Latent Simplex Position Model:
High Dimensional Multi-view Clustering
with Uncertainty Quantification

Leo L. Duan   LI.DUAN@UFL.EDU
Department of Statistics
University of Florida
Gainesville, FL 32611, USA

Abstract

High dimensional data often contain multiple facets, and several clustering patterns can co-exist under different variable subspaces, also known as the views. While multi-view clustering algorithms were proposed, the uncertainty quantification remains difficult — a particular challenge is in the high complexity of estimating the cluster assignment probability under each view, and sharing information among views. In this article, we propose an approximate Bayes approach — treating the similarity matrices generated over the views as rough first-stage estimates for the co-assignment probabilities; in its Kullback-Leibler neighborhood, we obtain a refined low-rank matrix, formed by the pairwise product of simplex coordinates. Interestingly, each simplex coordinate directly encodes the cluster assignment uncertainty. For multi-view clustering, we let each view draw a parameterization from a few candidates, leading to dimension reduction. With high model flexibility, the estimation can be efficiently carried out as a continuous optimization problem, hence enjoys gradient-based computation. The theory establishes the connection of this model to a random partition distribution under multiple views. Compared to single-view clustering approaches, substantially more interpretable results are obtained when clustering brains from a human traumatic brain injury study, using high-dimensional gene expression data.

Keywords: Co-regularized Clustering, Consensus, PAC-Bayes, Random Cluster Graph, Variable Selection

1. Introduction

High dimensional data are becoming increasingly common in areas such as genomics, computer vision, neuroscience, etc. They are characterized by the ambient dimension substantially larger than the sample size n. When clustering such data, canonical solutions tend to focus on finding one particular clustering pattern. For example, one idea is to use variable selection method to identify a small subset of variables with large discriminability, then using them as input for clustering algorithms (Law et al., 2003; Tadesse et al., 2005; Hoff, 2006; Witten and Tibshirani, 2010); another idea is to reduce the dimension onto latent linear subspaces, often via a Gaussian mixture model with low-rank covariance structure (Ghahramani and Hinton, 1996); recent work extends this to the variational autoencoder for nonlinear dimension reduction (Dilokthanakul et al., 2016). These methods have been successful when there is only one clear clustering result in the data. However, as high dimension

data often contain multiple facets of the observations, it is more natural to consider more than one clustering patterns — that is, different subspaces of variables can correspond to distinct clustering results. As a result, focusing on one clustering pattern — or, 'single-view' is often inadequate.

There has been active literature motivated for 'multi-view clustering'. Since there are two distinct definitions of this concept, to be clear, we will focus on the one finding multiple clustering patterns, as opposed to the other aiming for one consensus based on multiple data sources. Within our scope, early work includes combining random projection and spectral clustering to obtain clusters on a randomly projected space, repeating this several times to produce multiple clustering patterns (Fern and Brodley, 2003); using regularization framework by running clustering algorithm in each view, while minimizing the cross-view divergence (Kumar et al., 2011; Joshi et al., 2016).

As clusters tend to overlap, there is often substantial uncertainty in clustering. A major interest is on the randomness of cluster assignment, characterized by a categorical distribution. In the canonical single-view setting, one typically relies on the Bayesian framework by assigning a model-based likelihood (Fraley and Raftery, 2002). This requires putting a parametric assumption on each within-cluster distribution, then estimating the posterior of cluster assignment via the Markov-chain Monte Carlo (MCMC) algorithm. Among the Bayesian multi-view clustering literature (Niu et al., 2010; Li and Shafto, 2011; Kirk et al., 2012; Lock and Dunson, 2013; Niu et al., 2013; Mo et al., 2017), Guan et al. (2010) and Niu et al. (2012) use the Indian Buffet Process to combine relevant variables into several groups, and in each group, they use a Gaussian mixture model to carry out clustering. These Bayesian models give a generative perspective for the multi-view data; however, there are two major challenges in practice: 1) assigning a within-cluster distribution is prone to mis-specifying the model, which leads to breakdown of the parameter estimation (Hennig et al., 2004) and uncontrolled growth in the number of clusters (Miller and Dunson, 2018); 2) the MCMC computation suffers from a critically slow convergence/mixing as the dimension grows, limiting its high dimension application. For the former, it was recently shown that modeling the pairwise divergence has much better robustness compared to the original data (Duan and Dunson, 2018); for the latter, in general, it has become increasingly popular to replace sampling with an optimization-based approximation for the posterior distribution (El Moselhy and Marzouk, 2012).

In this article, we are motivated for an approximate Bayes approach that allows for a direct estimation of the cluster assignment and co-assignment probabilities, within the scope of having several distinct clustering patterns. This is inspired by the resemblance between a similarity matrix and a cluster graph, hence the former can be considered as a noisy version of the latter. In the community detection literature, one often learns a low-rank representation for each data point as the latent position in Euclidean/Stiefel space (Hoff et al., 2002), then cluster the coordinates into communities (Handcock et al., 2007). Instead of going through two modeling stages, we put the latent coordinates directly on the probability simplex, describing the probabilities for cluster assignment and allowing gradient-descent optimization; in the meantime, each co-assignment matrix is a random draw out of only a few candidate parameterizations, leading to dimension reduction and information sharing among views. The software is provided on `https://github.com/leoduan/LatentSimplexPosition`.

2. Method

Let $y_i \in \mathcal{Y}$ be the data over $i = 1, \ldots, n$. Each y_i is a multi-dimensional vector, and we can separate its elements into groups of sub-vectors $(y_i^{(1)}, y_i^{(2)}, \ldots, y_i^{(V)})$. We will now call the subspace $\mathcal{Y}^{(v)}$ for $y_i^{(v)}$ as a 'view' of the data. To help explain the idea, we use a running example of simulated data over 3 views (each in $\mathbb{R}^2$). Figure 1 shows the scatter plots.

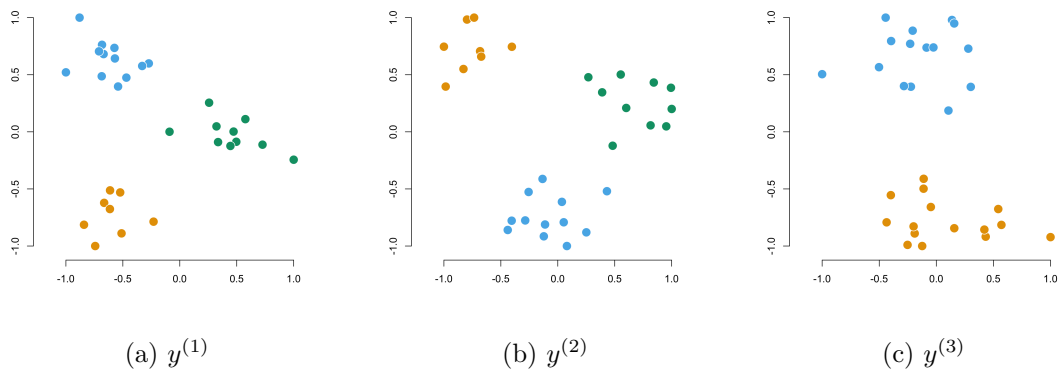(a) $y^{(1)}$ (b) $y^{(2)}$ (c) $y^{(3)}$

Figure 1: Simulated data under three views (each view is $\mathbb{R}^2$). The colors represent the true cluster assignments in each view.

Assuming the views are given, our goal is to obtain clustering for each view, in particular, a discrete label $c_i^{(v)} \in \{1, \ldots, g\}$ corresponding to the cluster assignment for each $y_i^{(v)}$. Equivalently, we can focus on the co-assignment for each pair of data

$$z_{i,j}^{(v)} = 1(c_i^{(v)} = c_j^{(v)}),$$

where $1(E)$ is the indicator function taking 1 if E is true, otherwise taking 0.

2.1. Latent Simplex Position Model

Treating $Z^{(v)} = \{z_{i,j}^{(v)}\}_{i,j}$ as an adjacency matrix, we can form a cluster graph: $G^{(v)} = (\mathcal{N}, \mathcal{E}^{(v)})$, with $\mathcal{N} = \{1, \ldots, n\}$ and $\mathcal{E}^{(v)} = \{e_{i,j}^{(v)} : z_{i,j}^{(v)} = 1\}$. In this graph, each cluster forms a complete sub-graph (all pairs of nodes within are connected), and the sub-graphs are disconnected. Figure 2 plots the cluster graph for each view.

3

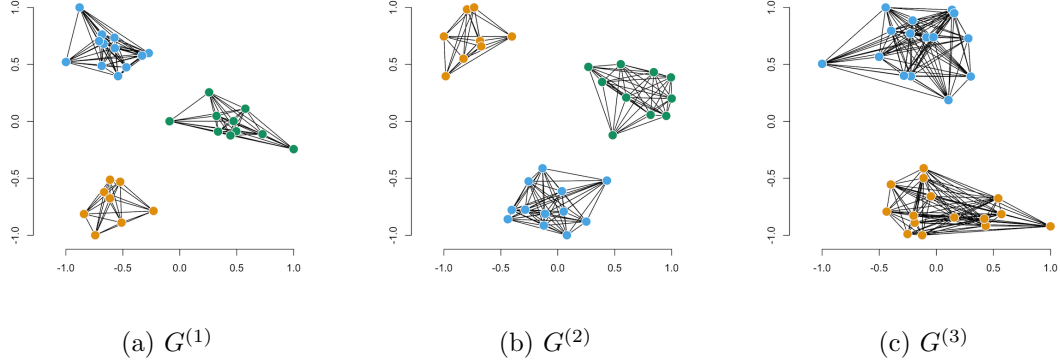(a) $G^{(1)}$ (b) $G^{(2)}$ (c) $G^{(3)}$

Figure 2: The oracle cluster graphs of the simulated data under three views: each of $G^{(1)}$ and $G^{(2)}$ has three disconnected sub-graphs; $G^{(3)}$ has only two sub-graphs.

