Clustering 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]
[1]Electrical and Computer Engineering Dept., Northeastern University, Boston, MA
[2]Brigham and Women's Hospital, Harvard Medical School, Boston, MA

Abstract

Utilizing expert input often improves clustering performance. However in a knowledge discovery problem, ground truth is unknown even to an expert. Thus, instead of one expert, we solicit the opinion from multiple experts. The key question motivating this work is: which experts should be assigned higher weights when there is disagreement on whether to put a pair of samples in the same group? To model the uncertainty in constraints from different experts, we build a probabilistic model for pairwise constraints through jointly modeling each expert's accuracy and the mapping from features to latent cluster assignments. After learning our probabilistic discriminative clustering model and accuracies of different experts, 1) samples that were not annotated by any expert can be clustered using the discriminative clustering model; and 2) experts with higher accuracies are automatically assigned higher weights in determining the latent cluster assignments. Experimental results on UCI benchmark datasets and a real-world disease subtyping dataset demonstrate that our proposed approach outperforms competing alternatives, including semi-crowdsourced clustering, semi-supervised clustering with constraints from majority voting, and consensus clustering.

1 Introduction

Given a dataset and a notion of similarity between samples, clustering aims to generate a partition on the dataset so that samples in the same group/cluster are similar and samples in different groups are dissimilar [9]. A key challenge in data clustering is how to define the similarity between samples. Depending on the user's interest (concept of similarity), the same dataset can be clustered from different perspectives. For example, a face dataset can be grouped based on either identity or pose; a set of marbles can be grouped based on either size or color. A way to address this challenge is to guide the clustering algorithm through human supervision, also called *semi-supervised clustering* [2]. Supervision is usually in the form of labels [1] or pairwise constraints between samples, including must-link (ML) and cannot-link (CL) constraints [20].

The accuracy of constraints is crucial to the performance of semi-supervised clustering. Instead of directly using constraints provided by a single expert, which might contain significant amount of noise and degrade the clustering performance, the combination of constraints provided by multiple experts usually lead to better clustering performance [8, 21, 22].

Crowdclustering combines constraints provided by multiple workers carrying out human intelligence tasks (HITs) to guide the clustering algorithm towards a better solution [8, 21]. The methods in [8, 21] assume each sample need to be annotated by at least one expert, limiting their use in practice because the required number of constraints will be too many if the sample size becomes large. *Semi-crowdsourced Clustering (SemiCrowd)* [22] combines constraints from multiple workers through computing the average similarity matrix between samples, applying matrix completion and then learning a distance metric. SemiCrowd assumes different workers have equal weights in generating the final solution since the constraints are built from the average similarity matrix between samples. This assumption is restrictive because different workers can have different levels of expertise in providing constraints.

In this paper, we consider a more practical scenario where there are multiple uncertain experts providing

Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54.

constraints for the same dataset. Note that our setup is close to *crowdclustering* and *SemiCrowd*, but we are using *experts* in a looser sense by allowing them to be constraints generated by either computer algorithms or human supervision. The uncertainties associated with multiple experts can be due to the following reasons: 1) The ground-truth cluster is unknown and need to be discovered; 2) There exist disagreements between experts; 3) Different experts may have varying levels of expertise. We also do not need every sample to be provided with constraints by an expert. *Our objective is to determine the best strategy to combine inconsistent constraints from multiple uncertain experts with different accuracies to improve clustering performance.*

This problem is motivated from the objective of subtyping a complex lung disease called Chronic Obstructive Pulmonary Disease. COPD is characterized by airflow limitation resulting from chronic inflammatory responses in the lungs to noxious particles or gases. COPD is currently the third leading cause of death in the United States [15]. It is widely accepted that COPD is a heterogeneous disease [4]; however, it is currently classified as one disease. Our goal is to discover the disease subtypes (clusters) in the hope of stratifying patients to enable personalized prognosis and treatment of patients. We would like to collect pairwise constraints provided by the experts to guide the clustering algorithm. One challenge is our clinicians have disagreements on whether to put two patients in the same subtype. Some investigators also applied machine learning algorithms on a subset of patients using variables they considered as important. We need to combine inconsistent constraints provided by clinicians and/or clustering labels generated from different machine learning algorithms. Intuitively, different experts should have varying levels of expertise. Furthermore, not all the patients are provided constraints by experts. This real-world problem is the primary motivation of this work.

To tackle the above problem, we build a probabilistic model for pairwise constraints from each expert by modeling each expert's accuracy. To avoid making assumptions on the generative process of clusters, we utilize a discriminative clustering model [7]. Compared to generative clustering methods, discriminative clustering approaches are more flexible and powerful in practice because they make fewer assumptions about the nature of clusters with fewer parameters needed and provides a natural out-of-sample extension. The learned discriminative clustering model can be used to cluster all the samples. Experts with higher accuracies are automatically assigned higher weights in determining the latent cluster assignments.

1.1 Contributions

In summary, the contributions of this work are: (1) we build a probabilistic model for constraints from multiple experts by explicitly modeling each expert's uncertainty; (2) we use a discriminative clustering model to achieve out-of-sample extension; (3) we demonstrate the proposed approach outperforms existing approaches on both UCI benchmark datasets and a real-world disease subtyping dataset.

