A Principled and Flexible Framework for Finding Alternative Clusterings

ZiJie Qi
Department of Computer Science
University of California, Davis
Davis, CA 95616
zqi@ucdavis.edu

Ian Davidson
Department of Computer Science
University of California, Davis
Davis, CA 95616
davidson@cs.ucdavis.edu

ABSTRACT

The aim of data mining is to find novel and actionable insights in data. However, most algorithms typically just find a single (possibly non-novel/actionable) interpretation of the data even though alternatives could exist. The problem of finding an alternative to a given original clustering has received little attention in the literature. Current techniques (including our previous work) are unfocused/unrefined in that they broadly attempt to find an alternative clustering but do not specify which properties of the original clustering should or should not be retained. In this work, we explore a principled and flexible framework in order to find alternative clusterings of the data. The approach is principled since it poses a constrained optimization problem, so its exact behavior is understood. It is flexible since the user can formally specify positive and negative feedback based on the existing clustering, which ranges from which clusters to keep (or not) to making a trade-off between alternativeness and clustering quality.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms

Algorithms, Experimentation

Keywords

Clustering

1. INTRODUCTION

The purpose of data mining is to find novel and actionable patterns. However, in many situations a practitioner already has knowledge of what is *not* actionable and *not* novel and unless this is somehow encoded, the algorithm may continue to find those patterns. Consider the clustering of loan applications in order to identify bad loans, but the clusters fall along racial lines. You may wish to find another alternative yet equally good clustering. Similarly, high dimensional data such as collections of images may naturally contain many plausible ways of clustering based on different subsets of pixels. Finally as previously showed, in even low dimensional data (the pen digit data set in our earlier work [5]) multiple explanations may exist if the underlying phenomenon is complex and the data is insufficient to justify just one explanation.

The recent innovation of finding an alternative clustering answers the question: "Given a clustering π, does there exist another clustering π' which is different from π but equally good in terms of objective function value?" Note that in this question there are two key factors of concern: alternativeness and quality. That is, we hope the new clustering not only interprets the data from an alternative perspective but also is of good quality in terms of the algorithm's objective function. Others [9, 2, 4] as well as ourselves [5] have tackled the problem which we term the *Singular Alternative Clustering Problem* described below.

PROBLEM 1. ***Singular Alternative Clustering Problem***. *Given an objective function f, an **existing** clustering π so that $f(\pi) = x$, does there exist another clustering π' that is different from π and where $f(\pi') \approx f(\pi)$?*

Note that the *Singular Alternative Clustering Problem* is a different problem from the one addressed in Jain, Meka and Dhillon's work [10]. In that paper, the authors deal with a problem of finding two disparate (alternative) clusterings **simultaneously**, while our work deals with finding an alternative clustering given an existing one.

There are two primary limitations in previous work on this topic. First, existing techniques aside from our prior work [5] are algorithm-dependent [9, 2, 4], as we shall describe in Section 2. Second, all the existing techniques including our own [5] do not specify which properties of the original clustering should be preserved (or not) in the new clustering. Instead, they bluntly find an alternative clustering with no guidance other than the new clustering must be an alternative to the original. However, in many circumstances we may not wish to find a complete alternative, but perhaps a partial alternative, and seek to precisely state which parts of the clustering to retain and which parts not to retain.

The main contribution of this paper is to propose a general framework for solving the *Singular Alternative Clustering Problem* where the expected properties of the new clustering can be specified. To find the new clustering in an **algorithm-independent** way, we create a transformation matrix to transform the data set into a new space while preserving the properties of the data set and respecting the users' feedback on the previous clustering. This allows any clustering algorithm to be applied to the transformed data. We will formally show that our approach is a solution to a **constrained**

KDD'09, June 28–July 1, 2009, Paris, France.

optimization problem. Our formulation **minimizes** the Kullback-Leibler divergence between two distributions: the original data and the transformed data, so that the data properties are not overly distorted. The **constraint** on the optimization allows us to specify which properties of the clustering should be kept. Formally, our aim is to create an approach that:

- Is general purpose and can address the *Singular Alternative Clustering Problem* for a variety of clustering algorithms;
- Can specifically identify which properties of the old clustering should (or should not) be maintained in the new clustering;
- Is efficient and easy to implement;
- Is feasible for both high (Figure 3, 4 and 5) and low dimensional data sets (Table 1, 2 and 3).

Note that this work does **not** build upon our previous work [5] apart from working on the same problem and also proposing an algorithm-independent approach.

