Multiple Clustering Views from Multiple Uncertain Experts

Yale Chang [1] **Junxiang Chen** [1] **Michael H. Cho** [2] **Peter J. Castaldi** [2] **Edwin K. Silverman** [2] **Jennifer G. Dy** [1]

Abstract

Expert input can improve clustering performance. In today's collaborative environment, the availability of crowdsourced multiple expert input is becoming common. Given multiple experts' inputs, most existing approaches can only discover one clustering structure. However, data is multi-faceted by nature and can be clustered in different ways (also known as views). In an exploratory analysis problem where ground truth is not known, different experts may have diverse views on how to cluster data. In this paper, we address the problem on *how to automatically discover multiple ways to cluster data given potentially diverse inputs from multiple uncertain experts*. We propose a novel Bayesian probabilistic model that automatically learns the multiple expert views and the clustering structure associated with each view. The benefits of learning the experts' views include 1) enabling the discovery of multiple diverse clustering structures, and 2) improving the quality of clustering solution in each view by assigning higher weights to experts with higher confidence. In our approach, the expert views, multiple clustering structures and expert confidences are jointly learned via variational inference. Experimental results on synthetic datasets, benchmark datasets and a real-world disease subtyping problem show that our proposed approach outperforms competing baselines, including meta clustering, semi-supervised clustering, semi-crowdsourced clustering and consensus clustering.

1. Introduction

As a cornerstone of unsupervised learning, clustering has been widely used in knowledge discovery problems (Jain et al., 1999). Given data matrix and a notion of similarity between samples, clustering aims to categorize data into different clusters so that samples in the same cluster are similar and samples in different clusters are dissimilar. Thus, depending on users' notions of similarity, the same dataset can be clustered in different ways, also known as *views* (Niu et al., 2010). For example, face images can be clustered based on pose or identity; marbles can be clustered based on shape or color. However, how to properly define similarity between samples in a knowledge discovery problem is nontrivial. To help solve this challenge, semi-supervised clustering utilizes expert supervision to guide the clustering towards the right solution (Wagstaff & Cardie, 2000; Basu et al., 2008). Supervision is usually in the form of pairwise constraints between samples, including must-link (ML) and cannot-link (CL) constraints.

Instead of supervision from one expert, it is becoming more common for supervision to be available from multiple experts as data can be shared and processed by increasingly larger audiences (e.g., crowdsourcing (Howe, 2008) mechanisms such as Amazon Mechanical Turk and large collaborative consortiums (Zhang et al., 2011)). In an exploratory data analysis setting, where ground truths are not known, different experts might provide supervision (pairwise contraints) with varying views in mind. For example, one expert might be thinking of similarity/clustering based on pose and another expert might be providing inputs based on identity on the face image problem. Moreover, because experts are not oracles, their inputs are prone to errors as well. In this paper, we address a new clustering paradigm: **how to discover multiple clustering structures in the data given potentially diverse constraints from multiple uncertain experts**.

Our objective of finding multiple clustering structures given inputs from multiple experts is motivated by discovering subtypes (clusters) of a complex lung disease called Chronic Obstructive Pulmonary Disease (COPD). COPD is currently the third leading cause of death in the US (Murphy et al., 2013). Although traditionally being called one disease, doctors believe there exist multiple disease subtypes and providing personalized clinical care to patients according to their disease subtypes can lead to more effective treatments. We have collected constraints provided by multiple experts in the consortium. The experts had var-

[1] Northeastern University, Boston, MA [2] Brigham and Women's Hospital, Harvard Medical School, Boston, MA. Correspondence to: Yale Chang <ychang@coe.neu.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

ied backgrounds: 1) clinicians tend to provide constraints by comparing patients' clinical measurements; and 2) radiologists tend to provide constraints based on examining patients' computed tomography (CT) images. On the one hand, experts have disagreements on whether to put a pair of patients in the same group because they are focusing on different aspects of patients. On the other hand, experts of similar backgrounds (clinicians, radiologists) tend to provide shared views of clustering data albeit with noisy constraints. Because the various experts might have different views of clustering data, it does not make sense to learn a single consensus clustering solution; rather, we need to discover the multiple consensus clustering solutions in the different views.

One naive way to generate multiple clustering solutions from multiple uncertain experts is to separately apply semi-supervised clustering (Bilenko et al., 2004; Davis et al., 2007; Basu et al., 2008) using constraints from each expert. The potential drawbacks of this strategy are 1) the clustering performance often drastically degrades in the presence of noisy constraints due to uncertainty of one expert; and 2) the resulting clustering solutions can be highly redundant due to the existence of similar expert views.

There are a few existing approaches that can combine constraints from multiple experts but they are mostly designed to generate one clustering solution. Semi-crowdsourced clustering (SemiCrowd) combines constraints provided by multiple experts through filtering out uncertain pairs in the average sample similarity matrix (Yi et al., 2012). However, this approach can only generate one clustering solution. Similar to SemiCrowd, most consensus clustering methods are designed to generate one clustering solution (Strehl & Ghosh, 2002; Fern & Brodley, 2004; Topchy et al., 2005; Ghosh & Acharya, 2011). Another disadvantage is that consensus clustering methods do not work when only a small number of pairwise constraints are available.

There are a few multiple alternative clustering approaches that can output multiple clustering solutions (Caruana et al., 2006; Cui et al., 2007; Jain et al., 2008; Niu et al., 2010; 2012). However, all these methods are unsupervised; none of these methods are able to utilize expert inputs.

There are a few existing crowdsourcing approaches that learns an underlying grouping of experts (Tian & Zhu, 2012; Kajino et al., 2013; Moreno et al., 2015). However, they are all designed for classification problems, where experts provide labels on samples queried. In our clustering task, experts can not provide labels because how to define different clusters is still unknown and yet to be discovered. Instead, they can provide pairwise constraints by comparing sample pairs using their domain knowledge. Therefore, these classification-based approaches above cannot be used in our clustering task.