This paper is organized as follows: in Section 2, we describe the problem and our approach; and in Section 3, we report experimental results on UCI benchmark datasets and a real-world disease subtyping dataset. Finally, we provide our conclusions in Section 4.

2 Proposed Approach

Given data matrix $X \in \mathbb{R}^{n \times d}$, where n is the number of samples and d is the number of features, and constraints provided by M experts, our objective is to categorize those n samples into K clusters by combining (possibly) inconsistent constraints from these M experts to improve clustering performance.

We assume the pairwise constraints from the m-th expert can be represented by an $n \times n$ similarity matrix $S^{(m)}$. In particular, if an expert provides must-link/cannot-link constraints, elements in $S^{(m)}$ take binary values: $S_{ij}^{(m)} = 1$ if the m-th expert gives must-link constraint for sample pair (x_i, x_j) and $S_{ij}^{(m)} = 0$ if cannot-link constraint is given. Note that $S_{ij}^{(m)}$ is denoted as unobserved if the m-th expert does not provide pairwise constraint for sample pair (x_i, x_j).

2.1 Discriminative Clustering Model

We have the following considerations when designing the clustering model: 1) It should be able to be used to model the uncertainties in the constraints provided by experts. Therefore, instead of hard-clustering, the cluster assignments of a sample to different clusters should be associated with probabilities. 2) To avoid making assumptions about the nature of clusters, which could be easily violated in real-world data, we decided not to use generative clustering models, such as Gaussian mixture model (GMM). 3) The clustering algorithm should be able to cluster samples that do not appear in the training set.

A discriminative clustering model satisfies all three requirements above. Assume the latent cluster assignments for n samples are denoted as $Z = (z_1, \cdots, z_n)^T$, where $z_i = k$ indicates the i-th sample belongs to the k-th cluster. In a discriminative clustering model, we

only need to specify $p(Z|X)$ and do not need to model $p(X)$. We assume $p(Z|X)$ follows a multiple logistic regression model:

$$p(z_i = k|x_i; W, \mathbf{b}) = \frac{\exp(\mathbf{w_k}^T x_i + b_k)}{\sum_{j=1}^{K} \exp(\mathbf{w_j}^T x_i + b_j)} \quad (1)$$

where $W = [\mathbf{w_1}, \cdots, \mathbf{w_K}] \in \mathbb{R}^{d\times K}$, $\mathbf{b} = [b_1, \cdots, b_K]^T \in \mathbb{R}^{K\times 1}$ are parameters needed to be learned. Each sample is assigned to the cluster associated with the largest probability value.

If only the data matrix X is available, the above discriminative model can be learned by maximizing $I(X, Z)$, the mutual information between data matrix X and cluster label Z with respect to (w.r.t.) parameters W and $\mathbf{b}$ [7]. Furthermore, to penalize the conditional models $p(Z|X)$ with complex decision boundaries, a regularization term $R(\lambda, W) = \lambda||W||_F^2$ can be added to the clustering objective to yield sensible clustering solutions. Instead of only maximizing $I(X, Z)$, the clustering quality associated with the data matrix, we also take into account the constraints provided by the M experts.

2.2 Experts' Constraint Model

We assume each expert is uncertain when providing constraints. As a result, the constraints provided by an expert do not necessarily agree with the latent cluster assignments Z. To model the uncertainty of the m-th expert, we assume the elements in matrix $S^{(m)}$, the pairwise constraints provided by the m-th expert, are generated from the following Bernoulli distributions.

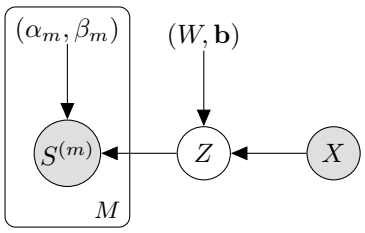
$$p(S_{ij}^{(m)}|z_i = z_j) = \alpha_m^{S_{ij}^{(m)}}(1-\alpha_m)^{1-S_{ij}^{(m)}} \quad (2)$$

$$p(S_{ij}^{(m)}|z_i \neq z_j) = \beta_m^{1-S_{ij}^{(m)}}(1-\beta_m)^{S_{ij}^{(m)}} \quad (3)$$

where α_m represents the m-th expert's *sensitivity*, i.e., the probability of assigning two samples that belong to the same cluster in the latent cluster assignments to the same cluster in the constraints. β_m represents the m-th expert's *specificity*, i.e., the probability of assigning two samples that belong to different clusters in the latent cluster assignments to different clusters in the constraints. Since different experts will naturally have different levels of expertise, we need to learn the individual sensitivity/specificity parameters for each of the M experts. We use $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_M)$, $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_M)$ to denote the sensitivity/specificity for all the M experts. When $\alpha_m = 0.5, \beta_m = 0.5$, it means that the constraints from the m-th expert is equivalent to random guess. On the other hand, when $\alpha_m = 1, \beta_m = 1$, it means perfect accuracy. We assume the constraints provided by the m-th expert cannot be worse than random guess by restricting the values of α_m, β_m to be lower bounded by 0.5 (and upper bounded by 1).

2.3 Graphical Model

Overall, the graphical model of our proposed approach is as follows:



where $p(Z|X; W, \mathbf{b})$ is defined by the discriminative clustering model based on multiple logistic regression described in subsection 2.1; $p(S^{(m)}|Z; \alpha_m, \beta_m)$ is defined by the Bernoulli distributions described in subsection 2.2.

