Clustering with Domain-Specific Usefulness Scores

Yale Chang* ychang@coe.neu.edu
Junxiang Chen* jchen@ece.neu.edu
Michael H. Cho† remhc@channing.harvard.edu
Peter J. Castaldi† repjc@channing.harvard.edu
Edwin K. Silverman† reeks@channing.harvard.edu
Jennifer G. Dy* jdy@ece.neu.edu

Abstract

Clustering is a challenging problem because given the same data set, it can be grouped in multiple different ways. Which of these clustering solutions is interesting depends on its domain application. Thus, incorporating domain expert input often improves clustering performance. However, most existing semi-supervised clustering techniques can only incorporate instance-level constraints (a few labels or must-link/cannot-link constraints), which domain experts may not be comfortable providing in knowledge discovery problems because categories are not known. Fortunately, domain experts often have an idea regarding properties that clustering solutions should have in order to be useful in domain application based on domain relevant scores. In this paper, we provide a framework for jointly optimizing the *usefulness* and *quality* of a clustering solution. Experiments on a synthetic data, a benchmark data, and a real-world disease subtyping problem demonstrate the usefulness of our proposed approach.

1 Introduction

A popular approach for exploratory data analysis is clustering. Cluster analysis is the process of grouping similar samples together and dissimilar samples in different groups based on some notion of *similarity* [1]. The results of clustering algorithms are highly dependent on how this notion of similarity is defined.

Similarity is typically defined by a metric or a probability model, which are based on the features representing the data. Depending on the feature sets used to compute similarity between samples, different clustering solutions can be generated. However, not all the features originally provided to represent the data are important. Instead of using all the input features to compute similarity, a good clustering algorithm should be able to learn a subspace that contains good clustering structure through dimensionality reduction.

However, which dimensions should we keep? The problem with clustering is that there is no one criterion that works best for all applications [2]. Moreover, a given data set can be grouped in multiple different ways [3]. For example, a face image data can be grouped together based on identity or pose. Furthermore, given the same data, what is interesting to an insurance agent is different from what a physician is interested in.

Fortunately, experts usually have some idea on the desired properties of the clustering solution they are interested in. In a knowledge discovery process, where categories of the data are not known (as they are yet to be discovered), it might be difficult for domain experts to provide supervision in the form of labels for a subset of samples and even providing must-link/cannot-link constraints for pairs of samples might also be difficult. Therefore, existing techniques [4, 5] that can only incorporate supervision at the instance level (labels or must-link/cannot-link constraints), such as semi-supervised classification [4] and semi-supervised clustering [5], cannot be directly used in this scenario. On the other hand, experts usually have some global level criterion of the *usefulness* of an overall clustering solution based on some domain relevant *scores*.

The idea of utilizing a score-based constraint was motivated from the objective of subtyping a complex lung disease called Chronic Obstructive Pulmonary Disease (COPD). Most clinicians agree that patients suffering from COPD should be clustered into different groups [6] so that each patient can receive personalized treatments. Our goal is to discover the disease subtypes (clusters). Because clusters are unknown, clinicians cannot provide any labeled supervision. When asked to provide pairwise must-link/cannot-link constraints, clinicians were reluctant to provide these because they do not agree on whether or not two patients belong to the same cluster. While the optimal clusters are not known, clinicians specify some properties that the candidate solutions must have in order to be clinically meaningful. For example, since it is known that COPD is related to risk scores derived from genetic variants, the mean-

*ECE Department, Northeastern University
†Brigham and Women's Hospital, Harvard Medical School

ingful clustering solution should contain clusters with significantly different genetic risk scores. This led us to design clustering algorithms that can utilize usefulness properties based on domain relevant scores. Note that the scores are provided by domain experts. For example, we use genetic risk scores in COPD subtypes discovery. Another example is that survival time can also be used as a score variable in cancer subtypes discovery [7]. A clustering solution should be related to the domain relevant scores in order to be meaningful in domain application.

In this paper, we introduce a framework for jointly optimizing the *usefulness* and *quality* of a clustering solution. We call our approach *Domain induced Score based Clustering (DiSC)*. We define the *usefulness* of a clustering solution as the global level property measured by score variable/s that domain experts care about. Cluster *quality* is the internal criterion optimized by the respective base clustering algorithm. Optimizing clustering quality helps limit the search space to subspaces that contain high-quality natural clusters. Furthermore, among all the candidate subspaces containing good clustering structure, optimizing usefulness helps find the subspace that is relevant to the scores that domain experts care about. As a concrete example of the proposed framework, we chose spectral clustering [8, 9] as our base clustering algorithm because this method has the advantage of allowing discovery of non-linear cluster structures. To optimize for usefulness, we utilize a measure that can also capture non-linear dependencies between the clustering structure and score variable/s. We chose the Hilbert Schmidt independence criterion (HSIC) [10] for this purpose. Experiments on a synthetic data, a benchmark data, and on the COPD disease subtyping application show that our proposed approach can discover more meaningful clusters compared to competing algorithms.

