Variational Inference for Nonparametric Multiple Clustering

Yue Guan, Jennifer G. Dy, Donglin Niu
Electrical & Computer Engineering Department
Northeastern University
Boston, MA 02115
{yguan, jdy, dniu}@ece.neu.edu

Zoubin Ghahramani
Engineering Department
University of Cambridge
Cambridge, UK
zoubin@eng.cam.ac.uk

ABSTRACT

Most clustering algorithms produce a single clustering solution. Similarly, feature selection for clustering tries to find one feature subset where one interesting clustering solution resides. However, a single data set may be multi-faceted and can be grouped and interpreted in many different ways, especially for high dimensional data, where feature selection is typically needed. Moreover, different clustering solutions are interesting for different purposes. Instead of committing to one clustering solution, in this paper we introduce a probabilistic nonparametric Bayesian model that can discover several possible clustering solutions and the feature subset views that generated each cluster partitioning simultaneously. We provide a variational inference approach to learn the features and clustering partitions in each view. Our model allows us not only to learn the multiple clusterings and views but also allows us to automatically learn the number of views and the number of clusters in each view.

Keywords

multiple clustering, non-redundant/disparate clustering, feature selection, nonparametric Bayes, variational inference

1. INTRODUCTION

Given unlabeled (or unsupervised) data, one of the first methods a data analyst might use to explore the data is clustering. Clustering is the process of grouping similar instances together. Most clustering algorithms find one partitioning of the data [12]. However, in high-dimensional data like text, image and gene data, instances can be grouped together in several different ways for different purposes. For example, face images of people can be grouped based on their pose or clustered based on the identity of the person. Web-pages collected from universities may have one clustering scheme based on the type of the web-page's owner: {*faculty, student, staff*}. Another clustering scheme could be based on the web-page's topic: {*physics, math, engineering, computer science*}. The third scheme could be based on which university the web-page belongs to. Often in real applications, the clustering solution found is not what the analyst or scientist is looking for, but the other possible groupings after removing the unwanted solution. In some cases, the analyst may simply want to discover all possible unique clustering solutions of the data.

Clustering is a difficult problem. Moreover, the difficulty is compounded by that data may be multi-faceted (i.e., there may be several possible clustering interpretations from a single data). In high-dimensional data, typically not all features are important; some features may be irrelevant and some may be redundant. Thus, the need for feature selection for clustering [9]. When designing a feature selection algorithm for clustering, one needs to define a criterion for selecting which of two or more alternative feature subsets is the relevant/interesting subset. Why choose one feature subset, when all the alternative feature subset views might be interesting to the user? Features irrelevant to one clustering interpretation, may be relevant to another clustering solution. In the web-page example, some of the words may be relevant to clustering the data based on university and other words may be more appropriate for grouping the data based on the web-page's owner. *In this paper, our goal is to discover multiple clustering solutions in different feature subset views simultaneously.*

To find multiple clustering partitions in different feature views, we model our data by a nonparametric Bayesian generative process: each feature is assumed to come from an infinite mixture of views, where each view generates an infinite mixture of clusters. This nonparametric Bayesian model allows us not only to learn the multiple clusterings and views but also allows us to automatically learn the number of views and the number of clusters in each view. In this paper, we introduce our probabilistic model and provide the variational inference steps necessary to learn all the parameters and hidden variables.

The rest of this paper is organized as follows. Section 2 reviews related work. In Section 3, we present our nonparametric model and variational method for learning. In Section 4, we report and discuss the results of our experiments on both synthetic and real data. And finally, we provide our conclusions and directions for future work in Section 5.

2. RELATED WORK

There are recent interests in finding more than one clustering interpretation. Given a data set with a clustering solution, Gondek and Hofmann in [10] suggest finding an alternative non-redundant clustering by a conditional information bottleneck approach [6]. Bae and Bailey in [1] utilize cannot-link constraints imposed on data points belonging to the same group by a previous clustering and agglomerative clustering in order to find an alternative non-redundant

clustering. In Qi et al. [20], they find an alternative projection of the original data that minimizes the Kullback-Leibler divergence between the distribution of the original space and the projection subject to the constraint that sum squared error between samples in the projected space with the means of the clusters they belong to is smaller than a pre-specified threshold. Their method approximates the clusters from mixtures of Gaussians with components sharing the same covariance matrix. These three [10, 1, 20] only addresses finding one alternative clustering. However, for complex data there may be more than one alternative clustering interpretation. In Cui et al. [7], their method finds multiple alternative views by clustering in the subspace orthogonal to the clustering solutions found in previous iterations. This approach discovers several alternative clustering solutions by iteratively finding one alternative solution given the previously found clustering solutions. All these methods find alternative clustering solutions sequentially (or iteratively). Another general way for discovering multiple solutions is by finding them simultaneously. Meta clustering in [4] generates a diverse set of clustering solutions by either random initialization or random feature weighting. Then these solutions are meta clustered using an agglomerative clustering based on a Rand index for measuring similarity between pairwise clustering solutions. Jain et al. in [13] also learn the non-redundant views together. Their method learns two disparate clusterings by minimizing two k-means type sum-squared error objective for the two clustering solutions while at the same time minimizing the correlation between these two clusterings. Like [7], [4] and [13], the approach we propose discovers multiple clustering solutions. Furthermore, like [4] and [13], our approach finds these solutions simultaneously. However, unlike all these methods, we provide a probabilistic generative nonparametric model which can learn the features and clustering solutions in each view simultaneously.