We begin this paper by describing the related work in Section 2. We present our framework to solve the *Singular Alternative Clustering Problem* in Section 3. In Section 4, we show the flexibility of our approach by discussing the variations to our problem formulation. In Section 5, we illustrate experimental results on UCI data sets which show that our approach provides genuinely (non-trivial) alternative clusterings with good quality. The experiments on image segmentation applications show that our approach can obtain alternative meaningful image partition results. Finally, we present the experiments where the desirable/undesirable clusters in the new clustering can be explicitly specified, and the results show that our approach not only achieves alternative clusterings but also maintains the desirable clusters.

2. RELATED WORK

The problem of finding alternative clusterings has received limited attention so far. Most of the approaches to address this problem are based on some specific clustering algorithm. Bae and Bailey [2] force an alternative clustering by generating cannot-linked constraints from all pairs of objects which are in the same cluster in π, the original clustering. However, their method is tied to a hierarchical clustering algorithm. Another approach combines k-means and PCA to project the data into an alternative subspace [4]. This has the limitation of not being appropriate for lower dimensional data sets such as spatial data, as we discussed and illustrated in previous work [5]. A third approach [9] explored the idea of using Conditional Information Bottleneck (CIB) to find an alternative clustering to a given non-novel clustering. This approach subtracts the background knowledge of the given clustering by maximizing conditional mutual information $I(C; Y|Z)$ (C, Y and Z denote the clusters of objects, relevant features and the background knowledge), which is difficult to implement since it requires modeling joint distribution between the cluster labels and the relevant features. The last approach, which is our own [5], first learns a distance metric D_π from the original clustering π and then interprets D_π from the geometric point of view. It then reverses the transformation of D_π using Moore-Penrose pseudo-inverse to get the new distance metric D'_π. Thus in the new data transformed by D'_π one will find a different clustering other than π.

The area of non-hierarchical clustering with constraints can potentially be used to find alternative clusterings. Consider a clustering π which can non-ambiguously be represented by a large conjunction of must-linked constraints between every two points in the same cluster and a large conjunction of cannot-linked constraints between every two points from different clusters. Since this represents the clustering π, we can guarantee that π is not found again by flipping the constraints (making must-linked constraints cannot-linked and vice-versa) and clustering to satisfy these flipped constraints. Consider if $\pi = \{(a, c), (b, d), (e, f)\}$ shown in Figure 1 (a) then this clustering can be uniquely represented as the constraints must-link(a,c), must-link(b,d), must-link(e,f), cannot-link(c,d), cannot-link(c,e) and cannot-link(d,f) (not all entailed constraints are provided for clarity) shown in Figure 1 (b). However, flipping these constraints for even this simple six point data set produces cannot-link(a,c), cannot-link(b,d), cannot-link(e,f), must-link(c,d), must-link(c,e), must-link(d,f) shown in Figure 1 (c) for which no clustering exists that satisfies all constraints. For similar reasons it is not desirable to learn a distance function from the flipped constraints due to the many inconsistent constraints that flipping could generate. Furthermore, even if a set of non-contradictory constraints could be generated, then trying to find just a single clustering to satisfy them is known to be **NP**-complete [7] for any constraint type combination involving cannot-linked constraints. Davidson and Ravi have shown that clustering under many cannot-linked constraints is intractable for batch [6], incremental [8] and even pruning-style algorithms [7]. This is a large hurdle since we most certainly wish to generate must-linked constraints from points in the same cluster but flipping them will produce the undesirable cannot-linked constraints. Finally, approaches that can deal with inconsistent constraints/advice in a principled manner were limited. For example, the work of Coleman *et al.* [3] that deals with embedding constraints into the spectral clustering algorithm only addresses the problem where no object is involved in more than 1 cannot-linked constraint, and only for $k = 2$.

As we discussed in the introduction section, the problem of finding two clustering simultaneously is a different problem from our problem setting. That problem has been formulated under the framework of the EM algorithm [10]. Their two approaches, Decorrelated-kmeans and Convolutional-EM in Jain, Meka and Dhillon's work are based on two separate assumptions. The first one assumes that if the "representative" vectors (which are different from the mean vectors and lack of intuitive interpretation) of the old clustering and the new clustering are mutually orthogonal, then the alternativeness of two clusterings should be guaranteed. The second approach interprets each clustering as a partial representation of the data and models the data as a sum of mixture distributions, each mixture corresponding to a clustering. Note that there is no transformation of the data involved in both methods.

