$ represents the probability of mapping (assigning) $x_i$ to cluster $c_i$. The conditional probability $q(y|c_1,..,c_d)$ represents the probability of assigning label $y$ to cell $\langle c_1,..,c_d\rangle$ in the cluster product space. The prediction model (18) corresponds to the graphical model in Figure 1.a. Note that this is a two-level randomized prediction model. The free parameters of the model are the conditional distributions $\{q(c_i|x_i)\}_{i=1}^{d}$ and $q(y|c_1,..,c_d)$. We denote these collectively by $Q = \left\{\{q(c_i|x_i)\}_{i=1}^{d}, q(y|c_1,..,c_d)\right\}$. In the next subsection we show that (18) corresponds to a randomized prediction strategy. We further denote:

$$L(Q) = \mathbb{E}_{p(x_1,..,x_d,y)} \mathbb{E}_{q(y'|x_1,..,x_d)} l(Y,Y')$$

and

$$\hat{L}(Q) = \mathbb{E}_{\hat{p}(x_1,..,x_d,y)} \mathbb{E}_{q(y'|x_1,..,x_d)} l(Y,Y'),$$

where $q(y|x_1,..,x_d)$ is defined by (18). We define

$$\bar{I}(X_i;C_i) = \frac{1}{n_i} \sum_{x_i,c_i} q(c_i|x_i) \ln \frac{q(c_i|x_i)}{\bar{q}(c_i)},$$

where $x_i \in \mathcal{X}_i$ are the possible values of $X_i$, $c_i \in \{1,..,m_i\}$ are the possible values of $C_i$, and

$$\overline{q}(c_i) = \frac{1}{n_i}\sum_{x_i} q(c_i|x_i)$$

is the marginal distribution over $C_i$ corresponding to $q(c_i|x_i)$ and a *uniform* distribution $u(x_i) = \frac{1}{n_i}$ over $\mathcal{X}_i$. Thus, $\overline{I}(X_i;C_i)$ is the mutual information corresponding to the joint distribution $\overline{q}(x_i,c_i) = \frac{1}{n_i}q(c_i|x_i)$ defined by $q(c_i|x_i)$ and the uniform distribution over $\mathcal{X}_i$.

With the above definitions we can state the following generalization bound for discriminative prediction with co-clustering.

**Theorem 7** *For any probability measure $p(x_1,..,x_d,y)$ over $\mathcal{X}_1 \times .. \times \mathcal{X}_d \times \mathcal{Y}$ and for any loss function $l$ bounded by 1, with probability of at least $1-\delta$ over a selection of an i.i.d. sample $S$ of size $N$ according to $p$, for all randomized classifiers $Q = \left\{\{q(c_i|x_i)\}_{i=1}^d, q(y|c_1,..,c_d)\right\}$:*

$$kl(\hat{L}(Q)\|L(Q)) \leq \frac{\sum_{i=1}^d (n_i\overline{I}(X_i;C_i) + m_i \ln n_i) + M\ln|\mathcal{Y}| + \frac{1}{2}\ln(4N) - \ln\delta}{N}, \tag{19}$$

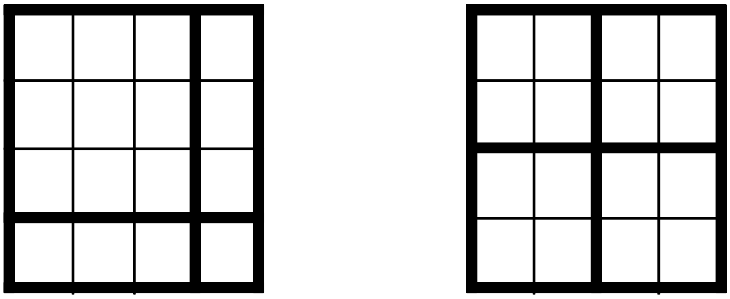
*where $M$ is the number of partition cells:*

$$M = \prod_{i=1}^d m_i.$$

**Remarks:** Of course, any bounded loss can be normalized to the [0,1] interval. Note that given a prediction strategy $Q = \left\{\{q(c_i|x_i)\}_{i=1}^d, q(y|c_1,..,c_d)\right\}$ both $\hat{L}(Q)$ and $\overline{I}(X_i;C_i)$ are computable exactly. $L(Q)$ can be bounded by numerical inversion of $kl$, as shown in equation (4). To minimize $L(Q)$ both $\hat{L}(Q)$ and $\overline{I}(X_i;C_i)$ should be minimized.

**Discussion:** There are two extreme solutions to the collaborative filtering task that provide good intuitions on the co-clustering approach to this problem. If we assign all of the data to a single large cluster, we can evaluate the empirical mean/median/most frequent rating of that cluster fairly well. In this situation the empirical loss $\hat{L}(Q)$ is expected to be large, because we approximate all the entries with the global average, but its distance to the true loss $L(Q)$ is expected to be small. If we take the other extreme and assign each row and each column to a separate cluster, $\hat{L}(Q)$ can be zero given that we can approximate every entry with its own value, but its distance to the true loss $L(Q)$ is expected to be large because each cluster has too little data to make a statistically reliable estimation. Thus, the goal is to optimize the trade-off between the locality of the predictions and their statistical reliability.

This trade-off is explicitly exhibited in bound (19): if we assign all $x_i$-es to a single cluster, then $\overline{I}(X_i;C_i) = 0$ and therefore $L(Q)$ is close to $\hat{L}(Q)$. And if we assign each $x_i$ to a separate cluster, then $\overline{I}(X_i;C_i)$ is large, specifically in this case $\overline{I}(X_i;C_i) = \ln n_i$, and $L(Q)$ is far from $\hat{L}(Q)$. But there are even finer observations we can draw from the bound. Bear in mind that $n_i\overline{I}(X_i;C_i)$ is linear in $n_i$, whereas $m_i \ln n_i$ is logarithmic in $n_i$. Thus, at least when $m_i$ is small compared to $n_i$ (which is a reasonable assumption when we cluster the values of $\mathcal{X}_i$) the leading term in (19) is $n_i\overline{I}(X_i;C_i)$. This term penalizes the *effective* complexity of a partition, rather than the raw number of clusters used. For example, the unbalanced partition of a $4\times 4$ matrix into $2\times 2$ clusters in Figure 2.a is simpler than the balanced partition into the same number of clusters in Figure 2.b. The reason, which will become clearer after we have defined the prior over the space of partitions in Subsection



(a) Unbalanced Partition (b) Balanced Partition

Figure 2: **Illustration of (a) an unbalanced and (b) a balanced partition of a 4×4 matrix into 2×2 clusters.** Note that there are 4 possible ways to group 4 objects into 2 unbalanced clusters and $\binom{4}{2} = 6$ possible ways to group 4 objects into 2 balanced clusters. Thus, the subspace of the unbalanced partitions is smaller than the subspace of the balanced partitions and the unbalanced partitions are simpler (it is easier to describe an unbalanced partition than a balanced one).

3.3, is that there are fewer unbalanced partitions than balanced ones. Therefore, the subspace of unbalanced partitions is smaller than the subspace of balanced partitions and it is easier to describe an unbalanced partition than a balanced one. Intuitively, the partition in Figure 2.a does not fully use the $2 \times 2$ clusters that it could use, and should therefore be penalized less. On a practical level, the bound makes it possible to operate at the optimization step with more clusters than are actually required and to penalize the final solution according to a de facto measure of cluster use. This claim is supported by our experiments. To summarize this point, the bound (19) suggests a trade-off between the empirical performance and the effective complexity of a partition.

Finally, consider the $M \ln |\mathcal{Y}|$ term in the bound. $M$ is the number of partition cells (in a hard partition) and $M \ln |\mathcal{Y}|$ corresponds to the size of the $\langle C_1, .., C_d, Y \rangle$ clique in the moral graph corresponding to the graph in Figure 1.a. The number of sample points $N$ should be comparable to the number of partition cells, so it is natural for this term to appear in the bound. This term grows exponentially with the number of dimensions $d$; thus we can apply the bound for low-dimensional problems like collaborative filtering, but when the number of dimensions grows a different approach is required. We suggest one possible way to handle high dimensional problems in Section 7.

**Proof of Theorem 7** The proof is a direct application of the PAC-Bayesian bound for classification in Theorem 1 (or, more precisely, its refinement in (3)). In order to apply the theorem, we have to define a hypothesis space $\mathcal{H}$, a prior over hypothesis space $\mathcal{P}$, a posterior over hypothesis space $\mathcal{Q}$, and calculate the KL-divergence $KL(\mathcal{Q}\|\mathcal{P})$. We define the hypothesis space in the next subsection and design a prior over it in Subsection 3.3. Substitution of the calculation of $KL(\mathcal{Q}\|\mathcal{P})$ in Lemma 9 into Theorem 1 completes the proof. ■

### 3.2 Grid Clustering Hypothesis Space

We define the hypothesis space $\mathcal{H}$ to be the space of hard grid partitions of the product space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ (as illustrated in Figure 2) augmented with label assignments to the partition cells. (In Subsection 3.4 we use grid partitions without labels on the partition cells; thus the discussion in this and the following subsection is kept general enough to hold in both cases.) In a hard grid partition each value $x_i \in \mathcal{X}_i$ is mapped deterministically to a single cluster $c_i \in \{1,..,m_i\}$. To operate on $\mathcal{H}$ we use the following notations:

- Let $\bar{m} = (m_1,..,m_d)$ be a vector counting the number of clusters along each dimension.
- We use $\mathcal{H}|_i$ to denote the space of partitions of $\mathcal{X}_i$. In other words, $\mathcal{H}|_i$ is a projection of $\mathcal{H}$ onto dimension $i$.
- Let $\mathcal{H}_{\bar{m}}$ denote the subspace of partitions of $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ in which the number of clusters used along each dimension matches $\bar{m}$. Obviously, for distinct $\bar{m}$-s, $\mathcal{H}_{\bar{m}}$-s are disjoint.
- We use $\mathcal{H}|_{(y|\bar{m})}$ or simply $\mathcal{H}|_{y|\bar{m}}$ to denote the space of possible assignments of labels to $\mathcal{H}_{\bar{m}}$. Then we can write $\mathcal{H} = \bigcup_{\bar{m}} \left(\mathcal{H}_{\bar{m}} \times \mathcal{H}|_{y|\bar{m}}\right)$.
- For each $h \in \mathcal{H}$ we write $h = h|_1 \times .. \times h|_d \times h|_{y|\bar{m}}$, where $h|_i$ denotes the partition induced by $h$ along dimension $i$ and $h|_{y|\bar{m}}$ denotes the assignment of labels to partition cells of $h$. In the discussion of density estimation with grid clustering in Subsection 3.4, $h$ is just $h = h|_1 \times .. \times h|_d$, without the labels assignment.

We show that $Q = \left\{\{q(c_i|x_i)\}_{i=1}^d, q(y|c_1,..,c_d)\right\}$ is a distribution over $\mathcal{H}$ and (18) corresponds to a randomized prediction strategy. More precisely, $Q$ is a distribution over $\mathcal{H}_{\bar{m}} \times \mathcal{H}|_{y|\bar{m}}$, where $\bar{m}$ matches the cardinalities of $C_i$-s in the definitions of $\left\{\{q(c_i|x_i)\}_{i=1}^d, q(y|c_1,..,c_d)\right\}$. In order to draw a hypothesis $h \in \mathcal{H}$ according to $Q$ we draw a cluster $c_i$ for each $x_i \in \mathcal{X}_i$ according to $q(c_i|x_i)$ and then draw a label for each partition cell according to $q(y|c_1,..,c_d)$. For example, we map each viewer to a cluster of viewers, map each movie to a cluster of movies, and assign ratings to the product space of viewer clusters by movie clusters. Then, in order to assign a label to a sample $\langle x_1,..,x_d\rangle$ we simply check which partition cell it has fallen into and return the corresponding label. Recall that in order to assign a label to another sample point, we have to draw a new hypothesis from $\mathcal{H}$.

Note that in (18) we actually skip the step of assigning a cluster for each $x_i \in \mathcal{X}_i$ and a label for each partition cell (in fact, the whole step of drawing a hypothesis) and assign a label to a given point $\langle x_1,..,x_d\rangle$ directly. Nevertheless, (18) corresponds to the randomized prediction process described above. This makes it possible to apply the PAC-Bayesian analysis.

### 3.3 Combinatorial Priors in PAC-Bayesian Bounds

In this section we design a combinatorial prior over the grid clustering hypothesis space and calculate the KL-divergence $KL(Q\|\mathcal{P})$ between the posterior defined earlier and the prior. An interesting point about this result is that combinatorial priors result in mutual information terms in the calculations of the KL-divergence. This can be contrasted with the $L_2$-norm and $L_1$-norm terms resulting from Gaussian and Laplacian priors respectively in the analysis of SVMs (Langford, 2005). Another important point to mention is that the posterior $Q$ returns a named partition of $\mathcal{X}_i$-s (the conditional

distribution $q(c_i|x_i)$ specifies the "name" $c_i$ of the cluster that $x_i$ is mapped to). However, the hypothesis space $\mathcal{H}$ and the prior $\mathcal{P}$ defined below operate with unnamed partitions: they only depend on the structure of a partition (the sizes of the clusters), but do not depend on the names assigned to the clusters. In this manner we account for all possible permutations of cluster names, which are irrelevant for the solution.

The statements in the next two lemmas are given in two versions, one for $\mathcal{H}$ augmented with labels, which is used in the proofs of Theorem 7, and the other for $\mathcal{H}_{\bar{m}}$ without the labels, which is used later for the proofs on density estimation with grid clustering.

**Lemma 8** *It is possible to define a prior $\mathcal{P}$ over $\mathcal{H}_{\bar{m}}$ that satisfies*:

$$\mathcal{P}(h) \geq \frac{1}{\exp\left[\sum_{i=1}^{d}\left(n_i H(hist(h|_i)) + (m_i - 1)\ln n_i\right)\right]}, \tag{20}$$

*where $hist(h|_i) = \{|c_{i1}|, .., |c_{im_i}|\}$ denotes the cardinality profile (histogram) of cluster sizes along dimension i of a partition corresponding to h and $H(hist(h|_i)) = -\sum_{j=1}^{m_i} \frac{|c_{ij}|}{n_i} \ln \frac{|c_{ij}|}{n_i}$ is the entropy of the (normalized) cardinality profile (note that $\sum_{j=1}^{m_i} |c_{ij}| = n_i$).*

*It is further possible to define a prior $\mathcal{P}$ over $\mathcal{H} = \bigcup_{\bar{m}} \left(\mathcal{H}_{\bar{m}} \times \mathcal{H}|_{y|\bar{m}}\right)$ that satisfies*:

$$\mathcal{P}(h) \geq \frac{1}{\exp\left[\sum_{i=1}^{d}\left(n_i H(hist(h|_i)) + m_i \ln n_i\right) + M \ln |\mathcal{Y}|\right]}. \tag{21}$$

**Remark:** The prior $\mathcal{P}$ over $\mathcal{H}$ that is defined explicitly in the proof of the lemma exploits structural asymmetries between more and less balanced grid partitions, as shown in Figure 2, without making assumptions on the data generating process. Note that the leading terms in the prior are $n_i H(hist(h|_i))$ that count the number of possible ways to assign $x_i$-s to $c_i$-s, which are invariant under permutation of $x_i$-s within each $\mathcal{X}_i$ (see the proof for details). Thus, it is impossible to design a significantly better prior without "strong" prior knowledge on the data generating process that can break the permutation symmetry on $x_i$-s. "Weak" prior knowledge on the number of clusters $m_i$ along each dimension and even on their sizes can only introduce an improvement that is logarithmic in $n_i$-s to the bounds. The PAC-Bayesian analysis enables us to operate with all possible grid partitions, while paying a very low (logarithmic) price for this generality.