Contributions. To address this new clustering paradigm, we build a Bayesian probabilistic model for learning multiple alternative consensus clustering views from experts' constraints, we call *Multiple Clustering Views from the Crowd* (**MCVC**). Multiple experts are automatically assigned to different latent views and constraints provided by each expert is assumed to be noisy perturbations of the clustering associated with that expert's view. Thus, multiple clustering structures can be discovered. Furthermore, by explicitly modeling the uncertainty of each expert, experts with higher accuracies are assigned higher weights, leading to improved quality of the learned clustering structure in each view. The clustering structure for each expert view is modeled by a discriminative clustering model (Gomes et al., 2010), which has the advantages of 1) naturally introducing uncertainties in cluster assignments; 2) avoiding making assumptions on the generative process of clusters; and 3) being able to cluster samples that do not appear in the training set. We demonstrate that our MCVC outperforms competing alternatives on synthetic, benchmark data, and a real-world disease subtyping problem.

2. Proposed Approach

We collect data matrix $X \in \mathbb{R}^{n\times d}$, where n is the number of samples and d is the number of features, and pairwise constraints provided by M experts $S^{(1:M)}$, where $S^{(m)} \in \{0, 1, \text{NULL}\}^{n\times n}$ represents the constraints provided by the m-th expert. For sample pair (x_i, x_j), $S_{ij}^{(m)} = 1$ means the m-th expert provides must-link (ML) constraint; $S_{ij}^{(m)} = 0$ means cannot-link (CL) constraint; $S_{ij}^{(m)} = \text{NULL}$ means no constraint is provided. Our objective is to utilize constraints collected from these M experts to guide the clustering algorithm to discover multiple clustering structures in the data.

2.1. Multiple Alternative Clustering Views

We assume there exist multiple alternative expert views and the constraints provided by experts in each view are perturbations of the clustering solution associated with that view [1]. Let c_m represent the latent view to which the m-th expert is assigned to. Furthermore, $Z^{(c_m)}$ are the latent clusters for each c_m view. Since we do not know the underlying number of possible expert views, we automatically learn the number of expert views by assuming a Dirichlet

[1] Note that "view" in multi-view clustering (Bickel & Scheffer, 2004) means coming from different sources or feature sets; whereas, "view" in our case follows the terminology in multiple alternative clusterings which means different interpretation or point of view of the data. Multi-view clustering only finds ONE clustering solution from multiple feature sets which are given. In contrast, our goal is to find MULTIPLE clustering solutions/views which are latent.

process (Ferguson, 1973) prior on c_m.

2.2. Model Uncertainties of Experts' Constraints

Since clustering is widely used for knowledge discovery, experts might not be certain on the constraints they provided. To incorporate the assumption that different experts may have different levels of expertises when providing constraints, we assume the uncertainty of the m-th expert can be characterized by accuracy parameters (α_m, β_m), where α_m represents the m-th expert's *sensitivity* and β_m represents *specificity*. Sensitivity is defined as the probability of providing ML constraints for sample pairs from the same cluster in the ground truth. Specificity is defined as the probability of providing CL constraints for sample pairs from different clusters in the ground truth. We model the conditional likelihood of $S^{(m)}$ given c_m (the latent view) and $Z^{(c_m)}$ (the latent cluster in each view) by a Bernoulli distribution as follows:

$$p(S_{ij}^{(m)} = 1 | Z_i^{(c_m)} = Z_j^{(c_m)}, \alpha_m) = \alpha_m \tag{1}$$

$$p(S_{ij}^{(m)} = 0 | Z_i^{(c_m)} \neq Z_j^{(c_m)}, \beta_m) = \beta_m \tag{2}$$

2.3. Discriminative Clustering

We consider the following when choosing the clustering model: 1) Since we need to model uncertainties of experts, instead of generating hard clustering results, the assignments to clusters should be associated with probabilities; 2) We should avoid making strong assumptions on the generative process of clusters, which can be easily violated in practice; 3) We should be able to cluster samples outside the training set. The discriminative clustering model (Gomes et al., 2010) satisfies all these requirements. Instead of assuming the generative process of data X, discriminative clustering directly models the conditional distribution of cluster label given data. We model the conditional distribution of the latent cluster label, $Z^{(c_m)}$, given weight $W^{(c_m)}$ and offset $b^{(c_m)}$ with a multiple logistic regression model:

$$p(Z_i^{(c_m)} = k | W^{(c_m)}, b^{(c_m)}; X) = \frac{e^{w_k^{(c_m)T} x_i + b_k^{(c_m)}}}{\sum_{j=1}^{K} e^{w_j^{(c_m)T} x_i + b_j^{(c_m)}}} \tag{3}$$

where $w_k^{(c_m)}$ is the k-th column of $W^{(c_m)} \in \mathbb{R}^{d \times K}$ and $b_k^{(c_m)}$ is the k-th row of $b^{(c_m)} \in \mathbb{R}^{K \times 1}$.

2.4. Prior Distributions

We describe the prior distributions of parameters used in our model. To incorporate the assumptions that experts' accuracies should be far away from random guess ($\alpha_m = \beta_m = 0.5$), we put Beta priors on α_m, β_m and set their parameters to make most of their probability densities be far away from 0.5 and close to 1.

$$p(\alpha_m) = \mathrm{Beta}(\tau_{\alpha^1 0}^{(m)}, \tau_{\alpha^2 0}^{(m)}) \tag{4}$$

$$p(\beta_m) = \mathrm{Beta}(\tau_{\beta^1 0}^{(m)}, \tau_{\beta^2 0}^{(m)}) \tag{5}$$

To automatically learn the number of expert views, we assume a Dirichlet process prior on c_m and utilize the stick-breaking construction as follows (Blei et al., 2006):

$$\nu_g \sim \mathrm{Beta}(1, \gamma);\ \pi_g \sim \nu_g \prod_{j=1}^{g-1} (1 - \nu_j);\ c_m \sim \mathrm{Cat}(\pi_{1:\infty})$$

Therefore, $p(c_m | \nu_{1:\infty})$ can be written as

$$p(c_m | \nu_{1:\infty}) = \prod_{g=1}^{\infty} \pi_g^{c_{mg}} = \prod_{g=1}^{\infty} \left(\nu_g \prod_{j=1}^{g-1} (1 - \nu_j)\right)^{c_{mg}} \tag{6}$$

where $c_{mg} = 1$ if $c_m = g$ and $c_{mg} = 0$ otherwise.