2.4 Maximum Likelihood of Experts' Constraints

Given dataset X and M experts' constraints $S^{(1:M)}$, we define our objective as maximizing the regularized likelihood of experts' constraints w.r.t. parameters $\theta = \{W, \mathbf{b}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ with constraint conditions on $\boldsymbol{\alpha}, \boldsymbol{\beta}$ as follows:

$$\max_{\theta} p(S^{(1:M)}; \theta) - R(W; \lambda) \quad (4)$$
$$\text{s.t.} \quad 0.5 \leq \alpha_m, \beta_m \leq 1 \quad (m = 1, \cdots, M)$$

Since we have missing variable Z when computing $p(S^{(1:M)}) = \sum_Z p(S^{(1:M)}, Z)$, we can use the expectation maximization (EM) algorithm to solve our objective.

E-step: Compute $q(Z) = p(Z|X; W, b)$.

M-step: Maximize the expected complete-data log likelihood with the regularization term

$$\max_{\theta} E_{q(Z)}\left[\log p(S^{(1:M)}, Z; \theta)\right] - R(W; \lambda) \quad (5)$$

According to the rule of probability, we have $p(S^{(1:M)}, Z; \theta) = p(Z)p(S^{(1:M)}|Z; \theta)$, where $p(Z)$ is the prior distribution of cluster indicator Z. Assume $p(Z)$ is balanced, i.e. $p(z_i = k) = 1/K$, we have

$$E_{q(Z)}[\log p(Z)] = \sum_{i=1}^{n}\sum_{k=1}^{K} p(z_i = k|x_i) \log p(z_i = k) \quad (6)$$
$$= -n \log K$$

Therefore, we only need optimize the expectation of the log conditional likelihood with the regularization

$$\max_{\theta} E_{q(Z)}\left[\log p(S^{(1:M)}|Z;\theta)\right] - R(W;\lambda) \quad (7)$$

In practice, we divide the first term by N, the total number of observed entries in $S^{(1:M)}$, to improve numerical stability. We set the regularization term $R(W;\lambda) = \lambda||W||_F^2$. We also change the sign of the objective to convert the optimization problem from maximization to minimization. Therefore, we can rewrite the objective as follows:

$$\min_{\theta} -\frac{1}{N}\left(E_{q(Z)}[\log p(S^{(1:M)}|Z;\theta)]\right) + \lambda||W||_F^2 \quad (8)$$
$$\text{s.t.} \quad 0.5 \leq \alpha_m, \beta_m \leq 1 \quad (m = 1, \cdots, M)$$

The expectation of the log conditional likelihood can be expanded as follows:

$$\begin{aligned}
&E_{q(Z)}[\log p(S^{(1:M)}|Z;\theta)] \quad (9)\\
&= \sum_{m=1}^{M} \sum_{(i,j)\in E^{(m)}} E_{q(Z)}[\mathbb{I}(z_i = z_j)\log p(S_{ij}^{(m)}|z_i = z_j) + \\
&\mathbb{I}(z_i \neq z_j)\log p(S_{ij}^{(m)}|z_i \neq z_j)] \\
&= \sum_{m=1}^{M} \sum_{(i,j)\in E^{(m)}} \gamma_{ij}\log p(S_{ij}^{(m)}|z_i = z_j) + (1-\gamma_{ij}) \\
&\log p(S_{ij}^{(m)}|z_i \neq z_j)
\end{aligned}$$

where $E^{(m)}$ represents the indices of observed entries in $S^{(m)}$, $\gamma_{ij} = E_{q(Z)}[\mathbb{I}(z_i = z_j)]$, which can be rewritten as $\gamma_{ij} = \sum_{k=1}^{K} p(z_i = k|x_i)p(z_j = k|x_j)$. The above formula can be further expanded by substituting the likelihood term with the Bernoulli model defined in subsection 2.2:

$$\log p(S_{ij}^{(m)}|z_i = z_j) = S_{ij}^{(m)}\log\alpha_m + (1 - S_{ij}^{(m)})\log(1-\alpha_m)$$
$$\log p(S_{ij}^{(m)}|z_i \neq z_j) = (1 - S_{ij}^{(m)})\log\beta_m + S_{ij}^{(m)}\log(1-\beta_m)$$

We can observe that the coefficient of $S_{ij}^{(m)}$, the constraints provided by the m-th expert, is $\omega_m = \log\frac{\alpha_m\beta_m}{(1-\alpha_m)(1-\beta_m)}$. Because of this property, experts with higher accuracies (the values of α_m, β_m are higher), will be assigned higher weights when computing the weighted similarity matrix.

2.5 Optimization

The proposed objective can be minimized via alternative optimization.