**Lemma 9** *For the prior defined in* (20) *and posterior $Q = \{q(c_i|x_i)\}_{i=1}^{d}$*:

$$KL(Q\|\mathcal{P}) \leq \sum_{i=1}^{d} \left(n_i \bar{I}(X_i;C_i) + (m_i - 1)\ln n_i\right). \tag{22}$$

*For the prior defined in* (21) *and posterior $Q = \left\{\{q(c_i|x_i)\}_{i=1}^{d}, q(y|c_1, .., c_d)\right\}$*:

$$KL(Q\|\mathcal{P}) \leq \sum_{i=1}^{d} \left(n_i \bar{I}(X_i;C_i) + m_i \ln n_i\right) + M \ln |\mathcal{Y}|. \tag{23}$$

### 3.3.1 PROOFS

**Proof of Lemma 8** To define the prior $\mathcal{P}$ over $\mathcal{H}_{\bar{m}}$ we count the hypotheses in $\mathcal{H}_{\bar{m}}$. There are $\binom{n_i-1}{m_i-1} \leq n_i^{m_i-1}$ possibilities to choose a cluster cardinality profile along a dimension $i$. (This is

because each of the $m_i$ clusters has a size of at least one. To define a cardinality profile we are free to distribute the "excess mass" of $n_i - m_i$ among the $m_i$ clusters. The number of possible distributions equals the number of possibilities to place $m_i - 1$ ones in a sequence of $(n_i - m_i) + (m_i - 1) = n_i - 1$ ones and zeros.) For a fixed cardinality profile $hist(h|_i)$ there are $\binom{n_i}{|c_{i1}|,..,|c_{im_i}|} \leq e^{n_i H(hist(h|_i))}$ possibilities to assign $x_i$-s to the clusters. Putting the combinatorial calculations together we can define a distribution $\mathcal{P}$ over $\mathcal{H}_{\bar{m}}$ that satisfies (20).

To prove (21) we further define a uniform prior over $\mathcal{H}|_{y|\bar{m}}$. Note that there are $|\mathcal{Y}|^M$ possibilities to assign labels to the partition cells in $\mathcal{H}_{\bar{m}}$. Finally, we define a uniform prior over the choice of $\bar{m}$. There are $n_i$ possibilities to chose the value of $m_i$ (we can assign all $x_i$-s to a single cluster, assign each $x_i$ to a separate cluster, and all the possibilities in between). Combining this with the combinatorial calculations performed for (20) yields (21). ■

**Proof of Lemma 9** We first handle bound (22). We use the decomposition $KL(Q\|\mathcal{P}) = -\mathbb{E}_Q \ln \mathcal{P}(h) - H(Q)$ and bound $-\mathbb{E}_Q \ln \mathcal{P}(h)$ and $H(Q)$ separately. We further decompose $\mathcal{P}(h) = \mathcal{P}(h|_1) \cdot .. \cdot \mathcal{P}(h|_d)$ and $Q(h)$ in a similar manner. Then $-\mathbb{E}_Q \ln \mathcal{P}(h) = -\sum_i \mathbb{E}_Q \ln \mathcal{P}(h|_i)$ and $H(Q) = \sum_i H(Q(h|_i))$. Therefore, we can treat each dimension separately.

By Lemma 8:

$$-\mathbb{E}_Q \ln \mathcal{P}(h|_i) \leq (m_i - 1) \ln n_i + n_i \mathbb{E}_Q H(hist(h|_i)). \tag{24}$$

Hence, in order to bound $-\mathbb{E}_Q \ln \mathcal{P}(h|_i)$ we have to bound the expected entropy of cluster cardinality profiles of the hypotheses generated by $Q$. Recall that $Q$ draws a cluster $C_i$ for each $x_i \in \mathcal{X}_i$ according to $q(c_i|x_i)$ and that this process results in marginal distribution $\bar{q}(c_i) = \frac{1}{n_i} \sum_{x_i} q(c_i|x_i)$ over the normalized cluster sizes (this is where the uniform distribution over $\mathcal{X}_i$ comes in). To bound $\mathbb{E}_Q H(hist(h|_i))$ we use the result on negative bias of empirical entropy estimates cited below, see (Paninski, 2003) for a proof.

**Theorem 10 (Paninski, 2003)** *Let $X_1, .., X_N$ be i.i.d. distributed by $p(x)$ and let $\hat{p}(x)$ be their empirical distribution. Then:*

$$\mathbb{E}_p H(\hat{p}) = H(p) - \mathbb{E}_p KL(\hat{p}\|p) \leq H(p). \tag{25}$$

By (25) $\mathbb{E}_Q H(hist(h|_i)) \leq H(\bar{q}(c_i))$. Substituting this into (24) yields:

$$-\mathbb{E}_Q \ln \mathcal{P}(h|_i) \leq n_i H(\bar{q}(c_i)) + (m_i - 1) \ln n_i. \tag{26}$$

Now we turn to bound $-H(Q(h|_i)) = \mathbb{E}_Q \ln Q(h|_i)$. To do so we bound $\ln Q(h|_i)$ from above. The bound follows from the fact that if we draw $n_i$ values of $C_i$ according to $q(c_i|x_i)$ the probability of the resulting type is bounded from above by $e^{-n_i \bar{H}(C_i|X_i)}$, where $\bar{H}(C_i|X_i) = -\frac{1}{n_i} \sum_{x_i, c_i} q(c_i|x_i) \ln q(c_i|x_i)$ (see Cover and Thomas, 1991, Theorem 12.1.2). Thus, $\mathbb{E}_Q \ln Q(h|_i) \leq -n_i \bar{H}(C_i|X_i)$, which together with (26) and the identity $\bar{I}(X_i; C_i) = H(\bar{q}(c_i)) - \bar{H}(C_i|X_i)$ completes the proof of (22).

To prove (23) we recall that $Q$ is defined for a fixed $\bar{m}$. Hence, $-\mathbb{E}_Q \ln \mathcal{P}(h|_{y|\bar{m}}) = M \ln |\mathcal{Y}|$ and $-H(Q(h|_{y|\bar{m}})) \leq 0$. Finally, since the prior $\mathcal{P}(\bar{m})$ over the selection of $\bar{m}$ is uniform we have $-\mathbb{E}_Q \ln \mathcal{P}(\bar{m}) = \sum_{i=1}^d \ln n_i$ and $H(Q(\bar{m})) = 0$, which is added to (22) by the additivity of $KL(Q\|\mathcal{P})$ completing the proof. ■

### 3.4 PAC-Bayesian Analysis of Density Estimation with Grid Clustering

In this subsection we derive a generalization bound for density estimation with grid clustering. This time we have no labels and the goal is to find a good estimator for an unknown joint probability distribution $p(x_1,..,x_d)$ over a $d$-dimensional product space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ based on a sample of size $N$ from $p$. As an illustrative example, think of estimating a joint probability distribution of words and documents ($X_1$ and $X_2$) from their co-occurrence matrix. The goodness of an estimator $q$ for $p$ is measured by $-\mathbb{E}_{p(x_1,..,x_d)} \ln q(X_1,..,X_d)$.

By Theorem 5, to obtain a meaningful bound for a direct estimation of $p(x_1,..,x_d)$ from $\hat{p}(x_1,..,x_d)$ we need $N$ to be exponential in $n_i$-s, since the cardinality of the random variable $\langle X_1,..,X_d \rangle$ is $\prod_i n_i$. To reduce this dependency to be linear in $\sum_i n_i$ we restrict the estimator $q(X_1,..,X_d)$ to be of the factor form:

$$q(x_1,..,x_d) = \sum_{c_1,..,c_d} q(c_1,..,c_d) \prod_{i=1}^{d} q(x_i|c_i) \tag{27}$$

$$= \sum_{c_1,..,c_d} q(c_1,..,c_d) \prod_{i=1}^{d} \frac{q(x_i)}{q(c_i)} q(c_i|x_i). \tag{28}$$

We emphasize that the above decomposition assumption is only on the estimator $q$ and not on the generating distribution $p$. A graphical model corresponding to equation (27) is given in Figure 1.b. Similar to the model for discriminative prediction, this is also a two-level randomized prediction model.

We select the hypothesis space $\mathcal{H}$ to be the space of hard partitions of the product space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$, as before; however, this time there are no labels to the partition cells. The general message of the following two theorems is that the empirical distribution over the coarse partition space $C_1 \times .. \times C_d$ converges to the true one faster than the empirical distribution over $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ converges to its true counterpart. We also show that (28) can be used to extrapolate the distribution over cluster space back to $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ space and obtain better generalization guarantees. Next we state this more formally.

As seen in the previous subsection, a distribution $Q = \{q(c_i|x_i)\}_{i=1}^{d}$ is a distribution over $\mathcal{H}_{\bar{m}}$. To obtain a hypothesis $h \in \mathcal{H}_{\bar{m}}$ we draw a cluster for each $x_i \in \mathcal{X}_i$ according to $q(c_i|x_i)$. The way we have written (28) enables us to view it as a randomized prediction process: we draw a hypothesis $h$ according to $Q$ and then predict the probability of $\langle x_1,..,x_d \rangle$ as $q(c_1^h(x_1),..,c_d^h(x_d)) \prod_i \frac{q(x_i)}{q(c_i^h(x_i))}$, where $c_i^h(x_i) = h(x_i)$ is the partition cell that $x_i$ fell within $h$. Although (28) skips the process of drawing the complete partition $h$ and returns the probability of $\langle x_1,..,x_d \rangle$ directly, the described randomized prediction process matches the predictions by (28) and thus enables its analysis with PAC-Bayesian bounds.

Let $h \in \mathcal{H}$ be a hard partition of $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ and let $h(x_i)$ denote the cluster to which $x_i$ is mapped in $h$. We define the distribution over the partition cells $\langle c_1,..,c_d \rangle$ induced by $p$ and $h$:

$$p_h(c_1,..,c_d) = \sum_{\substack{x_1,..,x_d: \\ \forall i \;\; h(x_i)=c_i}} p(x_1,..,x_d),$$

$$p_h(c_i) = \sum_{x_i : h(x_i)=c_i} p(x_i).$$

We further define the distribution over partition cells induced by the empirical distribution $\hat{p}(x_1,..,x_d)$ corresponding to the sample and $h$ by substitution of $\hat{p}$ instead of $p$ in the above definitions:

$$\hat{p}_h(c_1,..,c_d) = \sum_{\substack{x_1,..,x_d: \\ \forall i \;\; h(x_i)=c_i}} \hat{p}(x_1,..,x_d),$$
$$\hat{p}_h(c_i) = \sum_{x_i:h(x_i)=c_i} \hat{p}(x_i).$$

We also define the distribution over partition cells induced by $Q$ and $p$ and its empirical counterpart:

$$p_Q(c_1,..,c_d) = \sum_h Q(h)p_h(c_1,..,c_d) = \sum_{x_1,..,x_d} p(x_1,..,x_d)\prod_{i=1}^d q(c_i|x_i),$$
$$p_Q(c_i) = \sum_h Q(h)p_h(c_i) = \sum_{x_i} p(x_i)q(c_i|x_i),$$

$$\hat{p}_Q(c_1,..,c_d) = \sum_h Q(h)\hat{p}_h(c_1,..,c_d) = \sum_{x_1,..,x_d} \hat{p}(x_1,..,x_d)\prod_{i=1}^d q(c_i|x_i),$$
$$\hat{p}_Q(c_i) = \sum_h Q(h)\hat{p}_h(c_i) = \sum_{x_i} \hat{p}(x_i)q(c_i|x_i).$$

We extrapolate $p_h$, $p_Q$, $\hat{p}_h$ and $\hat{p}_Q$ to the whole space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ using (28):

$$p_h(x_1,..,x_d) = p_h(c_1^h(x_1),..,c_d^h(x_d))\prod_{i=1}^d \frac{p(x_i)}{p_h(c_i^h(x_i))},$$
$$p_Q(x_1,..,x_d) = \sum_{c_1,..,c_d} p_Q(c_1,..,c_d)\prod_{i=1}^d \frac{p(x_i)}{p_Q(c_i)}q(c_i|x_i),$$
$$\hat{p}_h(x_1,..,x_d) = \hat{p}_h(c_1^h(x_1),..,c_d^h(x_d))\prod_{i=1}^d \frac{\hat{p}(x_i)}{\hat{p}_h(c_i^h(x_i))},$$
$$\hat{p}_Q(x_1,..,x_d) = \sum_{c_1,..,c_d} \hat{p}_Q(c_1,..,c_d)\prod_{i=1}^d \frac{\hat{p}(x_i)}{\hat{p}_Q(c_i)}q(c_i|x_i).$$

Note that $p_Q(x_1,..,x_d)$ is a distribution over $\mathcal{X}_1 \times .. \times \mathcal{X}_d$, which has the form (28) and is the closest to the true distribution $p(x_1,..,x_d)$ under the constraint that $\{q(c_i|x_i)\}_{i=1}^d$ are fixed. Further, note that since we have no access to $p(x_1,..,x_d)$ we do not know $p_Q(x_1,..,x_d)$. In the next theorem we provide rates of convergence of the distributions $\hat{p}_Q(x_1,..,x_d)$, $\hat{p}_Q(c_1,..,c_d)$, and $\hat{p}_Q(x_i)$ based on the sample to their counterparts corresponding to the true distribution $p(x_1,..,x_d)$.

**Theorem 11** *For any probability measure $p$ over $\mathcal{X}_1 \times .. \times \mathcal{X}_d$ and an i.i.d. sample $S$ of size $N$ according to $p$, with probability of at least $1-\delta$ for all grid clusterings $Q = \{q(c_i|x_i)\}_{i=1}^d$ the following holds simultaneously*:

$$KL(\hat{p}_Q(c_1,..,c_d)\|p_Q(c_1,..,c_d)) \leq \frac{\sum_{i=1}^d n_i \bar{I}(X_i;C_i) + K_1}{N} \tag{29}$$

*and for all $i$*

$$KL(\hat{p}(x_i)\|p(x_i)) \leq \frac{(n_i-1)\ln(N+1) + \ln\frac{d+1}{\delta}}{N}, \tag{30}$$

*where*

$$K_1 = \sum_{i=1}^{d} m_i \ln n_i + (M-1)\ln(N+1) + \ln\frac{d+1}{\delta}. \tag{31}$$

*As well, with probability greater than* $1-\delta$:

$$KL(\hat{p}_Q(x_1,..,x_d)\|p_Q(x_1,..,x_d)) \leq \frac{\sum_{i=1}^{d} n_i \overline{I}(X_i;C_i) + K_2}{N}, \tag{32}$$

*where*

$$K_2 = \sum_{i=1}^{d} m_i \ln n_i + \left[M + \sum_{i=1}^{d} n_i - d - 1\right] \ln(N+1) - \ln\delta.$$

Before we discuss and prove the theorem we point out that although $\hat{p}_Q(x_1,..,x_d)$ converges to $p_Q(x_1,..,x_d)$ it still cannot be used to minimize $-\mathbb{E}_{p(x_1,..,x_d)} \ln \hat{p}_Q(X_1,..,X_d)$, because it is not bounded from zero. Also, we cannot construct a density estimator by smoothing $\hat{p}_Q(x_1,..,x_d)$ directly using Theorem 6, because the cardinality of the random variable $\langle X_1,..,X_d\rangle$ is $\prod_i n_i$ and this term will enter into the bounds. To circumvent this we take advantage of the factor form of $p_Q$ and use the bounds (29) and (30). We define an estimator $\tilde{p}_Q$, which is a smoothed version of $\hat{p}_Q$ in the following way:

$$\begin{aligned}
\tilde{p}_h(c_1,..,c_d) &= \frac{\hat{p}_h(c_1,..,c_d)+\gamma}{1+\gamma M}, \\
\tilde{p}(x_i) &= \frac{\hat{p}(x_i)+\gamma_i}{1+\gamma_i n_i}, \\
\tilde{p}_h(c_i) &= \sum_{x_i:h(x_i)=c_i} \tilde{p}(x_i), \\
\tilde{p}_h(x_1,..,x_d) &= \tilde{p}_h(c_1^h(x_1),..,c_d^h(x_d)) \prod_{i=1}^{d} \frac{\tilde{p}(x_i)}{\tilde{p}_h(c_i^h(x_i))}.
\end{aligned} \tag{33}$$

And for a distribution $Q$ over $\mathcal{H}$:

$$\tilde{p}_Q(c_1,..,c_d) = \frac{\hat{p}_Q(c_1,..,c_d)+\gamma}{1+\gamma M}, \tag{34}$$

$$\tilde{p}_Q(c_i) = \sum_{x_i} \tilde{p}(x_i) q(c_i|x_i) = \frac{\hat{p}_Q(c_i)+\gamma_i \overline{q}(c_i) n_i}{1+\gamma_i n_i}, \tag{35}$$

$$\begin{aligned}
\tilde{p}_Q(x_1,..,x_d) &= \sum_h Q(h)\tilde{p}_h(x_1,..,x_d) \\
&= \sum_{c_1,..,c_d} \tilde{p}_Q(c_1,..,c_d) \prod_{i=1}^{d} \frac{\tilde{p}(x_i)}{\tilde{p}_Q(c_i)} q(c_i|x_i).
\end{aligned} \tag{36}$$

In the following theorem we provide a bound on $-\mathbb{E}_{p(x_1,..,x_d)} \ln \tilde{p}_Q(X_1,..,X_d)$. Note that we take the expectation with respect to the true, unknown distribution $p$ that may have an arbitrary form (i.e., $p$ is not restricted to be of the factor form (27)).