1.1 Related Work There are several ways in which supervised input has been incorporated to improve the performance of clustering algorithms. One type of input is in the form of providing a few labeled samples [11]. The labeled samples are then used to initialize clustering assignments or force the clustering algorithm to respect those initial labels. However, in knowledge discovery problems (like the problems we are dealing with in this paper), the category labels may not be known. Thus, another type of supervision in the form of must-link/cannot-link constraints has been introduced [5]. There are various ways the must-link/cannot-link constraints can be used to improve clustering. In particular, constrained clustering algorithms learn cluster assignments such that the assignments respect the constraints provided; distance metric learning approaches learn a distance metric that satisfies the constraints. More recently, other forms of supervision have been suggested, such as more complex logical conjunctive normal form constraints [12]. Then again, in some applications, experts may still find it difficult to provide must-link/cannot-link constraints, or these complex constraints. In contrast to these existing work on semi-supervised clustering which incorporate instance-level constraints, our proposed approach improves clustering by incorporating expert guidance in the form of usefulness properties as measured by scores experts care about in a global manner.

In semi-supervised classification, given large amount of unlabeled data, human supervision is used to label a small set of samples to build better classifiers [4]. Note that semi-supervised classification focuses on the classification problem, with a classifier as the base learning algorithm; on the other hand, the DiSC focuses on the clustering problem, with a clustering algorithm as the base learning algorithm.

1.2 Contributions In summary, the contributions of this work are: (1) we propose a novel way of incorporating domain knowledge, which is through a domain-specific *usefulness* score criterion that provides global guidance regarding desired properties of a clustering solution as opposed to current approaches that utilize instance-level constraints; (2) we provide an optimization procedure to jointly optimize the quality of spectral clustering and the usefulness criterion; and (3) we demonstrate that the proposed method applied to a real-world problem of disease subtyping was able to discover meaningful clusters for stratifying COPD patients.

This paper is organized as follows: in Section 2, we provide the general formulation, background on spectral clustering and HSIC as well as optimization algorithms to solve the overall formulation; and in Section 3, we report experimental results on a synthetic dataset and a real-world dataset. Finally, we provide our conclusions in Section 4.

2 General Framework

Given dataset $X \in \mathbb{R}^{n \times d}$, where n is the number of samples and d is the number of features, and score $s \in \mathbb{R}^{n \times r}$ (where r is the number of scores) that is determined by experts based on their domain knowledge, our objective is to discover a subspace $\hat{X} = XW$, where $W \in \mathbb{R}^{d \times q}$ is the projection matrix that maps the data from its original d-dimensional space to its q-dimensional subspace, that contains good clustering structure and is useful as

evaluated by score s. The overall objective is as follows:

$$\max_{W,U} \mathcal{Q}(XW, U) + \mu\mathcal{R}(XW, s) \tag{2.1}$$

where U is the cluster assignment matrix and $\mathcal{Q}(XW, U)$ represents the quality of clustering structure in subspace XW, $\mathcal{R}(XW, s)$ represents the usefulness of subspace XW evaluated by score s. μ controls the trade-off between clustering quality and usefulness criterion.

The form of function $\mathcal{Q}$ is determined by the choice of clustering algorithm. There are many clustering algorithms available in the literature [1]. In this paper, we use spectral clustering because it can handle different data types and discover clusters of flexible (non-convex) shapes. Furthermore, many common clustering algorithms, including K-means and kernel K-means, can be viewed as special cases of spectral clustering [13].

The form of function $\mathcal{R}$ is defined by domain experts. Experts usually do have some features of interest (scores) $s \in \mathbb{R}^{n\times r}$ at a global level. The interesting subspace is expected to be related to these scores, i.e., *if two samples are close measured by their score values, they are expected to be close in the interesting subspace.* For example, in disease subtyping, clinicians aim to discover a subspace of observed clinical features (such as heart rate, blood pressure, weight, age, etc.) that contains good clustering structure (to provide personalized treatment according to the cluster a patient belongs to) and is related to disease risk scores derived from genetics (to confirm the clusters are indeed separated due to the disease rather than other factors). In this example, $s \in \mathbb{R}^{n\times 1}$ is given by disease risk scores based on genetics, where s_i indicates the risk scores of the i-th patient. Another example is to discover cancer subtypes from the gene expression data guided by score $s \in \mathbb{R}^{n\times 1}$, the survival time of n cancer patients. Two cancer patients with close survival time are more likely to suffer from the same type of cancer. Note that we allow $s \in \mathbb{R}^{n\times r}(r \geq 1)$ to have more than one dimension if multiple score variables are available. To incorporate this kind of domain knowledge, we should maximize the dependence between the desired subspace and the score variable/s.

In this paper, we use HSIC to measure the dependence between clustering structure and scores, which represents the usefulness of clustering. We choose HSIC because 1) it naturally capture our requirement that if two samples are close measured by their score values, they are expected to be close in the interesting subspace, 2) it has a closed-form estimator, which enables the easy use of numerical optimization methods. Although mutual information (MI) is also widely used to measure non-linear dependence between random variables, its estimator does not have a closed-form formula [14]. Therefore, MI has to be re-estimated for each update of the data subspace, leading to high computational cost. Below we give an overview on spectral clustering and HSIC.

2.1 Review on Spectral Clustering Given n data points $\{x_1, \cdots, x_n\}$ to be clustered and a notion of similarity $k(x_i, x_j)$ representing the similarity between two samples x_i and x_j. The data can be represented in the form of the *similarity graph* $G = (V, E)$. Each vertex v_i in the graph represents a data point x_i. The weight of edge $e_{i,j}$ is given by similarity value $k(x_i, x_j)$. The goal of spectral clustering is to group all the samples into c partitions $A_1, \cdots, A_c$, such that samples in the same partition (cluster) are similar and samples in different partitions are dissimilar.