Recently, there are several nonparametric Bayesian models introduced for unsupervised learning [19, 3, 11, 22]. The Chinese Restaurant Process (CRP) [19] and the stick-breaking Dirichlet Process Model [3] only assume one underlying partitioning of the data samples. The Indian Buffet Process (IBP) assumes that each sample is generated from an underlying latent set of features sampled from an infinite menu of features. There are also nonparametric Bayesian models for co-clustering [22]. None of these model multiple clustering solutions. There is, however, concurrent work that is independently developed that provides a nonparametric Bayesian model for finding multiple partitionings, called cross-categorization [17]. Their model utilizes the CRP construction and Gibbs sampling for inference. Here, we propose an approach that utilizes a multiple clustering stick-breaking construction and provide a variational inference approach to learn the model parameters and latent variables. Unlike Markov chain Monte Carlo samplers [18] including Gibbs sampling, variational methods provide a fast deterministic approximation to marginal probabilities.

3. NONPARAMETRIC MULTIPLE CLUSTERING MODEL

Given data $X \in \mathbb{R}^{n \times d}$, where n is the number of samples and d the number of features. $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^T = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_d]$, where $\mathbf{x}_i \in \mathbb{R}^d$ are the samples, the columns $\mathbf{f}_j \in \mathbb{R}^n$ are the features, and $(\cdot)^T$ is the transpose of a matrix. Our goal is to discover multiple clustering solutions and the feature views in which they reside. We formulate this problem as finding a partitioning of the data into a tile structure as shown in Figure 1. We want to find a partitioning that partitions the features into different views and the partitioning of the samples in each view. In the tile figure, the columns are permuted such that features that belong together in the same view are next to each other and the borders show partitions for the different views. In Figure 1, the samples in each view are clustered into groups where members from the same group are indicated by the same color. Note that the samples in different views are independently partitioned given the view; moreover, samples belonging to the same clusters in one view can belong to different clusters in other views. Here, we assumed that the features in each view are disjoint. In future work, we will explore models that can allow sharing of features between views.

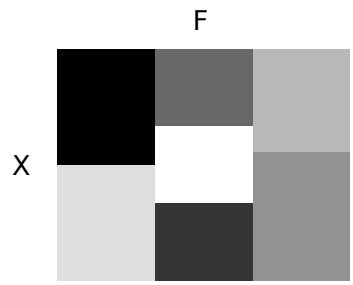


Figure 1: Tile structure partitioning of the data for multiple clustering in different feature views.

In this paper, we design a nonparametric prior model that can generate such a latent tile structure for data X. Let Y be a matrix of latent variables representing the partitioning of features into different views, where each element, $y_{j,v} = 1$ if feature $\mathbf{f}_j$ belongs to view v and $y_{j,v} = 0$ otherwise. And, let Z be a matrix of latent variables representing the partitioning of samples for all views, where each element, $z_{v,i,k} = 1$ if sample $\mathbf{x}_i$ belongs to cluster k in view v and $z_{v,i,k} = 0$ otherwise. We model both latent variables Y and Z from Dirichlet processes and utilize the stick-breaking construction [21] to generate these variables as follows.

1. Generate an infinite sequence w_v from a beta distribution: $w_v \sim Beta(1, \alpha),\ v = \{1, 2, \ldots\}$. α is the concentration parameter for a beta distribution.

2. Generate the mixture weights π_v for each feature partition v from a stick-breaking construction: $\pi_v = w_v \prod_{j=1}^{v-1}(1-w_j)$.

3. Generate the view indicator Y for each feature $\mathbf{f}_j$ from a multinomial distribution with weights π_v: $p(y_{j,v} = 1|\boldsymbol{\pi}) = \pi_v$, denoted $\mathbf{y}_j \sim Mult(\boldsymbol{\pi}) = \prod_{v=1}^{q} \pi_v^{y_{j,v}}$ and q is the number of views, which can be infinite.

4. For each view $v = \{1, 2, \ldots\}$, generate an infinite sequence $u_{v,k}$ from a beta distribution: $u_{v,k} \sim Beta(1, \beta),\ k = \{1, 2, \ldots\}$. Here, β is the concentration parameter for a beta distribution.

5. Generate the mixture weights $\eta_{v,k}$ for each cluster partition k in each view v from a stick-breaking construction: $\eta_{v,k} = u_{v,k} \prod_{\ell=1}^{k-1}(1 - u_{v,\ell})$.

6. Generate the cluster indicator Z for each view v from a multinomial distribution with weights $\eta_{v,k}$: $p(z_{v,i,k} = 1|\boldsymbol{\eta}) = \eta_{v,k}$ or $\mathbf{z}_{v,i} \sim Mult(\boldsymbol{\eta}) = \prod_{k=1}^{k_v} \eta_{v,k}^{z_{v,i,k}}$ and k_v is the number of clusters in view v, which can be infinite.

We are now ready to generate our observation variables X given the latent variables Y and Z. For each cluster in each view, we draw cluster parameter $\boldsymbol{\theta}$ from an appropriate prior distribution with hyperparameter $\boldsymbol{\lambda}$: $\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\boldsymbol{\lambda})$. Then, in each view v, we draw the value of the features in view v for sample i: $\mathbf{x}_{v,i} \sim p(\mathbf{x}_{v,i}|\theta_{v,z_{v,i}})$, where $z_{v,i}$ is equal to the cluster k that sample i belongs to in view v, and $\mathbf{x}_{v,i} = (x_{i,j} : y_{j,v} = 1)$ is the vector of features in view v.