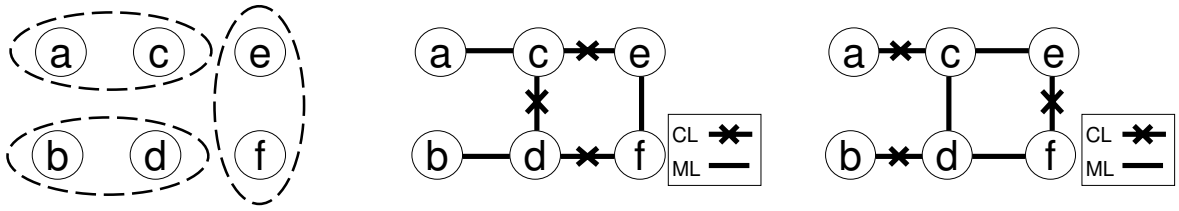
3. OUR APPROACH

3.1 Setting and Notation

Let $X = \{x_1, x_2, ..., x_n\} \subseteq \mathbb{R}^d$ denote the given d-dimensional data set which is represented by a $d \times n$ matrix. The original clustering π is found in X. The transformation matrix D is a $d \times d$ matrix while $Y = \{y_1, y_2, ..., y_n\} \subseteq \mathbb{R}^d$ refers to the transformed data set by transformation $Y = DX$. The alternative clustering π' is found in Y. Let X and Y follow the probability density functions $p_x(x)$ and $p_y(y)$.

The output of a clustering algorithm is a k-block set partition of the data set X which is referred to as a *clustering*. Each block forms a cluster, and they are referred to as $C_1, C_2, ..., C_k$. The size of cluster C_i is denoted as n_i. The cluster centroids are denoted as $m_1, m_2, ..., m_k$.

Please refer to Appendix B for the complete table of notations.



(a) Clustering $\pi = \{(a, c), (b, d), (e, f)\}$ (b) The minimum set of constraints to represent π non-ambiguously (c) The flipped constraints

Figure 1: A simple example where ML is must-linked constraint and CL is cannot-linked constraint.

3.2 Constrained Optimization Formulation

To achieve our goal of finding an alternative clustering with good quality in a general purpose manner we explore a data transformation approach that converts the original data X to Y using a distance metric represented by the transformation matrix D. Our formulation allows the user to choose any appropriate clustering algorithm to run on Y to achieve the new alternative clustering hence we term our approach algorithm-independent. Note that another option is to directly use the transformation matrix D generated in our approach along with the old data X by using the transformation matrix D as the distance metric in any distance based algorithm. The first option is more useful since we do not always have a distance based algorithm, and it is used in the experimental section of this paper.

There are two main factors in our work:

- The new data set Y preserves the characteristics of X as much as possible so that the new clustering π' found in the new space has good quality in the original space. **In all of our experiments we report VQE, DI and JI for both π and π' in X;**
- Conversely, π (or parts of π) should have a poor objective function value in Y so that it should not be found by a clustering algorithm.

According to these two factors, we formulate the problem as a constrained optimization problem, as shown in Eq.(1), where $B = D^T D$ and $\|\cdot\|_B$ denotes the Mahalanobis distance with this weight matrix B.

$$\min_{B \succeq 0} \; D_{KL}(p_y(y)||p_x(x)) \quad s.t. \; \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 \leq \beta \tag{1}$$

The objective function of the Kullback-Leibler divergence is a measure of the difference between two probability distributions. When $D_{KL}(p_y(y)||p_x(x)) = 0$, the two distributions $p_x(x)$ and $p_y(y)$ of X and Y are the same. We minimize KL divergence in Eq.(1) so that the probability density functions of X and Y are closely matched. This ensures that the inherent properties of X are not destroyed when being transformed to Y.

The constraint in Eq.(1) comes from the characteristics that we expect the new clustering π' to express. We now explain our initial and most general constraint. Variations in this constraint are discussed in Section 4. The constraint is best explained in a probabilistic framework. To simplify the problem formulation, assume that the clusters of π' follow a mixture model of multivariate Gaussian distributions $f_1(y), ..., f_k(y)$ with the same covariance matrix $\hat{\Sigma}$ but different means $\hat{m}_1, ..., \hat{m}_k$, respectively. In other words, we assume that each cluster in π' follows a multivariate Gaussian distribution with the same covariance matrix $\hat{\Sigma}$. Let $\hat{m}_1, ..., \hat{m}_k$ be the **projection of the original centroids** $m_1, ..., m_k$ in the new space. Note that these means are different from the centroids the algorithm will find. Let $C_1, C_2, ..., C_k$ denote the k clusters in π and $\widehat{C}_1, \widehat{C}_2, ..., \widehat{C}_k$ denote the k new clusters in π'. Then the probability density function of y is

$$p(y) = \sum_{i=1}^{k} \frac{\widehat{n}_i}{n} f_i(y) = \sum_{i=1}^{k} \frac{\widehat{n}_i}{n|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}} e^{-\frac{1}{2}\|(y-\hat{m}_i)\|^2_{\hat{\Sigma}^{-1}}} \tag{2}$$

where $\widehat{n}_i$ is the size of cluster $\widehat{C}_i$ in π'. Consequently we have $\sum_{i=1}^{k} \widehat{n}_i = n$.