2.5.1 Fix $\mathbf{b}, \boldsymbol{\alpha}, \boldsymbol{\beta}$, optimize w.r.t. W

To optimize the objective w.r.t. W, we apply a quasi-Newton algorithm, limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) because it has a good convergence rate and linear memory requirement [13, 17]. L-BFGS requires the gradient of the objective w.r.t. W. The key observation is only γ_{ij} and the regularization term $R(W;\lambda)$ are related to W. Since $\frac{\partial R(W;\lambda)}{\partial W} = 2\lambda W$, it will be sufficient to compute $\frac{\partial\gamma_{ij}}{\partial W}$. For clarity, we use p_{ik} to denote $p(z_i = k|x_i)$, then $\gamma_{ij} = \sum_{k=1}^{K} p_{ik}p_{jk}$, which leads to

$$\frac{\partial\gamma_{ij}}{\partial W} = \sum_{k=1}^{K}\left(\frac{\partial p_{ik}}{\partial W}p_{jk} + \frac{\partial p_{jk}}{\partial W}p_{ik}\right)$$

Since $\frac{\partial p_{ik}}{\partial \mathbf{w_t}} = p_{ik}(\mathbb{I}(k=t) - p_{it})x_i$, we have

$$\begin{aligned}
\frac{\partial p_{ik}}{\partial W} &= \left[\frac{\partial p_{ik}}{\partial \mathbf{w_1}}, \cdots, \frac{\partial p_{ik}}{\partial \mathbf{w_K}}\right] \\
&= p_{ik}x_i\left[\mathbb{I}(k=1) - p_{i1}, \cdots, \mathbb{I}(k=K) - p_{iK}\right]
\end{aligned}$$

2.5.2 Fix $W, \boldsymbol{\alpha}, \boldsymbol{\beta}$, optimize w.r.t. $\mathbf{b}$

Similar with W, it can be optimized with L-BFGS [13, 17]. Only γ_{ij} are related to $\mathbf{b}$, it is sufficient to compute $\frac{\partial\gamma_{ij}}{\partial \mathbf{b}}$. Since $\frac{\partial p_{ik}}{\partial b_t} = p_{ik}(\mathbb{I}(k=t) - p_{it})$, the gradient of γ_{ij} w.r.t. b_t can be written as

$$\begin{aligned}
\frac{\partial\gamma_{ij}}{\partial b_t} &= \sum_{k=1}^{K}\frac{\partial p_{ik}}{\partial b_t}p_{jk} + \frac{\partial p_{jk}}{\partial b_t}p_{ik} \\
&= \sum_{k=1}^{K} p_{ik}p_{jk}\left[2\mathbb{I}(k=t) - p_{it} - p_{jt}\right]
\end{aligned}$$

2.5.3 Fix $W, \mathbf{b}$, optimize w.r.t. $\boldsymbol{\alpha}, \boldsymbol{\beta}$

It can be optimized with the L-BFGS optimization with simple constraints proposed in [18]. Since only $\log p(S_{ij}^{(m)}|z_i = z_j)$ contains α_m in the objective, it is sufficient to compute

$$\frac{\partial\log p(S_{ij}^{(m)}|z_i = z_j)}{\partial\alpha_m} = S_{ij}^{(m)}\frac{1}{\alpha_m} + (1 - S_{ij}^{(m)})\frac{1}{\alpha_m - 1}$$

Similarly, since only $\log p(S_{ij}^{(m)}|z_i \neq z_j)$ contains β_m in the objective, it is sufficient to compute

$$\frac{\partial\log p(S_{ij}^{(m)}|z_i \neq z_j)}{\partial\beta_m} = (1 - S_{ij}^{(m)})\frac{1}{\beta_m} + S_{ij}^{(m)}\frac{1}{\beta_m - 1}$$

The objective function is nonconvex, therefore multiple initializations are required to help escape from local minima. In the experiments, we randomly initialize W and $\mathbf{b}$ by drawing each of their elements from a standard Gaussian distribution. $\boldsymbol{\alpha}, \boldsymbol{\beta}$ are initialized by drawing their elements from a uniform distribution between 0.5 and 1. We set the number of random initializations to be 20 and choose the one resulting in the minimal objective function value.

3 Experimental Results

In this section, we demonstrate our proposed approach can effectively combine constraints provided by multiple uncertain experts with varying levels of expertise to improve clustering performance.

3.1 Competing Alternatives

SemiCrowd: We use the SemiCrowd [22] as the first competing method. SemiCrowd handles inconsistencies between constraints from different experts by filtering out uncertain sample pairs in the average similarity matrix, and then applying matrix completion. In contrast, the crowdclustering method in [8] cannot be used because it requires every sample to be annotated by at least one expert. However, not all the samples are annotated by experts in our setting.

Semi-supervised Clustering: Another way to remove the inconsistencies in constraints collected from multiple experts is through majority voting. A pair of samples are assigned ML constraint if more than half of the experts provide ML constraints and they are assigned CL constraint otherwise. The resulting constraints can be combined with existing semi-supervised clustering algorithms to generate a clustering solution.

There are two different ways to improve clustering through incorporating constraints. In particular, metric learning approaches learn a distance metric so that the pairwise distances between samples in ML constraints become small and the pairwise distances between samples in CL constraints become large. We choose *Information-Theoretic Metric Learning (ITML)* [6] as the representative of metric learning approaches due to its superior performance compared to alternatives. On the other hand, constrained clustering reduces the searching space of clustering solution by respecting the constraints during the cluster discovery process. We choose *Constrained 1-Spectral Clustering (COSC)* [16] as the representative of constrained clustering due to its superior performance compared to alternatives. *Metric Pairwise Constrained KMeans (MPCKMeans)* [3], a widely used semi-supervised learning algorithm combining constrained clustering and metric learning, is also used as a competing method.