Figure 2 shows a graphical model of our nonparametric multiple clustering model. Our joint model $p(X, Y, Z, W, U, \theta)$ is:

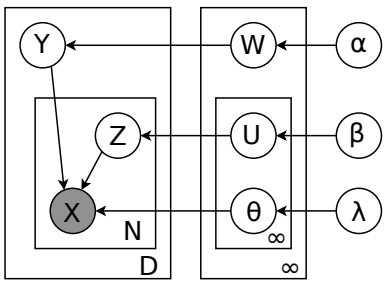


Figure 2: Graphical model for our nonparametric multiple clustering model.

$$
\begin{aligned}
p(X|Y, Z, \boldsymbol{\theta})p(Y|W)p(W|\alpha)p(Z|U)p(U|\beta)p(\boldsymbol{\theta}|\boldsymbol{\lambda}) \quad (1)\\
= \left[\prod_{v=1}^{q}\prod_{i=1}^{n} p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}})p(z_{v,i}|\eta_{v,i})\right]\\
\left[\prod_{v=1}^{q}\prod_{k=1}^{k_v} p(\boldsymbol{\theta}_{v,k}|\boldsymbol{\lambda})p(u_{v,k}|\beta)\right]\\
\prod_{j=1}^{d} p(\mathbf{y}_j|\boldsymbol{\pi}) \prod_{v=1}^{q} p(w_v|\alpha)
\end{aligned}
$$

where q is the number of views and k_v is the number of clusters in view v, n is the number of data points and d is the number of features. Here, in our nonparametric model, q and k_v can be infinite. Note that previous models for multiple clustering explicitly enforce non-redundancy, orthogonality or disparity among clustering solutions [10, 1, 7, 20, 13]; whereas, our model handles redundancy implicitly, since redundant clusterings offer no probabilistic modelling advantage and are penalized under the prior which assumes that each view is clustered independently.

3.1 Variational Inference

It is computationally intractable to evaluate the marginal likelihood, $p(X) = \int p(X, \boldsymbol{\phi})d\boldsymbol{\phi}$, where $\boldsymbol{\phi}$ represents the set of all parameters θ and latent variables, Y, Z, U, and W. Variational methods allow us to approximate the marginal likelihood by maximizing a lower bound, $\mathcal{L}(Q)$, on the true log marginal likelihood [15].

$$
\begin{aligned}
\ln p(X) &= \ln \int p(X, \boldsymbol{\phi})d\boldsymbol{\phi}\\
&= \ln \int Q(\boldsymbol{\phi})\frac{p(X, \boldsymbol{\phi})}{Q(\boldsymbol{\phi})}d\boldsymbol{\phi}\\
&\geq \int Q(\boldsymbol{\phi}) \ln \frac{p(X, \boldsymbol{\phi})}{Q(\boldsymbol{\phi})}d\boldsymbol{\phi} = \mathcal{L}(Q(\boldsymbol{\phi})),
\end{aligned}
$$

using Jensen's inequality [5]. The difference between the log marginal $\ln p(X)$ and the lower bound $\mathcal{L}(Q)$ is the Kullback-Leibler divergence between the approximating distribution $Q(\boldsymbol{\phi})$ and the true posterior $p(\boldsymbol{\phi}|X)$. The idea is to choose a $Q(\boldsymbol{\phi})$ distribution that is simple enough that the lower bound can be tractably evaluated and flexible enough to get a tight bound. Here, we assume a distribution for $Q(\boldsymbol{\phi})$ that factorizes over all the parameters $Q(\boldsymbol{\phi}) = \prod_i Q_i(\phi_i)$. For our model, this is

$$
Q(Y, Z, W, U, \boldsymbol{\theta}) = Q(Y)Q(Z)Q(W)Q(U)Q(\boldsymbol{\theta}) \quad (2)
$$

The $Q_i(\cdot)$ that minimizes the KL divergence over all factorial distributions is

$$
Q_i(\phi_i) = \frac{\exp \langle \ln P(X, \boldsymbol{\phi})\rangle_{k\neq i}}{\int \exp \langle \ln P(X, \boldsymbol{\phi})\rangle_{k\neq i} d\phi_i} \quad (3)
$$

where $\langle\cdot\rangle_{k\neq i}$ denotes averaging with respect to $\prod_{k\neq i} Q_k(\phi_k)$. Applying Equation 3, we obtain our factorial distributions, $Q_i(\phi_i)$ functions as:

$$
\begin{aligned}
Q(W) &= \prod_{v=1}^{q} Beta(\gamma_{v,1}, \gamma_{v,2}) && (4)\\
Q(Y) &= \prod_{j=1}^{d} Mult(\pi_j) && (5)\\
Q(U) &= \prod_{v=1}^{q}\prod_{k=1}^{k_v} Beta(\gamma_{v,k,1}, \gamma_{v,k,2}) && (6)\\
Q(Z) &= \prod_{v=1}^{q}\prod_{i=1}^{n} Mult(\eta_{v,i}) && (7)
\end{aligned}
$$

where $\gamma_{v,\cdot}$ are the beta parameters for the distributions on W, π_j are the multinomial parameters for the distributions on Y, $\gamma_{v,k,\cdot}$ are the beta parameters for the distributions on U, and $\eta_{v,i}$ are the multinomial parameters for the distributions on Z. Note that these variational parameters for the approximate posterior Q are not the same as the model parameters in Equation 1 although we have used similar notation.