**Theorem 12** *For the density estimator* $\tilde{p}_Q(x_1,..,x_d)$ *defined by equations* (33), (34), (35), *and* (36), $-\mathbb{E}_{p(x_1,..,x_d)} \tilde{p}_Q(X_1,..,X_d)$ *attains its minimum at* $\gamma(Q) = \frac{\sqrt{\varepsilon(Q)/2}}{M}$ *and* $\gamma_i = \frac{\sqrt{\varepsilon_i/2}}{n_i}$, *where* $\varepsilon(Q)$ *is*

*defined by the right-hand side of* (29) *and* $\varepsilon_i$*-s are defined by the right-hand side of* (30)*. At this optimal level of smoothing, with probability greater than* $1-\delta$ *for all* $Q=\{q(c_i|x_i)\}_{i=1}^d$ *simultaneously*:

$$-\mathbb{E}_{p(x_1,..,x_d)}\ln\tilde{p}_Q(X_1,..,X_d)\leq -I(\hat{p}_Q(c_1,..,c_d))+\ln(M)\sqrt{\frac{\sum_{i=1}^d n_i\bar{I}(X_i;C_i)+K_1}{2N}}+K_3, \qquad (37)$$

*where* $I(\hat{p}_Q(c_1,..,c_d))=\left[\sum_{i=1}^d H(\hat{p}_Q(c_i))\right]-H(\hat{p}_Q(c_1,..,c_d))$ *is the multi-information between* $C_1,..,C_d$ *with respect to* $\hat{p}_Q(c_1,..,c_d)$*,* $K_1$ *is defined by* (31)*,*

$$K_3=\phi(\varepsilon(Q))+\left[\sum_{i=1}^d H(\hat{p}(x_i))+2\sqrt{\varepsilon_i/2}\ln n_i+\phi(\varepsilon_i)+\psi(\varepsilon_i)\right],$$

*and the functions* $\phi$ *and* $\psi$ *are defined in Theorem 6.*

**Discussion:** We discuss Theorem 12 first. We point out that $\tilde{p}_Q(x_1,..,x_d)$ is directly related to $\hat{p}_Q(x_1,..,x_d)$ and that $\hat{p}_Q(x_1,..,x_d)$ is determined by the empirical frequencies $\hat{p}(x_1,..,x_d)$ of the sample and our choice of $Q=\{q(c_i|x_i)\}_{i=1}^d$. There are only two quantities in the bound (37) that depend on the choice of $Q$: $-I(\hat{p}_Q(c_1,..,c_d))$ and $\sum_i\frac{n_i}{N}\bar{I}(X_i;C_i)$ [note that the latter also appears in $\phi(\varepsilon(Q)$ in $K_3$]. Thus, Theorem 12 suggests that a good estimator $\tilde{p}_Q(x_1,..,x_d)$ of $p(x_1,..,x_d)$ should optimize the trade-off between $-I(\hat{p}_Q(c_1,..,c_d))$ and $\sum_i\frac{n_i}{N}\bar{I}(X_i;C_i)$. Similar to Theorem 7, the latter term corresponds to the mutual information that the hidden cluster variables preserve on the observed variables. Larger values of $\bar{I}(X_i;C_i)$ correspond to partitions of $\mathcal{X}_1,..,\mathcal{X}_d$, which are more complex. The first term, $-I(\hat{p}_Q(c_1,..,c_d))$, corresponds to the amount of structural information on $C_i$-s extracted by the partition. More precisely, we need to look at the value of $\sum_i H(\hat{p}(x_i))-I(\hat{p}_Q(c_1,..,c_d))$, where $\sum_i H(\hat{p}(x_i))$ is a part of $K_3$ and roughly corresponds to the performance we can achieve by approximating $p(x_1,..,x_d)$ with a product of empirical marginals $\prod_i\hat{p}(x_i)$. Thus, $-I(\hat{p}_Q(c_1,..,c_d))$ is the added value of the partition in estimating $p(x_1,..,x_d)$ and since $\sum_i H(\hat{p}(x_i))\geq I(\hat{p}_Q(c_1,..,c_d))$ the bound (37) is always positive.

The value of $I(\hat{p}_Q(c_1,..,c_d))$ increases monotonically with the increase of the partition complexity $Q$ (we can see this by the information processing inequality (Cover and Thomas, 1991)). Thus, the trade-off in (37) is analogous to the trade-off in (19): the partition $Q$ should balance its utility function $-I(\hat{p}_Q(c_1,..,c_d))$ and the statistical reliability of the estimate of the utility function, which is related to $\sum_i\frac{n_i}{N}\bar{I}(X_i;C_i)$. This trade-off suggests a modification to the original objective of co-clustering in (Dhillon et al., 2003), which is maximization of $I(C_1;C_2)$ alone (Dhillon et al. (2003) discuss the case of two-dimensional matrices). The trade-off in (37) can be applied to model order selection.

Now we make a few comments concerning Theorem 11. An interesting point about this theorem is that the cardinality of the random variable $\langle X_1,..,X_d\rangle$ is $\prod_i n_i$. Thus, a direct application of Theorem 2 to bound $KL(\hat{p}_Q(x_1,..,x_d)\|p_Q(x_1,..,x_d))$ would introduce this term into the bound. However, by using the factor form (28) of $\hat{p}_Q(x_1,..,x_d)$ and $p_Q(x_1,..,x_d)$ we are able to reduce this dependency to $(M+\sum_i n_i-d-1)$. This result reveals the great potential of applying PAC-Bayesian analysis to more complex graphical models, which we explore further in Section 7.

### 3.4.1 PROOFS

We conclude this section by presenting the proofs of Theorems 11 and 12.

**Proof of Theorem 11** The proof is based on PAC-Bayesian theorem on density estimation (Theorem 2). To apply the theorem we need to define a prior $\mathcal{P}$ over $\mathcal{H}$ and then calculate $KL(Q\|\mathcal{P})$. We note that for a fixed $Q$ the cardinalities of the clusters $\bar{m}$ are fixed. There are $\prod_i n_i$ disjoint subspaces $\mathcal{H}_{\bar{m}}$ in $\mathcal{H}$. We handle each $\mathcal{H}_{\bar{m}}$ independently and then combine the results to obtain Theorem 11.

By Theorem 2 and Lemma 9, for the prior $\mathcal{P}$ over $\mathcal{H}_{\bar{m}}$ defined in Lemma 8, with probability greater than $1-\frac{\delta}{(d+1)\prod_i n_i}$ we obtain (29) for each $\mathcal{H}_{\bar{m}}$. In addition, by Theorem 5 with probability greater than $1-\frac{\delta}{d+1}$ inequality (30) holds for each $X_i$. By a union bound over the $\prod_i n_i$ subspaces of $\mathcal{H}$ and the $d$ variables $X_i$ we obtain that (29) and (30) hold simultaneously for all $Q$ and $X_i$ with probability greater than $1-\delta$.

To prove (32), fix some hard partition $h$ and let $c_i^h = h(x_i)$. Then:

$$\begin{aligned}
&KL(\hat{p}_h(x_1,..,x_d)\|p_h(x_1,..,x_d)) \\
&\quad = KL(\hat{p}_h(x_1,..,x_d,c_1^h(x_1),..,c_d^h(x_d))\|p_h(x_1,..,x_d,c_1^h(x_1),..,c_d^h(x_d))) \\
&\quad = KL(\hat{p}_h(c_1,..,c_d)\|p_h(c_1,..,c_d)) \\
&\qquad + KL(\hat{p}_h(x_1,..,x_d|c_1^h(x_1),..,c_d^h(x_d))\|p_h(x_1,..,x_d|c_1^h(x_1),..,c_d^h(x_d))) \\
&\quad = KL(\hat{p}_h(c_1,..,c_d)\|p_h(c_1,..,c_d)) + \sum_{i=1}^d KL(\hat{p}_h(x_i|c_i^h(x_i))\|p_h(x_i|c_i^h(x_i))) \\
&\quad = KL(\hat{p}_h(c_1,..,c_d)\|p_h(c_1,..,c_d)) + \sum_{i=1}^d KL(\hat{p}(x_i)\|p(x_i)) - \sum_{i=1}^d KL(\hat{p}_h(c_i)\|p_h(c_i)) \\
&\quad \le KL(\hat{p}_h(c_1,..,c_d)\|p_h(c_1,..,c_d)) + \sum_{i=1}^d KL(\hat{p}(x_i)\|p(x_i)).
\end{aligned}$$

And:

$$\begin{aligned}
\mathbb{E}_S e^{NKL(\hat{p}_h(x_1,..,x_d)\|p_h(x_1,..,x_d))} &\le \left(\mathbb{E}_S e^{NKL(\hat{p}_h(c_1,..,c_d)\|p_h(c_1,..,c_d))}\right)\prod_{i=1}^d \mathbb{E}_S e^{NKL(\hat{p}(x_i)\|p(x_i))} \\
&\le (N+1)^{M+\sum_{i=1}^d n_i-(d+1)},
\end{aligned}$$

where the last inequality is by Theorem 3. From here we follow the lines of the proof of Theorem 2. Namely:

$$\begin{aligned}
\mathbb{E}_S\left[\mathbb{E}_{\mathcal{P}(h)} e^{NKL(\hat{p}_h(x_1,..,x_d)\|p_h(x_1,..,x_d))}\right] &= \mathbb{E}_{\mathcal{P}(h)}\left[\mathbb{E}_S e^{NKL(\hat{p}_h(x_1,..,x_d)\|p_h(x_1,..,x_d))}\right] \\
&\le (N+1)^{M+\sum_{i=1}^d n_i-(d+1)}.
\end{aligned}$$

Thus, by Markov's inequality $\mathbb{E}_{\mathcal{P}(h)} e^{NKL(\hat{p}_h(x_1,..,x_d)\|p_h(x_1,..,x_d))} \le \frac{1}{\delta}(N+1)^{M+\sum_i n_i-(d+1)}$ with probability of at least $1-\delta$ and (32) follows by the change of measure inequality (8) and convexity of the KL-divergence, when the prior $\mathcal{P}$ over $\mathcal{H}$ defined in Lemma 8 is selected (this time we give a weight of $(\prod_i n_i)^{-1}$ to each $\mathcal{H}_{\bar{m}}$ and obtain a prior over the whole $\mathcal{H}$). The calculation of $KL(Q\|\mathcal{P})$ for this prior is provided in Lemma 9. ■

**Proof of Theorem 12**

$$-\mathbb{E}_{p(x_1,..,x_d)} \ln \tilde{p}_Q(X_1,..,X_d) = -\mathbb{E}_{p(x_1,..,x_d)} \ln \mathbb{E}_{Q(h)} \tilde{p}_h(X_1,..,X_d)$$

$$
\begin{aligned}
&\leq -\mathbb{E}_{Q(h)}\mathbb{E}_{p(x_1,..,x_d)} \ln \tilde{p}_h(X_1,..,X_d) \\
&= -\mathbb{E}_{Q(h)}\mathbb{E}_{p(x_1,..,x_d)} \ln \tilde{p}_h(C_1^h(X_1),..,C_d^h(X_d)) \prod_i \frac{\tilde{p}(X_i)}{\tilde{p}_h(C_i^h(X_i))} \\
&= -\mathbb{E}_{Q(h)}\left[\mathbb{E}_{p_h(c_1,..,c_d)} \ln \tilde{p}_h(C_1,..,C_d)\right] - \sum_i \mathbb{E}_{p(x_i)} \ln \tilde{p}(X_i) + \sum_i \mathbb{E}_{Q(h)}\mathbb{E}_{p_h(c_i)} \ln \tilde{p}_h(C_i) \\
&\leq -\mathbb{E}_{Q(h)}\left[\mathbb{E}_{p_h(c_1,..,c_d)} \ln \tilde{p}_h(C_1,..,C_d)\right] - \sum_i \mathbb{E}_{p(x_i)} \ln \tilde{p}(X_i) + \sum_i \mathbb{E}_{p_Q(c_i)} \ln \tilde{p}_Q(C_i)
\end{aligned}
$$

At this point we use (15) to bound the first and the second term and the lower bound (17) to bound the last term and obtain (37). ■

## 4. Algorithms

In the previous section we presented generalization bounds for discriminative prediction and density estimation with co-clustering. The bounds presented in Theorems 7 and 12 hold for any prediction rule $Q$ based on grid clustering of the parameter space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$. In (Seldin, 2009) it is shown that for $d = 1$ the global minimum of bound (19) can be found efficiently. However, for $d \geq 2$ it can be exponentially hard to find the global minimum of both (19) and (37). Further, although we show in the applications section that bound (19) is remarkably tight, its tightness can still be insufficient for practical purposes. In this section we suggest how to replace the bounds with parameterized trade-offs that can be further fine-tuned, e.g., via cross-validation, to improve the usability in practice. The substitution of the bounds with parametrized trade-offs does not compromise on the rigor of the analysis: first, the bounds hold simultaneously for all the solutions found by minimization of the parameterized trade-offs and, second, the parameter of the trade-offs can also be tuned by substituting the result of trade-off minimization back into the corresponding bound, thus providing an estimate on a local minima of the bound.

### 4.1 Minimization of the PAC-Bayesian Bound for Discriminative Prediction with Grid Clustering

We start with minimization of the PAC-Bayesian bound for discriminative prediction based on grid clustering (19) suggested in Theorem 7. We rewrite the bound in a slightly different way in order to separate terms which are independent of the conditional distributions in $Q$:

$$kl(\hat{L}(Q)\|L(Q)) \leq \frac{\sum_{i=1}^d n_i \bar{I}(X_i;C_i) + K}{N}, \tag{38}$$

where

$$K = \sum_{i=1}^d m_i \ln n_i + M \ln|\mathcal{Y}| + \frac{1}{2}\ln(4N) - \ln\delta. \tag{39}$$

Note that $K$ depends on the number of clusters $m_i$ used along each dimension, but not on a specific form of a grid partition. Once the number of clusters used along each dimension has been selected, $K$ is constant.