Consensus Clustering: Most consensus clustering algorithms are not designed for our setting, where not all the samples are labeled by experts [11]. However, since the average similarity matrix between samples is available, *Cluster-based Similarity Partitioning Algorithm (CSPA)* [19], a consensus clustering algorithm that only need the sample similarity matrix to cluster the samples, can be applied.

KMeans Clustering: We also include KMeans [14] on the original data matrix without considering expert input as a baseline.

We put the detailed parameter settings for each approach in the supplementary material due to space constraint.

3.2 UCI Benchmark Experiments

We first run our approach and competing approaches on eleven datasets collected from the UCI machine learning repository [12]. Their detailed information, including sample size, the number of features and the number of clusters, is summarized in Table 1.

Table 1: Summary of UCI datasets.

Dataset	Sample Size	Dimension	# Clusters
BreastCancer	569	30	2
Cleveland	297	13	5
Column	310	6	3
Dermatology	358	34	6
Glass	214	9	6
Heart	270	13	2
Ionosphere	351	34	2
Mushroom	8124	22	2
Newthyroid	215	5	3
Satimage	6435	36	6
Wine	178	13	3

Generating Constraints for Experts Constraints from multiple experts are not directly available in the UCI benchmark datasets. Therefore, we need generate them from the ground-truth clustering solution, i.e., the class labels. Here we generate constraints provided by M different experts with sensitivity/specificity parameters $\boldsymbol{\alpha}, \boldsymbol{\beta}$ according to the following steps: 1) Randomly sample q ML constraint pairs and q CL constraint pairs from the ground-truth clustering solution; 2) For the m-th expert with sensitivity α_m and specificity β_m, randomly select $\lfloor q(1-\alpha_m) \rfloor$ ML constraint pairs and flip them to CL constraint pairs; also randomly select $\lfloor q(1-\beta_m) \rfloor$ CL constraint pairs and flip them to ML constraint pairs.

Note that here we restrict multiple experts to annotate the common set of sample pairs. There are three reasons to design experiments in this way: 1) To ensure that there are enough number of sample pairs associated with conflicting constraints from different experts to highlight the differences of competing approaches; 2) To ensure that there are sufficient number of experts annotating a sample pair so that majority voting and average similarity computation, which are required by competing approaches, are both feasible and reasonable; 3) To allow the number of constraints to vary between a large range.

An alternative design choice is to use different sample pair sets for different experts and use a much larger number of experts to satisfy the above requirements. For ease of explanation, we choose the above expert

input generation strategy. Our approach still outperforms competing approaches under the alternative design choice.

For each dataset, we generate equal number of ML and CL constraints and vary the total number of constraints from 100 to 2000. The constraints are randomly generated according to the ground-truth cluster label and accuracy parameters of multiple experts. For each dataset and a number of constraints, we repeat the constraint generation process 10 times to create 10 independent sets of constraints. Then we repeat running the proposed approach and competing approaches 10 times, each time with a different set of constraints. We evaluate the clustering solution by computing normalized mutual information (NMI) [19] with the ground truth cluster labels. The value of NMI is between 0 and 1. Higher values indicate better matches with the ground truth label. For each dataset, each constraint set, and a competing approach, there are 10 NMI values. We report the mean and standard deviation of these 10 NMI values.

Here we set the number of experts M to be 5 and consider two different sets of accuracy parameters: **Case 1:** The accuracies of different experts are designed to be unequal by setting $\boldsymbol{\alpha} = \boldsymbol{\beta} = \{0.95, 0.85, 0.75, 0.65, 0.55\}$. **Case 2:** The accuracies of different experts are designed to be equal and of good quality by setting $\boldsymbol{\alpha} = \boldsymbol{\beta} = \{0.9, 0.9, 0.9, 0.9, 0.9\}$.

The errorbar plots of NMI values against the number of constraints corresponding to eight UCI datasets and accuracy parameter $\boldsymbol{\alpha} = \boldsymbol{\beta} = \{0.95, 0.85, 0.75, 0.65, 0.55\}$ are shown in Figure 1. For accuracy parameter $\boldsymbol{\alpha} = \boldsymbol{\beta} = \{0.9, 0.9, 0.9, 0.9, 0.9\}$, the errorbar plots are shown in Figure 2. Note that black dash lines represent the results of KMeans clustering. The figures of *Dermatology, Glass, Heart* are put in the supplementary material due to space constraint. We use the Kruskal-Wallis test [10] on two sets of NMI values to compare the performance of competing approaches.

From Figures 1 and 2 and those in the supplemental, we have a few observations. When experts have unequal accuracies, our proposed approach consistently outperforms SemiCrowd, ITML, COSC, MPCKMeans on all the eleven datasets. However, in the case where experts have equal accuracies of good quality, our proposed approach outperforms SemiCrowd, ITML, COSC, MPCKMeans on some datasets (*BreastCancer, Column, Ionosphere, Mushroom, Newthyroid*), but does not show clear advantage on the remaining datasets, i.e., at least one of the baseline approaches is equally good or even better. This can