There are several ways to define the graph partition objective. In this paper, we use normalized cut function $\text{Ncut}(A_1, \cdots, A_c) = \sum_{i=1}^{c} \frac{\text{cut}(A_i, \bar{A}_i)}{\text{vol}(A_i)}$, where $\bar{A}_i$ is the complement of set A_i among the whole set V, $\text{cut}(A_i, \bar{A}_i)$ is defined as $\text{cut}(A_i, \bar{A}_i) = \sum_{v_j \in A_i, v_k \in \bar{A}_i} k(x_j, x_k)$ (i.e., the sum of the weights of the edges between samples in set A_i and in set $\bar{A}_i$), degree, $d_j = \sum_{k=1}^{n} k(x_j, x_k)$, is the degree of vertex v_j, and $\text{vol}(A_i)$ is defined as $\text{vol}(A_i) = \sum_{v_j \in A_i} d_j$ (i.e., the sum of the degrees of all vertices in set A_i). By minimizing Ncut, there will be less edge weights between samples in different sets, making samples in different partitions become dissimilar. Minimizing Ncut with respect to partitions $A_1, \cdots, A_c$ is an NP-hard combinatorial problem [15].

Spectral clustering relaxes the discrete optimization problem into a continuous optimization by introducing clustering assignment matrix $U \in \mathbb{R}^{n\times c}$ and convert the problem into solving the following optimization problem:

$$\min_{U \in \mathbb{R}^{n\times c}} \text{Tr}(U^T L U) \qquad \text{s.t.} \qquad U^T U = I \tag{2.2}$$

where $L = I_n - D^{-1/2} K D^{-1/2}$ is the normalized graph Laplacian matrix, I_n is the $n \times n$ identity matrix, $K \in \mathbb{R}^{n\times n}$ is the kernel matrix with $K_{i,j} = k(x_i, x_j)$, and $D \in \mathbb{R}^{n\times n}$ is a diagonal matrix with $D_{i,i} = d_i$. The clustering assignment matrix U can be solved by taking the eigenvectors of matrix L corresponding to the c smallest eigenvalues. To obtain the discrete clustering partition, each row of U is normalized to have unit length and K-means clustering is applied to group n rows of U into c categories. Each data point x_i is assigned to the same cluster that row u_i is assigned to.

2.2 Review on HSIC The Hilbert-Schmidt independence criterion (HSIC) is introduced to measure the

statistical dependence between random variables [10]. Given $Z := \{(x_1, y_1), \cdots, (x_n, y_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$ that are n independent samples from p_{xy}, an estimator of HSIC is given by

$$\text{HSIC}(X, Y) := (n-1)^{-2}\text{Tr}(KHLH) \tag{2.3}$$

where $K, H, L \in \mathbb{R}^{n\times n}$, $K_{ij} = k(x_i, x_j)$, $L_{ij} = l(y_i, y_j)$, $H_{ij} = \delta_{ij} - 1/n$ and $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. If sample pairs x_i, x_j are close (meaning K_{ij} becomes large), the corresponding sample pairs y_i, y_j should also be close (meaning L_{ij} is large) in order to maximize HSIC(X, Y). Due to this property, HSIC(XW, s) can be used to capture the requirement that if two samples are close measured by their score values, they are expected to be close in the interesting subspace,

2.3 Overall Formulation Given dataset X, not all the features are useful in defining clustering structure. In this paper, we learn the subspace instead of directly using the original data space to discover clustering structure of high quality. Among the subspaces that contain clustering structure of high quality, we further define the interesting subspace to be the one that is relevant to the scores that domain experts consider as important. From the framework given by Eq. 2.1, we define cluster quality $\mathcal{Q}$ as the negative objective of spectral clustering given in Eq. 2.2 and $\mathcal{R}$ as the dependence between subspace XW and score s measured by HSIC , giving us the following objective function

$$\max_{W,U} -\text{Tr}(U^T L_W U) + \mu\text{Tr}(KHK_sH) \quad \text{s.t.}\, U^T U = I_k,\, W^T W = I_q \tag{2.4}$$

or equivalently

$$\min_{W,U} \text{Tr}(U^T L_W U) - \mu\text{Tr}(KHK_sH) \quad \text{s.t.}\, U^T U = I_k,\, W^T W = I_q \tag{2.5}$$

where L_W is the graph Laplacian matrix constructed from data subspace $\hat{X} = XW$ and kernel function $k_c(\cdot,\cdot)$, i.e. $K_{i,j} = k_c(W^T x_i, W^T x_j), L_W = I - D^{-1/2}KD^{-1/2}$. K_s is the kernel matrix constructed from s and kernel function $k_s(\cdot,\cdot)$, i.e., $K_{si,j} = k_s(s_i, s_j)$. Note that $k_c(\cdot,\cdot)$, the kernel function used in the data subspace, and $k_s(\cdot,\cdot)$, the kernel function used for the s, do not necessarily have to be the same. $U \in \mathbb{R}^{n\times k}$ represents the clustering assignment matrix and k is the desired number of clusters.

2.4 Optimization Given a random initialization of W that satisfies orthogonality constraint, we can solve the optimization problem by alternatively optimizing for W and U in an iterative manner.

2.4.1 Fix W, Optimize U Since only the first term contains U, the optimal U can be obtained by the same way that is used in spectral clustering, i.e., taking the eigenvectors of graph Laplacian matrix L_W corresponding to its k smallest eigenvalues.