The update equations we obtain using variational inference are provided below:

$$
\begin{aligned}
\gamma_{v,1} &= 1 + \sum_{j=1}^{d} \pi_{j,v} && (8)\\
\gamma_{v,2} &= \alpha + \sum_{j=1}^{d}\sum_{l=v+1}^{q} \pi_{j,l} && (9)\\
\gamma_{v,k,1} &= 1 + \sum_{i=1}^{n} \eta_{v,i,k} && (10)\\
\gamma_{v,k,2} &= \beta + \sum_{i=1}^{n}\sum_{l=k+1}^{k_v} \eta_{v,i,l} && (11)\\
\pi_{j,v} &\propto \left[\prod_{i=1}^{n} p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}})\right] Mult(\pi_{j,v}|\gamma_v) && (12)\\
\eta_{v,i,k} &\propto p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}})Mult(\eta_{v,i,k}|\gamma_{v,k}) && (13)
\end{aligned}
$$

Note that all parameters on the right hand side of the equations above are based on the parameter estimates at the previous time step and those on the left hand side are the parameter updates at the current time step. The variational parameter $\eta_{v,i,k}$ can be interpreted as the posterior probability that view v of data point i is assigned to cluster k. The parameter $\pi_{j,v}$ can be interpreted as the posterior probability that feature j belongs to view v. We iterate these update steps until convergence.

3.2 Observation Models

In this paper, we provide two common observation probability models for modeling cluster components in mixture models: the Gaussian component and the multinomial component model. The Gaussian model is widely used for real-valued data where samples are assumed to be variations of some prototype. Multinomial model is appropriate for discrete data, such as text.

Gaussian Component. Assuming $p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}})$ comes from a Gaussian distribution, our parameter vector $\boldsymbol{\theta}_{v,z_{v,i}}$ comprises the mean $\boldsymbol{\mu}_{v,z_{v,i}}$ and covariance $\Sigma_{v,z_{v,i}}$ of our Gaussian distribution in view v and the cluster $z_{v,i}$ (the cluster sample $\mathbf{x}_{v,i}$ belongs to). We apply a normal-inverse Wishart distribution, the conjugate prior to a Gaussian distribution as our prior $p(\boldsymbol{\theta}_{v,z_{v,i}}|\boldsymbol{\lambda})$. The hyperparameter $\boldsymbol{\lambda}$ is a vector composed of the mean $\mathbf{m}_0$, covariance S_0, inverse scale matrix Ψ_0, and parameter p_0. The Gaussian likelihood distribution, $p(X|Y,Z,\boldsymbol{\theta})$ is:

$$\begin{aligned} & p(X|Y,Z,\theta) \\ = & \prod_{v=1}^{q}\prod_{i=1}^{n} \frac{1}{(2\pi)^{d_v/2}|\Sigma_{v,z_{v,i}}|^{1/2}} \\ & \exp\left(-\frac{1}{2}(x_{v,i}-\boldsymbol{\mu}_{v,z_{v,i}})^T\Sigma_{v,z_{v,i}}^{-1}(x_{v,i}-\boldsymbol{\mu}_{v,z_{v,i}})\right) \end{aligned}$$

where the indices indicate the corresponding view v and cluster component $z_{v,i}$ that sample i belongs to in each view and d_v is the number of features in view v.

Applying variational approximation, we have $Q(\boldsymbol{\theta}_{v,k})$ for each view v and cluster k as:

$$\frac{\Psi_{v,k}^{d_v/2}}{(2\pi)^{d_v/2}|\Sigma_{v,k}|^{1/2}}\exp(-\frac{1}{2}(\boldsymbol{\mu}_{v,k}-\mathbf{m}_{v,k})^T\Psi_{v,k}\Sigma_{v,k}^{-1}(\boldsymbol{\mu}_{v,k}-\mathbf{m}_{v,k})) \frac{|S_{v,k}|^{p_{v,k}/2}|\Sigma_{v,k}|^{-(p_{v,k}+d_v+1)/2}\exp(-\frac{1}{2}\text{trace}(S_{v,k}\Sigma_{v,k}^{-1}))}{2^{p_{v,k}d_v/2}\pi^{d_v(d_v-1)/4}\prod_{j=1}^{d_v}\Gamma((p_{v,k}+1-j)/2)}$$

where $\mathbf{m}_{v,k}$, $S_{v,k}$, $\Psi_{v,k}$ and $p_{v,k}$ are the parameters of the posterior normal-inverse Wishart distribution for cluster k in view v.

And the update equations are:

$$\mathbf{m}_{v,k} = \frac{n_{v,k}\bar{\mathbf{x}}_{v,k}+\Psi_0\mathbf{m}_0}{n_{v,k}+\Psi_0} \quad (14)$$

$$\Psi_{v,k} = \Psi_0 + n_{v,k} \quad (15)$$

$$p_{v,k} = p_0 + n_{v,k} \quad (16)$$

$$S_{v,k} = S_0 + \sum_{\mathbf{x}_{v,i}\in k}(\mathbf{x}_{v,i}-\bar{\mathbf{x}}_{v,k})(\mathbf{x}_{v,i}-\bar{\mathbf{x}}_{v,k})^T + \frac{n_{v,k}\Psi_0}{n_{v,k}+\Psi_0}(\bar{\mathbf{x}}_{v,k}-\mathbf{m}_0)(\bar{\mathbf{x}}_{v,k}-\mathbf{m}_0)^T \quad (17)$$

where $\bar{\mathbf{x}}_{v,k}$ is the sample mean of $\mathbf{x}$ in cluster k of view v and $n_{v,k}$ are the number of samples in cluster k in view v. The parameters $\boldsymbol{\mu}_{v,k}$ and $\Sigma_{v,k}$ are updated by their expected values under the variational distribution.