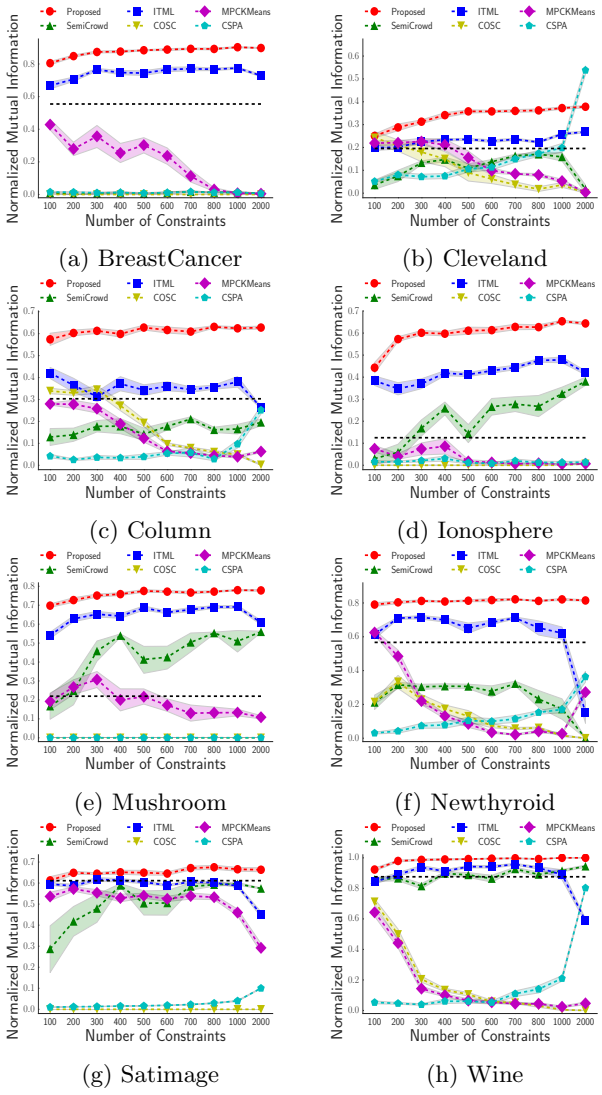
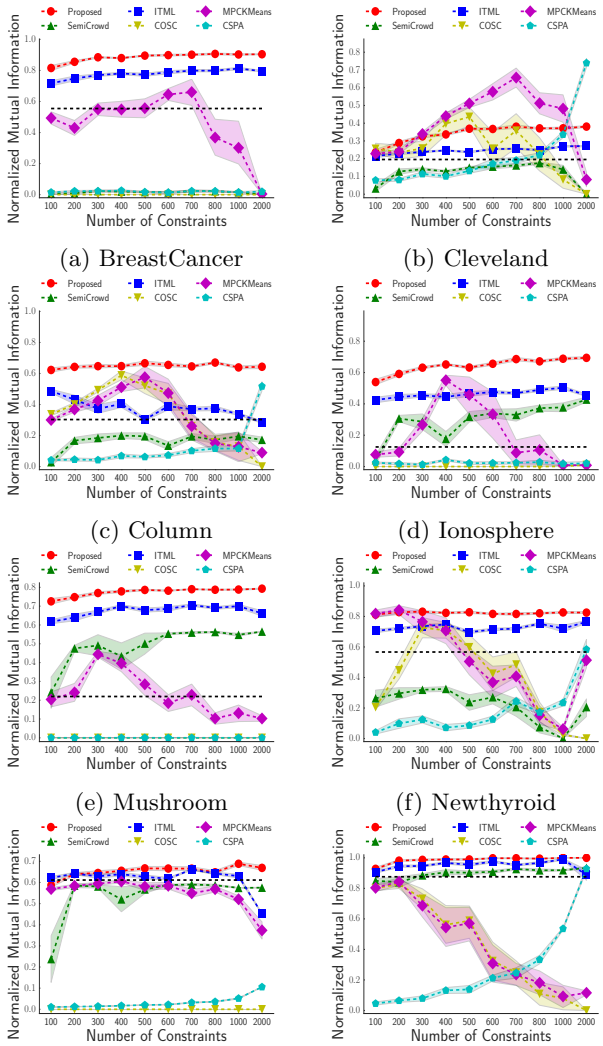


Figure 1: After setting accuracy parameter $\boldsymbol{\alpha} = \boldsymbol{\beta} = (0.95, 0.85, 0.75, 0.65, 0.55)$ to make different experts have unequal accuracies, each plot shows NMI against the number of constraints for competing approaches on one UCI dataset.

be explained by the fact that our proposed approach learns the accuracy of each expert. Higher accuracy of the m-th expert (α_m, β_m) leads to higher weight $\omega_m = \log \frac{\alpha_m \beta_m}{(1-\alpha_m)(1-\beta_m)}$ for the m-th expert.