We assume the prior distributions of both weight $W^{(g)}$ and offset $b^{(g)}$ are factorized Gaussian distributions.

$$p(W^{(g)}) = \prod_{i=1}^{d} \prod_{j=1}^{K} \mathcal{N}(\mu_{W_{ij}0}^{(g)}, \sigma_{W_{ij}0}^{(g)2}) \tag{7}$$

$$p(b^{(g)}) = \prod_{i=1}^{K} \mathcal{N}(\mu_{b_i 0}^{(g)}, \sigma_{b_i 0}^{(g)2}) \tag{8}$$

2.5. Joint Distribution

The overall joint distribution of observations and latent variables for our model is:

$$p(S^{(1:M)}, \alpha_{1:M}, \beta_{1:M}, c_{1:M}, \nu_{1:\infty}, Z^{(1:\infty)}, W^{(1:\infty)}, b^{(1:\infty)})$$

$$= \prod_{m=1}^{M} p(S^{(m)} | \alpha_m, \beta_m, c_m, Z^{(1:\infty)}) p(\alpha_m) p(\beta_m) \tag{9}$$

$$p(c_m | \nu_{1:\infty}) \prod_{g=1}^{\infty} p(Z^{(g)} | W^{(g)}, b^{(g)}; X) p(W^{(g)}) p(b^{(g)}) p(\nu_g)$$

The respective graphical model is shown in Figure 1.

3. Variational Inference

Our learning objective is to maximize the marginal likelihood of observed constraints, which is intractable. Thus, we apply variational inference. Given variational distribution $q(h; \theta)$, where h is the collection of latent variables and θ is their parameters. the log of the marginal likelihood $\log p(S^{(1:M)})$ can be decomposed as

$$\begin{aligned} \log p(S^{(1:M)}) &= \mathcal{L}_{q(h;\theta)} + KL \left[q(h; \theta) \,|\, p(h | S^{(1:M)}) \right] \\ &\geq \mathcal{L}_{q(h;\theta)} \end{aligned} \tag{10}$$

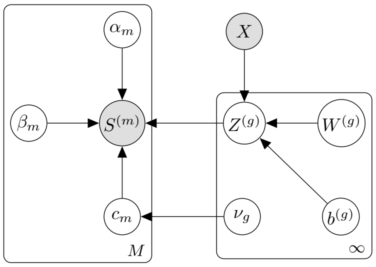


Figure 1. Overall Graphical Model

where the inequality holds due to the nonnegativity of the the Kullback-Liebler (KL) divergence. $\mathcal{L}_{q(h;\theta)}$ is called the evidence lower bound (ELBO) of $\log p(S^{(1:M)})$:

$$\mathcal{L}_{q(h;\theta)} = \mathrm{E}_{q(h;\theta)}\left[\log p(h, S^{(1:M)}) - \log q(h;\theta)\right] \quad (11)$$

In variational inference, the learning objective becomes maximizing $\mathcal{L}_{q(h;\theta)}$ w.r.t. variational parameters θ.

Let $h = \{\alpha_{1:M}, \beta_{1:M}, c_{1:M}, \nu_{1:G}, Z^{(1:G)}, W^{(1:G)}, b^{(1:G)}\}$, where G is the number of components of the truncated Dirichlet Process (Blei et al., 2006). We apply mean-field and assume $q(h;\theta)$ can be factorized as follows:

$$\begin{aligned} & q(\alpha_{1:M}, \beta_{1:M}, c_{1:M}, \nu_{1:G}, Z^{(1:G)}, W^{(1:G)}, b^{(1:G)}) \\ = & \prod_{m=1}^{M} \left[q(\alpha_m) \cdot q(\beta_m) \cdot q(c_m)\right] \cdot \\ & \prod_{g=1}^{G} \left[q(\nu_g) \prod_{i=1}^{n} q(Z_i^{(g)}) \prod_{i=1}^{d} \prod_{j=1}^{K} q(W_{ij}^{(g)}) \prod_{i=1}^{K} q(b_i^{(g)})\right] \quad (12) \end{aligned}$$

where the marginal distribution of each random variable is

$$q(\alpha_m) = \mathrm{Beta}(\tau^{(m)}_{\alpha^1}, \tau^{(m)}_{\alpha^2}) \quad (13)$$

$$q(\beta_m) = \mathrm{Beta}(\tau^{(m)}_{\beta^1}, \tau^{(m)}_{\beta^2}) \quad (14)$$

$$q(c_m) = \mathrm{Cat}(\phi_{m,:}) \quad (15)$$

$$q(\nu_g) = \mathrm{Beta}(\tau^{(g)}_{\nu^1}, \tau^{(g)}_{\nu^2}) \quad (16)$$

$$q(Z_i^{(g)}) = \mathrm{Cat}(\eta^{(g)}_{i,:}) \quad (17)$$

$$q(W_{ij}^{(g)}) = \mathcal{N}(\mu^{(g)}_{W_{ij}}, \sigma^{(g)2}_{W_{ij}}) \quad (18)$$

$$q(b_i^{(g)}) = \mathcal{N}(\mu^{(g)}_{b_i}, \sigma^{(g)2}_{b_i}) \quad (19)$$

We use θ to denote all the variational parameters, which consist of $\{\tau^{(m)}_{\alpha^1}, \tau^{(m)}_{\alpha^2}, \tau^{(m)}_{\beta^1}, \tau^{(m)}_{\beta^2}, \phi_{m,:}\}$ $(m = 1 \cdots M)$ and $\{\tau^{(g)}_{\nu^1}, \tau^{(g)}_{\nu^2}, \eta^{(g)}_{i,:}, \mu^{(g)}_{W_{ij}}, \sigma^{(g)}_{W_{ij}}, \mu^{(g)}_{b_i}, \sigma^{(g)}_{b_i}\}$ $(g = 1 \cdots G)$. Besides the simplex constraints on the parameters of the categorical distribution, both the parameters of the Beta distribution and the standard deviation of Gaussian distribution should have positive constraints.