2.4.2 Fix U, Optimize W This is an orthogonality-constrained minimization problem. The feasible set defined by the orthogonality constraint is called a Stiefel manifold, which is non-convex.

$$\min_{W\in\mathbb{R}^{d\times q}} g(W) \quad \text{s.t.}\, W^T W = I_q$$

Gradient descent on the Stiefel manifold [16] is generally used to satisfy the orthogonality constraint. However, this procedure is computationally inefficient due to the use of the matrix exponential. Therefore, we make use of a more efficient approach based on the Cayley transform [17].

Given a feasible point W and the gradient $G = \frac{\partial g}{\partial W}$, a skew-symmetric matrix A is defined as

$$A = GW^T - WG^T = (P_W G)W^T - W(P_W G)^T$$

where $P_W = I - \frac{1}{2}WW^T$. The update procedure is

$$W^{(t+1)} = QW^{(t)} \tag{2.6}$$

where $Q = (I + \frac{\tau}{2}A)^{-1}(I - \frac{\tau}{2}A)$, A is computed from $W^{(t)}$ and the step size τ can be obtained from the Armijo-Wolfe conditions [18]. The transformation defined by Q in Eq. 2.6 is called the Cayley transform. The key properties of this update are follows:

1) Constraint Preserving
$W^{(t+1)T}W^{(t+1)} = I$ holds if $W^{(t)T}W^{(t)} = I$ holds. Therefore, the orthogonality constraint can be satisfied during the update iterations if $W^{(0)}$, the initialization of W, is set to be an orthogonal matrix.

2) Projected Gradient Descent
It can be verified that $W^{(t+1)}|_{\tau=0} = W^{(t)}$, and $\frac{dW^{(t+1)}}{d\tau}|_{\tau=0}$ equals the projection of $-\frac{\partial g}{\partial W}$, the opposite direction of the gradient w.r.t. W, on the tangent space defined by the orthogonality constraint. Therefore, $W^{(t+1)}|_{\tau>0}$ will move along the direction derived from projected gradient descent and decrease the objective function value.

The computation of gradient $G = \frac{\partial g(W)}{\partial W}$ involves two terms: $J_1(W) = \text{Tr}(U^T L_W U)$ and $J_2(W) = \text{Tr}(KHK_sH)$. The first term $J_1(W)$ can be written as

$$\begin{aligned} J_1(W) &= \text{Tr}(U^T U) - \text{Tr}(U^T \tilde{K} U) \quad (\tilde{K} = D^{-1/2}KD^{-1/2}) \\ &= k - \sum_{i=1}^{k}\sum_{j=1}^{n}\sum_{l=1}^{n} U_{ji}U_{li}\tilde{K}_{jl} \end{aligned}$$

its gradient is determined by the gradient of $\tilde{K}_{jl}$

$$\frac{\partial J_1(W)}{\partial W} = -\sum_{i=1}^{k}\sum_{j=1}^{n}\sum_{l=1}^{n} U_{ji}U_{li}\frac{\partial \tilde{K}_{jl}}{\partial W}$$

The gradient of $\tilde{K}_{jl}$ can be computed as follows

$$\begin{aligned}\frac{\partial \tilde{K}_{jl}}{\partial W} &= \frac{\partial}{\partial W} d_j^{-1/2} K_{jl} d_l^{-1/2} \\ &= d_j^{-1/2} d_l^{-1/2} \frac{\partial K_{jl}}{\partial W} - \frac{1}{2} d_j^{-3/2} K_{jl} d_l^{-1/2} \frac{\partial d_j}{\partial W} \\ &\quad - \frac{1}{2} d_l^{-3/2} K_{jl} d_j^{-1/2} \frac{\partial d_l}{\partial W}\end{aligned}$$

where $\frac{\partial d_j}{\partial W} = \sum_{t=1}^{n} \frac{\partial K_{jt}}{\partial W}$. The computation of $\frac{\partial K_{jl}}{\partial W}$ depends on the form of the kernel function. For example, if a Gaussian kernel $k(W^T x_j, W^T x_l) = \exp(-\frac{||W^T x_j - W^T x_l||_2^2}{2\sigma^2})$ is used, the corresponding gradient is

$$\frac{\partial K_{jl}}{\partial W} = -\frac{1}{\sigma^2} K_{jl}(x_j - x_l)(x_j - x_l)^T W$$

The second term can be written as

$$\begin{aligned}J_2(W) &= \mathrm{Tr}(K\tilde{K}_s) \quad (\tilde{K}_s = HK_sH) \\ &= \sum_{j=1}^{n}\sum_{l=1}^{n} K_{jl}(\tilde{K}_s)_{ij}\end{aligned} \tag{2.7}$$

its gradient is determined by the gradient of K_{jl}. Combining the gradient of $J_1(W)$ and $J_2(W)$, the gradient of the overall objective function can be computed as

$$\frac{\partial g(W)}{\partial W} = \frac{\partial J_1(W)}{\partial W} - \mu \frac{\partial J_2(W)}{\partial W}$$

A pseudo-code of our algorithm, *Domain induced Score based Clustering (DiSC)* is shown in Algorithm 1.