To confirm accuracy/weight learned by our proposed approach actually works in this way, we plot ω_m $(m = 1, \cdots, 5)$, computed from the recovered α_m, β_m when the number of constraints is 500 and plot eight of them in Figure 3. As we can see, for datasets *BreastCancer, Dermatology, Mushroom, Wine*, the recovered expert weights are very close to their theoretical values. For other datasets, the recovered expert weights are smaller than their theoretical values. This can be explained by the fact that different clusters in those datasets cannot be perfectly separated by the linear



(a) BreastCancer (b) Cleveland

(c) Column (d) Ionosphere

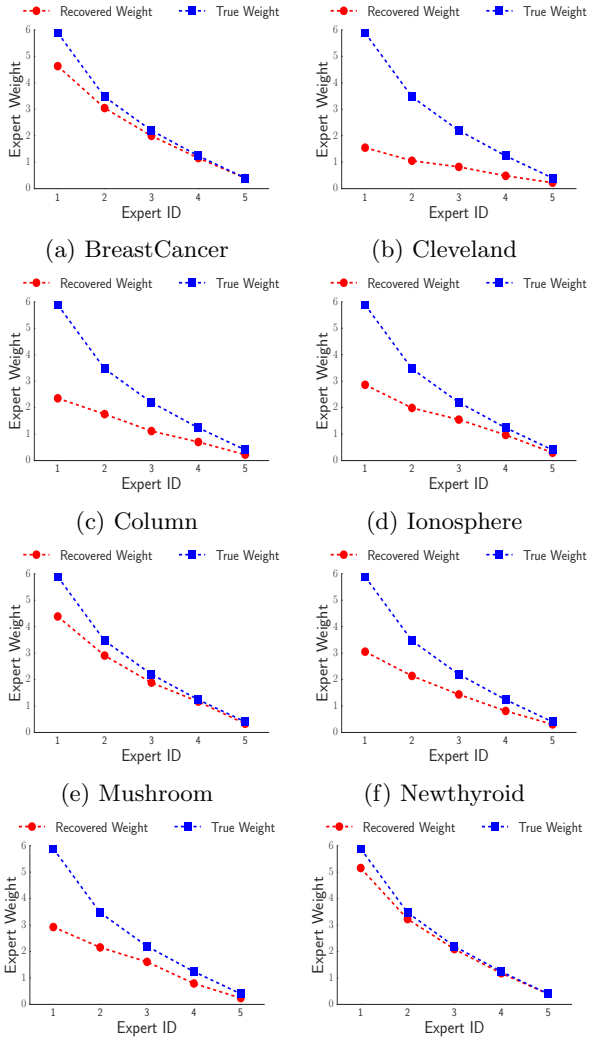
(e) Mushroom (f) Newthyroid

(g) Satimage (h) Wine

Figure 2: After setting accuracy parameter $\boldsymbol{\alpha} = \boldsymbol{\beta} = (0.9, 0.9, 0.9, 0.9, 0.9)$ to make different experts have equal accuracies of good quality, each plot shows NMI against the number of constraints for competing approaches on one UCI dataset.

discriminative clustering approach used in our model. However, the experts with higher accuracies are still assigned higher weights, which can explain the advantage of our proposed approach compared to all the competing methods in the unequal accuracies scenario.

The performance of CSPA is only comparable to our proposed approach when the number of constraints is sufficiently large. For example, on the *Wine* dataset, the performance of CPSA is comparable only when the number of constraints reaches 2000, an arguably large number of constraints for 178 samples. A similar behavior can be observed on a few other datasets (*Cleveland, Dermatology, Glass, Newthyroid*). However, for all three datasets with more than 500 samples (*BreastCancer, Mushroom, Satimage*), the performance of CSPA is still not comparable to our approach when the number of constraints reaches 2000. Only with a much larger number of constraints, can the performance of CSPA be comparable to our approach.



(a) BreastCancer (b) Cleveland

(c) Column (d) Ionosphere

(e) Mushroom (f) Newthyroid

(g) Satimage (h) Wine

Figure 3: After setting accuracy parameter $\boldsymbol{\alpha} = \boldsymbol{\beta} = (0.95, 0.85, 0.75, 0.65, 0.55)$ to make different experts have unequal accuracies, each plot shows the recovered expert weights (in red) and true expert weights (in blue) for all experts on a UCI dataset.

3.3 COPD Subtyping Experiments

Chronic Obstructive Pulmonary Disease (COPD) is a complex lung disease characterized by airflow limitation resulting from chronic inflammatory responses in the lungs to noxious particles or gases. It is currently classified as one disease; however, COPD is known to be a heterogeneous disease and our objective is to discover subtypes (clusters). We collected a dataset containing 987 patients with COPD. For each patient, we extracted 39 features, including demographics, clinical information, lung function, and measures from com-

puted tomography (CT) chest imaging.

We have a cohort of investigators (pulmonologists, radiologists, data analysts working with clinicians) and each investigator (expert) defined subtypes, and therefore provided constraints, according to what they consider as important for subtyping this set of patients. For example, a set of clinicians defined subtypes based primarily on CT images. Another set of subtypes was generated from data scientists working with clinicians to apply machine learning algorithms. Each expert only used a subset of samples and features to define subtypes. Therefore, their outputs are in the form of partial labels, which have an equivalent representation of pairwise constraints. We use constraints provided by 27 experts.

Evaluation Since there is no ground truth available for this dataset, we can no longer compare competing approaches by computing NMI. While the characteristics of optimal clusters in COPD are not known, disease experts may consider properties that the solutions must have in order to be meaningful. For example, in COPD and other diseases, *mortality* is a critical outcome. Other key measures in COPD include decline in lung function (*FEV1 decline*), and differences in risk scores derived from genetic variants (*copdScore*) [5].

We randomly split the patients into two sets of equal sizes, one for training and the other for validation. We first run competing approaches on the training set, samples in the validation set are then assigned cluster labels. To check whether patients in different clusters show significant difference on those three key variables described above, including *FEV1 decline, mortality* and *copdScore*, we compute the p-values on those variables. Both *FEV1 decline* and *copdScore* are real-valued, therefore we compute their p-values using the Kruskal-Wallis test. In contrast, since *mortality* is a binary variable, we compute its p-value using χ^2 test.