In order to handle a large number of views, we consider the following generative process: assuming there are d ways to parameterize the distribution for $Z^{(v)}$, in each view, we draw one of d candidate parameterizations; then we proceed to draw the cluster assignments and form a cluster graph.

$$x^{(v)} \overset{iid}{\sim} \text{Categorical}(\lambda_1, \dots, \lambda_d),$$
$$c_i^{(v)} \mid x^{(v)} = l \overset{indep}{\sim} \text{Categorical}(w_{i,1}^{(l)}, \dots, w_{i,g}^{(l)}),$$
$$z_{i,j}^{(v)} = 1(c_i^{(v)} = c_j^{(v)}),$$

where $(\lambda_1, \dots, \lambda_d) \in \Delta^{d-1}$ and $(w_{i,1}^{(l)}, \dots, w_{i,g}^{(l)}) \in \Delta^{g-1}$; $\Delta^{g-1} = \{(\nu_1, \dots, \nu_g) : \nu_k \geq 0, \sum_{k=1}^{g} \nu_k = 1\}$ is the probability simplex. Equivalently, for $j < i$,

$$z_{i,j}^{(v)} \mid p_{i,j}^{(v)}, x^{(v)} \sim \text{Bernoulli}(p_{i,j}^{(v)}),$$
$$p_{i,j}^{(v)} = \sum_{l=1}^{d} 1(x^{(v)} = l) \sum_{k=1}^{g} w_{i,k}^{(l)} w_{j,k}^{(l)}. \tag{1}$$

In addition, we could further consider the data as generated from $y_i^{(v)} \mid c_i^{(v)} = k \sim \mathcal{F}_{v,k}$, with $\mathcal{F}_{v,k}$ a certain distribution. However, $\mathcal{F}_{v,k}$ is often unknown and challenging to estimate. Instead, we will focus on a pairwise transform as a surrogate for $y_i^{(v)}$'s.

Intuitively, if $y_i^{(v)}$ and $y_j^{(v)}$ are close to each other, it is more likely that they are from the same $\mathcal{F}_{v,k}$. In machine learning, we have the similarity score to quantify such a proximity:

$$s_{i,j}^{(v)} = \mathcal{K}(y_i^{(v)}, y_j^{(v)}), \tag{2}$$

where $\mathcal{K}$ is a positive semi-definite kernel that maps to $(0, 1)$. This can be taken as an approximate

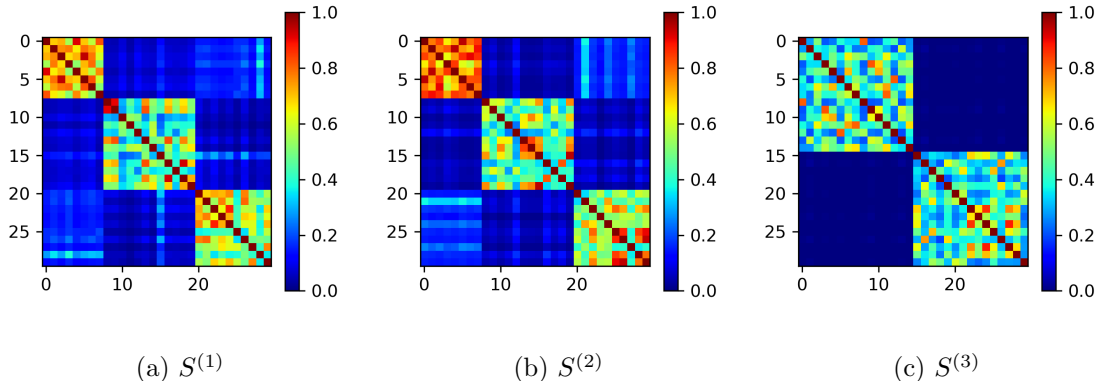$$s_{i,j}^{(v)} \approx \Pr(z_{i,j}^{(v)} = 1). \tag{3}$$

(a) $S^{(1)}$ (b) $S^{(2)}$ (c) $S^{(3)}$

Figure 3: Similarity as an approximation for $\Pr\big(z_{i,j}^{(v)} = 1 \mid y_i^{(v)}, y_j^{(v)}\big)$ under three views.

We will also use matrix notations $S^{(v)} = \{s_{i,j}^{(v)}\}_{i,j}$ an $n \times n$ matrix, $P^{(v)} = \{p_{i,j}^{(v)}\}_{i,j}$ an $n \times n$ matrix, and $W^{(l)} = \{w_{i,k}^{(l)}\}_{i,k}$ an $n \times g$ matrix. In this article, we use a popular similarity $s_{i,j}^{(v)} = \exp(-\|y_i^{(v)} - y_j^{(v)}\|/b_{i,j}^{(v)})$ with $b_{i,j}^{(v)} > 0$ the local bandwidth parameter formed by the row quantiles in $S^{(v)}$, according to Zelnik-Manor and Perona (2005). Figure 3 plots the similarity matrices computed from the simulated data.

In order to connect (3) with (1), we propose a generative model for $S^{(v)}$

$$s_{i,j}^{(v)} \mid p_{i,j}^{(v)} \sim H(p_{i,j}^{(v)}), \tag{4}$$

where H is a distribution such that $s_{i,j}^{(v)}$ is a noisy version of $p_{i,j}^{(v)}$. To choose its density h, because both $s_{i,j}^{(v)}$ and $p_{i,j}^{(v)}$ are Bernoulli probabilities, we use a pseudo-likelihood based on the Kullback-Leibler divergence (we will justify this choice in the theory section)

$$
\begin{aligned}
h(s_{i,j}^{(v)}; p_{i,j}^{(v)}) &\propto \exp\big[-KL\big(p_{i,j}^{(v)}\|s_{i,j}^{(v)}\big)\big], \\
KL\big(p_{i,j}^{(v)}\|s_{i,j}^{(v)}\big) &= p_{i,j}^{(v)} \log \frac{p_{i,j}^{(v)}}{s_{i,j}^{(v)}} + (1 - p_{i,j}^{(v)}) \log \frac{1 - p_{i,j}^{(v)}}{1 - s_{i,j}^{(v)}}.
\end{aligned}
\tag{5}
$$

Using (4), we can obtain the posterior distribution $\Pi(P^{(v)} \mid S^{(v)})$, as a surrogate for $\Pi(P^{(v)} \mid y_1^{(v)}, \ldots, y_n^{(v)})$. Although it is an approximation, a key benefit is that the posterior mode of $(P^{(v)} \mid S^{(v)})$ directly estimates the clustering uncertainty $\Pr(c_i^{(v)} = c_j^{(v)})$; and the mode of $(W^{(x^{(v)})} \mid S^{(v)})$ estimates $\Pr(c_i^{(v)} = k)$. And note that

$$S^{(v)} \approx W^{(x^{(v)})} W^{(x^{(v)})\mathrm{T}} = P^{(v)},$$

where the right hand side has the rank less or equal to $(g-1)$, providing a low-rank smoothing. Therefore, the optimization for the mode is close to a simple matrix factorization, hence it is computationally more efficient, compared to the costly MCMC algorithm.

5

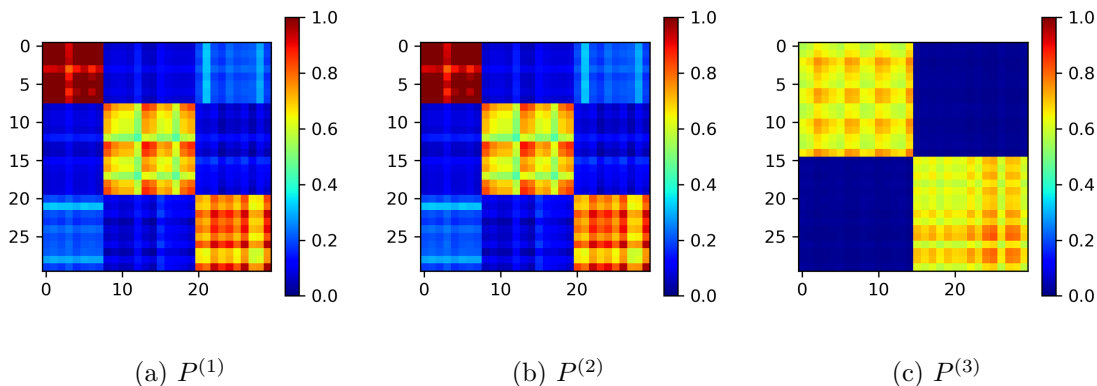(a) $P^{(1)}$ (b) $P^{(2)}$ (c) $P^{(3)}$

Figure 4: The matrix $P^{(v)}$ provides a low-rank smoothing for $S^{(v)}$: each pixel of the heatmap represents the marginal probability $\Pr(z_{i,j}^{(v)} = 1)$. The first two matrices have the same parameterization due to the same $x^{(1)} = x^{(2)}$.

Lastly, if we consider a 'similarity graph' with $S^{(v)}$ as its adjacency matrix, then the ith row of $W^{(x^{(v)})}$

$$(w_{i,1}^{(x^{(v)})}, \ldots, w_{i,g}^{(x^{(v)})}) \in \Delta^{g-1}$$

is a latent position for the node i. Therefore, our model is a special case of the latent position model (Hoff et al., 2002); and we name it as the latent simplex position (LSP) model.