We derive the closed-form formula of $\mathcal{L}_{q(h;\theta)}$ as a function of θ and put the detailed steps in the supplementary materials due to space constraints. Since ELBO $\mathcal{L}_{q(h;\theta)}$ can be written as a function of variational parameters θ, we can directly maximize $\mathcal{L}_{q(h;\theta)}$ using gradient-based optimization approaches. The gradient $\frac{\partial \mathcal{L}_{q(h;\theta)}}{\partial \theta}$ can be automatically computed using reverse-mode differentiation (Maclaurin et al., 2015). We choose to use a limited-memory projected quasi-Newton algorithm (PQN) to optimize our objective because it has both superlinear convergence rate and linear memory requirement (Schmidt et al., 2009). Because our objective $\mathcal{L}_{q(h;\theta)}$ is not concave, we provide multiple initializations $\theta^{(0)}$ to the optimization algorithm and choose the one resulting in the maximal objective value. We set the number of random initializations to be 50 in all experiments and the results are stable across different runs.

4. Experimental Results

In this section, we aim to demonstrate our MCVC can automatically 1) assign multiple experts to different views; and 2) improve the quality of clustering solution in each view by assigning higher weights to uncertain experts of higher accuracies. We also analyze how 3) the settings of the number of clusters; and 4) the constraints provided by irrelevant experts affect the performance of MCVC.

4.1. Competing Alternatives

For aim 1), we construct two views of experts based on two different ways to cluster the data. These two expert views are treated as the ground truth of assigning multiple experts to different views. Then we compare our MCVC against an adapted version of meta clustering (Caruana et al., 2006).

Meta Spectral Clustering (MetaClust): The original meta clustering approach cannot handle multiple experts' constraints. Instead of using the data matrix to generate multiple clustering solutions as input, we directly use the constraint sets provided by multiple experts as input. Meta clustering first computes the similarity between experts by computing the rand index (Rand, 1971) between the constraint sets they provide. Given the resulting similarity matrix between experts, instead of hierarchical clustering as in the original paper, we apply spectral clustering (Ng et al., 2001) to assign multiple experts to different views. We determine the number of expert views by maximizing the gap between the consecutive eigenvalues of the graph Laplacian (Von Luxburg, 2007).

For aim 2), we first construct one view of experts based on one way to cluster the data. The underlying clustering structure is treated as the ground truth of clustering samples. Then we compare our MCVC against the following alternatives in generating clusters of high quality.

SemiCrowd: SemiCrowd (Yi et al., 2012) combines multiple expert constraints by filtering out uncertain pairs in the average similarity matrix, applying matrix completion, and then learning a distance metric for clustering.

Semi-supervised Clustering: Given a set of pairwise constraints, semi-supervised clustering either learns a better distance metric for clustering or guides the clustering algorithm to satisfy those constraints. We use Information-theoretic Metric Learning (ITML) (Davis et al., 2007) and Metric Pairwise Constrained KMeans (MPCKMeans) (Bilenko et al., 2004) as the representatives of those two strategies due to their superior performances compared to alternatives. Since semi-supervised clustering can only take one set of constraints as input, we combine constraints from multiple experts through majority voting (a sample pair is given ML constraint if the majority of experts provide ML constraints for them and CL constraint otherwise).

Consensus Clustering: Most consensus clustering algorithms only work with cluster labels instead of pairwise constraints. However, Cluster-based Similarity Partitioning Algorithm (CSPA) (Strehl & Ghosh, 2002), a consensus clustering approach that only need average similarity matrix between samples as input, can be used in our setup.

We provide the parameter setting details for all methods in the supplementary materials due to space constraint.

4.2. Synthetic and Benchmark Experiments

Synthetic Dataset: To help understand the algorithms, we generate a synthetic data that has multiple alternative clustering views. We generate a synthetic dataset containing 600 samples and six features. The scatterplots between pairwise features in this dataset are shown in Figure 2. First, there exist three clusters in the subspace spanned by the first two features, which we denote as Y_1. In all these three figures, the red, blue and green colors represent the true cluster indicator of Y_1. Second, there exist an alternative clustering structure in the subspace spanned by the third and fourth features, which we denote as Y_2. Note that Y_2 and Y_1 are very distinct. Third, the subspace spanned by the fifth and sixth feature does not contain well-separated clusters and we consider these as noisy features.

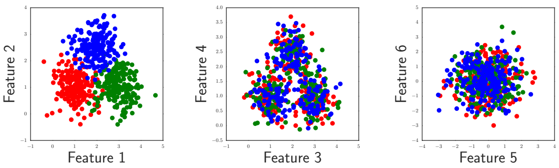


Figure 2. Scatterplots of pairwise features in the synthetic dataset.

WebKB Dataset: The WebKB dataset (web, 1998) contains webpages collected from four universities. After removing stop words and extracting the top 200 words with most frequent occurrences, we obtain a data matrix with 1041 samples and 200 features. The data can be clustered according to either owner types (*course, faculty, project, student*), which we treat as Y_1, or universities (*Cornell, Austin, Washington, Wisconsin*), which we use as Y_2.

Face Dataset: The Face dataset (Lichman, 2013) consists of 640 face images of people taken with varying poses (straight, left, right, up). Each image has 960 raw pixels. We apply principal component analysis (PCA) and keep 20 principal components (explaining 80% variance). Thus, we obtain a data matrix with 640 samples and 20 features. The data can be clustered based on pose (Y_1) or identity (Y_2).