Results Since there is no out-of-sample extension for COSC and CSPA, we can only compare our proposed approach against the other four competing methods: including SemiCrowd, ITML, MPCKMeans and KMeans in the COPD subtyping experiment. The p-values on three key variables corresponding to each approach are shown in Table 2. We observe that only our proposed approach obtains significant p-values (less than 0.05) on all three variables. In particular, our proposed approach is the only one to achieve a significant p-value on *FEV1 decline*, which means that our proposed approach can identify 4 COPD subtypes with significantly different lung function decline.

We also plot the recovered weights of the 27 experts in Figure 4. We noticed that most of the top-raking experts are contributed by clinicians, who use their domain knowledge to provide constraints, which is consistent with our expectation.

Table 2: P-values on three key variables (*FEV1 decline, mortality, copdScore*) of competing approaches

Approach	*FEV1 decline*	*mortality*	*copdScore*
Proposed	**1.41e-2**	**7.59e-6**	**4.52e-4**
SemiCrowd	6.93e-2	**2.14e-4**	**4.00e-2**
ITML	0.43	8.15e-2	**4.75e-3**
MPCKMeans	0.44	**3.97e-7**	**1.99e-3**
KMeans	0.50	**2.14e-6**	**2.29e-4**

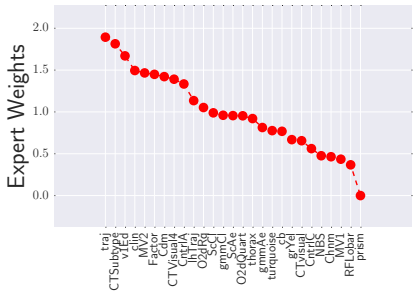


Figure 4: Recovered weights of 27 experts providing constraints for the COPD subtyping

4 Conclusions

In this paper, we have introduced a novel probabilistic model for clustering from multiple uncertain experts who may have varying levels of accuracies. Through learning a discriminative clustering model, samples that do not have constraints provided by an expert can be assigned cluster labels using the discriminative clustering model. After recovering the accuracy of each expert, constraints provided by experts with higher accuracies are assigned higher weights in determining the latent cluster assignments. Experimental results on UCI benchmark datasets and a real world disease subtyping dataset demonstrate our proposed approach outperforms competing alternatives, including semi-crowdsourced clustering, semi-supervised clustering with constraints obtained through majority voting, and consensus clustering.

5 Acknowledgements

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

References

[1] S. Basu, A. Banerjee, and R. Mooney. Semi-supervised clustering by seeding. In *ICML*. Citeseer, 2002.

[2] S. Basu, I. Davidson, and K. Wagstaff. *Constrained clustering: Advances in algorithms, theory, and applications*. CRC Press, 2008.

[3] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *ICML*. ACM, 2004.

[4] P. J. Castaldi, J. Dy, J. Ross, Y. Chang, G. R. Washko, D. Curran-Everett, A. Williams, D. A. Lynch, B. J. Make, J. D. Crapo, et al. Cluster analysis in the copdgene study identifies subtypes of smokers with distinct patterns of airway disease and emphysema. *Thorax*, 2014.

[5] M. H. Cho, M.-L. N. McDonald, X. Zhou, M. Mattheisen, P. J. Castaldi, C. P. Hersh, D. L. DeMeo, J. S. Sylvia, J. Ziniti, N. M. Laird, 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.

[6] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216. ACM, 2007.

[7] R. G. Gomes, A. Krause, and P. Perona. Discriminative clustering by regularized information maximization. In *NIPS*, pages 775–783, 2010.

[8] R. G. Gomes, P. Welinder, A. Krause, and P. Perona. Crowdclustering. In *NIPS*, pages 558–566, 2011.

[9] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323, 1999.

[10] W. H. Kruskal and W. A. Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621, 1952.

[11] T. Li, C. Ding, and M. I. Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *ICDM*, pages 577–582. IEEE, 2007.

[12] M. Lichman. UCI machine learning repository, 2013.

[13] D. C. Liu and J. Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[14] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA., 1967.

[15] S. L. Murphy, J. Xu, and K. D. Kochanek. 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.

[16] S. S. Rangapuram and M. Hein. Constrained 1-spectral clustering. In *AISTATS*, pages 1143–1151, 2012.

[17] M. Schmidt. minfunc: unconstrained differentiable multivariate optimization in matlab. 2012.

[18] M. W. Schmidt, E. Berg, M. P. Friedlander, and K. P. Murphy. Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm. In *AISTATS*, page None, 2009.

[19] A. Strehl and J. Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *JMLR*, 3:583–617, 2003.

[20] K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl, et al. Constrained k-means clustering with background knowledge. In *ICML*, volume 1, pages 577–584, 2001.

[21] J. Yi, R. Jin, A. K. Jain, and S. Jain. Crowdclustering with sparse pairwise labels: A matrix completion approach. In *AAAI Workshop on Human Computation*, volume 2. Citeseer, 2012.

[22] J. Yi, R. Jin, S. Jain, T. Yang, and A. K. Jain. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In *NIPS*, pages 1772–1780, 2012.