The minimization problem corresponding to (38) can be stated as follows:

$$\min_Q L \quad s.t. \quad kl(\hat{L}(Q)\|L) = \frac{\sum_{i=1}^d n_i \bar{I}(X_i;C_i) + K}{N}. \tag{40}$$

It is generally possible to find a local minimum of the minimization problem (40) directly using alternating projection methods - see, e.g., (Germain et al., 2009) for such an approach to solving a similar minimization problem for linear classifiers. We take a slightly different approach that further enables us to compensate for the imperfection of the bounds. Since $K$ is constant, $L(Q)$ depends on a parameterized trade-off between $\hat{L}(Q)$ and $\sum_{i=1}^{d} n_i \bar{I}(X_i;C_i)$, which can be written as follows:

$$\mathcal{F}(Q,\beta) = \beta N \hat{L}(Q) + \sum_{i=1}^{d} n_i \bar{I}(X_i;C_i), \tag{41}$$

where $\beta$ is the trade-off parameter. The corresponding minimization problem is:

$$\mathcal{F}_{min}(\beta) = \min_{Q} \beta N \hat{L}(Q) + \sum_{i=1}^{d} n_i \bar{I}(X_i;C_i). \tag{42}$$

In general, every value of $\beta$ yields a different solution to the minimization problem (42). The optimum of (40) (which is computationally hard to find) corresponds to some specific value of $\beta$. Hence, by scanning the possible values of $\beta$, minimizing (42), and substituting the resulting $\hat{L}(Q)$ and $\bar{I}(X_i;C_i)$ back into (19) it is virtually possible to find the optimum of (40) (only virtually, because finding the global optimum of (42) is computationally hard as well). However, the trade-off (41) provides an additional degree of freedom. In cases where the bound (19) is not sufficiently tight for practical applications it is possible to tune the trade-off by determining the desired value of $\beta$ via cross-validation instead of back-substitution into the bound.

The minimization problem (42) is closely related to the rate distortion trade-off in information theory (Cover and Thomas, 1991). To find a local minimum of $\mathcal{F}(Q,\beta)$ we adopt an EM-like alternating projection procedure, very similar to the Blahut-Arimoto algorithm for minimization of the rate distortion function (Arimoto, 1972; Blahut, 1972; Cover and Thomas, 1991). We note that for $d \geq 2$ the alternating projections involve more than two convex sets and hence only a local minimum can be achieved. (For $d = 1$ the procedure achieves the global minimum.) For the sake of simplicity of the notations we restrict ourselves to the case of $d = 2$, but it is straightforward to extend the algorithm to higher dimensions.

The Lagrangian corresponding to the minimization problem (42) is:

$$\mathcal{L}(Q,\beta) = \beta N \hat{L}(Q) + \sum_{i=1}^{2} n_i \bar{I}(X_i;C_i) + \sum_{i=1}^{2} \sum_{x_i \in \mathcal{X}_i} \nu(x_i) \sum_{c_i} q(c_i|x_i) + \sum_{c_1,c_2} \nu(c_1,c_2) \sum_{y} q(y|c_1,c_2),$$

where $\nu$-s are Lagrange multipliers corresponding to normalization constraints on $\{q(c_i|x_i)\}_{i=1}^{2}$ and $q(y|c_1,c_2)$. In order to minimize $\mathcal{L}(Q,\beta)$ we write $\hat{L}(Q)$ explicitly:

$$\begin{aligned}
\hat{L}(Q) &= \sum_{x_1,x_2,y} \hat{p}(x_1,x_2,y) \sum_{y'} q(y'|x_1,x_2) l(y,y') \\
&= \sum_{x_1,x_2,y} \hat{p}(x_1,x_2,y) \sum_{y',c_1,c_2} q(y'|c_1,c_2) q(c_1|x_1) q(c_2|x_2) l(y,y') \\
&= \sum_{y,y'} l(y,y') \sum_{c_1,c_2} q(y'|c_1,c_2) \sum_{x_1,x_2} q(c_1|x_1) \hat{p}(x_1,x_2,y) q(c_2|x_2).
\end{aligned}$$

We further derive $\hat{L}(Q)$ with respect to $q(c_1|x_1)$. The derivative with respect to $q(c_2|x_2)$ is similar.

$$\frac{\partial \hat{L}(Q)}{\partial q(c_1|x_1)} = \sum_{y,y'} l(y,y') \sum_{x_2,c_2} q(y'|c_1,c_2) \hat{p}(x_1,x_2,y) q(c_2|x_2). \tag{43}$$

Recall that $\bar{I}(X_i;C_i) = \frac{1}{n_i}\sum_{x_i,c_i} q(c_i|x_i)\ln\frac{q(c_i|x_i)}{\bar{q}(c_i)}$ and $\bar{q}(c_i) = \frac{1}{n_i}\sum_{x_i} q(c_i|x_i)$. Hence the derivative of $n_i\bar{I}(X_i;C_i)$ is:

$$\frac{\partial n_i\bar{I}(X_i;C_i)}{\partial q(c_i|x_i)} = \ln\frac{q(c_i|x_i)}{\bar{q}(c_i)}.$$

Derivatives of the remaining terms in $\mathcal{L}(Q,\beta)$ provide normalization for the corresponding variables. Thus, by taking the derivative of $\mathcal{L}(Q,\beta)$ with respect to $q(c_i|x_i)$, equating it to zero and reorganizing the terms we obtain a set of self-consistent equations that can be iterated until convergence:

$$\begin{aligned}
\bar{q}_t(c_i) &= \frac{1}{n_i}\sum_{x_i} q_t(c_i|x_i),\\
q_{t+1}(c_i|x_i) &= \frac{\bar{q}_t(c_i)}{Z_{t+1}(x_i)} e^{-\beta N\frac{\partial\hat{L}(Q_t)}{\partial q(c_i|x_i)}},\\
Z_{t+1}(x_i) &= \sum_{c_i} q_{t+1}(c_i|x_i),\\
y^*_{t+1}(c_1,c_2) &= \arg\min_{y'}\sum_{y} l(y,y')\sum_{x_1,x_2} q_{t+1}(c_1|x_1)\hat{p}(x_1,x_2,y)q_{t+1}(c_2|x_2), \qquad (44)\\
q_{t+1}(y|c_1,c_2) &= \delta[y,y^*_{t+1}(c_1,c_2)], \qquad (45)
\end{aligned}$$

where $\delta[\cdot,\cdot]$ is the Kronecker delta function, $\frac{\partial\hat{L}(Q)}{\partial q(c_i|x_i)}$ is given by (43), and the subindex $t$ denotes the iteration number. Equations (44) and (45) correspond to minimization of $\hat{L}(Q)$ with respect to $q(y|c_1,c_2)$ and generally depend on the loss function. For the zero-one loss, $y^*(c_1,c_2)$ is the most frequent value of $y$ appearing in the $\langle c_1,c_2\rangle$ partition cell; for the absolute loss it is the median value; for the quadratic loss it is the average value. We summarize the algorithm in the Algorithm 1 box[2]. We note that for the quadratic loss the loss minimizer $y^*(c_1,c_2)$, which is the average value in this case, can fall out of the finite space of labels $\mathcal{Y}$. However, the algorithm can still be applied and a bound can be obtained by post-process quantization, see appendix B for details.

### 4.2 Minimization of the PAC-Bayesian Bound for Density Estimation

Similar to the PAC-Bayesian bound for discriminative prediction, the PAC-Bayesian bound for density estimation (37) depends on the trade-off:

$$\mathcal{G}(Q,\beta) = -\beta N I(\hat{p}_Q(c_1,c_2)) + \sum_{i=1}^{2} n_i\bar{I}(X_i;C_i).$$

All other terms in (37) do not depend on the specific form of grid partition $Q$. (As in the previous subsection we restrict ourselves to $d=2$.) Unfortunately, $-I(\hat{p}_Q(c_1,c_2))$ is concave in $q(c_i|x_i)$-s, whereas $\bar{I}(X_i;C_i)$ is convex in $q(c_i|x_i)$. Therefore, alternating projection methods are hard to apply. Instead, $\mathcal{G}(Q,\beta)$ can be minimized (with respect to $Q$) using sequential minimization (Slonim et al., 2002; Dhillon et al., 2003). The essence of the sequential minimization method is that we start with some random assignment $q(c_i|x_i)$ and then iteratively take $x_i$-s out of their clusters and reassign them to new clusters, so that $\mathcal{G}(Q,\beta)$ is minimized. This approach leads to a hard partition of the data (i.e., each $x_i$ is deterministically assigned to a single $c_i$). The algorithm is given in Algorithm 2 box.

2. Matlab implementation of the algorithm is available at `http://www.kyb.mpg.de/~seldin`.

**Algorithm 1** Algorithm for minimization of $\mathcal{F}(Q,\beta) = \beta N\hat{L}(Q) + \sum_{i=1}^{2} n_i \overline{I}(X_i;C_i)$ by alternating projections.

**Input:** $\hat{p}(x_1,x_2,y)$, $N$, $n_1$, $n_2$, $m_1$, $m_2$, $l(y,y')$, $|\mathcal{Y}|$, $\beta$.
**Initialize:** $q_0(c_i|x_i)$ and $q_0(y|c_1,c_2)$ randomly.
$t \leftarrow 0$
$\overline{q}_t(c_i) \leftarrow \frac{1}{n_i}\sum_{x_i} q_t(c_i|x_i)$
**repeat**
  **for** $i = 1,2$ **do**
    $q_{t+1}(c_i|x_i) \leftarrow \overline{q}_t(c_i) e^{-\beta N \frac{\partial \hat{L}(Q_t)}{\partial q(c_i|x_i)}}$
    $Z_{t+1}(x_i) \leftarrow \sum_{c_i} q_{t+1}(c_i|x_i)$
    $q_{t+1}(c_i|x_i) \leftarrow \frac{q_{t+1}(c_i|x_i)}{Z_{t+1}(x_i)}$
    $\overline{q}_{t+1}(c_i) \leftarrow \frac{1}{n_i}\sum_{x_i} q_{t+1}(c_i|x_i)$
    $y^*_{t+1}(c_1,c_2) \leftarrow \arg\min_{y'} \sum_y l(y,y') \sum_{x_1,x_2} q_{t+1}(c_1|x_1)\hat{p}(x_1,x_2,y)q_{t+1}(c_2|x_2)$
    $q_{t+1}(y|c_1,c_2) \leftarrow \delta[y,y^*_{t+1}(c_1,c_2)]$
    $t \leftarrow t+1$
  **end for**
**until** convergence
**return** $\{q_t(c_i|x_i)\}_{i=1}^2, q_t(y|c_1,c_2)$ from the last iteration.

**Algorithm 2** Algorithm for sequential minimization of $\mathcal{G}(Q,\beta) = -\beta N I(\hat{p}_Q(c_1,c_2)) + \sum_{i=1}^{2} n_i \overline{I}(X_i;C_i)$.

**Input:** $\hat{p}(x_1,x_2)$, $N$, $n_1$, $n_2$, $m_1$, $m_2$, $\beta$.
**Initialize:** $q_0(c_i|x_i)$ randomly.
**repeat**
  **for** all $x_1 \in \mathcal{X}_1$ and all $x_2 \in \mathcal{X}_2$ according to some random order over $\mathcal{X}_1$ and $\mathcal{X}_2$ **do**
    **for** $i = 1,2$ **do**
      Select $x_i \in \mathcal{X}_i$ according to the order selected above.
      Compute $\mathcal{G}(Q,\beta)$ for each possible assignment of $x_i$ to $c_i \in \{1,..,m_i\}$
      Reassign $x_i$ to $c_i$ such that $\mathcal{G}(Q,\beta)$ is minimized.
      Update $\hat{p}_Q(c_1,c_2) \leftarrow \sum_{x_1,x_2} q(c_1|x_1)\hat{p}(x_1,x_2)q(c_2|x_2)$.
    **end for**
  **end for**
**until** no reassignments further minimize $\mathcal{G}(Q,\beta)$.
**return** $\{q(c_i|x_i)\}_{i=1}^2$ from the last iteration.

## 5. Applications

In this section we illustrate an application of the PAC-Bayesian bound for discriminative prediction based on co-clustering (19) and Algorithm 1 for minimization of the corresponding trade-off (41) on the problem of collaborative filtering. The problem of collaborative filtering was discussed in the previous sections. The goal of collaborative filtering is to complete the missing entries in a viewers-by-movies ratings matrix. This problem attracted a great deal of attention recently thanks to the Netflix challenge[3]. Since our goal here is mainly to illustrate our approach to co-clustering via the PAC-Bayesian bounds rather than to solve the large-scale challenge we concentrate on a much smaller MovieLens 100K dataset[4]. The dataset consists of 100,000 ratings on a five-star scale for 1,682 movies by 943 users. We take the five non-overlapping splits of the dataset into 80% training and 20% test subsets provided at the MovieLens website. We stress that the training data are extremely sparse - only 5% of the training matrix entries are populated, whereas 95% of the values are missing.

To measure the accuracy of our algorithm we use the mean absolute error (MAE) measure, which is commonly used for evaluation on this dataset (Herlocker et al., 2004). Let $\check{p}(x_1,x_2,y)$ be the distribution over $\langle X_1,X_2,Y\rangle$ in the test set. The mean absolute error is defined as:

$$MAE = \sum_{x_1,x_2,y} \check{p}(x_1,x_2,y) \sum_{y'} q(y'|x_1,x_2)|y-y'|.$$

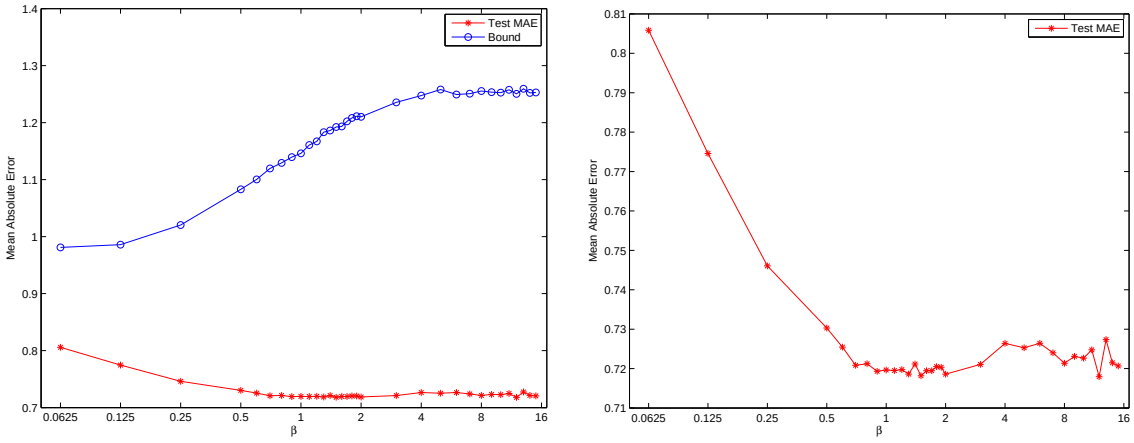
In previous work the best MAE reported for this dataset was 0.73 (Herlocker et al., 2004). It is worth recalling that the ratings are on a five-star scale, thus a MAE of 0.73 means that, on average, the predicted rating is 0.73 stars (less than one star) far from the observed rating. The maximal possible error is 4 (which occurs if we predict one star instead of five or vice versa), which determines the scale on which all the results should be judged.