2.2. Regularization in Overfitted Model

Often we do not know the minimally needed number of clusters g_0 (or the 'truth'), instead, we assign an overfitted model with an over-specified $g \gg g_0$. It is useful to consider regularization: suppose that the cluster k is redundant, we can use some regularization term to force the kth column in $W^{(l)}$ to be close to zero, $w_{i,k}^{(l)} = \Pr(c_i^{(l)} = k) \approx 0$ for $i = 1, \ldots, n$. Similarly, we want to over-specify d and use regularization to force some redundant $\lambda_l \approx 0$.
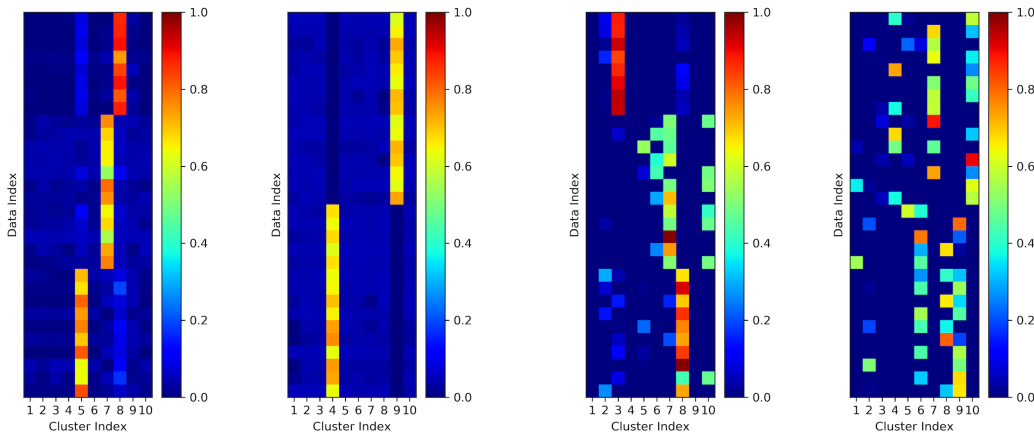
We use a regularized loss function

$$RegLoss = KL(P\|S) + n \sum_{l=1}^{d} R(W^{(l)}) + \sum_{l=1}^{d} T(\lambda_l), \tag{6}$$

with $KL(P\|S) = \sum_{v=1}^{V} \sum_{j<i} KL(p_{i,j}^{(v)}\|s_{i,j}^{(v)})$ and two regularization terms $R(W^{(l)})$ and $T(\lambda_l)$.

Inspired by the group lasso variable selection in regression (Yuan and Lin, 2006; Meier et al., 2008), we use a group regularization to induce *column* sparsity in each $W^{(l)}$.

$$R(W^{(l)}) = \sum_{k=1}^{g} \sqrt{\sum_{i=1}^{n} \left(\log \frac{w_{i,k}^{(l)}}{\epsilon}\right)_{+}^{2}}.$$

6

Since each $w_{i,k}^{(l)} \leq 1$ and is already on a small scale, we first divide it by a close-to-zero and positive ϵ, and take a logarithmic transform; those $w_{i,k}^{(l)}$ above and away from $\epsilon \approx 0$ are penalized using $(x)_+ = \max(x,0)$; those below ϵ are not penalized because they are negligibly small (in this article we use $\epsilon = 10^{-3}$ as the threshold). To achieve a group regularization, a 2-norm is used on each column; and $R(W^{(l)})$ is multiplied with n so that it grows in the same order as $KL(P\|S)$ (which contains $O(n^2)$ terms). This regularization shows good empirical performance, as it recovers the true number of clusters in all of our simulation studies. Figure 5(a) shows the estimated $W^{(l)}$ in the previous simulation.

(a) Estimated $W^{(l)}$ matrix using group regularization.

(b) Estimated $W^{(l)}$ matrix using Dirichlet prior $\mathrm{Dir}(0.1,\ldots,0.1)$ on each row.

Figure 5: Group regularization on $W^{(l)}$ in an overfitted clustering model with $g = 10$.

We also consider some classical shrinkage prior on the simplex, such as the Dirichlet prior with the concentration parameter smaller than 1. However, a drawback is that the shrinkage is applied independently on multiple simplex vectors, and there is no control on the *joint* distribution of all rows of $W^{(l)}$. As a result, many spurious small clusters appear even though each row is sparse [Figure 5(b)]. We expect more advanced models such as hierarchical Dirichlet mixture (Teh et al., 2005; Zhou, 2014; Ohama et al., 2017) might solve this problem as well; we use the group regularization for computational convenience.

On the second regularization $T(\lambda_l)$, since there is only one simplex vector $(\lambda_1, \ldots, \lambda_d)$, it is easier to handle compared to $W^{(l)}$. We apply Dirichlet prior $\mathrm{Dir}(\alpha_\lambda, \ldots, \alpha_\lambda)$, equivalently,

$$T(\lambda_l) = (1 - \alpha_\lambda)\log(\lambda_l),$$

and we use $\alpha_\lambda = 1/d$ as a common choice for approximating the infinite mixture (Rasmussen, 2000).

With those two regularizations, we can choose d and g as large as possible (if the ground truth of the cluster number is not known). For example, we can choose the maximal d and g according to the computing budget, such as the memory limit.

2.3. Producing Consensus via Combining Views

In our model, the views with different $x^{(v)}$'s have distinct clustering patterns; on the other hand, sometimes there is still an interest to combine the information from those views together to form a 'consensus'.

(a) The first two views (each view is in $\mathbb{R}$), each corresponds to different clustering structure.

(b) Two noisy views, neither contains any clustering structure.

(c) Estimated $P^{(1)}$ based on the first view.

(d) Estimated $P^{(2)}$ based on the second view.

(e) Consensus formed by combining the first two views, while ignoring the other ones.
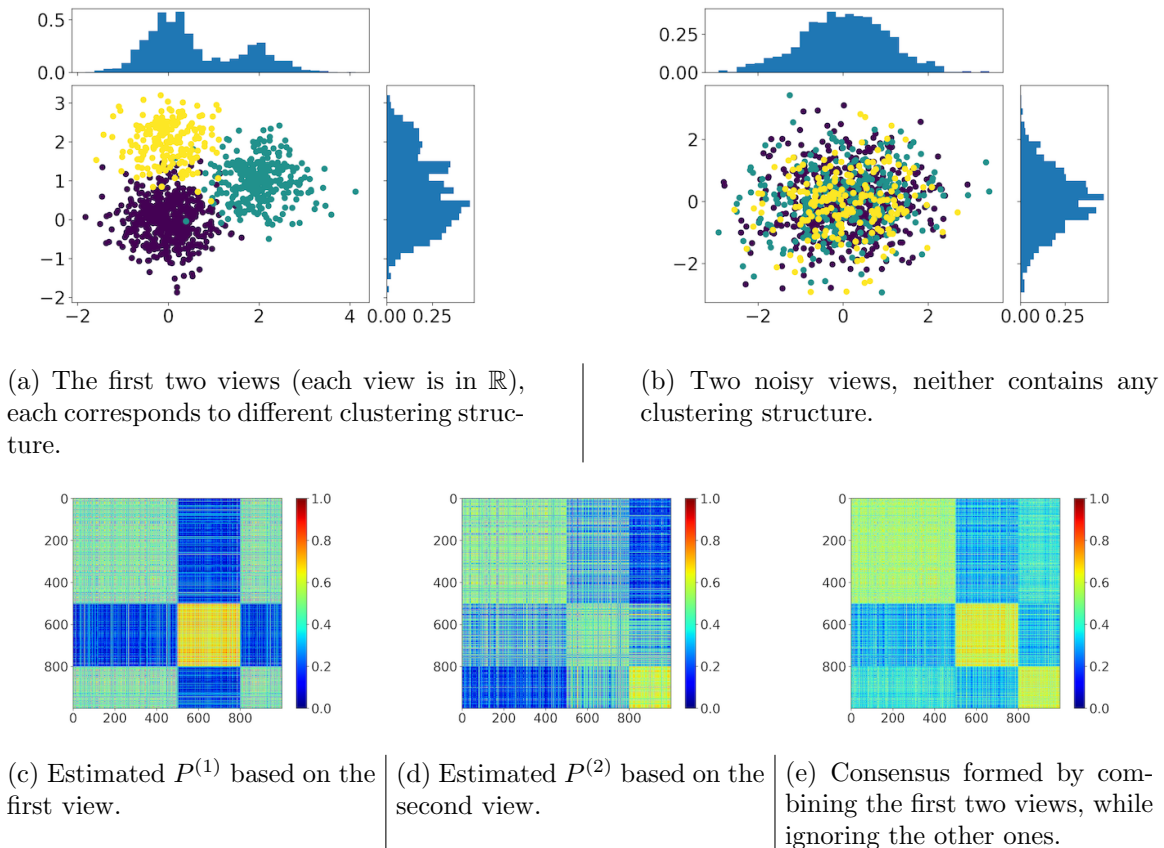
Figure 6: Illustration on how to combine different views to form a consensus (each view in $\mathbb{R}$: the estimated $P^{(v)}$'s from the first two views (two marginal histograms of panel a) are taken, while the ones with no clustering structures (panel b) are excluded, producing an average as the consensus co-assignment probability matrix (panel e).

Since a convex combination of $P^{(v)}$'s is still positive semi-definite, we consider the weighted average as the 'consensus' co-assignment probability

$$\bar{P} = \sum_{v=1}^{V} \frac{u^{(v)} P^{(v)}}{\sum_{v=1}^{V} u^{(v)}}. \tag{7}$$

with the weight $u^{(v)} \geq 0$, and $\sum_{v=1}^{V} u^{(v)} > 0$. In this article, we take a simple strategy for choosing $u^{(v)}$: for a view, if its most probable $\hat{x}^{(v)}$ (which can be computed from (10) in Section 3) is equal to l and $W^{(l)}$ is a matrix with one column filled by 1's and others by 0's,

then there is no clustering pattern in this view; hence, we set its $u^{(v)}$ to zero. For the other views, we set equal weights $u^{(v)} = 1$.