Complexity Analysis When optimizing U, the naive implementation of the eigen-decomposition of the Laplacian matrix would introduce time complexity that is cubic w.r.t. the sample size, which is too costly. This probem can be addressed by applying the combinatorial multigrid solver proposed in [19]. The proposed solver in [19] utilizes the supported theory of graphs related to the Laplacian matrix and only introduce time complexity $\mathcal{O}(K \cdot n \cdot \log^2 n)$, where K is the number of clusters and n is the number of samples. When optimizing W: 1) the time complexity of computing the kernel matrix is $\mathcal{O}(d \cdot n^2)$; and 2) the time complexity of computing the gradient w.r.t. W and Cayley transform is $\mathcal{O}\left((d+q)n^2 + Kn + d^3\right)$. Since $q < d$, $K < n$ and $\log^2 n$ can be upper bounded by n when n becomes large, the overall time complexity is $\mathcal{O}\left((d+K)\cdot n^2 + d^3\right)$, where n is the number of samples, d is the number of features and K is the number of clusters. The space complexity is dominated by the storage of kernel matrix and Laplacian matrix, which has space complexity $\mathcal{O}(n^2)$.

Algorithm 1 Domain induced Score based Clustering (**DiSC**)

Input: dataset $X \in \mathbb{R}^{n\times d}$, score $s \in \mathbb{R}^{n\times 1}$, number of clusters k, data kernel $k_c(\cdot,\cdot)$, score kernel $k_s(\cdot,\cdot)$, weight μ, initial projection matrix $W^{(0)}$, maximal number of iteration T, precision of convergence ϵ.
procedure DiSC
  Compute kernel matrix K_s for score s
  Set *converged* = FALSE
  Set iteration number $t = 0$
  for $t <= T$ **do**
    Set $W \leftarrow W^{(t)}$, optimize U to get $U^{(t)}$
    Compute objective $F^{(t)}$ with $W^{(t)}$ and $U^{(t)}$
    Set $U \leftarrow U^{(t)}$, optimize W to get $W^{(t+1)}$
    Compute objective $F^{(t+1)}$ with $W^{(t+1)}$
    Compute $r^{(t)} \leftarrow \frac{|F^{(t+1)} - F^{(t)}|}{|F^t|}$
    if $||W^{t+1} - W^t||_F < \epsilon$ and $r^{(t)} < \epsilon$ **then**
      converged = TRUE
      break
    end if
    $t \leftarrow t + 1$
  end for
  if *converged* = True **then**
    Set $W_{opt} \leftarrow W^{(t)}$
    Compute kernel matrix K for subspace XW_{opt}
    Apply spectral clustering with K to get label Y
  end if
end procedure
Output: clustering solution Y, optimal projection W_{opt}

This algorithm will achieve a local minimum given any random initialization of orthogonal projection matrix W. W is initialized by drawing its elements from a standard Gaussian distribution and then applying Gram-Schmidt process to make it orthogonal. Due to the non-convex feasible region introduced by the orthogonality constraint of W, multiple initializations are required to escape from local minima. In the experiments, we randomly initialize W 20 times and choose the one resulting in the minimal objective function value.

3 Experimental Results

To demonstrate that DiSC can discover the subspace containing good clustering structure and relevant to the score variable defined by the domain experts, we first test the approach on a synthetic dataset. We then illustrate its performance on a publicly available WebKB benchmark data with a simulated score that guides the clustering solution. Finally, we show how to apply our approach for exploratory clustering on subtyping COPD, where a domain expert's score of

interest is available.

We compare our proposed *Domain induced Score based Clustering (DiSC)* approach against the following competing approaches.

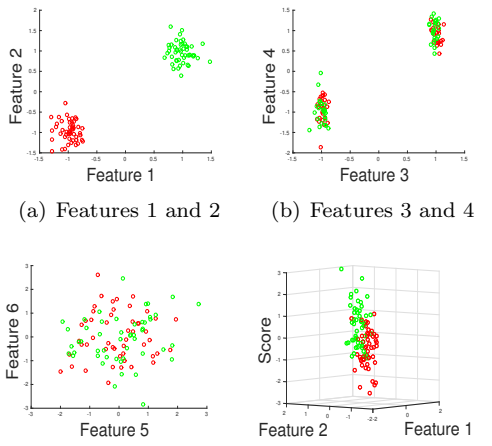
1) SC (spectral clustering) [9]: We apply spectral clustering on the original dataset and this can serve as a baseline.

2) DRSC (dimensionality reduction for spectral clustering) [20]: There are a few subspace learning approaches aiming at learning a subspace by optimizing clustering quality [21, 22, 20]. Existing dimensionality reduction algorithms, including principal component analysis (PCA) and Fisher discriminative analysis with kernels (KFD), can also be used for learning a subspace followed by applying existing clustering algorithms (KMeans or spectral clustering). DRSC learns a subspace through optimizing the clustering quality of spectral clustering in the subspace. We choose DRSC as a competing approach because their experiments show that DRSC outperforms all the subspace learning approaches for clustering described above.

3) DRHSIC (dimensionality reduction using HSIC): Because the score variable is used as a target in DiSC, we also need to compare DiSC against supervised subspace learning algorithms. In particular, given a target variable, *sufficient dimension reduction* aims to learn a subspace of input features that is *sufficient* for predicting the target [23]. We choose dimensionality reduction using HSIC as a representative of supervised subspace learning because its performance is comparable to competing alternatives and it can capture nonlinear dependencies [23]. After learning the subspace, we apply spectral clustering [9] in the learned subspace to obtain clustering result.