In (Seldin et al., 2007) we improved the MAE on this dataset to 0.72 by using a Minimum Description Length (MDL, Grünwald, 2007) formulation of co-clustering. In the MDL formulation the co-clustering solutions are evaluated by the total description length, which includes the length of the description of assignments of $x_i$-s to $c_i$-s together with the length of the description of the ratings given the assignments. For fixed numbers of clusters ($m_i$-s) used along each dimension, the MDL solution corresponds to optimization of the trade-off (41) with logarithmic loss $\hat{L}(Q) = \hat{I}(Y;C_1,C_2)$ and $\beta = 1$ (where $\hat{I}(Y;C_1,C_2)$ is the empirical mutual information between the clusters and the label). In the MDL formulation of co-clustering developed in (Seldin et al., 2007) only hard (deterministic) assignments of $x_i$-s to $c_i$-s were considered. The best performance of 0.72 was achieved at $m_1 = 13$ and $m_2 = 6$ with below 1% sensitivity to small changes in $m_1$ and $m_2$ both in the description length and in the prediction accuracy. The deviation in prediction accuracy between the five splits of the MovieLens dataset was below 0.01.

In the present work we implemented Algorithm 1 for minimization of $\mathcal{F}(Q,\beta)$ as a function of $Q$ and applied it to the MovieLens dataset. There are four major differences between Algorithm 1 and the MDL algorithm suggested in (Seldin et al., 2007) that should be highlighted:

- Algorithm 1 directly optimizes a given loss function (MAE in the case of MovieLens) rather than the description length, which is only indirectly related to the loss function.

3. See `http://www.netflixprize.com/rules`.
4. Available at `http://www.grouplens.org`.



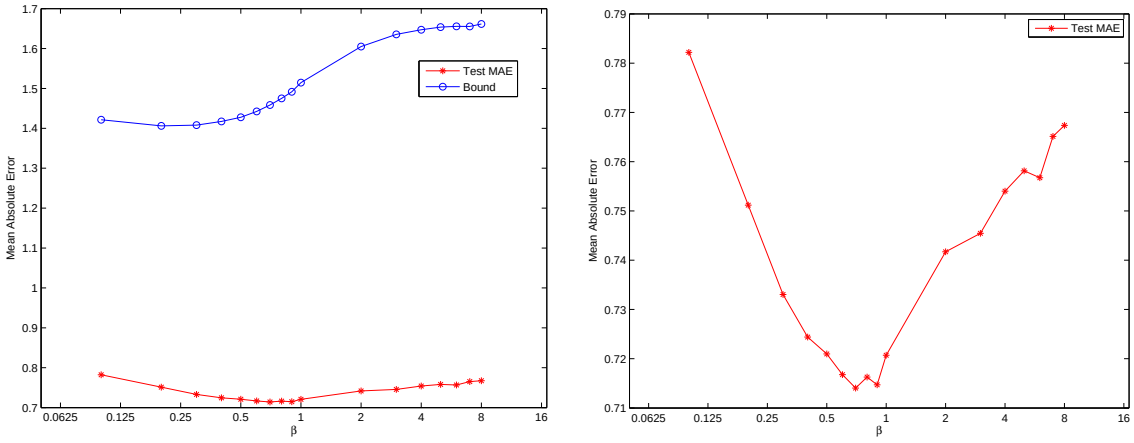

(a) Bound (19). (b) Test Loss (zoom into subfigure a.).

Figure 3: **Co-clustering of the MovieLens dataset into 13x6 clusters.** Figure (a) shows the value of bound (19) together with the MAE on the test set as a function of β. Figure (b) zooms into MAE on the test set. The values of β are on a log scale. See text for further details.

- Algorithm 1 considers soft assignments of $x_i$-s to $c_i$-s.
- Algorithm 1 is an iterative projection algorithm rather than the sequential optimization algorithm suggested in (Seldin et al., 2007). Note that this point is neither positive nor negative, since sequential optimization algorithms are very powerful and especially in hard cases can outperform iterative projection methods. The advantage of iterative projection methods is in their mathematical elegance, faster convergence (although in the hard cases it may be fast convergence to trivial, but strong attractors), and the ability to handle soft assignments.
- Algorithm 1 considers arbitrary values of β. (However, the algorithm in (Seldin et al., 2007) can be easily extended to handle arbitrary values of β.) As we will show below, the value of $\beta = 1$ dictated by the MDL formulation is not always optimal and MDL solutions can overfit the data. This observation was already made previously in a context of other learning problems by Kearns et al. (1997).

We conducted three experiments with Algorithm 1. In all three experiments we fixed the numbers of clusters $m_1$ and $m_2$ used along both dimensions and analyzed the MAE on the test set and the value of bound (19) as a function of β. In each experiment, for each of the five splits of the dataset into training and test sets mentioned earlier, and for each value of β we applied 10 random initializations of the algorithm. The solution $Q$ corresponding to the best value of $\mathcal{F}(Q,\beta)$ per each data split and per each value of β was then selected. We further calculated the average of the results over the dataset splits to produce the graphs of the bound values and test MAE as functions of β.

In the first experiment we verified that we are able to reproduce the results achieved previously in (Seldin et al., 2007). We set $m_1 = 13$ and $m_2 = 6$, as the best values obtained in (Seldin et al., 2007) and applied Algorithm 1. The results are presented in Figure 3. We make the following conclusions from this experiment:

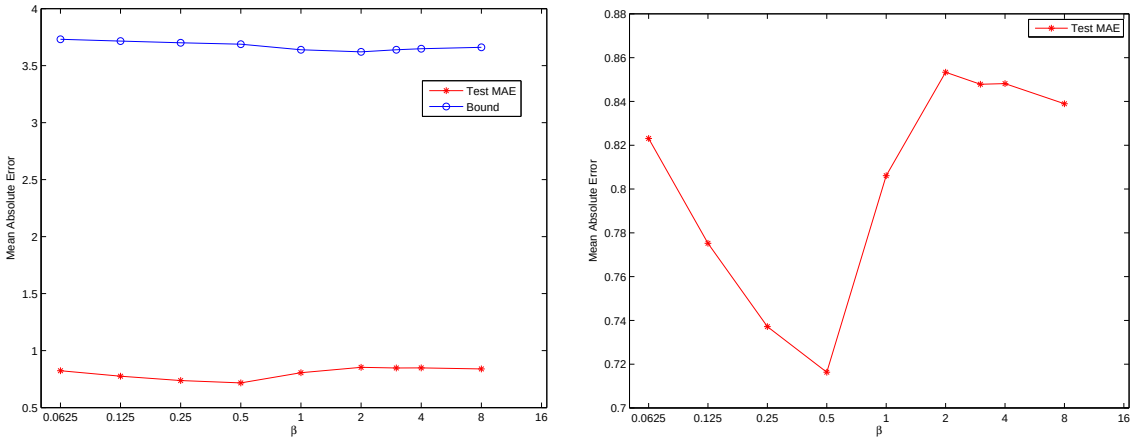



(a) Bound (19).

(b) Test Loss (zoom into subfigure a.).

Figure 4: **Co-clustering of the MovieLens dataset into 50x50 clusters.** Figure (a) shows the value of bound (19) together with the MAE on the test set as a function of β. Figure (b) zooms into MAE on the test set. The values of β are on a log scale. See text for further details.

- The performance of Algorithm 1 is comparable to the performance achieved in (Seldin et al., 2007) with sequential optimization.
- The optimal performance is achieved at β close to one, which corresponds to the MDL functional optimized in (Seldin et al., 2007).
- The values of the bound are meaningful (recall that the maximal possible loss is 4; thus the bound value of $\sim 1.25$ is informative).
- The bound is 25%-75% far from the test error.
- The bound does not follow the shape of the test loss. According to the bound in this task it is best to assign all the data to one big cluster. This is explained by the fact that this is a hard problem and the improvement in the empirical loss $\hat{L}(Q)$ achieved by co-clustering is relatively small. For the best co-clustering solution found $\hat{L}(Q) \approx 0.67$, whereas if we assign all the data to one big cluster $\hat{L}(Q) \approx 0.89$. Thus, the improvement in $\hat{L}(Q)$ achieved by the clustering is only about 30% while the tightness of the bound is 25%-75%. This is clearly insufficient to apply the bound as the main guideline for model order selection. However, it is possible to set the value of β in the trade-off $\mathcal{F}(Q,\beta)$ via cross-validation and obtain remarkably good results. It should be pointed out that the trade-off $\mathcal{F}(Q,\beta)$ was derived from the bound, thus even though the analysis is not perfectly tight it produced a useful practical result.
- Note that in the setting of this experiment the small values of $m_1$ and $m_2$ provide "natural regularization"; thus there is no significant decrease in performance when we increase β beyond 1. This will change in the following experiments.




(a) Bound (19). (b) Test Loss (zoom into subfigure a.).

Figure 5: **Co-clustering of the MovieLens dataset into 283x283 clusters.** Figure (a) shows the value of bound (19) together with the MAE on the test set as a function of β. Figure (b) zooms into MAE on the test set. The values of β are on a log scale. See text for further details.

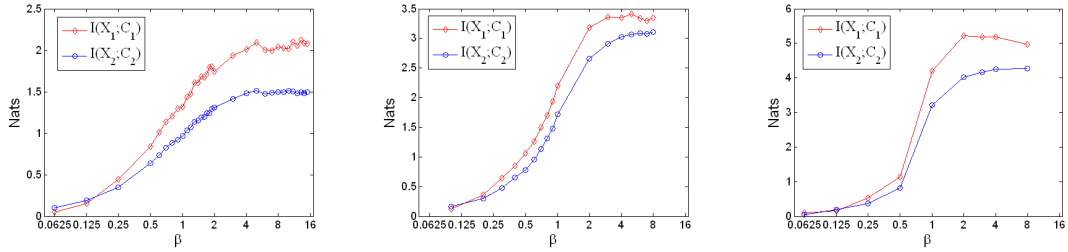
The power of bound (19) and the trade-off $\mathcal{F}(Q,\beta)$ derived from the bound is that it penalizes the effective complexity of the solution rather than the gross number of clusters used. The practical implication of this property is that we can initialize the optimization algorithm with more clusters than are actually required to solve a problem, and the algorithm will automatically adjust the extent to which it uses said available clusters. This property is verified in the following two experiments. In the first experiment we initialized Algorithm 1 with $m_1 = m_2 = 50$ clusters along each dimension. The result of optimization of $\mathcal{F}(Q,\beta)$ as a function of β is presented in Figure 4. We make the following observations based on this experiment:

- The best performance (the 0.72 test MAE) achieved in the previous setting with $m_1 = 13$ and $m_2 = 6$ is achieved in the new setting with $m_1 = m_2 = 50$ as well. This supports the ability of the algorithm to operate with more clusters than are actually required by the problem and to adjust the complexity of the solution automatically.

- Note that the optimal value of β in this setting is below 1. In particular, this implies that the MDL formulation, which corresponds to $\beta = 1$ would overfit in this case. The role of the regularization parameter β is also more evidently expressed here compared to the preceding experiment.

- The values of the bound, although less tight than in the previous case, are still meaningful. The shape of the bound becomes closer to the shape of the test loss, although in light of the preceding experiment we would not attribute importance to it, and would still prefer to set the value of β via cross-validation.

In our last experiment we went to the extreme case of taking $m_1 = m_2 = 283 = \sqrt{N}$. Note that the size of the cluster space $M = m_1 m_2$ in this case is $80,089$ and is equal to the size of the




(a) 13x6 clusters (b) 50x50 clusters (c) 283x283 clusters

Figure 6: **The values of $\overline{I}(X_i;C_i)$ in the minimized $\mathcal{F}(Q,\beta)$ corresponding to Figures 3, 4, 5.** We show the values of $\overline{I}(X_i;C_i)$ that were obtained after minimization of $\mathcal{F}(Q,\beta)$ by $Q$ when clustering the MovieLens dataset into 13x6, 50x50, and 283x283 clusters.

training set, $N = 80,000$. The implication is that extensive use of all available clusters can result in a situation where each partition cell contains an order of a single observation, which is clearly insufficient for statistically reliable predictions. Thus, in this experiment the number of clusters provides no regularization at all and the only parameter responsible for regularization of the model is the trade-off parameter $\beta$. The result of the experiment is presented in Figure 5. We highlight the following points regarding this experiment:

- The best performance (the 0.72 test MAE) is achieved in this experiment as well. This further stresses the ability to have full control over regularization of the model via parameter $\beta$ of the trade-off $\mathcal{F}(Q,\beta)$.
- The role of the regularization parameter $\beta$ is further increased in this experiment compared to the previous two. The optimal value of $\beta$ here is clearly below 1 (the optimal $\beta \approx 0.5$), suggesting that the MDL solution would be overfitting.
- The value of the bound still remains meaningful, although it is already quite far from the test error. The shape of the bound does not seem to provide useful information and the value of $\beta$ should be set via cross-validation.

In Figure 6 we show how the mutual information $\overline{I}(X_1;C_1)$ and $\overline{I}(X_2;C_2)$ changed in the three experiments as we optimized $\mathcal{F}(Q,\beta)$ by $Q$ for increasing values of $\beta$. An important observation to be made from these graphs (by relating them to Figures 3, 4, and 5) is that in all three experiments the best prediction performance was achieved at roughly the same values of the mutual information $\overline{I}(X_1;C_1)$ and $\overline{I}(X_2;C_2)$. For clustering into 13x6 clusters prediction performance of MAE equal to 0.72 and slightly lower was achieved at $\beta$ values starting from 0.7 and larger, when $\overline{I}(X_1;C_1)$ was in the range between 1.1 and 2.1 and $\overline{I}(X_2;C_2)$ was in the range between 0.8 and 1.5; for clustering into 50x50 clusters prediction performance of 0.72 and slightly lower was achieved for $\beta$ values in the range between 0.5 and 1.0, when $\overline{I}(X_1;C_1)$ was in the range between 1.1 and 2.2 and $\overline{I}(X_2;C_2)$ was in the range between 0.8 and 1.7; and for clustering into 283x283 clusters the optimal prediction performance of slightly below 0.72 was achieved for $\beta = 0.5$ and at this value of $\beta$ we had $\overline{I}(X_1;C_1) = 1.1$ and $\overline{I}(X_2;C_2) = 0.8$. We see that although the three experiments were initialized

with different numbers of clusters $m_1$ and $m_2$, the optimal prediction performance was achieved at roughly the same *effective complexity* of the solution (measured by $\overline{I}(X_i;C_i)$-s) and that the trade-off parameter $\beta$ took care of regularization of the model.

## 6. Probabilistic Matrix Tri-Factorization

For $d = 2$ the discriminative prediction and density estimation models considered in this paper can be seen as two forms of matrix tri-factorization and for higher dimensions $d > 2$ as Tucker decompositions. In this section we discuss this relation in more detail, starting from the discriminative prediction problem.

### 6.1 Probabilistic Matrix Tri-Factorization for Discriminative Prediction

For $d = 2$ equation (18) accepts the form:

$$q(y|x_1,x_2) = \sum_{c_1,c_2} q(c_1|x_1)q(y|c_1,c_2)q(c_2|x_2).$$

If $q(y|c_1,c_2)$ is restricted to be a delta distribution then it can be replaced by a function $f(c_1,c_2)$. This means that instead of drawing $y$ in the cluster product space according to a distribution $q(y|c_1,c_2)$ it is predicted deterministically by $f(c_1,c_2)$. Note that the assignment of $x_i$-s to $c_i$-s remains stochastic. The restriction of deterministic $q(y|c_1,c_2)$ does not limit the model significantly, since for many loss functions, such as zero-one, absolute, or quadratic losses the optimal prediction rule is deterministic in the cluster product space in any case. At the same time the bounds derived previously are still valid, since they are valid for any distribution $q(y|c_1,c_2)$ and in particular for the delta distribution. The restricted model accepts the form:

$$f(x_1,x_2) = \sum_{c_1,c_2} q(c_1|x_1)\, f(c_1,c_2) q(c_2|x_2),$$

which can be written as a matrix product:

$$A = Q_1^T F Q_2, \tag{46}$$

where

$$Q_i = [q(c_i|x_i)] \qquad (i = 1,2)$$

are $m_i \times n_i$ *left* stochastic matrices[5] mapping $x_i$-s to their clusters $c_i$-s, and

$$F = [f(c_1,c_2)]$$

is an $m_1 \times m_2$ matrix describing what happens in the cluster product space.