Multinomial Component. Assuming $p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}})$ comes from a multinomial distribution, our parameter vector $\boldsymbol{\theta}_{v,z_{v,i}}$ comprises of the probability of occurrence for each feature in view v and cluster $z_{v,i}$:

$$\rho_{v,z_{v,i},1},\ldots,\rho_{v,z_{v,i},d_v},$$

where d_v is the number of features in view v. We use a Dirichlet distribution, the conjugate prior to a multinomial distribution as our prior $p(\boldsymbol{\theta}_{v,z_{v,i}}|\boldsymbol{\lambda})$, with hyperparameters $\boldsymbol{\lambda}=\{\alpha_{0,1},\ldots,\alpha_{0,d_v}\}$. The multinomial likelihood distribution, $p(X|Y,Z,\boldsymbol{\theta})$ is:

$$\prod_{v=1}^{q}\prod_{i=1}^{n}p(\mathbf{x}_{v,i}|\boldsymbol{\theta}_{v,z_{v,i}}) = \prod_{v=1}^{q}\prod_{i=1}^{n}\rho_{v,z_{v,i},1}^{x_{v,i,1}}\cdots\rho_{v,z_{v,i},d_v}^{x_{v,i,d_v}} \quad (18)$$

Here, we use the notation $x_{v,i,l}$ to be the value of the lth feature in view v in sample i.

Applying variational approximation, we have $Q(\boldsymbol{\theta}_{v,k})$ as:

$$Q(\boldsymbol{\theta}_{v,k}) = \frac{\Gamma(\sum_{l=1}^{d_v}\alpha_{v,k,l})}{\prod_{l=1}^{d_v}\Gamma(\alpha_{v,k,l})}\rho_{v,k,1}^{\alpha_{v,k,1}-1}\cdots\rho_{v,k,d_v}^{\alpha_{v,k,d_v}-1} \quad (19)$$

where $\alpha_{v,k,\cdot}$ is the parameter for the posterior Dirichlet distribution in cluster k of view v. And the update equation is:

$$\alpha_{v,k,l} = \alpha_{0,l} + x_{v,k,l} \quad (20)$$

where $x_{v,k,l}$ is the count of the lth feature in cluster k of view v. The parameters $\rho_{v,k,l}$ are updated by their means under the variational distribution.

4. EXPERIMENTS

In this section, we perform experiments on both synthetic and real data to investigate whether our algorithm gives reasonable multiple clustering solutions. We first test our method on two synthetic data sets in Section 4.1 to get an understanding of the performance of our method. Then we test our method on three real data in Section 4.2: a face image, a machine sound and a text data. We compare our method, we call nonparametric multi-clust, against two recently proposed algorithms for discovering multiple clusters: orthogonal projection clustering [7] and de-correlated k-means [13]. We also compare our method against related single-view clustering algorithms: a probabilistic approach that performs feature selection and clustering simultaneously (FSC) [16], and a baseline nonparametric Dirichlet process mixture model (DP-Gaussian or DP-Multinomial) [3].

In orthogonal projection clustering, they first reduced the dimensionality by principal component analysis [14] (retaining 90% of the total variance)). Then, instances are clustered in the principal component space by k-means clustering algorithm to find a dominant clustering. Because the means of clusters represent the clustering solution, data are projected to the subspace that is orthogonal to the subspace spanned by the means. In the orthogonal subspace, they use PCA followed by the clustering algorithm again to find an alternative clustering solution. This process is repeated until all the possible views are found. In de-correlated k-means [13], they simultaneously minimize the sum-squared errors in two clustering views and the correlation of the mean vectors and representative vectors between the two views. They apply gradient descent to find the clustering solutions. In de-correlated k-means, both views minimize sum-squared errors in all the original dimensions. In FSC [16], they add a feature saliency, a measure of feature relevance, as a missing variable to a probabilistic objective function. To add feature saliency, they assumed that the features are conditionally independent. They then derived an expectation-maximization (EM) [8] algorithm to estimate the feature saliency for a mixture of Gaussians. They are able to find the features and clusters simultaneously. They can also automatically determine the number of clusters based on a minimum message length penalty. We also compare with a nonparametric Dirichlet process mixture [3] to serve as

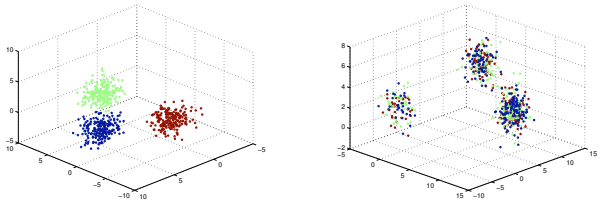
a baseline. In all our experiments, we apply the Gaussian cluster component for the synthetic, image and sound data and the multinomial cluster component for text data for our method, FSC and the DP mixture. We set the concentration parameter α and β in our experiment to 10. We set $q = d$ and $k_v = n/2$ for all views. We similarly set the concentration parameter in the DP mixture to 10 and the maximum number of clusters $k = n/2$. We set our hyperparameters for the Gaussian model as $\mathbf{m}_0 = 0$ and $S_0 = I$ identity, and for the multinomial model as $\alpha_{0,l} = 1$. For FSC, we set the maximum number of clusters equal to twice the true number of clusters. For both orthogonal projection and de-correlated k-means, we set the number of clusters and views equal to the known number of clusters and views.