Suppose object x_i belongs to the cluster C_j in π, which means that C_j is its most probable cluster. In order to find a different clustering we must transform the data so that x_i is more likely to be assigned to a different cluster other than C_j in the new space. Then the probability of object y_i belonging to $\widehat{C}_j$ (being closest to $\hat{m}_j$) in the new clustering π' should be small, which is written as $(0 \leq \alpha \leq 1)$:

$$\frac{\widehat{n}_j}{n|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}} e^{-\frac{1}{2}\|(y_i-\hat{m}_j)\|^2_{\hat{\Sigma}^{-1}}} \leq \alpha \tag{3}$$

This is equal to Eq.(4).

$$\sum_{j=1, x_i \notin C_j}^{k} \frac{\widehat{n}_j}{n|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}} e^{-\frac{1}{2}\|(y_i-\hat{m}_j)\|^2_{\hat{\Sigma}^{-1}}} \geq 1 - \alpha \tag{4}$$

When there is no specific assumption of the sizes of clusters in π', we can assume that each cluster has the same size. Thus $\frac{\widehat{n}_j}{n} = \frac{1}{k}$ for $(1 \leq j \leq k)$. In the multivariate Gaussian model, we assume that $\widehat{\Sigma}$ has the same variance along each dimension and dimensions are highly independent. Then the off diagonal entries are very small and can be ignored, and $\widehat{\Sigma}$ can be approximated by a multiplication of a scaler and an identity matrix $\sigma^2 I$ $(\sigma > 0)$. Therefore Eq.(4) becomes:

$$\frac{e^{-\frac{1}{2\sigma^2}}}{k|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}} \sum_{j=1, x_i \notin C_j}^{k} e^{-\frac{1}{2}\|(y_i-\hat{m}_j)\|^2} \geq 1 - \alpha \tag{5}$$

$$\sum_{j=1,x_i\notin C_j}^{k} e^{-\frac{1}{2}\|(y_i-\hat{m}_j)\|^2} \geq (1-\alpha)k|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}e^{\frac{1}{2\sigma^2}} \quad (6)$$

Since $(a_1 + a_2 + \ldots + a_n)/n \geq \sqrt[n]{a_1 a_2 \ldots a_n}$ when $a_i > 0$ $(1 \leq i \leq n)$, Eq.(6) must hold if Eq.(7) is true:

$$(k-1)\sqrt[k-1]{\prod_{j=1,x_i\notin C_j}^{k} e^{-\frac{1}{2}\|(y_i-\hat{m}_j)\|^2}} \geq (1-\alpha)k|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}e^{\frac{1}{2\sigma^2}} \quad (7)$$

Therefore, we have

$$\sum_{j=1,x_i\notin C_j}^{k} \|(y_i-\hat{m}_j)\|^2 \leq -2\ln\left[\left(\frac{(1-\alpha)k|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}e^{\frac{1}{2\sigma^2}}}{k-1}\right)^{k-1}\right] \quad (8)$$

Let β be $-2ln[(\frac{(1-\alpha)k|\hat{\Sigma}|^{\frac{1}{2}}(2\pi)^{\frac{d}{2}}e^{\frac{1}{2\sigma^2}}}{k-1})^{k-1}]$. For every y_i, the constraint becomes ($\beta > 0$):

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1,x_i\notin C_j}^{k}\|(y_i-\hat{m}_j)\|^2 \leq \beta \quad (9)$$

Since $Y = DX$, $\hat{m} = Dm$ and $B = D^T D$, for x we have

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1,x_i\notin C_j}^{k}\|(x_i-m_j)\|_B^2 \leq \beta \quad (10)$$

To derive the solution, we define an auxiliary covariance matrix $\widetilde{\Sigma}$, shown in Eq.(11). We see that $\widetilde{\Sigma}$ is a $d \times d$ matrix and can be interpreted as the variance of the data with respect to $k-1$ centroids since the centroid which each instance is assigned to in π is excluded.

$$\widetilde{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1,x_i\notin C_j}^{k}(x_i-m_j)(x_i-m_j)^T \quad (11)$$

Then the solution to our constrained optimization problem defined in Eq.(1) is $B = \widetilde{\Sigma}^{-1}$ and since $B = D^T D$ we have our transformation matrix $D = \widetilde{\Sigma}^{-\frac{1}{2}}$. Details of the solution are described in Appendix A. We see that $\widetilde{\Sigma}$ is essentially the summation over n (the number of instances) $d \times d$ matrices. Each of these n matrices is in turn a summation of further $(k-1)$ $d \times d$ matrices. Each of these $n(k-1)$ matrices measures the variability caused by a point for a given centroid which this point unlikely belongs to in π'. We then see that the solution to Eq.(1) is to transform the data so as to reduce this variability which in turns satisfies the upper bound in the equation.