Given a data matrix $A$ (probably sparse) and a trade-off parameter $\beta$ Algorithm 1 provides a locally optimal approximation of $A$ in the form of (46) regularized by the mutual information preserved in $Q_1$ and $Q_2$. Note that Algorithm 1 naturally handles the missing entries in $A$. The product $Q_1^T F Q_2$ can then be used to complete the missing entries.

5. A *left* stochastic matrix is a matrix of non-negative real numbers with each column summing up to 1. In a *right* stochastic matrix each row sums up to 1.

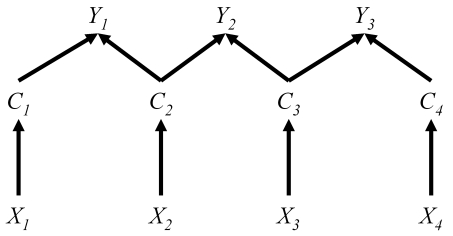



Figure 7: **A graphical model for simultaneous tri-factorization of multiple matrices (equations (47)-(49)).** The mapping of $X_i$ to $C_i$ is modeled by a corresponding matrix $Q_i$ and labels $Y_i$ correspond to prediction tasks in the respective matrices $A_i$.

Matrix factorization of the form (46) was already considered in (Banerjee et al., 2007) without regularization. However, in (Banerjee et al., 2007) the matrices $Q_1$ and $Q_2$ are restricted to deterministic assignments of $x_i$-s to $c_i$-s (the entries of $Q_1$ and $Q_2$ are in $\{0,1\}$), whereas in the factorization proposed here $Q_1$ and $Q_2$ are stochastic matrices and $F$ is arbitrary. Matrix tri-factorization considered in (Ding et al., 2006; Yoo and Choi, 2009a) is more closely related to matrix tri-factorization for density estimation discussed in the next subsection. We note that the recent Bayesian approaches to matrix factorization (Shan and Banerjee, 2008; Salakhutdinov and Mnih, 2008) are three-level stochastic models and unlike the two-level stochastic model in (46) cannot be written as a simple product of matrices. The following list of positive properties of probabilistic matrix tri-factorization suggested here further distinguishes it from other forms of matrix factorization, including singular value decomposition (SVD) (Strang, 2009; Golub and Loan, 1996), non-negative matrix factorization (Lee and Seung, 1999, 2001), low-rank matrix factorization (Srebro et al., 2005a), and maximum margin matrix factorization (Srebro et al., 2005b; Srebro, 2004), that satisfy only parts of the list:

- It has a clear probabilistic interpretation.
- It naturally handles missing values.
- The factorized matrix $A$ can be arbitrary (not necessarily positive or positive definite).
- Overfitting can be controlled via the regularization parameter $\beta$.
- The generalization bound derived for co-clustering applies to this form of matrix factorization.
- It is a two-level stochastic model of the data.
- The model can be optimized by iterative projections.
- The model achieves state-of-the-art results in prediction of missing matrix entries.

We leave a wider practical comparison of the different matrix factorization methods as a subject for future work.

A promising direction for future research suggested in (Seldin, 2009) and independently in (Yoo and Choi, 2009b) is to apply matrix tri-factorization in tasks, where multiple related datasets are

considered. For example, let $A_1$ be a matrix of viewers-by-viewers properties, $A_2$ be a collaborative filtering matrix, and $A_3$ be a matrix of movies-by-movies properties. We can look for simultaneous tri-factorizations, such that:

$$A_1 \approx Q_1^T F_1 Q_2 \tag{47}$$
$$A_2 \approx Q_2^T F_2 Q_3 \tag{48}$$
$$A_3 \approx Q_3^T F_3 Q_4. \tag{49}$$

In other words, the clustering of viewers into clusters of viewers is shared between factorizations of $A_1$ and $A_2$ and the clustering of movies into clusters of movies is shared between factorizations of $A_2$ and $A_3$. An unregularized form of simultaneous tri-factorization for collaborative filtering was already explored in (Yoo and Choi, 2009b). Problems of a similar form are also frequent in bioinformatics, when multiple experiments with partial relations are considered. For example, Alter et al. (2003) applied generalized SVD (GSVD) to compare yeast and human cell-cycle gene expression datasets. In their experiment it is natural to create separate systems of clusters for yeast and human genes, but a common system of clusters for the cell-cycle time points. As already pointed out, probabilistic matrix tri-factorization suggested here has several advantages over SVD (and consequently over GSVD). Hence, it would be interesting to apply it to this type of problem.

### 6.2 Probabilistic Matrix Tri-Factorization for Density Estimation

The model for discrete density estimation based on co-clustering in equation (27) can also be written as matrix tri-factorization (for $d = 2$):

$$A = R_1^T G R_2, \tag{50}$$

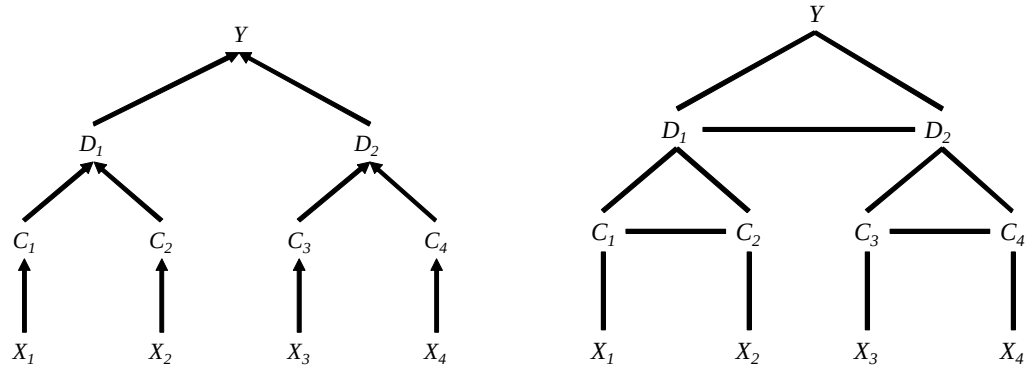
where

$$R_i = [q(x_i|c_i)] = \left[\frac{q(x_i)}{q(c_i)} q(c_i|x_i)\right] \qquad (i = 1,2)$$

are $m_i \times n_i$ *right* stochastic matrices of probabilities of generating $x_i$-s given $c_i$-s and

$$G = [q(c_1, c_2)]$$

is an $m_1 \times m_2$ matrix of joint probability distribution of $c_1$ and $c_2$. As already mentioned, this model is appropriate for co-occurrence data analysis, such as word-document co-occurrence matrices. An unregularized form of such decomposition was already considered in (Ding et al., 2006; Yoo and Choi, 2009a). Here, $A$ is assumed to be a joint probability matrix (i.e., the entries of $A$ are non-negative and sum up to 1). In practice $A$ is an empirical joint probability distribution matrix and factorization (50) regularized by the mutual informations $I(X_i;C_i)$ can be used to regularize the estimation of the joint probability distribution. Algorithm 2 can be used to find a local optimum of such factorization given the regularization parameter $\beta$. The generalization bound developed in Theorem 12 holds for this factorization. We remind the reader that Algorithm 2 operates with deterministic assignments of $x_i$-s to clusters $c_i$-s. Although the resulting reverse conditional distribution $q(x_i|c_i)$ is not deterministic, the algorithm does not explore all possible solutions to this problem. Developing an algorithm for finding a local optimum of the regularized decomposition with mixed memberships of $x_i$-s is a challenging direction for future research.




(a) Discriminative Prediction Model (b) Moralization of (a)

Figure 8: **Illustration of (a) a graphical model for discriminative prediction and (b) its moralized counterpart.** The illustration is for $d = 4$.

## 7. PAC-Bayesian Analysis of Graphical Models

The analysis of co-clustering presented in Section 3 holds for any dimension $d$. However, the dependence of the bounds (19), (32), and (37) on $d$ is exponential because of the $M = \prod_i m_i$ term that they involve. This term is reasonably small when the number of dimensions is small (two or three), as in the example of co-clustering. However, as the number of dimensions grows, this term grows exponentially. Thus, high dimensional tasks require a different treatment. Some improvements are also possible if we consider discriminative prediction based on a single parameter $X$ (i.e., in the case of $d = 1$), but the one-dimensional case is beyond the scope of this paper and we refer the interested reader to (Seldin and Tishby, 2008; Seldin, 2009) for further details. In this section we suggest a hierarchical approach to handle high-dimensional problems. We then show that this approach can also be applied to generalization analysis of graphical models.

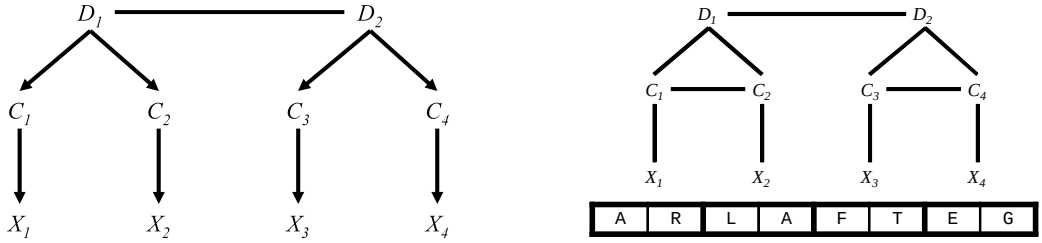
### 7.1 Hierarchical Approach to High Dimensional Problems ($d > 2$)

One possible way to handle high dimensional problems is to use hierarchical partitions, as shown in Figures 8 and 9. For example, the discriminative prediction rule corresponding to the model in Figure 8.a is:

$$q(y|x_1,..,x_4) = \sum_{d_1,d_2} q(y|d_1,d_2) \sum_{c_1,..,c_4} \prod_{i=1}^{2} q(d_i|c_{2i-1},c_{2i}) \prod_{j=1}^{4} q(c_j|x_j). \tag{51}$$

And the corresponding randomized prediction strategy is
$Q = \left\{ \{q(c_i|x_i)\}_{i=1}^{4}, \{q(d_i|c_{2i-1},c_{2i})\}_{i=1}^{2}, q(y|d_1,d_2) \right\}$. In this case the hypothesis space is the space of all hard partitions of $x_i$-s to $c_i$-s and of the pairs $\langle c_{2i-1}, c_{2i} \rangle$ to $d_i$-s. By repeating the analysis in Theorem 7 we obtain that with probability greater than $1-\delta$:

$$kl(\hat{L}(Q)\|L(Q)) \leq \frac{B_1 + B_2 + |D_1||D_2| \ln|\mathcal{Y}| + \frac{1}{2}\ln(4N) - \ln\delta}{N}, \tag{52}$$




(a) Density Estimation Model (b) Moralization and Application of (a)

Figure 9: **Illustration of (a) a directed model for density estimation and (b) its moralization and application to sequence modeling.** The sequence in subfigure (b) is an imaginary subsequence of length 8 of a protein sequence. Each $X_i$ corresponds to a pair of amino acids in the subsequence.

where

$$B_1 = \sum_{i=1}^{4} \left( n_i \bar{I}(X_i; C_i) + m_i \ln n_i \right),$$

$$B_2 = \sum_{i=1}^{2} \left( (m_{2i-1} m_{2i}) \bar{I}(D_i; C_{2i-1}, C_{2i}) + |D_i| \ln(m_{2i-1} m_{2i}) \right).$$

Observe that the $M \ln |\mathcal{Y}|$ term in (19), which corresponds to the clique $\langle C_1, C_2, C_3, C_4, Y \rangle$, is replaced in (52) with terms which correspond to much smaller cliques $\langle C_1, C_2, D_1 \rangle$, $\langle C_3, C_4, D_2 \rangle$, and $\langle D_1, D_2, Y \rangle$. This factorization makes it possible to control the complexity of the partition and the tightness of the bound. In a similar way it is possible to derive factorized analogs to bounds (32) and (37) that apply to density estimation hierarchies as in Figure 9.

We provide an illustration of a possible application of the models in Figure 9. Imagine that we intend to analyze protein sequences. Protein sequences are sequences over the alphabet of 20 amino acids. Subsequences of length 8 can reach $20^8 = 256 \cdot 10^8$ instances. Instead of studying this space directly, which would require an order of $256 \cdot 10^8$ samples, we can associate each $X_i$ with a pair of amino acids - see Figure 9. The subspace of pairs of amino acids is only $20^2 = 400$ instances and local interactions between adjacent pairs of amino acids can easily be studied. We can cluster the pairs of amino acids into, say, 20 clusters $C$. Interactions between adjacent pairs of $C$-s in such a construction correspond to interactions between quadruples of amino acids. The subspace of quadruples is $20^4 = 16 \cdot 10^4$ instances. However, the reduced subspace of pairs of $C_i$-s is only $20^2 = 400$ instances. Thus, we have doubled the range of interactions, but remained at the same level of complexity. We can further cluster pairs of $C_i$-s (which correspond to quadruples of amino acids) into $D_i$-s and study the space of 8-tuples of amino acids while remaining at the same level of complexity.

The above approach shares the same basic principle already discussed in the collaborative filtering task: by clustering together similar pairs (and then quadruples) of amino acids we increase the statistical reliability of the observations, but reduce the resolution at which we process the data. Bound (52) suggests how the trade-off between model resolution and statistical reliability can be optimized.

### 7.2 PAC-Bayesian Analysis of Graphical Models

The result in the previous subsection suggests a new approach to learning graphical models by providing a way to evaluate the expected performance of a graphical model on new data. Thus, instead of constructing a graphical model that fits the observed data it serves to construct a model with good generalization properties. The analysis used to derive bound (52) can be applied to any directed graphical model in the form of a tree (as in Figures 8.a, 9.a) or its moralized counterpart (as in Figures 8.b, 9.b). The analysis shows that the generalization power of these graphical models is determined by a trade-off between empirical performance and the amount of mutual information that is propagated up the tree. It is important to note that the PAC-Bayesian bound is able to take advantage of the factor form of distribution (51) and that bound (52) depends on the sizes of the tree cliques, but not on the size of the parameter space $\mathcal{X}_1 \times .. \times \mathcal{X}_d$. Further, a prior can be added over all possible directed graphs under consideration to obtain a PAC-Bayesian bound that will hold for all of them simultaneously. Development of efficient algorithms for optimization of the tree structure and extension of the results to more general graphical models are key directions for future research.

## 8. PAC-Bayesian Analysis of Graph Clustering and Pairwise Clustering

In this section we show that our approach to predictive formulation of unsupervised learning problems and their subsequent PAC-Bayesian analysis can also be applied to weighted graph clustering (and, consequently, to pairwise clustering, which can be regarded as clustering of a graph with edge weights corresponding to pairwise distances). Graph clustering is an important tool in data analysis with a wide variety of applications including social networks analysis, bioinformatics, image processing, and many more. As a result, a multitude of different approaches to graph clustering were developed. Examples include graph cut methods (Shi and Malik, 2000), spectral clustering (Ng et al., 2001), information-theoretic approaches (Slonim et al., 2005), to name just a few. Comparing the different approaches is usually a painful task, mainly because the goal of each of these clustering methods is formulated in terms of the solution: most clustering methods start by defining some objective functional and then minimize it. But, for a given problem, how can we choose whether to apply a graph cut method, spectral clustering, or an information-theoretic approach?