This consensus is closely related to the variable-selection based clustering (Witten and Tibshirani, 2010). Indeed, the latter is equivalent to directly using $d = 2$ latent parameterizations (one of them having no clustering).

To illustrate the consensus, we generate data with 10 views, each in $\mathbb{R}$: each of the first two views contains more than one clusters [Figure 6(a), generated from $N(0,1), N(2,1)$ and $N(0,1), N(1,1), N(2,1)$], respectively; while the views $3-10$ have no clustering structure [Figure 6(b), all generated from $N(0,1)$]. In the estimation of the LSP model, $P^{(3)}, \ldots, P^{(10)}$ are linked to an $W^{(l)}$ without clustering structure. Using (7), the consensus combines $P^{(1)}$ and $P^{(2)}$ and shows that there are three clusters.

3. Computation

As $x^{(v)}$ is a latent variable, we use the Expectation-Maximization (EM) algorithm. Letting $\eta_l^{(v)} = \mathbb{E}1(x^{(v)} = l)$, in the E step, we update

$$\eta_l^{(v)} = \frac{\lambda_l \exp\left[-\sum_{j<i} KL\left(p_{i,j}^{*(l)} \| s_{i,j}^{(v)}\right)\right]}{\sum_{l'=1}^{d} \lambda_{l'} \exp\left[-\sum_{j<i} KL\left(p_{i,j}^{*(l')} \| s_{i,j}^{(v)}\right)\right]}, \tag{8}$$

where $p_{i,j}^{*(l)} = (W^{(l)}W^{(l)\mathrm{T}})_{i,j}$.

In the M step, we minimize the expected loss function over the parameter $W^{(l)}$, using the ADAM gradient descent algorithm (Kingma and Ba, 2014):

$$\mathbb{E}_{x^{(1)},\ldots,x^{(V)}} RegLoss = \sum_{v=1}^{V} \sum_{l=1}^{d} \eta_l^{(v)} \sum_{j<i} \left[p_{i,j}^{*(l)} \log \frac{p_{i,j}^{*(l)}}{s_{i,j}^{(v)}} + (1 - p_{i,j}^{*(l)}) \log \frac{1 - p_{i,j}^{*(l)}}{1 - s_{i,j}^{(v)}} \right]$$
$$+ n \sum_{l=1}^{d} R\left(W^{(l)}\right) + \sum_{l=1}^{d} T(\lambda_l), \tag{9}$$

and set λ_l to its mode

$$\hat{\lambda}_l \propto \max(0, \alpha_\lambda - 1 + \sum_{v=1}^{V} \eta_l^{(v)}) \text{ for } l = 1, \ldots, d \text{ such that } \sum_{l=1}^{d} \hat{\lambda}_l = 1.$$

The vector $(\eta_1^{(v)}, \ldots, \eta_d^{(v)})$ gives the scores on how likely the vth view is generated from each parameterization. As a point estimate for $x^{(v)}$, the most probable one is

$$\hat{x}^{(v)} = \arg\max_{l \in \{1,\ldots,d\}} \eta_l^{(v)}. \tag{10}$$

Similarly, we have the point-wise optimal $\tilde{c}_i^{(v)} = \arg\max_k w_{i,k}^{(\hat{x}^{(v)})}$. We can use those two quantities to determine the effective numbers of parameterizations and clusters: $\hat{d}$ as the number of unique $\hat{x}^{(v)}$'s for $v = 1, \ldots, V$, and $\hat{g}^{(v)}$ as the number of unique $\tilde{c}_i^{(v)}$'s for $i = 1, \ldots, n$.

On the other hand, as shown by Wade and Ghahramani (2018), the point-wise clustering estimate $\tilde{c}_i^{(v)}$ is not necessarily optimal for the overall clustering. Instead, we use the estimated $P^{(\hat{x}^{(v)})}$ and $\hat{g}^{(v)}$ as the input matrix and cluster number in the spectral clustering, to produce a joint point estimate $(\hat{c}_1^{(v)}, \ldots, \hat{c}_n^{(v)})$. As shown in the data experiments, this results in much more accurate clustering than using $S^{(v)}$ directly in the spectral clustering.

3.1. Scalability and Initialization

In our optimization algorithm, the M step is the most computationally intensive one, since we need several gradient descents in each EM iteration. Fortunately, we can substantially reduce its computing complexity and make it scalable to a very large V.

Before using gradient descent, we first compute two matrices, with their (i,j)th elements

$$\kappa_{i,j} = -\sum_{v=1}^{V} \eta_l^{(v)} \log \frac{s_{i,j}^{(v)}}{1 - s_{i,j}^{(v)}}, \qquad \gamma_{i,j} = \sum_{v=1}^{V} \eta_l^{(v)}.$$

By changing the order of summation, the expected loss in (9) becomes

$$\sum_{l=1}^{d} \sum_{j<i} \left\{ \kappa_{i,j} p_{i,j}^{*(l)} + \gamma_{i,j} \left[p_{i,j}^{*(l)} \log \frac{p_{i,j}^{*(l)}}{1 - p_{i,j}^{*(l)}} + \log(1 - p_{i,j}^{*(l)}) \right] \right\}$$

$$+ n \sum_{l=1}^{d} R(W^{(l)}) + \sum_{l=1}^{d} T(\lambda_l) + C,$$

where $C = -\sum_{v=1}^{V} \sum_{l=1}^{d} \eta_l^{(v)} \sum_{j<i} \log(1 - s_{i,j}^{(v)})$ is a constant free from $W^{(l)}$; hence it can be ignored during the M step. Notice that this alternative form reduces the computational complexity from $O(Vn^2 d)$ to $O(n^2 d)$ for each gradient descent.

Similar to the conventional mixture models, when starting the EM algorithm, it is crucial to have good initial values for the parameters. Therefore, we now develop an initialization strategy. Note that if ignoring the low-rank constraint in $p_{i,j}^{*(l)}$'s, the loss for those $x^{(v)} = l$, $\sum_{v:x^{(v)}=l} KL(p_{i,j}^{*(l)} \| s_{i,j}^{(v)})$ is minimized at

$$\log \frac{\widehat{p}_{i,j}^{*(l)}}{1 - \widehat{p}_{i,j}^{*(l)}} = \left[\sum_{v} 1(x^{(v)} = l) \right]^{-1} \sum_{v:x^{(v)}=l} \log \frac{s_{i,j}^{(v)}}{1 - s_{i,j}^{(v)}},$$

which is the group mean of the log odds. Therefore, we first use a simple K-means (with K set to d) on V matrices $\{\log[s_{i,j}^{(v)}/(1 - s^{(v)})]\}_{i,j}$, putting them into d groups and treating the K-means labels as the initial estimates for $\hat{x}^{(v)}$'s. Then setting $\eta_l^{(v)} = 1(\hat{x}^{(v)} = l)$ and $\lambda_l = 1/d$, we run the M step to obtain the initial values for $W^{(l)}$'s.

We track the expected loss $\mathbb{E}_{\{x^{(v)}\}} RegLoss$ for convergence, and consider the algorithm as converged if the decrease in the expected loss is less than one percent over 100 iterations. Since the loss function is non-convex, we run the algorithm multiple times under random initializations with K-means++(Arthur and Vassilvitskii, 2007). We choose the ones with the lowest loss as the final estimates.

4. Theory

In this section, we provide a theoretical justification for the LSP model, by establishing a link to the random partition distribution using pairwise information/distances (Blei and Frazier, 2011; Dahl et al., 2017).

We first briefly review the idea of random partition distribution. Given a matrix of co-assignment probabilities $P = \{p_{i,j}\}_{i,j}$, with $p_{i,j} \in (0,1)$, we can sample a cluster graph. Starting with an initial set containing one index $\mathcal{N}_0 = \{i\}$ (with i randomly chosen), each time, we draw another j randomly from $\mathcal{N} \setminus \mathcal{N}_0$ and assign

$$
\begin{aligned}
\hat{z}_{i,j}^{(v)} &= 1, & \text{if } \forall \hat{z}_{i,i'}^{(v)} = 1, \hat{z}_{j,i'}^{(v)} = 1, \\
\hat{z}_{i,j}^{(v)} &= 0, & \text{if } \forall \hat{z}_{i,i'}^{(v)} = 0, \hat{z}_{j,i'}^{(v)} = 1, \\
\hat{z}_{i,j}^{(v)} &= 0, & \text{if } \forall \hat{z}_{i,i'}^{(v)} = 1, \hat{z}_{j,i'}^{(v)} = 0, \\
\hat{z}_{i,j}^{(v)} &\sim \text{Bernoulli}(p_{i,j}), & \text{otherwise,}
\end{aligned}
\tag{11}
$$

for $i, i' \in \mathcal{N}_0$ and $i \neq i'$. That is, sampling new Bernoulli if it is not determined by the pairwise constraints of a cluster graph. After updating $\hat{z}_{i,j}^{(v)}$, we add j into $\mathcal{N}_0$ and go to the next loop. Eventually, this forms an $n \times n$ binary matrix $\hat{Z}^{(v)}$. These procedures are associated with a random partition distribution, that we denote by $\hat{Z}^{(v)} \sim \phi$.

Now under the multi-view setting, let us focus on the sub-group of views with the same parameterization (that is, having equal $x^{(v)}$'s). Without loss of generality, we assume they have indices $v = 1, \ldots, M$.