To show how well our method compares against competing methods in discovering the "true" labeling, we report the normalized mutual information (NMI) [23] between the clusters found by these methods with the "true" class labels. Let C represent the clustering results and L the labels, $NMI = \frac{MI(C,L)}{\sqrt{H(C)H(L)}}$, where $MI(C, L)$ is the mutual information between random variables C and L and $H(\cdot)$ is the entropy of $(\cdot)$. Note that in all our experiments, labeled information are not used for training. We only use them to measure the performance of our clustering algorithms. Higher NMI values mean the more similar the clustering results are with the labels; and it only reaches its maximum value of one when both clustering and labels are perfectly matched.

4.1 Experiments on Synthetic Data

To get a better understanding of our method and test its applicability, we first perform our approach on two synthetic data sets. The first synthetic data is a data set with two alternative views. The second synthetic data is a high dimensional data with three independent clustering views. We want to test whether or not our algorithm can deal with high dimensionality and more than two views.

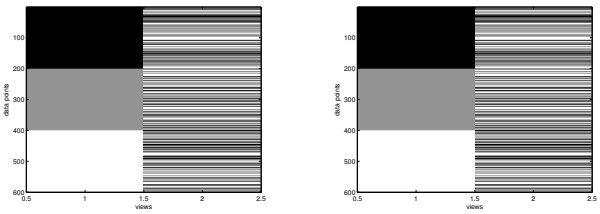
The first synthetic data is generated from six features with 600 instances. Three Gaussian clusters are generated in feature subspace $\{F_1, F_2, F_3\}$ with 200, 200 and 200 instances in each cluster with identity covariances. We call this view 1. The other three Gaussian clusters are generated in feature subspace $\{F_4, F_5, F_6\}$ with 200, 100 and 300 instances respectively and with identity covariances. Similarly, we call this view 2. Figure 3 shows 3-D scatterplots of the two views. The colors in both plots reflect the labels for view 1 (one color for each cluster label). Note that the clusters in view 2 are independent of the labels in view 1 as shown by the mixing of the colors in each cluster in subfigure (b).



(a) View 1 (b) View 2

Figure 3: Scatter plots for synthetic data 1.

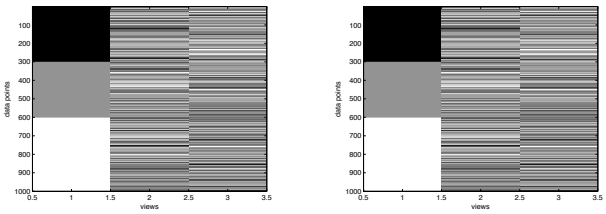
Figure 4(a) is the tile structure for the true labeling in each view for synthetic data 1. Figure 4(b) shows the tile structure discovered by our method. Here, our algorithm was able to correctly learn the number of views and number of clusters in each view as shown by the number of tile partitions in the figure. Our algorithm successfully discovers the two views in the dataset as shown by the similarity between the two tile structures. Our algorithm finds the feature subspaces corresponding to the two views (i.e., feature subset F_1, F_2, F_3 for the first view and feature subset F_4, F_5, F_6 for the second view). In the tile figure, the samples/rows were permuted such that samples that belong together in view one are close together. Note that the samples that belong together in view one need not belong together in view two. Hence, the random structure in view two.



(a) True Labeling (b) Algorithm Result

Figure 4: Tile partitioning structure for the first synthetic data: (a) the tile structure from the true labels and (b) the result of our algorithm.

The second synthetic data is 100-dimensional with 1000 instances and three independent views. In each feature set ($F_{(1..30)}$, $F_{(31..60)}$ and $F_{(61..100)}$), random vectors with three Gaussian components are generated. Figure 5(a) is the tile structure for the true labeling in each view. Figure 5(b) shows the tile structure discovered by our method. Our algorithm successfully discovers the three views in the dataset as shown by the similarity between the two tile structures. Our algorithm is able to find the feature subsets corresponding to the three views and also the correct number of views and number of clusters in each view.



(a) True Labeling (b) Algorithm Result

Figure 5: Tile partitioning structure for the second synthetic data: (a) the tile structure from the true labels and (b) the result of our algorithm.

Table 4.1 reports the NMI values obtained by the different methods on both synthetic data. For the multiple clustering methods, for each known view, we report the clustering that has the highest NMI with that view. For the single clustering methods, we report the NMI of that clustering with each of the true labeling views. The best results are highlighted in bold. The results show that our

method obtained consistently the best NMI for all views. Orthogonal projection is able to find view 1, but degrades it's performance in finding the second and third views. De-correlated k-means is able to find the two views for synthetic data 1, but fails to find the third view in synthetic data 2. Note that the single clustering methods, DP-Gaussian and FSC, only found the most dominant view, view 1 on both data sets and performed poorly on the other views.