Here we formulate weighted graph clustering as a prediction problem[6]. Given a subset of edge weights we analyze the ability of graph clustering to predict the remaining edge weights. The rationale behind this formulation is that if a model (not necessarily cluster-based) is able to predict with high precision all edge weights of a graph given a small subset of edge weights then it is a good model of the graph. The advantage of this formulation of graph modeling is that it is independent of the specific way chosen to model the graph and can be used to compare any two solutions, either by comparison of generalization bounds or by cross-validation. The generalization bounds or cross-validation also address the finite-sample nature of the graph clustering problem and provide a clear criterion for model order selection. For very large datasets, where computational constraints can prevent considering all edges of a graph, as for example in (Yom-Tov and Slonim, 2009), the generalization bounds can be used to resolve the trade-off between computational workload and precision of graph modeling.

Below we provide a PAC-Bayesian analysis of graph clustering in the case, where independent sampling of edge weights is possible. For example, in the analysis of flow graphs, such as loads

6. Unweighted graphs can be modeled by setting the weight of present edges as 1 and absent edges as 0.

on links in traffic or communication networks, we can repeatedly sample one or more non-adjacent edge weights at a time (non-adjacent edges have no common vertices). In the analysis of snapshot graphs, for example image segmentation, the dependencies between adjacent edge weight samples should be taken into account, but this is again beyond the scope of this paper.

### 8.1 PAC-Bayesian Analysis of Graph Clustering

Assume that $\mathcal{X}$ is a space of $|\mathcal{X}|$ nodes and denote by $w_{x_1x_2}$ the weight of an edge connecting nodes $x_1$ and $x_2$.[7] We assume that the weights $w_{x_1x_2}$ are generated according to an unknown probability distribution $p(w|x_1,x_2)$. We further assume that the space of nodes $\mathcal{X}$ is known and we are given a sample of size $N$ of edge weights, generated according to $p(x_1,x_2,w)$. The goal is to build a regression function $q(w|x_1,x_2)$ that will minimize the expected prediction error of the edge weights $\mathbb{E}_{p(x_1,x_2,w)}\mathbb{E}_{q(w'|x_1,x_2)}l(W,W')$ for some externally given loss function $l(w,w')$. Note that this formulation does not assume any specific form of $q(w|x_1,x_2)$ and enables comparison of all possible approaches to this problem.

Here we analyze generalization abilities of $q(w|x_1,x_2)$ based on clustering:

$$q(w|x_1,x_2)=\sum_{c_1,c_2}q(w|c_1,c_2)q(c_1|x_1)q(c_2|x_2). \tag{53}$$

One can immediately see the relation between (53) and the co-clustering model for discriminative prediction (18). The only difference is that in (53) the nodes $x_1,x_2$ belong to the same space of nodes $\mathcal{X}$ and the conditional distribution $q(c|x)$ is shared for the mapping of endpoints of an edge. Let $\hat{p}(x_1,x_2,w)$ be the empirical distribution over edge weights. The empirical loss of prediction strategy $Q=\{q(c|x),q(w|c_1,c_2)\}$ corresponding to (53) can then be written as:

$$\hat{L}(Q)=\mathbb{E}_{\hat{p}(x_1,x_2,w)}\mathbb{E}_{q(w'|x_1,x_2)}l(W,W').$$

The following generalization bound for graph clustering can be proved by a minor adaptation of the proof of Theorem 7.

**Theorem 13** *For any probability measure $p(x_1,x_2,w)$ over the space of nodes and edge weights $\mathcal{X}\times\mathcal{X}\times\mathcal{W}$ and for any loss function $l$ bounded by 1, with probability of at least $1-\delta$ over a selection of an i.i.d. sample $S$ of size $N$ according to $p$, for all graph clustering models defined by $Q=\{q(c|x),q(w|c_1,c_2)\}$:*

$$kl(\hat{L}(Q)\|L(Q))\le\frac{n\overline{I}(X;C)+m\ln n+m^2\ln|\mathcal{W}|+\frac{1}{2}\ln(4N)-\ln\delta}{N}, \tag{54}$$

*where $m=|C|$, $n=|\mathcal{X}|$, and $|\mathcal{W}|$ is the number of distinct edge weights.*

Continuous edge weights can be handled by post-process quantization, as shown in appendix B.

As in the case of co-clustering, in practice we can replace (54) with a trade-off:

$$\mathcal{J}(Q,\beta)=\beta N\hat{L}(Q)+n\overline{I}(X;C) \tag{55}$$

and tune $\beta$ either by substituting $\hat{L}(Q)$ and $\overline{I}(X;C)$ resulting from the solution of (55) back into the bound or via cross-validation.

7. All the results can be straightforwardly extended to hyper-graphs.

If the distribution $q(w|c_1,c_2)$ is restricted to be a delta distribution equation (53) can be rewritten as:

$$f(x_1,x_2) = \sum_{c_1,c_2} q(c_1|x_1) f(c_1,c_2) q(c_2|x_2)$$

and the corresponding matrix form is:

$$A \approx Q^T F Q,$$

where $Q = [q(c|x)]$, $F = [f(c_1,c_2)]$, and $A$ is an input matrix (probably sparse) providing a sample of graph edge weights. This form of symmetric matrix tri-factorization is briefly mentioned in (Ding et al., 2006). Let $\mathbb{I}$ be an indicator matrix for the entries present in $A$. For quadratic loss the empirical approximation error $\hat{L}(Q)$ can be written as:

$$\hat{L}(Q) = \frac{1}{N} \|\mathbb{I} \circ (A - Q^T F Q)\|_2^2, \tag{56}$$

where $\circ$ denotes entrywise product of two matrices. From (56) it is easy to see that $\hat{L}(Q)$ is not convex in $Q$ (unlike in the case of co-clustering, where the dependence on $Q_i$-s was quadratic, here the power of $Q$ is 4). Hence, although alternating projections similar to those in Algorithm 1 can be applied to $\mathcal{J}(Q,\beta)$ they are not guaranteed to converge to a local minimum. Nevertheless, according to some preliminary experiments they can achieve reasonably good solutions and bound (54) provides reasonably tight guarantees on the expected approximation quality (Seldin, 2010). Alternatively, trade-off $\mathcal{J}(Q,\beta)$ can be optimized by sequential minimization. Unlike alternating projections, sequential minimization (similar to the one in Algorithm 2) is guaranteed to converge to a local minimum, but can operate with deterministic assignments of graph nodes $x_i$-s to the clusters $c_i$-s only. A convergent algorithm that will be able to explore stochastic assignments still awaits to be developed.

### 8.2 Related Work on Pairwise Clustering

The regularization of pairwise clustering by mutual information $\bar{I}(X;C)$ was already applied in practice by Slonim et al. (2005). They maximized a parameterized trade-off $\beta\langle s\rangle - \bar{I}(X;C)$, where $\langle s\rangle = \sum_c \bar{q}(c) \sum_{x_1,x_2} q(x_1|c) q(x_2|c) w_{x_1x_2}$ measured average pairwise similarities within a cluster[8]. Their algorithm demonstrated superior results in cluster coherence compared to 18 other clustering methods. The regularization by mutual information was motivated by information-theoretic considerations inspired by the rate distortion theory (Cover and Thomas, 1991). Namely, the authors drew a parallel between $\langle s\rangle$ and distortion and $\bar{I}(X;C)$ and compression rate of a clustering algorithm. Further, Yom-Tov and Slonim (2009) showed that the algorithm can be run in parallel mode, where each parallel worker operates with a subset of pairwise relations at each iteration rather than all of them. Such a mode of operation was motivated by inability to consider all pairwise relations in very large datasets due to computational constraints. Yom-Tov and Slonim (2009) reported only minor empirical degradation in clustering quality, but no formal analysis or guarantees were suggested. The results presented here can help to analyze such problems and help to address the trade-off between computational workload and approximation quality in analysis of very large graphs.

8. The loss $L(Q)$ is slightly more general than $\langle s\rangle$ since it also considers edges between the clusters. Although, this generality breaks the convexity of $\hat{L}(Q)$ in (56).

## 9. Related Work on Clustering

The idea of considering clustering in the context of a higher level task was inspired by the Information Bottleneck (IB) principle (Tishby et al., 1999; Slonim, 2002; Slonim et al., 2006). The IB principle considers the problem of extracting information from a random variable $X$ that is relevant for prediction of a random variable $Y$. The relevance variable $Y$ defines the high-level task. For example, $X$ might be a speech signal and the task might be identification of the speaker or transcription of the signal. The extraction of relevant information from $X$ is done by means of clustering of $\mathcal{X}$ into clusters $\tilde{\mathcal{X}}$ that preserve the information on $\mathcal{Y}$ (Tishby et al., 1999). Clearly, each relevance variable $\mathcal{Y}$ corresponds to a different partition (clustering) of $\mathcal{X}$. The IB principle was further extended to graphical models in (Slonim et al., 2006).

The idea to consider clustering as a proxy to solution of a prediction task was further developed in (Krupka and Tishby, 2005, 2008; Krupka, 2008). Krupka and Tishby analyze a scenario wherein each object has multiple properties, but only a fraction of the properties is observed. Consider the following illustration: assume we are presented with multiple fruits and we observe their parameters, such as size, color, and weight. We can cluster the fruits by their observed parameters in order to facilitate prediction of unobserved parameters, such as taste and toxicity. This approach enables one to conduct a formal analysis and derive generalization bounds for prediction rules based on clustering.

In recent years extensive attempts have been made to address the question of model order selection in clustering through evaluation of its stability (Lange et al., 2004; von Luxburg and Ben-David, 2005; Ben-David et al., 2006; Shamir and Tishby, 2009; Ben-David and von Luxburg, 2008). This perspective suggests that for two random samples generated by the same source, clustering of the samples should be similar (and hence stable). Otherwise the obtained clustering is unreliable. Although it has been proven that in a large sample regime stability can be used for model order selection (Shamir and Tishby, 2009), no upper bounds on the minimal sample size required for stability estimates to hold can be proved. Moreover, in certain cases stability indices based on arbitrarily large samples can be misleading (Ben-David and von Luxburg, 2008). Since in any practical application the amount of data available is limited, currently existing stability indices cannot be used for reliable model order selection and it is not clear whether the stability indices can be used to compare solutions based on different optimization objectives.

**Gaussian ring example.** We use the following example from (Seldin, 2009) to illustrate that generalization and stability criteria for evaluation of clustering are not equivalent. Assume points in $\mathbb{R}^2$ are generated according to the following process. First, we select a center $\mu$ of a Gaussian according to a uniform distribution on a unit circle in $\mathbb{R}^2$. Then we generate a point $x \sim \mathcal{N}(\mu, \sigma^2 I)$ according to a Gaussian distribution centered at $\mu$ with a covariance matrix $\sigma^2 I$ for a fixed $\sigma$ ($I$ is a 2 by 2 identity matrix). Given a sample generated according to the above process we can apply a mixture of Gaussians clustering in order to learn the generating distribution. Note that:

1. Due to the circular symmetry in the generating process and model redundancy, the solution will always be unstable (the centers of Gaussians in the mixture of Gaussians model can move arbitrarily along the unit circle and their variance in the direction tangential to the circle is loosely constrained).

2. By increasing the sample size and the number of Gaussians in the mixture of Gaussians model the true density of the points can be approximated arbitrarily well.

Hence, models with good generalization properties are not necessarily stable. This point should be kept in mind when using generalization as an evaluation criterion in clustering.

## 10. Discussion and Future Work

This paper underlines the importance of external evaluation of unsupervised learning, such as clustering or more general structure learning, based on the context of its potential application. Such a form of evaluation is important for delivery of better structure learning algorithms as well as for better understanding of their outcome. We argue that structure learning does not occur for its own sake, but rather in order to facilitate a solution of some higher level goal. In any non-trivial data many structures co-exist simultaneously and it is a matter of the subsequent usage of the outcome of the learning algorithm to determine which of the structure elements are valuable and which are not. Therefore, unsupervised learning cannot be analyzed in isolation from its potential application. In our opinion, one of the main obstacles in theoretical advancement of unsupervised learning is an absence of a good mathematical definition of the context of its application. The analysis of co-clustering presented here is a first step toward context-based analysis of more complex models.

The work presented here started with an attempt to improve our understanding of clustering. We note that clustering is tightly related to the object naming process in human language. In a sense, a cluster is an entity that can be assigned a name. By clustering objects we ignore their irrelevant properties and concentrate on the relevant ones. And of course, this division can change according to our needs. For example, we can divide animals into birds and mammals or into flying and notatorial or into domestic and wild. Whereas the classification into birds and mammals or flying and notatorial may be considered intrinsic, the classification into domestic and wild is definitely application-oriented. In order to design successful clustering and categorization algorithms it is important to understand the basic principles behind this process. It is not a-priori clear that, if we restrict ourselves to pure prediction tasks, clustering the underlying sample space helps. As shown in (Seldin and Tishby, 2008; Seldin, 2009), in classification by a single parameter there is no need to cluster the parameter space, but rather simple smoothing performs better. In classification in higher dimensional spaces, kernel-based methods can be superior to clustering-based approaches. However, we know that as humans we communicate by using a clustered representation of the world rather than by kernel matrices. Thus, there should be advantages for such form of communication. Identification, understanding, and analysis of these advantages is an important future direction for the design of better clustering and higher structure learning algorithms.

## Acknowledgements

We are very grateful to the anonymous reviewers for their valuable comments that helped us to improve this manuscript significantly.

## Appendix A. Proof of Theorem 6

**Proof of Theorem 6** First we prove inequality (15):

$$\begin{aligned}-\mathbb{E}_{Q(h)}\mathbb{E}_{p_h(z)}\ln\tilde{p}_h(Z) &= \mathbb{E}_{Q(h)}\mathbb{E}_{[\hat{p}_h(z)-p_h(z)]}\ln\tilde{p}_h(Z) - \mathbb{E}_{Q(h)}\mathbb{E}_{\hat{p}_h(z)}\ln\tilde{p}_h(Z)\\ &= \mathbb{E}_{Q(h)}\mathbb{E}_{[\hat{p}_h(z)-p_h(z)]}\ln\frac{\hat{p}_h(Z)+\gamma}{1+\gamma|\mathcal{Z}|} - \mathbb{E}_{Q(h)}\mathbb{E}_{\hat{p}_h(z)}\ln\frac{\hat{p}_h(Z)+\gamma}{1+\gamma|\mathcal{Z}|}\end{aligned}$$

$$\leq -\frac{1}{2}\|\hat{p}_Q(z) - p_Q(z)\|_1 \ln \frac{\gamma}{1+\gamma|\mathcal{Z}|} + \mathbb{E}_{Q(h)} H(\hat{p}_h(z)) + \ln(1+\gamma|\mathcal{Z}|)$$
$$\leq H(\hat{p}_Q(z)) - \sqrt{\varepsilon/2} \ln \frac{\gamma}{1+\gamma|\mathcal{Z}|} + \ln(1+\gamma|\mathcal{Z}|). \quad (57)$$

The last inequality is justified by the concavity of the entropy function $H$ and the KL-divergence bound on the $L_1$ norm (Cover and Thomas, 1991):

$$\|\hat{p}_Q(z) - p_Q(z)\|_1 \leq \sqrt{2KL(\hat{p}_Q(z)\|p_Q(z))} \leq \sqrt{2\varepsilon}.$$

By differentiation (57) is minimized by $\gamma = \frac{\sqrt{\varepsilon/2}}{|\mathcal{Z}|}$. By substitution of this value of $\gamma$ into (57) we obtain (15). Inequality (16) is justified by (15) and the concavity of the ln function. Finally, we prove the lower bound (17):

$$\begin{aligned} -\mathbb{E}_{p_Q(z)} \ln \tilde{p}_Q(Z) &= \mathbb{E}_{[\hat{p}_Q(z) - p_Q(z)]} \ln \tilde{p}_Q(Z) - \mathbb{E}_{\hat{p}_Q(z)} \ln \tilde{p}_Q(Z) \\ &\geq -\frac{1}{2}\|\hat{p}_Q(z) - p_Q(z)\|_1 \ln \frac{1+\gamma|\mathcal{Z}|}{\gamma} + H(\hat{p}_Q(z)) \\ &\geq H(\hat{p}_Q(z)) - \sqrt{\varepsilon/2} \ln \frac{|\mathcal{Z}|(1+\sqrt{\varepsilon/2})}{\sqrt{\varepsilon/2}}. \end{aligned}$$

■

## Appendix B. Treatment of Continuous Label Spaces $\mathcal{Y}$ via Quantization

In Theorems 7 and 13 it was assumed that the label space $\mathcal{Y}$ (or the edge weight space $\mathcal{W}$) is finite. However, for quadratic loss the minimization of $\mathcal{F}(Q,\beta)$ by Algorithm 1 can return a solution that falls out of this finite space. Furthermore, the input space $\mathcal{Y}$ itself does not have to be finite (e.g., gene expression levels in bioinformatics can be given on a continuous scale). Here we show that the bound can be easily generalized to handle this case via quantization of $\mathcal{Y}$. The analysis can be seen as post-processing and does not require modifications of the trade-off $\mathcal{F}(Q,\beta)$ and of Algorithm 1, since the algorithm does not assume finiteness of $\mathcal{Y}$.