There are a few free parameters that need to be set for our proposed DiSC and competing approaches. The number of cluster k is set to be the number of ground-truth clusters on synthetic dataset and WebKB dataset. For COPD subtyping dataset, we set this value according to a recent study on COPD [6]. The dimentionality of subspace q is set to be $q = k$. The weight of the usefulness term μ can be adjusted so that the value of the first term and the second term are at the same scale. We applied the Gaussian kernel for both k_c and k_s in these experiments and the scale parameters are set using the median heuristic [24]. Since the optimization is non-convex, each initialization may result in a local optimum. Therefore, we run the algorithm multiple times with m different random initializations. We set $m = 20$ by default. However, m can be increased as d, the number of features in the original dataset, becomes larger. The clustering that results in the minimal objective function value is selected.

3.1 Synthetic Dataset We generate a synthetic dataset consisting of six features as is shown in Figure 1. There are two clusters in the subspace spanned by features 1 and 2. This clustering solution can be denoted as $C_{1,2}$. There are also two clusters residing in the subspace spanned by features 3 and 4 and this clustering solution can be denoted as $C_{3,4}$. There is no clear clustering structure in the subspace spanned by features 5 and 6. The red and green colors represent the labelling according to $C_{1,2}$. As we can see, $C_{1,2}$ and $C_{3,4}$ are quite different. While $C_{3,4}$ has the same between-cluster distance with $C_{1,2}$, it has smaller average within cluster distance, making clusters in $C_{3,4}$ more compact. $C_{3,4}$ therefore has a better clustering structure measured by the objective of spectral clustering. However, the clustering solution that domain experts are interested in discovering is $C_{1,2}$ because a hypothetical domain defined score s the experts consider important is related to the features as follows $s = X_1 + X_2 + 3X_5 + 3X_6 + \varepsilon$, where ε represents additive noise.



(a) Features 1 and 2 (b) Features 3 and 4

(c) Features 5 and 6 (d) Score vs Features 1,2

Figure 1: Scatter Plot of Features and Score in Synthetic Data

Since ground truth ($C_{1,2}$) is available on this synthetic data set, we compute the normalized mutual information (NMI) [25] and adjusted Rand index (ARI) [26] between predicted label and the true label. The range of NMI between predicted label and true label is between 0 and 1. Rand index (RI) computes the percentage of sample pairs that belong to the same cluster. ARI corrects RI for chance. The maximal value of ARI is 1. For both NMI and ARI, higher value indicates better clustering quality.

Evaluation results of different approaches based on NMI and ARI are provided in Table 1. As we can see from Table 1, both SC and DRSC discover the dominant

	NMI($C_{1,2}$)	ARI($C_{1,2}$)	NMI($C_{3,4}$)	ARI($C_{3,4}$)
SC	2.60e-3	-6.6e-3	0.81	0.88
DRSC	1.16e-3	-8.60e-3	0.88	0.92
DRHSIC	0.30	0.38	7.33e-3	0
DiSC	**1.00**	**1.00**	0	1.02e-2

Table 1: NMI between Predicted Labels and $C_{1,2}, C_{3,4}$

clustering structure $C_{3,4}$. Although DRHSIC will not prefer $C_{3,4}$, the desired structure $C_{1,2}$ is also ignored by DRHSIC because clustering quality criterion is not optimized. In contrast, our proposed approach DiSC can discover $C_{1,2}$ because the clustering quality and relevance to the score variable s are jointly optimized.

3.2 WebKB Benchmark Dataset In addition to synthetic data, we illustrate how our framework can be utilized to guide the clustering solution on a publicly available benchmark dataset – WebKB [27]. This dataset is a sub-sample of 1039 web pages collected from four universities: *Cornell University, University of Texas, Austin, University of Washington* and *University of Wisconsin, Madison*. These web pages can be clustered according to the university they are collected from. They can also be clustered according to four topics: *student, course, faculty* and *project*. After removing stop words and words of low variances, we keep 200 words in the end. We selected this dataset because it can be interpreted or clustered in two different ways: by university or by topic. The dominant clustering solution in this dataset is by clustering according to the four topics. However, if hypothetically, the domain expert is interested in the grouping related to university, they can guide the algorithm by providing a score. Note that there is no score variable in this dataset. To simulate this hypothetical scenario, we artificially create a score by summing up the frequencies of words related to university. Note that we run our method on this dataset with a simulated score only to illustrate how it works. On real-world data, as we will show in the next subsection, that score variable is determined/selected by domain experts.

We denote the clustering solution according to topic as C_T and university as C_U. We compare our approach against SC, DRSC and DRHSIC by computing NMI/ARI between each solution and C_T as well as C_U. The results are provided in Table 2.

	NMI(C_U)	ARI(C_U)	NMI(C_T)	ARI(C_T)
SC	5.86e-2	1.70e-3	2.84e-2	1.50e-2
DRSC	3.17e-2	2.45e-2	0.27	0.20
DRHSIC	0.32	0.27	0.16	0.11
DiSC	**0.40**	**0.31**	0.16	0.10

Table 2: NMI between Predicted Labels and C_U, C_T

As we can see from Table 2, DRSC discovers the clustering structure C_T. Both DRHSIC and DiSC prefer C_U to C_T due to the guidance provided by the score variable. Our approach DiSC results in higher NMI/ARI value with C_U because the clustering quality is also optimized.