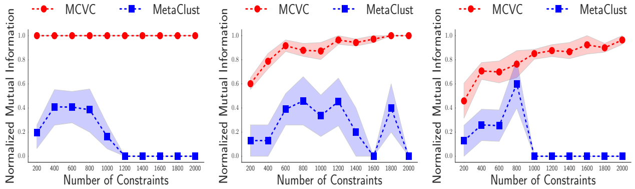
Simulating Constraints from Multiple Experts: For synthetic and benchmark datasets, we do not have access to real-world constraints provided by experts. Therefore, we simulate these constraints. Given a ground-truth clustering solution Y, the number of ML constraints n_{ML}, the number of CL constraints n_{CL}, and accuracy parameters of M experts $\alpha_{1:M}, \beta_{1:M}$, we can generate the constraints provided by the m-th expert as follows: 1) randomly sample n_{ML} ML constraints and n_{CL} CL constraints from Y; 2) randomly flip $n_{ML}(1-\alpha_m)$ ML pairs to CL pairs and flip $n_{CL}(1-\beta_m)$ CL pairs to ML pairs.

4.2.1. Task 1: Learning the Various Latent Views from Multiple Experts

In this subsection, we test the performance of our MCVC on automatically learning the latent views from multiple experts. We simulate noisy constraints provided by multiple expert from two latent views, Y_1 and Y_2 as follows:
1) the first view consists of experts 1-5, who provide constraints based on clustering solution Y_1 and have accuracy parameters $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.9, 0.85, 0.8, 0.75)$;
2) the second view consists of experts 6-10, who provide constraints based on clustering solution Y_2 and have accuracy parameters $\alpha_{6:10} = \beta_{6:10} = (0.75, 0.8, 0.85, 0.9, 0.95)$.

We compare the performance of our MCVC and meta spectral clustering (MetaClust) in recovering the ground-truth expert views as the number of constraints, n_{con}, is varied from 200 to a large number that makes the performances of both approaches become stable. We repeat the constraints generation process ten times to avoid the randomness of a single run. As a result, we obtain ten constraint sets for a fixed number of constraints. Given one constraint set, we run MCVC and MetaClust to generate two possibly different ways to group experts, which are denoted as $L_{\textbf{MCVC}}$ and $L_{\textbf{MetaClust}}$ respectively. We measure performance based on the normalized mutual information (NMI) (Strehl & Ghosh, 2002) between $L_{\textbf{MCVC}}$, $L_{\textbf{MetaClust}}$

and $L_{\textbf{True}}$, the ground-truth expert views. NMI measures the similarity between two partitions. In our case, higher NMI values indicate better performance. For a fixed number of constraints, one constraint set, and one approach, we obtain ten NMI values. We plot the mean and standard deviation for every set of ten NMI values of each approach as we vary the number of constraints as shown in Figure 3.



(a) Synthetic (b) WebKB (c) Face

Figure 3. Compare our MCVC against meta clustering (MetaClust) in assigning multiple experts to different views on (a) synthetic, (b) WebKB and (c) Face datasets.

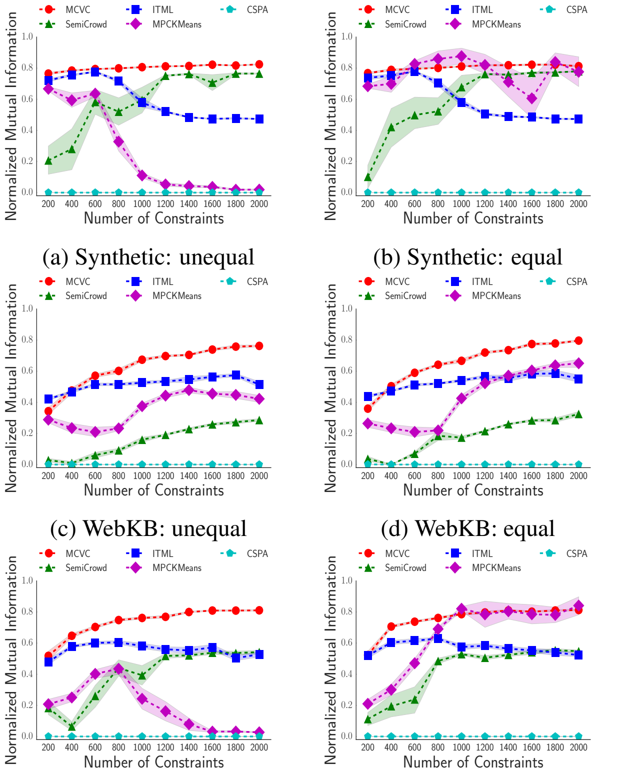
We have the following observations: 1) On the synthetic dataset, our MCVC can consistently recover the ground-truth expert views; 2) On WebKB and Face, the performance of MCVC improves as the number of constraints increases and can recover the ground-truth expert views when the number of constraints becomes large enough; 3) On WebKB and Face, when the number of constraints is too small, as is shown in the left part of each figure, MCVC will be dominated by the priors and therefore cannot perfectly recover the expert views. 4) On all three datasets, meta clustering fails to recover the ground-truth expert views.

4.2.2. Task 2: Discover Clustering Solution in an Expert View

In this subsection, we test the performance of our MCVC against competing methods in learning the clustering structure given constraints provided by multiple experts from one view. We simulate noisy constraints provided by multiple expert based on one ground-truth view, Y_1. We consider two different settings for their accuracy parameters: 1) experts have unequal accuracies $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.9, 0.85, 0.8, 0.75)$; 2) experts have equal and high accuracies $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.95, 0.95, 0.95, 0.95)$.

Our objective is to show that through learning different accuracies of multiple uncertain experts, our MCVC can generate better clustering results compared to competing alternatives (SemiCrowd, ITML, MPCKMeans and CSPA).