Table 1: NMI Results on Synthetic Data 1 and 2

	DATA 1		DATA 2		
	VIEW 1	VIEW 2	VIEW 1	VIEW 2	VIEW 3
OUR METHOD	**0.89**	**0.91**	**0.85**	**0.83**	**0.88**
ORTHO. PROJ.	0.87	0.66	0.83	0.61	0.53
DEC. K-MEANS	0.84	0.87	0.83	0.75	0.46
FSC	0.86	0.20	0.83	0.14	0.17
DP-GAUSSIAN	0.85	0.21	0.84	0.13	0.18

4.2 Experiments on Real-World Data

We now test our method on three real-world data sets to see whether we can find meaningful clustering views that correspond to human labeling. We select data that have high-dimensionality and multiple possible partitionings. In particular, we test our method on a face image, a sound data and a text data.

4.2.1 Experiments on Face Data

The face dataset from UCI KDD repository [2] consists of 640 face images of 20 people taken at varying poses (straight, left, right, up), expressions (neutral, happy, sad, angry), eyes (wearing sunglasses or not). The two dominant views of this face data are: identity of the person and their pose. These two views are non-redundant, meaning that with the knowledge of identity, no prediction of pose can be made. We test our method to see whether we can find these two clustering views. Each person has 32 images with four equally distributed poses. The image resolution is 32×30 pixels, resulting in a data set with 640 instances and 960 features. For this data, we first applied principal components analysis [14] to extract appearance-based features [24] and reduce the dimensionality by keeping only the first 100 eigenvectors correponding to the largest eigenvalues. The first 100 eigenvectors retains a total of 99.24% of the overall variance.

Figure 6 shows the mean image of each cluster discovered by our method in view 1. Notice that view 1 corresponds to the identity of the face image. Similarly, in Figure 7, we show the mean image of each cluster discovered by our method in view 2. Here, view 2 corresponds to pose of the face image. To provide us with a summary of the features selected by our algorithm in each view, we display the top two features (eigenvectors) corresponding to the largest variance in each view and show them as images, also known as eigenfaces [24]. Figure 8 shows the first two eigenfaces for view 1 and Figure 9 shows the first two eigenfaces for view 2. Note that the eigenfaces (images) in view 1 as shown in Figure 8 make sense. It captures the face and background pixel information important for distinguishing identity. The eigenfaces in view 2 as shown in Figure 9 captures information necessary for distinguishing pose. Table 2 presents the NMI results for all methods on the three real data. The results show that our method successfully finds the two clustering views of the face data with the highest NMI values compared to all the competing methods. Again, FSC and DP-Gaussian only find one view; their NMI values for view 2 are very low.

Figure 6: Mean image corresponding to identity from the face data.

Figure 7: Mean image corresponding to pose from the face data.

4.2.2 Experiments on Machine Sound Data

This data is comprised of accelerometer acoustic signals of varying machines. The purpose of this project is to recognize the different machine types from these sound signals. In this data, there are three kinds of machine sounds: $pump, fan, motor$. Each instance of sound can be from one machine, or mixture of two machines, or mixture of three machines. The three views in this data are: pump or not pump, fan or not fan, and motor or not motor. There are 280 sound samples with 100000 FFT features. We used only the 1000 highest FFT points as our features. Table 2 shows that our method obtained the best NMI results for all views. We observe that the dominant motor view is not that dominant as reflected by the poor NMI results for FSC and DP-Gaussian. This also shows that the features for the other views can hurt the performance of single clustering algorithms, even ones that can also perform feature selection together with clustering simultaneously (FSC).

4.2.3 Experiments on WebKB Data

This data set [1] contains html documents from four universities: Cornell University, University of Texas, Austin, University of Washington and University of Wisconsin, Madison. We removed the miscellaneous pages and subsampled a total of 1041 pages from four web-page owner types: course, faculty, project and student. We pre-processed the data by removing rare words, stop words, and words with low variances, resulting in 350 word features. The two views are either based on university or based on owner type.

[1] http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/

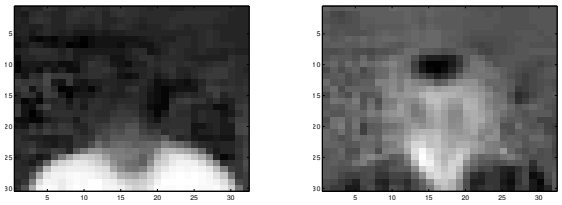


Figure 8: First two eigenfaces selected by our method for view 1 (based on identity) on the face data.

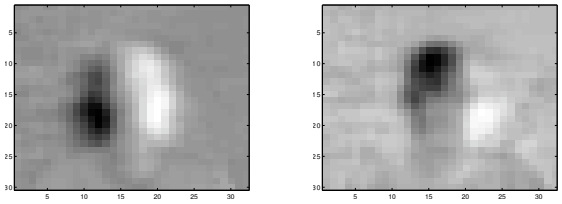


Figure 9: First two eigenfaces selected by our method for view 2 (based on pose) on the face data.