To understand how different methods output different clustering results on the same dataset, we evaluate the importance of the i-th original feature in each approach by computing the ℓ_2-norm of the i-th row of the learned projection matrix W. After ranking all the original features (words), we show the ten most important features for DRSC, DRHSIC and DiSC. We put the table in the supplementary due to space constraints. Comparing DRSC and DRHSIC, the top feature sets has little overlap. The score variable guides DRHSIC to discover the subspace related to university. The top-rank features discovered by DRHSIC and DiSC are highly consistent. However, DiSC can discover clustering structure of higher quality, as is confirmed by the NMI and ARI values shown in Table 2.

3.3 Application to COPD Subtyping In this paper, we are specifically motivated by the objective of discovering clinically relevant subtypes (clusters) of Chronic Obstructive Pulmonary Disease (COPD), a lung disease that is currently the third leading cause of death in the United States [28]. Patients suffering from COPD is currently lumped into one cluster, which cannot capture the complexities of the disease. It is widely accepted by clinicians that there exist different disease subtypes of COPD [6]. The identification of different clusters can lead to tailored medical care for each individual. However, clinicians do not have consensus on how those disease subtypes should be defined. Therefore, this is a knowledge discovery problem.

We used a dataset with 1906 patients. For each patient, we collected 39 features, including demographics, clinical information, lung function, and variables extracted from computed tomography (CT) chest images. These are features clinicians consider important for describing COPD patients. There are multiple ways to partition this data. Which partitioning should we select?

The standard semi-supervised setting of providing sample labels do not work because this is a discovery process and labels are not known. Providing pairwise constraints is also not easy for clinicians because different clinicians do not agree on which pair to go together in a cluster. In our collaborative meetings, we realized that our domain experts do have some indicators/scores that guide them on which clustering solution is desirable. For example, a property that they desire is for the clustering solution to be related to biology and in COPD they have an indicator called *copdScore*. Ge-

netic variants can increase the risk of COPD, and act through specific biologic mechanisms. *copdScore* is a sum of risk variants identified from a genome-wide association study [29]. In the experiment, we used *copdScore* as our domain-specific score to guide DiSC.

Since the ground-truth clusters are unknown and yet to be discovered, we can no longer evaluate the clustering performance by computing NMI. In COPD and other diseases, *mortality*, is an indicator which represents whether a patient dies in the end due to the disease, and is a critical outcome. We can evaluate a clustering solution by computing the p-value of the χ^2 test on *mortality* as an external measure of quality. It tests whether there exists significant difference on *mortality* between different clusters. A better clustering solution is expected to generate a lower p-value. In addition to *mortality*, we also report the p-value of the Kruskal-Wallis test on *copdScore* on a separate test set.

We apply SC, DRSC, DRHSIC and our proposed approach DiSC on the dataset. We put the comparison between solutions generated by different approaches in the supplementary material. The solutions generated by SC and DRSC are close, which also holds for the synthetic dataset. We call this solution the dominant clustering solution. On the other hand, the clustering solutions output by DRHSIC and DiSC are quite different from the dominant clustering solution. This means the additional information provided by *copdScore* changes the cluster solution. Also note that the clustering solutions generated by DRHSIC and DiSC are also different because DiSC maximizes both the clustering quality and the association to *copdScore*, whereas DRHSIC only maximizes for *copdScore*. For each approach, we put its clustering quality (lower value indicates better quality) and association to *copdScore* (higher value indicates higher relevance) in Table 3. As we can see, SC and DRSC have the best clustering quality and DRHSIC has the highest association to the score variable. DiSC achieves tradeoff between optimizing clustering quality and association to the score variable.

To understand how different methods output different clustering results on the same dataset, we evaluate the importance of the i-th original feature in each approach by computing the ℓ_2-norm of the i-th row of the learned projection matrix W. After ranking all the original features, we show the ten most important features for DRSC, DRHSIC and DiSC and put the table in the supplementary material. The dominant clustering structure resides in the subspace spanned by features related to lung function. The important feature sets in DRHSIC and DiSC have little overlap with those lung function features considered as important by DRSC.

To validate whether the solution generated by our proposed approach is interesting, we randomly split the dataset into training set and testing set of equal size. The various algorithms output the projection matrix W and a form of spectral clustering solution. To classify test data, we first project the data to a low-dimensional subspace using the learned projection matrix W and assign each sample to the same cluster with its nearest neighbor in the training set. Note that the nearest neighbor computation is done in the learned subspace. Then we run a χ^2 test to evaluate whether different clusters in the testing set are significantly different on *mortality*. In addition, we performed a Kruskal-Wallis test on *copdScore* to see whether different groups have the same median value on this feature. The resulting p-values are shown in Table 3. Note that lower p-values of a feature indicate more heterogeneity (cluster dissimilarity) across different clusters on this feature. We observe that DRHSIC performs the best (as evidenced by the smallest p-values) with respect to *copdScore*, which is expected as this method was optimized for *copdScore*. This is followed closely by our method DiSC. Interestingly, even though *mortality* was not utilized in the learning (training phase), our proposed method DiSC performed the best in terms of *mortality* compared to competing methods. We believe it is because our method optimizes for both cluster quality from the data features and relevance to the score domain experts deemed useful. It demonstrates that our proposed approach can generate a useful clustering solution, as measured by heterogeneity on *mortality*.