The constraint in our formulation (Eq.(1)) is exchangeable with different specifications of the expected properties in the new clusterings. We will discuss the details of variations of the problems in Section 4.

An illustrative example. We use the following simple example to illustrate our techniques. Figure 2 (a) shows that the data set X is composed of four multivariate Gaussian distributions at four corners of a square with the same variance along each dimension. The given clustering π with two horizontal clusters is shown in Figure 2 (b). We see that $\widetilde{\Sigma}$

$$\widetilde{\Sigma} = \begin{bmatrix} 9.7419 & 0.1801 \\ 0.1801 & 36.6461 \end{bmatrix}, D = \widetilde{\Sigma}^{-\frac{1}{2}} = \begin{bmatrix} 0.3204 & -0.0010 \\ -0.0010 & 0.1652 \end{bmatrix}$$

indicating that there is more variability between the points along the y-axis than the x-axis. Then the resultant transformation D is to compress more along the y-axis than the x-axis. Therefore, when X is transformed to $Y = DX$, the new clustering π' with two vertical clusters as shown in Figure 2 (c) is more likely to be found.

4. VARIATIONS OF THE PROBLEM

Our basic formulation of the constraint part of the optimization problem in Eq.(1) essentially transforms the data but makes sure that each point is not assigned to the same cluster as before. In this section we will discuss other variations to guide the data transformation. In particular we shall explore three main variations of general use, but there may be others of more specific use. We will empirically verify our approach to the first and second variations of the problem in Section 5.

The three variations allow:

1. Specifying a trade-off between the alternativeness and quality of the new clustering with respect to the original clustering.
2. Specifying which clusters in the original clustering to keep and which clusters not to keep.
3. Finding an alternative clustering in a subspace.

Recall that the constraint in Eq.(1) takes the form

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1,x_i\notin C_j}^{k}\|(x_i-m_j)\|_B^2 \leq \beta.$$

Each of the above variations involves changing some aspect of this basic form, as we now describe.

4.1 Specifying the Trade-off between Alternativeness and Quality

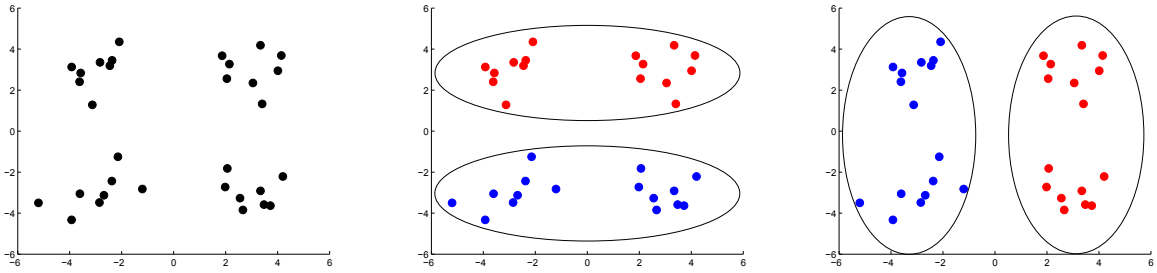
We add the parameter $a \geq 1$ to quantify the trade-off between alternativeness and quality by redefining the constraint as follows:

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1,x_i\notin C_j}^{k}\|(x_i-m_j)\|_B^a \leq \beta \quad (12)$$

We see that the larger a is, the stronger the constraint of assigning an object to a different cluster in the new clustering will be (see our probabilistic interpretation of this constraint in the previous section). Hence the optimization is focused/biased more towards alternativeness. Conversely if a is made small then the constraint is weaker. The solution to the modified optimization problem is then $B = \widetilde{\Sigma}^{-\frac{a}{2}}$, that is, $D = \widetilde{\Sigma}^{-\frac{a}{4}}$.