For each graph, we assume that there is a ground truth cluster graph $Z_0^{(v)}$. If it is known, we can compare it with the sampled $\hat{Z}^{(v)}$ and compute a loss function in $[0,1]$ (such as the 1-minus normalized mutual information)

$$
loss\left(Z_0^{(v)}, \hat{Z}^{(v)}\right).
$$

To assess the quality of ϕ in recovering the ground truth, theoretically, we would hope to take average over infinite samples of $\hat{Z}^{(v)} \sim \phi$, and then take average over M views,

$$
R(\Pi_M, \phi) = \frac{1}{M} \sum_{v=1}^{M} \mathbb{E}_{\hat{Z}^{(v)} \sim \phi} loss\left(Z_0^{(v)}, \hat{Z}^{(v)}\right),
$$

where Π_M denotes the empirical distribution for M views. Taking one step further, suppose $Z_0^{(v)} \overset{iid}{\sim} \Pi_0$, we can define the generalization risk

$$
R(\Pi_0, \phi) = \mathbb{E}_{Z_0^{(v)} \sim \Pi_0} \mathbb{E}_{\hat{Z}^{(v)} \sim \phi} loss\left(Z_0^{(v)}, \hat{Z}^{(v)}\right).
$$

Since Π_0 is not fully known, we cannot directly minimize $R(\Pi_0, \phi)$; however, we can use some assumptions on Π_0 (such as having at most g clusters) and use $R(\Pi_M, \phi)$ as an approximate. Therefore, it is imperative to optimize approximation. We have the following bound based on the Probably Approximately Correct (PAC)-Bayes theory (Seldin and Tishby, 2010; Guedj, 2019).

Theorem 1 *For $M \in [2, \infty)$, if $(1/M)\sum_{v=1}^{M} loss\left(Z_0^{(v)}, \hat{Z}^{(v)}\right) \in (0,1)$, with probability greater than $1 - \delta$ based on Π_0 (the true distribution for $Z_0^{(v)}$),*

$$KL\big[R(\Pi_0, \phi) \,||\, R(\Pi_M, \phi)\big]$$
$$\leq (1/M)\bigg\{ \sum_{v=1}^{M}\sum_{j<i} KL(p_{i,j}||s_{i,j}^{(v)})/M + \log\left[\exp(\frac{1}{12M})\sqrt{\frac{\pi M}{2}} + 2\right] - \log \delta \bigg\},$$

where $p_{i,j}$ is absolutely continuous with respect to $s_{i,j}^{(v)}$.

Combining the KL terms on the right hand side over the different parameterizations, we obtain the $KL(P\|S)$ function in our LSP model.

Therefore, estimating the LSP model can be considered as a procedure to optimize the multi-view random partition distribution ϕ, in terms of improving the finite-view performance. Specifically, minimizing the difference between $R(\Pi_0, \phi)$ and $R(\Pi_M, \phi)$ reduces the chance of overfitting, and is known as 'reducing generalization error' in the PAC-Bayes literature (Seldin and Tishby, 2010). Compared to the canonical random partition distribution, we also gain in the computation since we do not have to sample $\hat{Z}^{(v)}$.

5. Data Experiments

5.1. Single View Simulations

Since most clustering approaches are based on a single view, we first compare our model with them using simulations. For a clear visualization, we generate data from the two-component mixture distribution in a single view $y_i \in \mathbb{R}^2$, with $n = 400$. Figure 7(a-f) plots the generated data under 6 different settings.
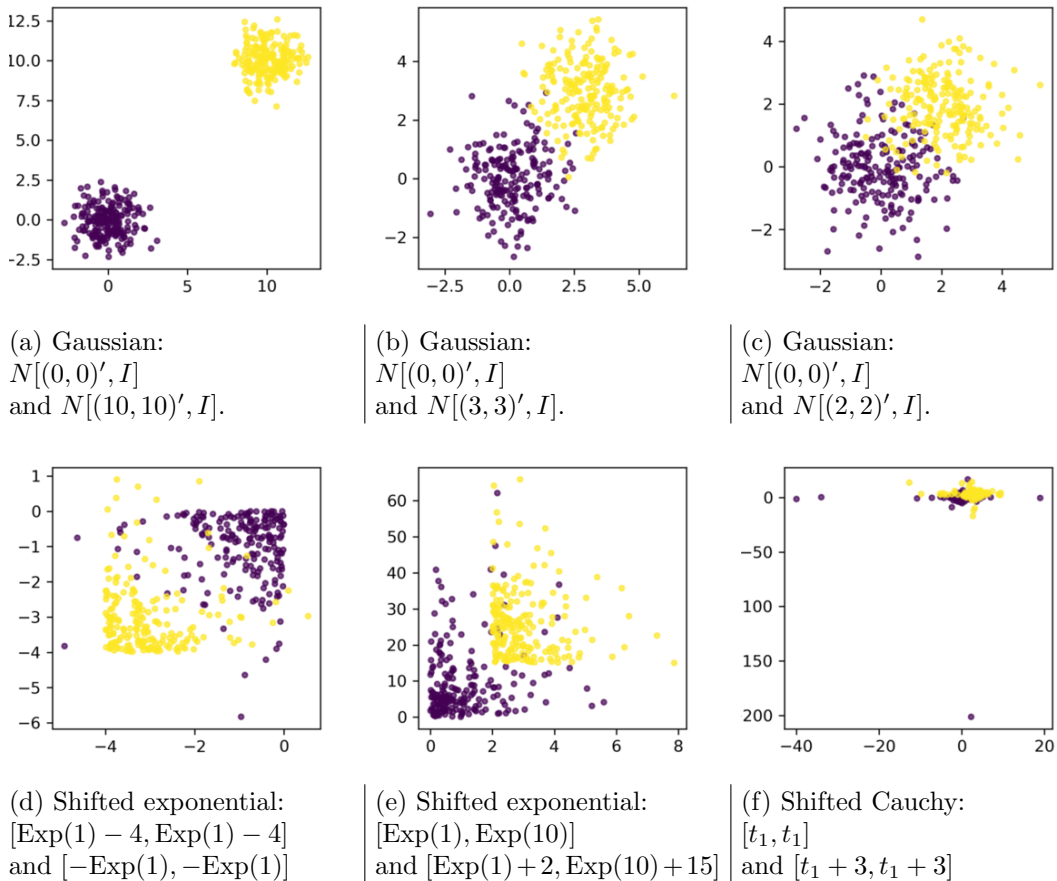
(a) Gaussian:
$N[(0,0)', I]$
and $N[(10, 10)', I]$.

(b) Gaussian:
$N[(0,0)', I]$
and $N[(3, 3)', I]$.

(c) Gaussian:
$N[(0,0)', I]$
and $N[(2, 2)', I]$.

(d) Shifted exponential:
$[\text{Exp}(1) - 4, \text{Exp}(1) - 4]$
and $[-\text{Exp}(1), -\text{Exp}(1)]$

(e) Shifted exponential:
$[\text{Exp}(1), \text{Exp}(10)]$
and $[\text{Exp}(1) + 2, \text{Exp}(10) + 15]$

(f) Shifted Cauchy:
$[t_1, t_1]$
and $[t_1 + 3, t_1 + 3]$

Figure 7: Generated data in the single view clustering experiments (the view is in $\mathbb{R}^2$). In each setting, two clusters are simulated, with the color representing the ground truth labels.

For comparison, we test 9 other clustering algorithms, provided by the Scikit-Learn Cluster package. We use the Normalized Mutual Information (NMI) as a benchmark score: for two discrete vectors of the same length n, $\text{NMI}(x, y) = 2\{H(x) + H(y) - H[(x, y)]\}/[H(x) + H(y)]$, with $H(w) = -\sum_k \hat{p}_k \log \hat{p}_k$, $\hat{p}_k = \sum_i 1(w_i = w_k^*)/n$ and w_k^* the kth unique value among $(w_1, \ldots, w_n)$. NMI measures the accuracy of the estimated cluster labels with respect to the ground truth labels and is invariant to label-switching. The result is listed in Table 1. In the Gaussian cases (a-c), the Gaussian mixture as the true model has the best performance. When the symmetric and Gaussian-tail assumptions are violated (d-f), more recent methods such as spectral clustering start to show their advantage. The performance of the LSP model (with $g = 2$ and $d = 1$) is very close to the spectral clustering; while in (f) with the heavy-tailed distribution creating many 'outliers', the spectral clustering fails completely due to the large noise in the raw similarity matrix, and LSP does not have this issue thanks to the low-rank smoothing.

13

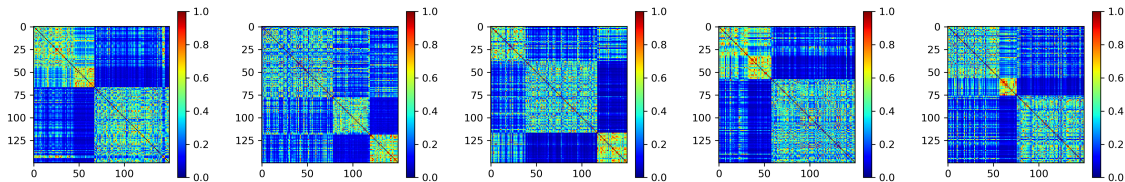|	(a)	(b)	(c)	(d)	(e)	(f)
K-means	**1**	0.89	0.68	0.56	0.25	0
Affinity propagation	0.57	0.37	0.30	0.31	0.30	0.23
Mean-shift	**1**	0.89	**0.70**	0.56	0.40	0.05
Agglomerative clustering	**1**	0.77	0.62	0.54	0.17	0
DBSCAN	0.92	0	0.01	0	0.43	0
OPTICS	0.23	0.20	0.17	0.21	0.19	0.17
Gaussian mixture	**1**	**0.91**	**0.70**	0.58	0.37	0
Birch	0.99	0.70	0.49	0.41	0.27	0.01
Spectral clustering	**1**	0.89	**0.70**	**0.59**	0.55	0.0
LSP	**1**	0.89	**0.70**	**0.59**	**0.58**	**0.42**

Table 1: Normalized mutual information showing the accuracy of single view clustering, using the data simulated in Figure 7.