We vary the total number of ML/CL constraints provided by each expert from 200 to 2000 (as described in the previous subsection). For each fixed number of constraints, we randomly generate 10 constraint sets. For each constraint set, we run all approaches and obtain their clustering solutions. We measure performance based on NMI between the resulting clustering solutions and the ground-truth solution Y_1. For a fixed number of constraints, one constraint set, and one approach, we obtain 10 NMI values. We plot the mean and standard deviation for every set of 10 NMI values for each approach as we vary the number of constraints as shown in Figure 4. To compare the performance of different approaches, we apply the Kruskal-Wallis test (Kruskal & Wallis, 1952), which can be used to test whether two groups of samples are drawn from the same distribution, on their corresponding groups of NMI values.



(a) Synthetic: unequal (b) Synthetic: equal

(c) WebKB: unequal (d) WebKB: equal

(e) Face: unequal (f) Face: equal

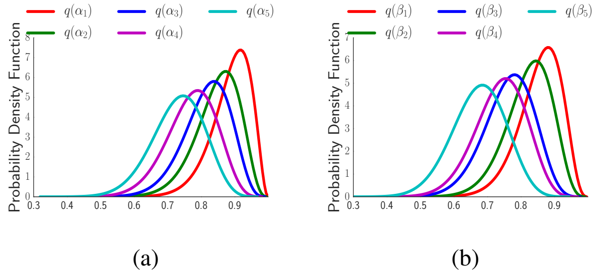
Figure 4. Compare our MCVC against SemiCrowd, ITML, MPCKMeans, CSPA in generating better clustering solutions on (a,b) synthetic, (c,d) WebKB and (e,f) Face datasets: in the left figures (a,c,e), the accuracy parameters are set to be unequal: $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.9, 0.85, 0.8, 0.75)$; in the right figures (b,d,f), the accuracy parameters are set to be equal and have high values: $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.95, 0.95, 0.95, 0.95)$.

From the left figures, where the accuracy parameters are set to be unequal, we have the following observations: I) Our MCVC consistently outperforms all competing alternatives. II) The performances of semi-supervised clustering approaches, including ITML and MPCKMeans, do not consistently improve as the number of constraints increases. III) The performance of SemiCrowd increases as the number of constraints increases. This means it can effectively reduce the noise in the constraints. However, it does not work well when the number of constraints is too small. IV) consensus clustering (CSPA) fails because the number of constraints is too small to be used to construct accurate sample similarity matrix.

From the right figures, where the accuracy parameters are set to be equal and have high values, we have the following observations: V) Our MCVC consistently outperforms SemiCrowd, ITML and CSPA but does not consistently outperform MPCKMeans.

By comparing the right against the left figures, we have the following observations: VI) MPCKMeans works well only when all the experts have high accuracies; it fails when not all experts have high accuaracies (as shown on the left figures). VII) Both ITML and SemiCrowd do not benefit much from higher percentage of correct constraints; VIII) Our MCVC perfoms well in both settings.

Explanation: Our MCVC has good performance in the unequal accuracies case because it can learn the accuracy parameters of different experts and assign higher weights to more accurate experts and lower weights to uncertain experts respectively. On the synthetic dataset, the posterior distributions of accuracy parameters $\alpha_{1:5}, \beta_{1:5}$ computed from MCVC are shown in Figure 5.



(a) (b)

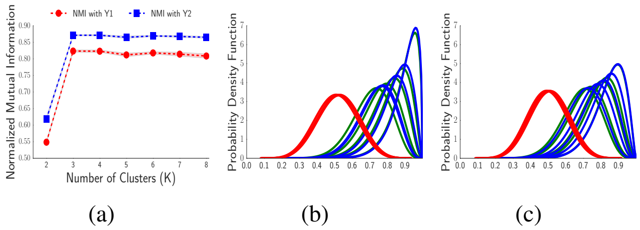
Figure 5. On the synthetic dataset, the posterior distributions of sensitivities $\alpha_{1:5}$ (a) and those of specificities $\beta_{1:5}$ (b) when experts have unequal accuracy parameters $\alpha_{1:5} = \beta_{1:5} = (0.95, 0.9, 0.85, 0.8, 0.75)$.

As we can see, the modes of their distributions are very close to their true values. In our MCVC, the constraints from the m-th expert are assigned weight $\omega_m = \log \frac{\alpha_m \beta_m}{(1-\alpha_m)(1-\beta_m)}$ when combining the constraints from M experts (we put the derivation details in the supplementary materials). Therefore, higher accuracies (α_m, β_m) naturally lead to higher weights. Thus, our MCVC becomes robust to noisy constraints in the unequal accuracies case.

4.2.3. Sensitivity Analysis: to the Number of Clusters and to Irrelevant Experts

Number of Clusters: To investigate how the setting of K, the number of clusters, affects the performance of our MCVC, we vary K from 2 to 8 and run MCVC on the synthetic dataset using two expert views constructed from the procedures described in subsection 4.2.1. The number of constraints is fixed to be 2000 and the constraints generation process is repeated 10 times. First, for any value of K, MCVC can correctly assign experts to two views and also generate two clustering solutions. Second, to evaluate how the clustering performances are affected by K, we compute the NMI between the two output clustering solutions and Y_1, Y_2 respectively.

The errorbar plot is shown in Figure 6 (a). As we can see, our MCVC is very robust to the setting of K as long as $K \geq K^*$, where K^* is the maximal number of clusters among all clustering views and $K^* = 3$ for our synthetic data. In problems where K^* is unknown, we can set K to be some large value to increase the possibility that $K \geq K^*$. However, in practice, we also observe that more constraints are needed to effectively train the model when K becomes large because there will be a larger number of parameters.



(a) (b) (c)

Figure 6. (a) shows the NMI between output clustering solutions and Y_1, Y_2 as K increases in the experiment studying the setting of the number of clusters; (b) and (c) show the posterior distributions of sensitivities $\alpha_{1:15}$ and specificities $\beta_{1:15}$ in the experiment studying the existence of irrelevant experts: green means experts 1-5, blue means experts 6-10, red means experts 11-15.