4.2 Specifying Which Clusters to Keep and Not to Keep

To allow this we can have multiple constraints (summations) for different clusters. For a cluster (say C_j) we wish not to keep we employ the same constraint as in Eq.(1) except the summation is limited to points only in C_j. For a cluster (say C_l) we wish to retain we then have the constraint: $\sum_{x_i\in C_l}\|(x_i-m_l)\|_B^2 \leq \delta$ where δ is some small constant value. The new form of the constraint in



(a) Data X

(b) Clustering π with 2 clusters at top and bottom

(c) The new clustering π' with 2 clusters at left and right

Figure 2: An illustrative 2D example.

our problem formulation is shown in Eq.(13) where the clusters of $C_Y = \{C_1, ..., C_r\}(1 \leq r < k)$ are retained and the clusters of $C_N = \{C_{r+1}, ..., C_k\}$ are not retained. Note that the first summation is related to the points in the clusters to be maintained, and the second summation is related to the points in the clusters not to be maintained.

$$\sum_{x_i \in C_Y} \sum_{l=1, x_i \in C_l}^{r} \|(x_i - m_l)\|_B^2 + \sum_{x_i \in C_N} \sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 \leq \beta \tag{13}$$

The solution to the modified optimization problem is then $B = \widetilde{\Sigma}_1^{-1}$, that is, $D = \widetilde{\Sigma}_1^{-\frac{1}{2}}$, where $\widetilde{\Sigma}_1$ is defined in Eq.(14).

$$\widetilde{\Sigma}_1 = \frac{1}{n}\Big(\sum_{x_i \in C_Y} \sum_{l=1, x_i \in C_l}^{r} (x_i - m_l)(x_i - m_l)^T + \sum_{x_i \in C_N} \sum_{j=1, x_i \notin C_j}^{k} (x_i - m_j)(x_i - m_j)^T \Big) \tag{14}$$

There are some cases where the partial clustering that needs to be kept is not composed of whole clusters but some small chunklets of objects. We can specify the information that objects x_i and x_j should be in the same cluster as a constraint $s(x_i, x_j)$. Suppose $C_Y = \{c_1, ..., c_t\}$ includes all the chunklets $c_1, ..., c_t (t > 0)$ that is supposed to be retained and the objects in $C_N = X \setminus A$ should change their assignment in the new clustering. We generate a constraint set $S = \{s(x_i, x_j), ...\}$ which includes all the pairs of points that should be in the same cluster. The constraint formulation is as in Eq.(15), and the auxiliary matrix $\widetilde{\Sigma}_2$ is redefined in Eq.(16).

$$\sum_{x_i \in C_Y} \sum_{s(x_i, x_l) \in S} \|(x_i - x_l)\|_B^2 + \sum_{x_i \in C_N} \sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 \leq \beta \tag{15}$$

The solution to the modified optimization problem in Eq.(15) is then $B = \widetilde{\Sigma}_2^{-1}$, that is, $D = \widetilde{\Sigma}_2^{-\frac{1}{2}}$, where $\widetilde{\Sigma}_2$ is as in Eq.(16).

$$\widetilde{\Sigma}_2 = \frac{1}{n}\Big(\sum_{x_i \in C_Y} \sum_{s(x_i, x_l) \in S} (x_i - x_l)(x_i - x_l)^T + \sum_{x_i \in C_N} \sum_{j=1, x_i \notin C_j}^{k} (x_i - m_j)(x_i - m_j)^T \Big) \tag{16}$$

4.3 Finding an Alternative Clustering in a Subspace

In the original formulation we transformed the data using all entries/dimensions in B, but in this variation we normalize over only a subspace in B. For example, we may find that the given clustering π is most compact in some subset of dimensions and wish to find an alternative clustering in the complement of this subset. This is effectively finding an alternative clustering in the complementary subspace that π is most compact in. It can be achieved by fixing the row and column entries in B to be zero for all dimensions that the clustering π is most compact in. Then it makes all the points along each of these dimensions mapped to the dimension origin, effectively making these dimensions useless for differentiating points into clusters.

5. EXPERIMENTAL RESULTS

We present three sets of experimental results. We will now sketch the results and in later subsections provide full details that will allow their repetition. Note that the source code in MATLAB used to reproduce these results will be made available, and we have posted the source code at `www.constrained-clustering.org`. In all of the experiments, we use three measurements: Dunn Index (DI), Vector Quantization Error (VQE) and Jaccard Index (JI) to evaluate the results. The DI is a quality measure of the ratio of the minimum distance between two clusters (when measured as the average link distance) to the maximum cluster diameter. The larger the DI the better. We also report the VQE for clusterings as it is the objective function that k-means minimizes. The smaller the VQE the better. Note that the DI is a measure of separation between clusters normalized by the cluster diameters, while the VQE only measures cluster compactness, not their separation. The Jaccard Index (JI) measures the similarity between two clusterings, the smaller the JI, and the more dissimilar the two clusterings are. **All of the measurements are calculated based on the original data X.**

All of the clusterings related to the UCI data are obtained by the k-means algorithm. The experiments on image segmentation

uses the spectral clustering algorithm as defined by Shi and Malik [11]. As before, π is the given clustering and the new alternative clustering is π'.

5.1 UCI Data Set

Our first set of results is on standard UCI data [1] sets and compares our work against others [2], including our previous work [5]. The comparison to the work of Cui et.al. [4] can be referred to in our previous paper [5], but their approach does not work well for lower dimensional data.