For uncertainty quantification, we compute the co-assignment probability using the ground truth distribution and compare it with the estimated P from the LSP model. The raw S and estimated P are provided in the appendix. The median absolute deviations (MAD) between P and the oracle are 0.03, 0.07, 0.08, 0.04, 0.05 and 0.06. In addition, to assess the limitation of the LSP model (and similarity-based algorithms in general), we modify (c) and make the two clusters closer. When two clusters are generated from $N[(0,0)', I]$ and $N[(1,1)', I]$, the LSP model could only discover one large cluster. This is not surprising since heavily overlapped clusters can be alternatively taken as one cluster; in such cases, mixture models with stronger assumptions would work better, such as the ones with a parametric density for each cluster.
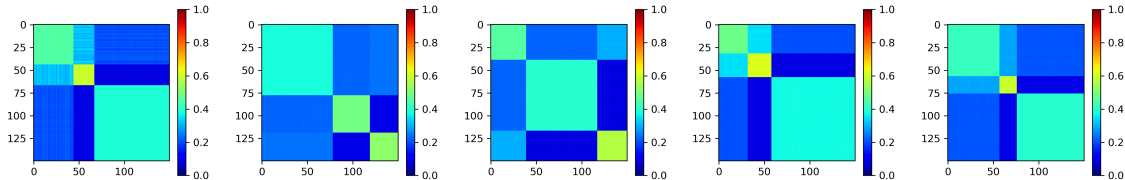
5.2. Multi View Experiments
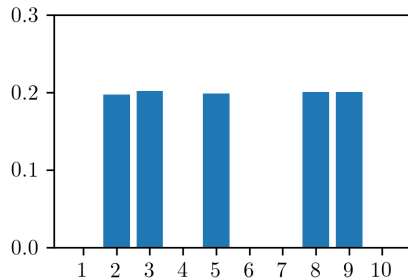
5.2.1. SCALING TO A LARGE NUMBER OF VIEWS

We first use a simulation to assess the multi-view clustering performance, under a large $V \gg n$. We use $V = 50,000$ views, where each view has $n = 150$. To produce distinct clustering patterns, we simulate 5 different $W^{(l)}$'s, with each row generated from a Dirichlet distribution in a 3-element simplex; for a better visualization, the rows are re-ordered and grouped by the index $\arg_k \max w_{i,k}^{(l)}$ (the most probable cluster). Then in each view, we randomly choose one of the five matrices (denoting the choice by $x_0^{(v)} \in \{1, \dots, 5\}$) as the parameterization, and sample the cluster labels $c_i^{(v)} \mid x^{(v)} = l \sim \text{Categorical}(w_{i,1}^{(l)}, w_{i,2}^{(l)}, w_{i,3}^{(l)})$. Lastly, we generate each data point $y_i^{(v)} \mid c_i^{(v)} = k \sim N(\mu_k, I_2)$, with $\mu_1 = (0,0)'$, $\mu_2 = (2,2)'$ and $\mu_3 = (-2,-2)'$, so that there is moderate overlap between clusters. Figure 8(a) plots five similarity matrices representative for those distinct patterns.

14

(a) Five similarity matrices corresponding to different clustering patterns.

(b) Estimated parameterization $W^{(l)}W^{(l)\mathrm{T}}$.

(c) Estimated λ_l for the 10 latent parameterizations.

Figure 8: Simulation shows the LSP model recovers the latent parameterizations in multi-view clustering.

When fitting the LSP model, we use $d = g = 10$. To examine the quality of initialization, we compare the initialized $\hat{x}^{(v)}$ and the oracle $x_0^{(v)}$: they show a high NMI at 0.83, indicating that that the K-means on the log-odds of $S^{(v)}$'s gives a very good initialization. After running the EM algorithm, we do see 5 of the 10 estimated λ_l's are shrunk to near zero [Figure 8(c)], and the estimated $W^{(l)}W^{(l)\mathrm{T}}$'s with non-trivial λ_l's indeed recover the five patterns [Figure 8(b)]. For these high dimensional data, the algorithm takes about 10 minutes to finish on a CUDA GPU with 11Gb of memory.

5.2.2. CLUSTERING UCI HAND WRITTEN DIGITS

Since the software is not readily available for most of the existing multi-view clustering methods, we use a dataset that was previously used for benchmark and reported by Kumar et al. (2011). The dataset is the UCI Dutch utility maps handwritten digits data (https://archive.ics.uci.edu/ml/datasets/Multiple+Features), and does not require specific data processing; hence it can provide a fair comparison. The data have six views: (1) 76 Fourier coefficients of the character shapes, (2) 216 profile correlations, (3) 64 Karhunen-Love coefficients, (4) 240-pixel averages in 2×3 windows, (5) Zernike moments and (6) 6

morphological features. For each digit, there are 200 samples; the NMI is calculated as an evaluating criterion.

We compare our model with two other methods: using single-view spectral clustering independently in each view (SV-SC), and co-regularized spectral clustering (C-SC) (Kumar et al., 2011). When fitting the LSP model, we use $d = V$ as its possible max value, and $g = 10$ as the known ground truth. The model converges to 2 effective parameterizations: $\hat{x}^{(v)} = 1$ for $v = 1, \dots, 5$ and $\hat{x}^{(6)} = 2$.

Single View	1	2	3	4	5	6
SV-SC	0.571	0.618	0.646	0.635	0.523	**0.474**
LSP	**0.697**	**0.706**	**0.697**	**0.705**	**0.705**	**0.474**
Combining Multiple Views						
SV-SC (feature concat)			0.619			
C-SC			**0.768**			
LSP consensus			0.742			

Table 2: Normalized mutual information for clustering UCI hand-written digits. Single view spectral clustering (SV-SC) and co-regularized spectral clustering (C-SC) are included for comparison. The NMIs for C-SC are obtained from Kumar et al. (2011).

(a) $P^{(v)}$ for the 1st to the 5th views.
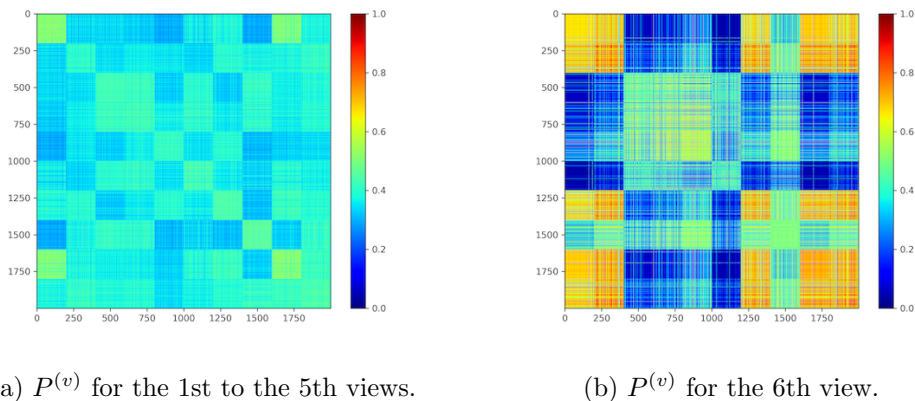
(b) $P^{(v)}$ for the 6th view.

Figure 9: Uncertainty for clustering hand written digit data, shown in the two estimated co-assignment probability matrices. The rows and columns are ordered according to the true labels of the digits.

We first compute the point estimates $\hat{c}_i^{(v)}$ under each view. Table 2 shows that, compared to SV-SC, our model produces higher NMI in almost every view (except for the 6th view). This is likely due to the sharing of information among the first few views in the LSP model. Then we combine views to produce a consensus. LSP has a better performance than using SV-SC on the concatenated features from all views, while it is slightly worse compared to C-SC. On the other hand, LSP has a unique advantage in the uncertainty quantification for each view. As shown in Figure 9, in the first parameterization, the main source of

uncertainty is due to the overlap between the 1st and the 9th clusters; whereas the second parameterization has larger overlap among clusters $1, 2, 7, 9$ and 10.
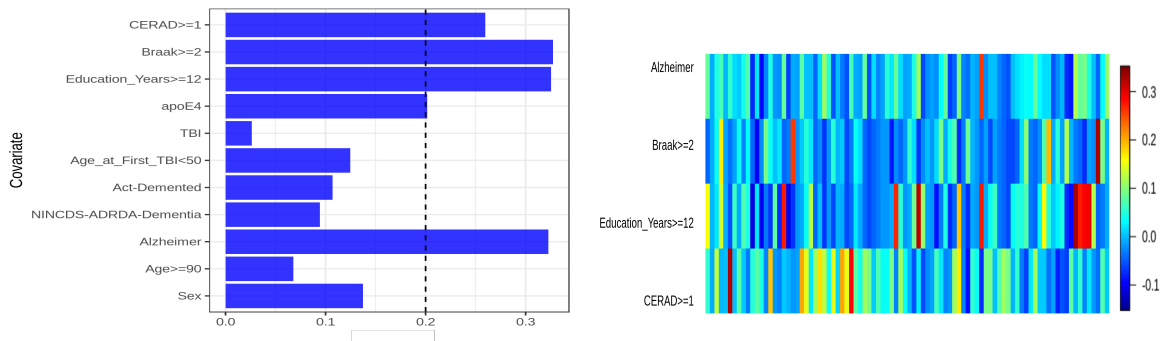
5.2.3. CLUSTERING BRAINS VIA RNA-SEQUENCING DATA

We now consider a scientific application with the RNA Sequencing data originated from the human aging, dementia, and traumatic brain injury (TBI) study. The data are obtained from the Allen Institute for Brain Science (Miller et al., 2017) (https://aging.brain-map.org/download/index), and the hippocampus region is chosen for its important role in aging-related disease. Among the 107 brains, there are $n = 94$ containing gene expression data in the hippocampus. The age of the subjects at death has an average of 90 and a standard deviation of 6. There are $50,281$ genes, each with normalized gene-level FPKM values. Since most of the genes contain very little discriminability, a screening step is first carried out: the genes are ranked by their standard deviation divided by the median, with the top $V = 1,000$ chosen for the downstream modeling.