Approach	*cluster quality*(↓)	*score association*(↑)	p-*copdScore*(↓)	p-*mortality*(↓)
SC	**2.73**	1.67	2.50e-3	4.66e-9
DRSC	**2.73**	1.93	6.18e-8	9.51e-10
DRHSIC	2.82	**2.12**	**9.73e-11**	1.87e-12
DiSC	2.78	2.07	3.56e-8	**1.62e-13**

Table 3: For different approaches, the four columns show their cluster quality, score association, p-values on *copdScore* and *mortality* respectively. ↓ means lower value is better and ↑ means higher value is better.

4 Conclusions

In this paper, we provide a framework that allows learning a low-dimensional subspace and a clustering solution that both contains clustering structure of high quality and is informative about score variable/s defined by domain experts. We encountered the need for such a formulation on a real-world exploratory data clustering problem of subtyping COPD, where different clustering structures reside in different subspaces and not all of them are interesting. Existing approaches cannot be directly applied to solve this problem. In particular, dimensionality reduction for clustering can only discover the dominant clustering solution, which may not be interesting. Semi-supervised clustering can only utilize

instance level constraints, such as must-link and cannot-link constraints. However, our domain experts were not comfortable in providing such constraints. Luckily, our domain experts have a general idea of global properties that the clustering solution should have. In particular, according to our experts, the desired clustering should be relevant to the genetic risk score for each patient (which we can obtain). Our experiments demonstrate that the incorporation of domain-specific global score criterion can guide exploratory clustering through our new clustering approach discover more meaningful clustering structure compared to competing methods.

5 Acknowledgements

We would like to acknowledge support for this project from the NIH grant NIH/NHLBI RO1HL089856 and RO1HL089857.

References

[1] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[2] J. A. Hartigan, "Clustering algorithms," 1975.

[3] D. Niu, J. G. Dy, and M. I. Jordan, "Multiple non-redundant spectral clustering views," in *ICML*, pp. 831–838, 2010.

[4] O. Chapelle, B. Schölkopf, A. Zien, *et al.*, "Semi-supervised learning," 2006.

[5] I. Davidson and S. Basu, "A survey of clustering with instance level constraints," *ACM Transactions on Knowledge Discovery from Data*, pp. 1–41, 2007.

[6] P. J. Castaldi, J. Dy, J. Ross, *et al.*, "Cluster analysis in the copdgene study identifies subtypes of smokers with distinct patterns of airway disease and emphysema," *Thorax*, pp. thoraxjnl–2013, 2014.

[7] E. Bair and R. Tibshirani, "Semi-supervised methods to predict patient survival from gene expression data," *PLoS Biol*, vol. 2, no. 4, p. e108, 2004.

[8] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[9] A. Y. Ng, M. I. Jordan, Y. Weiss, *et al.*, "On spectral clustering: Analysis and an algorithm," *NIPS*, vol. 2, pp. 849–856, 2002.

[10] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with hilbert-schmidt norms," in *Algorithmic learning theory*, pp. 63–77, Springer, 2005.

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

[12] W. Zhi, X. Wang, B. Qian, P. Butler, N. Ramakrishnan, and I. Davidson, "Clustering with complex constraints-algorithms and applications.," in *AAAI*, 2013.

[13] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *KDD*, pp. 551–556, ACM, 2004.

[14] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical review E*, vol. 69, no. 6, p. 066138, 2004.

[15] T. N. Bui and C. Jones, "Finding good approximate vertex and edge partitions is NP-hard," *Information Processing Letters*, vol. 42, no. 3, pp. 153–159, 1992.

[16] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303–353, 1998.

[17] Z. Wen and W. Yin, "A feasible method for optimization with orthogonality constraints," *Mathematical Programming*, vol. 142, no. 1-2, pp. 397–434, 2013.

[18] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[19] I. Koutis, G. L. Miller, and D. Tolliver, "Combinatorial preconditioners and multilevel solvers for problems in computer vision and image processing," *Computer Vision and Image Understanding*, vol. 115, no. 12, pp. 1638–1646, 2011.

[20] D. Niu, J. G. Dy, and M. I. Jordan, "Dimensionality reduction for spectral clustering," in *AISTATS*, pp. 552–560, 2011.

[21] C. Ding and T. Li, "Adaptive dimension reduction using discriminant analysis and k-means clustering," in *ICML*, pp. 521–528, ACM, 2007.

[22] J. Chen, Z. Zhao, J. Ye, and H. Liu, "Nonlinear adaptive distance metric learning for clustering," in *KDD*, pp. 123–132, ACM, 2007.

[23] T. Suzuki and M. Sugiyama, "Sufficient dimension reduction via squared-loss mutual information estimation," *Neural computation*, vol. 25, no. 3, pp. 725–758, 2013.

[24] A. J. Smola and B. Schölkopf, *Learning with kernels*. Citeseer, 1998.

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

[26] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

[27] "Webkb dataset:http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/," 1998.

[28] S. L. Murphy, J. Xu, and K. D. Kochanek, "Deaths: final data for 2010.," *National Vital Statistics System*, vol. 61, no. 4, pp. 1–117, 2013.

[29] M. H. Cho, M.-L. N. McDonald, X. Zhou, *et al.*, "Risk loci for chronic obstructive pulmonary disease: a genome-wide association study and meta-analysis," *The Lancet Respiratory Medicine*, vol. 2, no. 3, pp. 214–225, 2014.