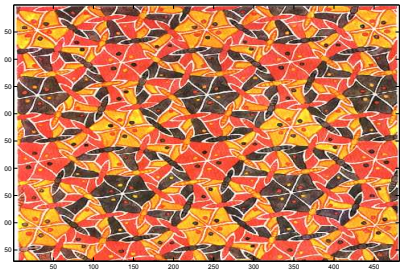
We show (see Table 1) that our work is comparable to similar work with respect to quality of clustering found (when measured by the VQE) and diversity between the original and alternative clustering (when measured using the Jaccard index). The approach of Bae and Bailey [2] obtains better DI results but worse VQE results than our own. This can be explained by the fact that the objective function of their algorithm is the DI and for k-means it is the VQE. However, our approach has the advantage of being usable with a variety of distance based clustering algorithms, being able to provide both positive (keep a cluster) or negative (don't keep a cluster) feedback (which will be discussed in Section 5.3) and being able to trade-off the two parts (alternativeness and quality) of the constrained optimization problem. By modifying the exponent a we can favor making the alternative clustering more different than π but typically of worse quality and vice-versa, as shown in Table 2.

These types of experiments are typically performed to show that the approach finds a clustering of reasonable quality and is different from the original clustering. However, they do not show if the second clustering is truly an interpretable alternative to the original clustering. To show that, we need to focus on data sets where the results are readily interpretable, as we do for our next two sets of experiments.

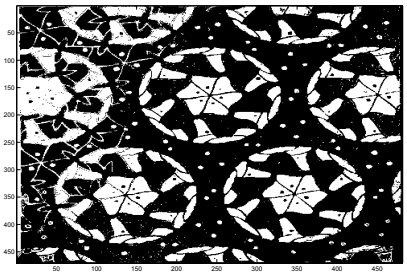
5.2 Image Segmentation

Our second set of experimental results is for image segmentation using spectral clustering. We focus on several Escher images which are known to have multiple interpretations to the human eye. Consider Figure 3 (a) which has two interpretations. If the eye focuses on the black sections then there is a segmentation of the image into black and non-black as found by spectral clustering in Figure 3 (b). This is the dominant segmentation since the contrast between the two clusters is great as one cluster includes the black parts and the other cluster includes the orange and yellow parts. However, our approach is able to discover the second and more subtle two cluster segmentation in Figure 3 (c) where the orange is in one cluster and the black and yellow are in the other. Similar results are found for Figure 4 (a) which contains three types of butterflies (red, green and blue). Spectral clustering first finds for $k = 2$ a clustering of the blue butterflies by themselves and the green and red butterflies together in Figure 4 (b). The alternative clustering found by our approach is the red butterflies by themselves and the blue and green butterflies together in Figure 4 (c). In Figure 5, the original clustering π partitions the image based on the blocked background and subsumes the mandolin, but the new clustering π' separates the object of the mandolin from the background.

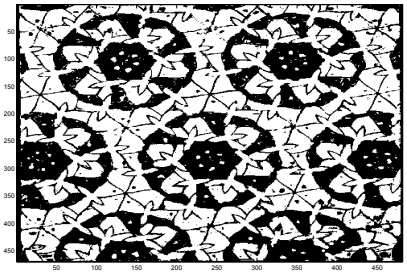
We use the normalized spectral method of Shi and Malik [11] as the clustering algorithm. Each object in an image is a pixel with two kinds of information: RGB value and position. The similarity between two pixels is the weighted sum of the Euclidean distance between their RGB values and position. The transformation is only carried out in the RGB value space. In Figure 3 and 4 we see that for the Escher images the new clustering π' finds a different texture in the images.



(a) The original flower image



(b) π found by the spectral clustering



(c) π' found by our approach given π

Figure 3: Escher flower example of an image with textures with alternative interpretations, $k = 2$.

5.3 Specifying Which Clusters to Keep and Not to Keep

Finally in our last set of experimental results we show how our approach can focus on which clusters to keep and which not to keep. The cluster C_l we chose to maintain is the one with the maximum size/cardinality in π which is derived from the inherent labels in the UCI data set. Because the data set *Ionosphere* only has two clusters inherently it is not possible to find a much different clustering while maintaining one cluster. So we only focus on three data sets: *Glass*, *ESL* and *Vehicle*. Let function $g(C_l, \pi')$ return the largest number of points in C_l which are in the same cluster in π'. For instance, if the data set has 6 objects $X = \{x_1, ..., x_6\}$, $C_l = \{x_1, x_2, x_3\}$, and the clustering π' partitions X into two clusters $C_1 = \{x_1, x_2\}$ and $C_2 = \{x_3, ..., x_6\}$, then $g(C_l, \pi') = 2$. The size of subset C_l is denoted as n_l.