This experiment treats each gene as a view, and clusters the 94 brains using the gene expression. The LSP model was initialized at $d = 30$ and $g = 30$. It converges to 10 effective latent parameterizations and at most 4 non-trivial clusters.

For validation, the multi-view results are compared against 11 observed clinical covariates — such as sex, whether had TBI before, dementia evaluation scores, etc. For each clinical covariate, the NMIs are computed by comparing it against the estimated $\hat{c}_i^{(v)}$ for $v = 1, \ldots, 1000$. To see if there is a possible link between the gene(s) and clinical covariates, for each covariate, we take the maximum of 1000 NMIs (MaxNMI) and plot it in Figure 10(a). As this involves multiple comparisons, to show the findings are unlikely to be false positives, we consider two additional baseline MaxNMIs: 1) we randomly draw 1000 Bernoulli random vectors, each of length $n = 94$, and compute the MaxNMI to each covariate. We repeat the simulation with different Bernoulli probabilities (ranging from 0.1 to 0.9), the largest MaxNMI is 0.15; 2) it was previously reported in a meta-analysis study (Tan et al., 2016), that the difference in the covariate sex has no clear effect on the hippocampus area, in our experiment, it has MaxNMI 0.13. Therefore, we choose MaxNMI≥ 0.2 as a cut-off. For clarification, these results are mainly exploratory due to the small n; more data are needed before making any statistical claims.

Among all the covariates, the CERAD score (measuring the progression of Alzheimer's disease), the Braak stage (measuring the progression of Parkinson's disease and Alzheimer's disease) and the confirmed diagnose of Alzheimer's disease appear linked to a subset of gene expression. For the covariates seemingly less relevant to gene expression, the length of education also shows large MaxNMI, whereas the experience of traumatic brain injury (TBI), aging and dementia-related score show surprisingly low NMIs. Besides the MaxNMI, Figure 10(b) plots the NMIs of the top 100 genes associated with the selected four covariates. Clearly, each covariate seems more correlated with a distinct set of views/genes.

17

(a) Maximum NMI over $1,000$ genes, comparing the clustering labels estimated from the LSP model and each clinical covariate.

(b) The NMIs of the 100 selected genes associated with the four covariates.

Figure 10: Clustering aging brains using gene expression. The results are compared with 11 clinical covariates using NMIs.

6. Discussion

In this article, we propose a method to directly estimate the cluster assignment probabilities for each data under multiple views. There are several interesting extensions that could be pursued. 1) The similarity matrix can be computationally prohibitive to handle when n is large; therefore, a random feature map (Rahimi and Recht, 2008) can be considered. A similar solution has recently been proposed for spectral clustering using random binning features (Wu et al., 2018). 2) We have assumed the views are given; in practice, if they are not known, we could use some domain-specific knowledge to estimate views. For example, in image processing, we could use edge detection and convolution of pixels to form each view. It is useful to study how they would impact the clustering results. 3) It is interesting to combine the LSP model with another loss function such as the one from a regression task, as an extension to our PAC-Bayes theory result. This could create generalized Bayes models and form new insights about the semi-supervised learning.

Appendix A. Proof of the main theorem

Proof Let $\text{KL}(P||S)$ be the Kullbeck-Leibler divergence between two generating distributions for a cluster graph using P or S as the co-assignment probability matrix. The sample space of the distribution is a sub-space $\mathcal{Z} \subset \{0,1\}^{n(n-1)/2}$ subject to constraints described in (11). We assume that S is given.

Step 1. Change of measure:

$$
Mkl\left[\frac{1}{M}\sum_{v=1}^{M}\mathbb{E}_{\hat{Z}^{(v)}\sim P}loss\left(Z_0^{(v)},\hat{Z}^{(v)}\right),\mathbb{E}_{Z_0^{(v)}\sim\Pi_0}\mathbb{E}_{\hat{Z}^{(v)}\sim P}loss\left(Z_0^{(v)},\hat{Z}^{(v)}\right)\right]
$$

$$
= Mkl\left[\frac{1}{M}\sum_{v=1}^{M}\mathbb{E}_{\hat{Z}\sim P}loss\left(Z_0^{(v)},\hat{Z}\right),\mathbb{E}_{Z_0^{(v)}\sim\Pi_0}\mathbb{E}_{\hat{Z}\sim P}loss\left(Z_0^{(v)},\hat{Z}\right)\right]
$$

$$
\leq \mathbb{E}_{\hat{Z}\sim P}Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),\mathbb{E}_{Z_0^{(v)}\sim\Pi_0}loss\left(Z_0^{(v)},\hat{Z}\right)\right]
$$

$$
\leq \mathrm{KL}(P\|S)+\log\mathbb{E}_{\hat{Z}\sim S}\exp\left\{Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),\mathbb{E}_{Z_0^{(v)}\sim\Pi_0}loss\left(Z_0^{(v)},\hat{Z}\right)\right]\right\},
$$

where the first equality is because $\hat{Z}^{(v)}$'s are iid, the first inequality uses the convexity of the kl function; the second inequality uses the change of measure inequality [Lemma 4 in (Seldin and Tishby, 2010)].

Due to the Markov's inequality, with probability greater than $1-\delta$,

$$
\mathbb{E}_{\hat{Z}\sim S}\exp\left\{Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),\mathbb{E}_{Z_0^{(v)}\sim\Pi_0}loss\left(Z_0^{(v)},\hat{Z}\right)\right]\right\}
$$

$$
\leq \frac{1}{\delta}\mathbb{E}_{Z_0^{(v)}\overset{iid}{\sim}\Pi_0}\mathbb{E}_{\hat{Z}\sim S}\exp\left\{Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),loss\left(Z_0^{(v)},\hat{Z}\right)\right]\right\}
$$

$$
= \frac{1}{\delta}\mathbb{E}_{\hat{Z}\sim S}\mathbb{E}_{Z_0^{(v)}\overset{iid}{\sim}\Pi_0}\exp\left\{Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),loss\left(Z_0^{(v)},\hat{Z}\right)\right]\right\},
$$

and the last equality is due to $kl(.)$ is upper bounded, hence we can use Fubini's theorem.

Step 2. Bounding the exponential kl function by a constant:

Using Theorem 1 of (Maurer, 2004),

$$
\mathbb{E}_{Z_0^{(v)}\overset{iid}{\sim}\Pi_0}\exp\left\{Mkl\left[\frac{1}{M}\sum_{v=1}^{M}loss\left(Z_0^{(v)},\hat{Z}\right),loss\left(Z_0^{(v)},\hat{Z}\right)\right]\right\}
$$

$$
\leq \exp(\frac{1}{12M})\sqrt{\frac{\pi M}{2}}+2.
$$

Step 3. Relaxing the KL-divergences between two cluster graph distributions to total element-wise kl divergences:

DUAN

$$
\begin{aligned}
\mathrm{KL}(P||S) = \sum_{Z \in \mathcal{Z}} \{ & 1(z_{i,j} = 1) p_{i,j} \log \frac{p_{i,j}}{s_{i,j}} \\
& + 1(z_{i,j} = 0)(1 - p_{i,j}) \log \frac{1 - p_{i,j}}{1 - s_{i,j}} \} \\
\leq \sum_{Z \in \{0,1\}^{n(n-1)/2}} \{ & 1(z_{i,j} = 1) p_{i,j} \log \frac{p_{i,j}}{s_{i,j}} \\
& + 1(z_{i,j} = 0)(1 - p_{i,j}) \log \frac{1 - p_{i,j}}{1 - s_{i,j}} \} \\
= \sum_{j < i} & kl(p_{i,j}||s_{i,j}),
\end{aligned}
$$

where the inequality is due to the non-negativity of each kl function and $\mathcal{Z} \subseteq \{0,1\}^{n(n-1)/2}$.
Combining the results,

$$
KL\big[R(\Pi_0, \phi) \ || \ R(\Pi_M, \phi) \big] \leq \frac{\sum_{j<i} KL(p_{i,j}||s_{i,j}) + \log\{\exp(\frac{1}{12M})\sqrt{\frac{\pi M}{2}} + 2\} - \log \delta}{M}.
$$

Taking $S = S^{(v)}$, summing both sides over $v = 1, \ldots, M$ and dividing by M yields the result. ∎

Appendix B. Single View Simulation

(a) S for (a).

(b) S for (b).

(c) S for (c).

(d) Estimated P for (a).

(e) Estimated P for (b).

(f) Estimated P for (c).

(g) S for (d).

(h) S for (e).

(i) S for (f).

(j) Estimated P for (d).