Irrelevant Experts: We define *irrelevant experts* as those who provide constraints based on a notion of similarity that is not supported by any feature in the data. To demonstrate how the existence of irrelevant experts affect the performance of our MCVC, we simulate three expert views using the synthetic data: 1) the first two views are the same as those described in subsection 4.2.1; and 2) the third view consists of five irrelevant experts who provide constraints based on a random clustering solution Y_3 and have accuracy parameters $\alpha_{11:15} = \beta_{11:15} = (0.95, 0.9, 0.85, 0.8, 0.75)$. We generate 2000 constraints from each expert. First, the first two expert views can still be perfectly recovered by our MCVC. Second, the irrelevant experts (experts 11-15) are either assigned to the first two expert views or new expert views.

We plot the posterior distributions of accuracy parameters for all 15 experts in Figure 6 (b,c). As we can see, the densities of irrelevant experts' accuracy parameters (red curves) concentrate around 0.5, which indicates the constraints from those experts are treated as random guesses and assigned near zero weights by MCVC. In practice, we can identify irrelevant experts by checking whether the modes of their accuracy parameters' posteriors are close to 0.5. Therefore, our MCVC is robust to irrelevant experts.

4.3. COPD Subtyping Experiment

Chronic Obstructive Pulmonary Disease (COPD) is a complex lung disease characterized by increasing breathlessness. Although being called one disease, doctors believe there exist different disease subtypes. The identification of different disease subtypes (clusters) can lead to tailored medical care for each patient.

Dataset: We collected 39 features from 987 COPD patients, including clinical measurements, demographics, lung function and measures from CT chest imaging.

Experts' Constraints: We also collected constraints provided by 29 experts, including clinicians and radiologists. We need to utilize experts' constraints to guide the clustering algorithm. However, the key challenge is that different experts disagree on whether to put a pair of patients in the same cluster. We suspect there exist different expert views and experts in each view provide constraints based on a shared way to cluster the data. The discovery of these expert views and their corresponding clustering solutions can provide more options for further investigation.

Evaluation: Since ground truth is not known, we can no longer use NMI to compare the performances of competing approaches. However, there are some key genetic variables that are known to be related to COPD, including *copdScore (Busch et al., 2017), HHIP (Pillai et al., 2009), MMP12 (Cho et al., 2014)*. A clustering solution is considered as useful/relevant if patients in different clusters show significant differences on these genetic variables.

We randomly split the dataset into half training set and half testing set. All approaches are learned using the training set and the constraints from multiple experts. The learned models can be used to cluster test samples and generate clustering solutions. To evaluate each solution, we compute its associated p-values on these genetic variables by applying Kruskal-Wallis test on *copdScore* (continuous) and χ^2 test on *HHIP* and *MMP12* (discrete). We use $P < 0.05$ to identify significant differences in these genetic variables.

Methods and Results: We first apply our MCVC and obtain 12 expert views. After removing irrelevant experts and views containing only one expert, there are 5 expert views left. Our physician collaborators identify 2 interesting expert views by analyzing the cluster characteristics of their associated clustering solutions, which are denoted as MCVC-A and MCVC-B respectively: 1) Solution MCVC-A contains emphysema-dominant and airway-dominant clusters, where emphysema cluster means the destruction of lung tissue and airway cluster means the increase of airway wall thickness. 2) Solution MCVC-B contains clusters of different levels of disease severity.

For competing approaches, we first run meta clustering and all 29 experts were lumped into one view. Then we apply SemiCrowd, ITML and MPCKMeans to combine the data matrix and constraints from all experts.

Table 1. p-values on three key COPD-related genetic variables.

Solutions	*copdScore*	*HHIP*	*MMP12*
MCVC-A	**3.47e-2**	**3.29e-3**	**4.41e-2**
MCVC-B	**3.49e-5**	**6.47e-3**	**3.72e-3**
SemiCrowd	1.30e-1	2.80e-1	1.47e-1
ITML	**4.67e-5**	**6.68e-3**	6.42e-2
MPCKMeans	**5.13e-4**	**1.46e-2**	**2.66e-2**

The p-values of solutions provided by our MCVC and competing approaches are shown in Table 1. As we can see, solutions provided by MCVC and MPCKMeans contain different clusters that show significant differences on all three COPD-related genetic variables. In contrast, both ITML and SemiCrowd are not significantly correlated with all three genetic variables. There's some overlap between solution MPCKMeans and solution MCVC-B (with NMI value 0.39). However, solution MCVC-A can only be discovered by our approach. This way of clustering COPD patients is consistent with some COPD investigators' latest discovery of COPD subtypes (Castaldi et al., 2014).

5. Conclusions

In this paper, we build a probabilistic model to discover multiple ways to cluster the data given potentially diverse inputs from multiple uncertain experts. This is achieved by automatically assigning multiple experts to different views and learning the clustering structure associated with each expert view. The quality of clustering solution in each expert view are improved by assigning higher weights to experts of higher accuracies. Experimental results on synthetic data, benchmark datasets and a real-world disease subtyping problem demonstrate that our MCVC outperforms its competing alternatives, including meta clustering, semi-supervised clustering, semi-crowdsourced clustering and consensus clustering.

6. Acknowledgements

We would like to acknowledge support for this project from the NIH grant NIH/NHLBI RO1HL089856, RO1HL089857 and NSF/IIS-1546428.

References

Webkb dataset, 1998. URL http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/.

Basu, Sugato, Davidson, Ian, and Wagstaff, Kiri. *Constrained clustering: Advances in algorithms, theory, and applications*. CRC Press, 2008.

Bickel, Steffen and Scheffer, Tobias. Multi-view clustering. In *IEEE International Conference on Data Mining*, volume 4, pp. 19–26, 2004.

Bilenko, Mikhail, Basu, Sugato, and Mooney, Raymond J. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning*, pp. 11, 2004.

Blei, David M, Jordan, Michael I, et al. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1:121–143, 2006.