Assume that $\mathcal{Y}$ is limited in [0,1] interval and apply uniform quantization of $\mathcal{Y}$ at intervals $\Delta$, then $|\mathcal{Y}_\Delta| = \frac{1}{\Delta}$ ($\mathcal{Y}_\Delta$ is the quantized copy of $\mathcal{Y}$ and we assume that the quantization starts at $\frac{1}{2}\Delta$ and ends at $1-\frac{1}{2}\Delta$). By rounding the continuous values of $y$ obtained by Algorithm 1 toward the closest quantization both the empirical and the expected loss are increased by at most $2\Delta+\Delta^2$ (in the case of quadratic loss). This is because quantization can shift the true $y$ and the prediction $y'$ by at most $\frac{1}{2}\Delta$ and then $l(y-\frac{1}{2}\Delta, y'+\frac{1}{2}\Delta) = (y-y'-\Delta)^2 = (y-y')^2 - 2(y-y')\Delta + \Delta^2 \leq l(y,y') + 2\Delta + \Delta^2$, where the last inequality follows from the assumption that $\mathcal{Y}$ is limited in [0,1]. Hence, we have

$$L(Q) \leq kl^{-1}\left(\hat{L}(Q) + 2\Delta + \Delta^2, \frac{\sum_{i=1}^d n_i \bar{I}(X_i;C_i) + K}{N}\right) + 2\Delta + \Delta^2,$$

where $K$, defined previously in (39), becomes:

$$K = \sum_{i=1}^d m_i \ln n_i - M \ln \Delta + \frac{1}{2}\ln(4N) - \ln\delta.$$

As a rule of thumb one can choose $\Delta = kM/N$ for $k \approx 5$, so that the contribution of $\Delta$ to the two operands of the inverse KL-divergence is approximately equivalent. In general this correction for quantization has no significant influence on the bound (Seldin, 2010).

## References

Orly Alter, Patrick O. Brown, and David Botstein. Generalized singular value decomposition for comparative analysis of genome-scale expression data sets of two different organisms. In *Proceedings of the National Academy of Science*, 2003.

Amiran Ambroladze, Emilio Parrado-Hernández, and John Shawe-Taylor. Tighter PAC-Bayes bounds. In *Advances in Neural Information Processing Systems (NIPS)*, 2007.

S. Arimoto. An algorithm for computing the capacity of discrete memoryless channel. *IEEE Transactions on Information Theory*, 18, 1972.

Jean-Yves Audibert and Olivier Bousquet. Combining PAC-Bayesian and generic chaining bounds. *Journal of Machine Learning Research*, 2007.

Arindam Banerjee. On Bayesian bounds. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2006.

Arindam Banerjee, Inderjit Dhillon, Joydeep Ghosh, Srujana Merugu, and Dhamendra Modha. A generalized maximum entropy approach to Bregman co-clustering and matrix approximation. *Journal of Machine Learning Research*, 8, 2007.

Peter Bartlett and Shahar Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 2001.

Peter Bartlett, Stéphane Boucheron, and Gábor Lugosi. Model selection and error estimation. *Machine Learning*, 2001.

Peter Bartlett, Michael Collins, Ben Taskar, and David McAllester. Exponentiated gradient algorithms for large-margin structured classification. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

Shai Ben-David and Ulrike von Luxburg. Relating clustering stability to properties of cluster boundaries. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 2008.

Shai Ben-David, Ulrike von Luxburg, and David Pál. A sober look on clustering stability. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

R. Blahut. Computation of channel capacity and rate distortion functions. *IEEE Transactions on Information Theory*, 18:460–473, 1972.

Gilles Blanchard and François Fleuret. Occam's hammer. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 2007.

David Blei and John Lafferty. Topic models. In A. Srivastava and M. Sahami, editors, *Text Mining: Theory and Applications*. Taylor and Francis, 2009.

Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Theory of classification: a survey of recent advances. *ESAIM: Probability and Statistics*, 2005.

Olivier Catoni. PAC-Bayesian supervised classification: The thermodynamics of statistical learning. *IMS Lecture Notes Monograph Series*, 56, 2007.

Yizong Cheng and George M. Church. Biclustering of expression data. In *Proceedings of the $8^{th}$ International Conference on Intelligent Systems for Molecular Biology (ISMB)*, 2000.

Hyuk Cho and Inderjit S. Dhillon. Co-clustering of human cancer microarrays using minimum sum-squared residue co-clustering. *IEEE/ACM Trans. on Computational Biology and Bioinformatics (TCBB)*, 5:3, 2008.

Hyuk Cho, Inderjit S. Dhillon, Yuqiang Guan, and Suvrit Sra. Minimum sum-squared residue co-clustering of gene expression data. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004.

Thomas M. Cover. Admissibility properties of Gilberts encoding for unknown source probabilities. *IEEE Transactions on Information Theory*, 18, 1972.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 1991.

Koby Crammer, Mehryar Mohri, and Fernando Pereira. Gaussian margin machines. In *Proceedings on the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009.

Philip Derbeko, Ran El-Yaniv, and Ron Meir. Explicit learning curves for transduction and application to clustering and compression algorithms. *Journal of Artificial Intelligence Research*, 22, 2004.

Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, 1996.

Inderjit S. Dhillon, Subramanyam Mallela, and Dharmendra S. Modha. Information-theoretic co-clustering. In *ACM SIGKDD*, 2003.

Chris Ding, Tao Li, Wei Peng, and Haesun Park. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining (ACM SIGKDD)*, 2006.

Miroslav Dudík, Steven J. Phillips, and Robert E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, 8, 2007.

Ran El-Yaniv and Oren Souroujon. Iterative double clustering for unsupervised and semi-supervised learning. In T.G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2001.

Dayane Freitag. Trained named entity recognition using distributional clusters. In *Proceedings of EMNLP*, 2004.

Thomas George and Srujana Merugu. A scalable collaborative filtering framework based on co-clustering. In *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM05)*, 2005.

Pascal Germain, Alexandre Lacasse, François Laviolette, and Mario Marchand. PAC-Bayesian learning of linear classifiers. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

Edgar N. Gilbert. Codes based on inaccurate source probabilities. *IEEE Transactions on Information Theory*, 17(3), 1971.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, $3^{rd}$ edition, 1996.

Peter Grünwald. *The Minimum Description Length Principle*. MIT Press, 2007.

Isabelle Guyon, Ulrike von Luxburg, and Robert C. Williamson. Clustering: Science or art? NIPS Workshop on Clustering Theory, 2009.

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

Jonathan Herlocker, Joseph Konstan, Loren Terveen, and John Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 2004.

Matthew Higgs and John Shawe-Taylor. A PAC-Bayes bound for tailored density estimation. In *Proceedings of the International Conference on Algorithmic Learning Theory (ALT)*, 2010.

Edwin Thompson Jaynes. Information theory and statistical mechanics. *Physical Review*, 106, 1957.

Michael Kearns, Yishay Mansour, Andrew Ng, and Dana Ron. An experimental and theoretical comparison of model selection methods. *Machine Learning*, 1997.

Yong-Deok Kim and Seungjin Choi. Nonnegative Tucker decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

Yuval Kluger, Ronen Basri, Joseph T. Chang, and Mark Gerstein. Spectral biclustering of microarray data: Coclustering genes and conditions. *Genome Research*, 2003.

Vladimir Koltchinskii. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 2001.

Rafail E. Krichevskiy. Laplaces law of succession and universal encoding. *IEEE Transactions on Information Theory*, 44(1), 1998.

Eyal Krupka. *Generalization from Observed to Unobserved Features*. PhD thesis, The Hebrew University of Jerusalem, 2008.

Eyal Krupka and Naftali Tishby. Generalization in clustering with unobserved features. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

Eyal Krupka and Naftali Tishby. Generalization from observed to unobserved features by clustering. *Journal of Machine Learning Research*, 9, 2008.

Tilman Lange, Volker Roth, Mikio L. Braun, and Joachim M. Buhmann. Stability based validation of clustering solutions. *Neural Computation*, 2004.

John Langford. Tutorial on practical prediction theory for classification. *Journal of Machine Learning Research*, 2005.

John Langford and John Shawe-Taylor. PAC-Bayes & margins. In *Advances in Neural Information Processing Systems (NIPS)*, 2002.

Danial Lashkari and Polina Golland. Co-clustering with generative models. Technical report, MIT-CSAIL-TR-2009-054, 2009.

Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401, 1999.

Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2001.

Hang Li and Naoki Abe. Word clustering and disambiguation based on co-occurrence data. In *Proceedings of the 17$^{th}$ international conference on Computational linguistics*, 1998.

Sara C. Madeira and Arlindo L. Oliveira. Biclustering algorithms for biological data analysis: A survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, January 2004.

Yishay Mansour and David McAllester. Generalization bounds for decision trees. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 2000.

Andreas Maurer. A note on the PAC-Bayesian theorem. www.arxiv.org, 2004.

David McAllester. Some PAC-Bayesian theorems. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 1998.

David McAllester. PAC-Bayesian model averaging. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 1999.

David McAllester. Simplified PAC-Bayesian margin bounds. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 2003.

David McAllester. Generalization bounds and consistency for structured labeling. In Gökhan Bakir, Thomas Hofmann, Bernhard Schölkopf, Alexander Smola, Ben Taskar, and S.V.N. Vishwanathan, editors, *Predicting Structured Data*. The MIT Press, 2007.

Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (NIPS)*, 2001.

Liam Paninski. Estimation of entropy and mutual information. *Neural Computation*, 2003.

Liam Paninski. Variational minimax estimation of discrete distributions under kl loss. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

Richard Rohwer and Dayne Freitag. Towards full automation of lexicon construction. In Dan Moldovan and Roxana Girju, editors, *HLT-NAACL 2004: Workshop on Computational Lexical Semantics*, 2004.

Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using Markov chain monte carlo. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

Matthias Seeger. PAC-Bayesian generalization error bounds for Gaussian process classification. *Journal of Machine Learning Research*, 2002.

Matthias Seeger. *Bayesian Gaussian Process Models: PAC-Bayesian Generalization Error Bounds and Sparse Approximations*. PhD thesis, University of Edinburgh, 2003.

Yevgeny Seldin. *A PAC-Bayesian Approach to Structure Learning*. PhD thesis, The Hebrew University of Jerusalem, 2009.

Yevgeny Seldin. A PAC-Bayesian analysis of graph clustering and pairwise clustering. Technical report, http://arxiv.org/abs/1009.0499, 2010.

Yevgeny Seldin and Naftali Tishby. Multi-classification by categorical features via clustering. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

Yevgeny Seldin and Naftali Tishby. PAC-Bayesian generalization bound for density estimation with application to co-clustering. In *Proceedings on the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009.

Yevgeny Seldin, Noam Slonim, and Naftali Tishby. Information bottleneck for non co-occurrence data. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2007.

M. Mahdi Shafiei and Evangelos E. Milios. Model-based overlapping co-clustering. In *Proceeding of SIAM Conference on Data Mining*, 2006.

Ohad Shamir and Naftali Tishby. On the reliability of clustering stability in the large sample regime. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

Hanhuai Shan and Arindam Banerjee. Bayesian co-clustering. In *IEEE International Conference on Data Mining (ICDM)*, 2008.

John Shawe-Taylor and David Hardoon. Pac-bayes analysis of maximum entropy classification. In *Proceedings on the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2009.

John Shawe-Taylor and Robert C. Williamson. A PAC analysis of a Bayesian estimator. In *Proceedings of the International Conference on Computational Learning Theory (COLT)*, 1997.

John Shawe-Taylor, Peter L.Bartlett, Robert C.Williamson, and Martin Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 44(5), 1998.

Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), 2000.

Noam Slonim. *The Information Bottleneck: Theory and Applications*. PhD thesis, The Hebrew University of Jerusalem, 2002.

Noam Slonim and Naftali Tishby. Document clustering using word clusters via the information bottleneck method. In *The 23$^{rd}$ Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2000.

Noam Slonim, Nir Friedman, and Naftali Tishby. Unsupervised document classification using sequential information maximization. In *Proceedings of the* 25$^{th}$ *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2002.

Noam Slonim, Gurinder Singh Atwal, Gasper Tracik, and William Bialek. Information-based clustering. *Proceedings of the National Academy of Science*, 102(51), 2005.

Noam Slonim, Nir Friedman, and Naftali Tishby. Multivariate information bottleneck. *Neural Computation*, 18, 2006.

Nathan Srebro. *Learning with Matrix Factorizations*. PhD thesis, MIT, 2004.

Nathan Srebro, Noga Alon, and Tommi S. Jaakkola. Generalization error bounds for collaborative prediction with low-rank matrices. In *Advances in Neural Information Processing Systems (NIPS)*, 2005a.

Nathan Srebro, Jason Rennie, and Tommi Jaakkola. Maximum margin matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2005b.

Mark Steyvers and Tom Griffiths. Probabilistic topic models. In T. Landauer, D. McNamara, S. Dennis, and W. Kintsch, editors, *Latent Semantic Analysis: A Road to Meaning*. Laurence Erlbaum, 2006.

Gilbert Strang. *Introduction to linear algebra*. Wellesley-Cambridge Press, 4$^{th}$ edition, 2009.

Ilya Sutskever, Ruslan Salakhutdinov, and Josh B. Tenenbaum. Modelling relational data using Bayesian clustered tensor factorization. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

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

Naftali Tishby, Fernando Pereira, and William Bialek. The information bottleneck method. In *Allerton Conference on Communication, Control and Computation*, 1999.

L. G. Valiant. A theory of the learnable. *Communications of the Association for Computing Machinery*, 27(11), 1984.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Soviet Math. Dokl.*, 9, 1968.

V. N. Vapnik and A. Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2), 1971.

Vladimir N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.

Ulrike von Luxburg and Shai Ben-David. Towards a statistical theory of clustering. In *PASCAL Workshop on Statistics and Optimization of Clustering*, 2005.

Pu Wang, Carlotta Domeniconi, and Kathryn Blackmond Laskey. Latent Dirichlet Bayesian co-clustering. In *Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, 2009.

Elad Yom-Tov and Noam Slonim. Parallel pairwise clustering. In *SIAM International Conference on Data Mining (SDM)*, 2009.

Jiho Yoo and Seungjin Choi. Probabilistic matrix tri-factorization. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009a.

Jiho Yoo and Seungjin Choi. Weighted nonnegative matrix co-tri-factorization for collaborative prediction. In *Proceedings of the Asian Conference on Machine Learning (ACML)*, 2009b.