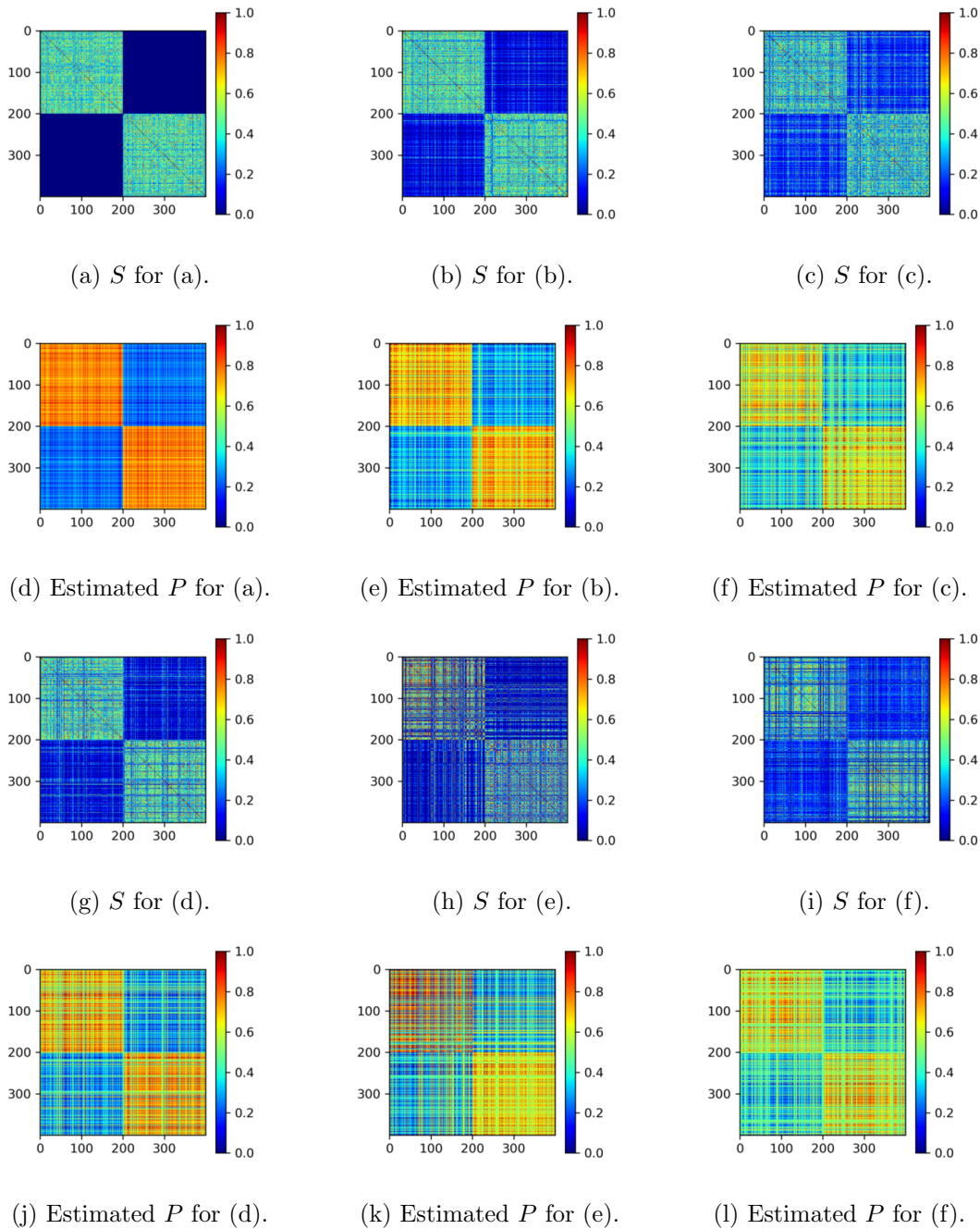(k) Estimated P for (e).

(l) Estimated P for (f).

Figure 11: Computed similarity matrices and estimated co-assignment probability matrices in the single view clustering experiments (Section 5.1).

References

David Arthur and Sergei Vassilvitskii. K-means++: the advantages of careful seeding. In *Proceedings of the 18th annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

David M Blei and Peter I Frazier. Distance dependent Chinese restaurant processes. *Journal of Machine Learning Research*, 12(Aug):2461–2488, 2011.

David B Dahl, Ryan Day, and Jerry W Tsai. Random partition distribution indexed by pairwise information. *Journal of the American Statistical Association*, 112(518):721–732, 2017.

Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. Deep unsupervised clustering with Gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

Leo L Duan and David B Dunson. Bayesian distance clustering. *arXiv preprint arXiv:1810.08537*, 2018.

Tarek A El Moselhy and Youssef M Marzouk. Bayesian inference with optimal maps. *Journal of Computational Physics*, 231(23):7815–7850, 2012.

Xiaoli Z Fern and Carla E Brodley. Random projection for high dimensional data clustering: a cluster ensemble approach. In *Proceedings of the 20th International Conference on Machine Learning*, pages 186–193, 2003.

Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, June 2002.

Zoubin Ghahramani and Geoffrey E Hinton. The EM algorithm for mixtures of factor analyzers. Technical report, Technical Report CRG-TR-96-1, University of Toronto, 1996.

Yue Guan, Jennifer G Dy, Donglin Niu, and Zoubin Ghahramani. Variational inference for nonparametric multiple clustering. In *MultiClust Workshop, Knowledge Discovery and Data Mining*, 2010.

Benjamin Guedj. A primer on PAC-bayesian learning. *arXiv preprint arXiv:1901.05353*, 2019.

Mark S Handcock, Adrian E Raftery, and Jeremy M Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354, 2007.

Christian Hennig et al. Breakdown points for maximum likelihood estimators of location–scale mixtures. *The Annals of Statistics*, 32(4):1313–1340, 2004.

Peter D Hoff. Model-based subspace clustering. *Bayesian Analysis*, 1(2):321–344, 2006.

Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

Shalmali Joshi, Joydeep Ghosh, Mark Reid, and Oluwasanmi Koyejo. Rényi divergence minimization based co-regularized multiview clustering. *Machine Learning*, 104(2-3): 411–439, 2016.

Diederik P Kingma and Jimmy Ba. ADAM: a method for stochastic optimization. In *International Conference on Learning Representations*, pages 3455–3463, 2014.

Paul Kirk, Jim E Griffin, Richard S Savage, Zoubin Ghahramani, and David L Wild. Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics*, 28(24): 3290–3297, 2012.

Abhishek Kumar, Piyush Rai, and Hal Daume. Co-regularized multi-view spectral clustering. In *Advances in Neural Information Processing Systems*, pages 1413–1421, 2011.

Martin H Law, Anil K Jain, and Mário Figueiredo. Feature selection in mixture-based clustering. In *Advances in Neural Information Processing Systems*, pages 641–648, 2003.

Dazhuo Li and Patrick Shafto. Bayesian hierarchical cross-clustering. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pages 443–451, 2011.

Eric F Lock and David B Dunson. Bayesian consensus clustering. *Bioinformatics*, 29(20): 2610–2616, 2013.

Andreas Maurer. A note on the PAC Bayesian theorem. *arXiv preprint cs/0411099*, 2004.

Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):53–71, 2008.

Jeffrey W Miller and David B Dunson. Robust Bayesian inference via coarsening. *Journal of the American Statistical Association*, pages 1–13, 2018.

Jeremy A Miller, Angela Guillozet-Bongaarts, Laura E Gibbons, Nadia Postupna, Anne Renz, Allison E Beller, Susan M Sunkin, Lydia Ng, Shannon E Rose, and Kimberly A Smith. Neuropathological and transcriptomic characteristics of the aged brain. *Elife*, 6: e31126, 2017.

Qianxing Mo, Ronglai Shen, Cui Guo, Marina Vannucci, Keith S Chan, and Susan G Hilsenbeck. A fully Bayesian latent variable model for integrative clustering analysis of multi-type omics data. *Biostatistics*, 19(1):71–86, 2017.

Donglin Niu, Jennifer G Dy, and Michael I Jordan. Multiple non-redundant spectral clustering views. In *Proceedings of the 27th International Conference on Machine Learning*, pages 831–838, 2010.

Donglin Niu, Jennifer Dy, and Zoubin Ghahramani. A nonparametric Bayesian model for multiple clustering with overlapping feature views. In *Artificial Intelligence and Statistics*, pages 814–822, 2012.

Donglin Niu, Jennifer G Dy, and Michael I Jordan. Iterative discovery of multiple alternative clustering views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36 (7):1340–1353, 2013.

Iku Ohama, Issei Sato, Takuya Kida, and Hiroki Arimura. On the model shrinkage effect of gamma process edge partition models. In *Advances in Neural Information Processing Systems*, pages 397–405, 2017.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2008.

Carl Edward Rasmussen. The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*, pages 554–560, 2000.

Yevgeny Seldin and Naftali Tishby. PAC-Bayesian analysis of co-clustering and beyond. *Journal of Machine Learning Research*, 11(Dec):3595–3646, 2010.

Mahlet G Tadesse, Naijun Sha, and Marina Vannucci. Bayesian variable selection in clustering high-dimensional data. *Journal of the American Statistical Association*, 100(470): 602–617, 2005.

Anh Tan, Wenli Ma, Amit Vira, Dhruv Marwha, and Lise Eliot. The human hippocampus is not sexually-dimorphic: meta-analysis of structural MRI volumes. *Neuroimage*, 124: 350–366, 2016.

Yee W Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Sharing clusters among related groups: hierarchical Dirichlet processes. In *Advances in Neural Information Processing Systems*, pages 1385–1392, 2005.

Sara Wade and Zoubin Ghahramani. Bayesian cluster analysis: point estimation and credible balls. *Bayesian Analysis*, 13(2):559–626, 2018.

Daniela M Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726, 2010.

Lingfei Wu, Pin-Yu Chen, Ian En-Hsu Yen, Fangli Xu, Yinglong Xia, and Charu Aggarwal. Scalable spectral clustering using random binning features. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2506–2515. ACM, 2018.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems*, pages 1601–1608, 2005.

Mingyuan Zhou. Beta-negative binomial process and exchangeable random partitions for mixed-membership modeling. In *Advances in Neural Information Processing Systems*, pages 3455–3463, 2014.