Busch, Robert, Hobbs, Brian D, Zhou, Jin, Castaldi, Peter J, McGeachie, Michael J, Hardin, Megan E, Hawrylkiewicz, Iwona, Sliwinski, Pawel, Yim, Jae-Joon, Kim, Woo Jin, et al. Genetic association and risk scores in a copd meta-analysis of 16,707 subjects. *American Journal of Respiratory Cell and Molecular Biology*, 2017.

Caruana, Rich, Elhawary, Mohamed, Nguyen, Nam, and Smith, Casey. Meta clustering. In *IEEE International Conference on Data Mining*, pp. 107–118, 2006.

Castaldi, Peter J, Dy, Jennifer, Ross, James, Chang, Yale, Washko, George R, Curran-Everett, Douglas, Williams, Andre, Lynch, David A, Make, Barry J, Crapo, James D, et al. Cluster analysis in the copdgene study identifies subtypes of smokers with distinct patterns of airway disease and emphysema. *Thorax*, 2014.

Cho, Michael H, McDonald, Merry-Lynn N, Zhou, Xiaobo, Mattheisen, Manuel, Castaldi, Peter J, Hersh, Craig P, DeMeo, Dawn L, Sylvia, Jody S, Ziniti, John, Laird, Nan M, et al. Risk loci for chronic obstructive pulmonary disease: a genome-wide association study and meta-analysis. *The Lancet Respiratory Medicine*, 2(3): 214–225, 2014.

Cui, Ying, Fern, Xiaoli Z, and Dy, Jennifer G. Non-redundant multi-view clustering via orthogonalization. In *IEEE International Conference on Data Mining*, pp. 133–142, 2007.

Davis, Jason V, Kulis, Brian, Jain, Prateek, Sra, Suvrit, and Dhillon, Inderjit S. Information-theoretic metric learning. In *Proceedings of the Twenty-Fourth International Conference on Machine Learning*, pp. 209–216, 2007.

Ferguson, Thomas S. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, pp. 209–230, 1973.

Fern, Xiaoli Zhang and Brodley, Carla E. Solving cluster ensemble problems by bipartite graph partitioning. In *Proceedings of the Twenty-first International Conference on Machine learning*, pp. 36, 2004.

Ghosh, Joydeep and Acharya, Ayan. Cluster ensembles. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(4):305–315, 2011.

Gomes, Ryan G, Krause, Andreas, and Perona, Pietro. Discriminative clustering by regularized information maximization. In *Advances in Neural Information Processing Systems*, pp. 775–783, 2010.

Howe, Jeff. *Crowdsourcing: How the power of the crowd is driving the future of business*. Random House, 2008.

Jain, Anil K, Murty, M Narasimha, and Flynn, Patrick J. Data clustering: a review. *ACM Computing Surveys*, 31 (3):264–323, 1999.

Jain, Prateek, Meka, Raghu, and Dhillon, Inderjit S. Simultaneous unsupervised learning of disparate clusterings. *Statistical Analysis and Data Mining*, 1(3):195–210, 2008.

Kajino, Hiroshi, Tsuboi, Yuta, and Kashima, Hisashi. Clustering crowds. In *AAAI Conference on Artificial Intelligence*, 2013.

Kruskal, William H and Wallis, W Allen. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47(260):583–621, 1952.

Lichman, M. UCI machine learning repository, 2013. URL http://archive.ics.uci.edu/ml.

Maclaurin, D, Duvenaud, D, Johnson, M, and Adams, RP. Autograd: Reverse-mode differentiation of native python. *http://github. com/HIPS/autograd*, 2015.

Moreno, Pablo G, Artés-Rodríguez, Antonio, Teh, Yee Whye, and Perez-Cruz, Fernando. Bayesian nonparametric crowdsourcing. *Journal of Machine Learning Research*, 16:1607–1627, 2015.

Murphy, Sherry L, Xu, Jiaquan, and Kochanek, Kenneth D. Deaths: final data for 2010. *National vital statistics reports: from the Centers for Disease Control and Prevention, National Center for Health Statistics, National Vital Statistics System*, 61(4):1–117, 2013.

Ng, Andrew Y, Jordan, Michael I, Weiss, Yair, et al. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, volume 14, pp. 849–856, 2001.

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

Niu, Donglin, Dy, Jennifer G, and Ghahramani, Zoubin. A nonparametric bayesian model for multiple clustering with overlapping feature views. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, pp. 814–822, 2012.

Pillai, Sreekumar G, Ge, Dongliang, Zhu, Guohua, Kong, Xiangyang, Shianna, Kevin V, Need, Anna C, Feng, Sheng, Hersh, Craig P, Bakke, Per, Gulsvik, Amund, et al. A genome-wide association study in chronic obstructive pulmonary disease (copd): identification of two major susceptibility loci. *PLoS Genetics*, 5(3), 2009.

Rand, William M. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

Schmidt, Mark, Berg, Ewout, Friedlander, Michael, and Murphy, Kevin. Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, pp. 456–463, 2009.

Strehl, Alexander and Ghosh, Joydeep. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(12):583–617, 2002.

Tian, Yuandong and Zhu, Jun. Learning from crowds in the presence of schools of thought. In *Proceedings of the Eighteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 226–234, 2012.

Topchy, Alexander, Jain, Anil K, and Punch, William. Clustering ensembles: Models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1866–1881, 2005.

Von Luxburg, Ulrike. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

Wagstaff, Kiri and Cardie, Claire. Clustering with instance-level constraints. *AAAI Conference on Artificial Intelligence*, 1097, 2000.

Yi, Jinfeng, Jin, Rong, Jain, Shaili, Yang, Tianbao, and Jain, Anil K. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In *Advances in Neural Information Processing Systems*, pp. 1772–1780, 2012.

Zhang, Junjun, Baran, Joachim, Cros, Anthony, Guberman, Jonathan M, Haider, Syed, Hsu, Jack, Liang, Yong, Rivkin, Elena, Wang, Jianxin, Whitty, Brett, et al. International cancer genome consortium data portala one-stop shop for cancer genomics data. *Database*, 2011, 2011.