Table 2: NMI Results on Real-World Data

FACE DATA			
	IDENTITY		POSE
OUR METHOD	**0.86**		**0.63**
ORTHOGONAL PROJECTION	0.67		0.38
DE-CORRELATED K-MEANS	0.70		0.40
FSC	0.82		0.06
DP-GAUSSIAN	0.84		0.03
MACHINE SOUND DATA			
	PUMP	MOTOR	FAN
OUR METHOD	**0.82**	**0.87**	**0.83**
ORTHOGONAL PROJECTION	0.73	0.68	0.47
DE-CORRELATED K-MEANS	0.64	0.58	0.75
FSC	0.12	0.42	0.26
DP-GAUSSIAN	0.25	0.32	0.16
WEBKB TEXT DATA			
	OWNER		SCHOOL
OUR METHOD	**0.62**		**0.68**
ORTHOGONAL PROJECTION	0.43		0.53
DE-CORRELATED K-MEANS	0.48		0.57
FSC	0.16		0.38
DP-MULTINOMIAL	0.26		0.39

The results in Table 2 show that our method outperformed all the other methods in finding the two views in terms of NMI. Similar to the machine sound data, in this data, the dominant school view is not as dominant compared to the owner-type view, resulting in poor NMI results for FSC and DP-multinomial. Again this shows that when data is multi-faceted, the features for the other views can hurt the performance of single clustering algorithms, even ones that can also perform feature selection and clustering simultaneously (FSC).

We also analyzed the words in each view discovered by our algorithm. The top ten most frequent words in view 1 (based on owner-type) is: student, project, research, theorem, department, report, group, science, faculty, system. The top ten most frequent words in view 2 (based on school) is: Cornell, finance, social, Washington, Texas, visual, define, Wisconsin, program, Madison. Note that these words make sense and are descriptive of their corresponding views.

5. CONCLUSIONS AND FUTURE WORK

Traditional clustering algorithms output a single clustering solution. In this paper, we introduced a new method for discovering multiple clustering solutions. A difficulty with clustering in high-dimensional spaces is that not all features are important to find the interesting cluster structure. To add to this difficulty, there is no one criterion for clustering that works well for all applications. Thus, finding the best feature subset and clusters usually has the dilemma of selecting the best solution out of several reasonable alternatives. In the multiple clustering paradigm, we avoid this dilemma by allowing the algorithm to discover different possible clustering solutions in different feature subset views.

Data may be multi-faceted, such that different feature views may provide different possible clustering interpretation. We introduced a probabilistic nonparametric Bayesian model for this type of richly structured multi-faceted data. Moreover, we provided a variational inference approach to learn the features and clusters in each view for this model. Besides learning the latent features and clusters per view, our Bayesian formulation also allows us to automatically learn the number of views and the number of clusters in each view. Our results on synthetic and real-world data show that our method can find multiple alternative views of the data with accuracies competitive to other multiple clustering methods. In addition, our approach is able to discover the feature subsets in each view with cluster accuracies competitive to a state-of-the-art unsupervised feature selection method [16]. Our current model assumes that the features in each view are disjoint; however, the interesting multiple clustering views may share some common features. One of our future research directions is to explore models that allow feature sharing. Another direction we are currently working on are hierarchical clustering extensions of our model.

6. REFERENCES

[1] E. Bae and J. Bailey. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *IEEE International Conference on Data Mining*, pages 53–62, 2006.

[2] S. D. Bay. The UCI KDD archive, 1999.

[3] D. M. Blei and M. I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2006.

[4] R. Caruana, M. Elhawary, N. Nguyen, and C. Smith. Meta clustering. In *IEEE International Conference on Data Mining*, pages 107–118, 2006.

[5] D. Chandler. *Introduction to Modern Statistical Mechanics*. Oxford University Press, USA, 1987.

[6] G. Chechik and N. Tishby. Extracting relevant structures with side information. In *Advances in Neural Information Processing Systems 15 (NIPS-2002)*, pages 857–864, 2003.

[7] Y. Cui, X. Z. Fern, and J. Dy. Non-redundant multi-view clustering via orthogonalization. In *IEEE Intl. Conf. on Data Mining*, pages 133–142, 2007.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum

likelihood from incomplete data via the em algorithm. *Journal Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[9] J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *Journal of Machine Learning Research*, 5:845–889, August 2004.

[10] D. Gondek and T. Hofmann. Non-redundant data clustering. In *Proceedings of the IEEE International Conference on Data Mining*, pages 75–82, 2004.

[11] T. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 475–482. MIT Press, Cambridge, MA, 2006.

[12] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[13] P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learing of disparate clustering. In *SIAM Intl. Conf. on Data Mining*, pages 858–869, 2008.

[14] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New-York, 1986.

[15] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[16] M. H. Law, M. Figueiredo, and A. K. Jain. Feature selection in mixture-based clustering. In *Advances in Neural Information Processing Systems 15*, Vancouver, December 2002.

[17] V. Mansinghka, E. Jonas, C. Petschulat, B. Cronin, P. Shafto, and J. Tenenbaum. Cross-categorization: A method for discovering multiple overlapping clusterings. In *Nonparametric Bayes Workshop at NIPS*, 2009.

[18] R. Neal. Probabilistic inference using markov chain monte carlo methods. Technical report, Dept. of Computer Science, University of Toronto, 1993.

[19] J. Pitman. Combinatorial stochastic processes. Technical report, U.C. Berkeley, Department of Statistics, August 2002.

[20] Z. J. Qi and I. Davidson. A principled and flexible framework for finding alternative clusterings. In *ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2009.

[21] J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

[22] H. Shan and A. Banerjee. Bayesian co-clustering. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM)*, pages 530–539, Pisa, Italy, 2008.

[23] A. Strehl and J. Ghosh. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *Journal on Machine Learning Research*, 3:583–617, 2002.

[24] M. Turk and A. Pentland. Face recognition using eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–591, 1991.