We define the hit rate as the percentage of objects kept in the same cluster in π', as follows:

$$\text{Hit Rate} = \frac{g(C_l, \pi')}{n_l} \tag{17}$$

Table 1: Results of comparing several alternative clustering approaches for Non-Hierarchical clustering, where π is given and π' is the new clustering found by each approach. Note DI=Dunn Index, JI=Jaccard Index, VQE = Vector Quantization Error (results are averaged over ten random restarts of each approach).

Approach	Data with Labels		[2]			[5]			This Approach $a = \frac{5}{4}$		
Measurement	DI (π)	VQE(π)	JI	DI(π')	VQE(π')	JI	DI(π')	VQE(π')	JI	DI(π')	VQE(π')
Glass	0.21	911	0.26	0.83	855	0.24	0.38	505	0.29	0.43	407
Ionosphere	0.65	3086	0.54	1.21	3207	0.43	0.98	2421	0.46	0.77	2716
ESL	0.38	1374	0.28	0.62	1085	0.24	0.73	1787	0.13	0.67	1277
Vehicle	0.56	$2.4*10^7$	0.26	1.05	$5.5*10^6$	0.18	0.57	$5.4*10^6$	0.22	0.77	$5.0*10^6$

Table 2: Results of our approach parameterized with a, where π is given and π' is the new clustering found by each approach. Note that as a increases the diversity between π and π' increases and clustering quality suffers (results are averaged over ten random restarts of our approach).

a	$a = \frac{5}{4}$			$a = \frac{3}{2}$			$a = 2$		
Measurement	JI	DI(π')	VQE(π')	JI	DI(π')	VQE(π')	JI	DI(π')	VQE(π')
Glass	0.29	0.43	407	0.28	0.32	412	0.20	0.33	822
Ionosphere	0.46	0.77	2716	0.47	0.74	2813	0.47	0.68	3126
ESL	0.13	0.67	1277	0.12	0.65	1514	0.10	0.62	2216
Vehicle	0.22	0.77	$5.0*10^6$	0.19	0.67	$6.5*10^6$	0.16	0.53	$1.6*10^7$

In Table 3, we shows that the new clustering π' in three data sets is of good quality as measured by DI and VQE. Meanwhile, π' is not only different to π as compared by JI but also maintains the cluster we want to keep as indicated by the high Hit Rates. We can increase the Hit Rate by increasing the exponent a in Eq.(12).

6. CONCLUSION

Data mining aims to find novel and actionable patterns with most algorithms typically returning just one such set of results. However, in some circumstances we wish to find multiple alternative explanations of the data. In this paper we study the following problem: given a clustering, find a good quality alternative which we have termed the singular alternative clustering problem. This allows the domain expert who already has a not useful clustering to encode this knowledge so that the algorithm does not find the same clustering again. Multiple sequential solutions to this problem can also be used to find many possible alternative patterns in the data.

Previous works to address this problem, including our own, were limited in several ways. Firstly, they were (except our own [5]) algorithm-dependent; secondly, they were all (including our own [5]) unrefined in the sense that they only allowed the domain expert to say "find a completely alternative clustering" but did not allow them to specify what properties of the given clustering to keep or not keep.

In this paper we formulate a solution to this problem as a constrained optimization problem that minimizes the Kullback-Leibler divergence between the probability density functions of the original data set and a new transformed data set. This ensures that the properties of the transformed data closely match those of the original data set. The constraints specify which properties of clustering should or should not be maintained and can be modified to multiple variations of the problem in a principled and flexible way. Variations include finding alternative clusterings in subspaces as well as trading off clustering quality with alternativeness to the original clustering.

There are several advantages to our approach. It is general purpose since it can be used with many clustering algorithms based on a distance function, such as k-means and agglomerative algorithms. The transformation of the data is specified in closed-form and is easy to implement. Our approach can specify which parts of the clustering are desirable and which are not. Furthermore, the tradeoff between alternativeness and quality can be controlled by the parameter a in Eq.(12).

To validate our approach we performed a set of experiments on the low dimensional UCI data set (see Table 1) to compare our approach against the techniques of others [2] and our previous work [5]. As discussed and shown in our earlier work, the approach of Cui et.al. [4] is not designed for lower dimensional data and hence does not perform well. It is important to note that all of the clustering quality measurements we present are calculated based on the original data X. **That is, even though we transform X to an alternative space we measure the alternative clusterings' properties in the original space.** We illustrated that our approach not only achieves clusterings of comparable quality but also finds a diverse set of clusterings. We were able to focus on alternativeness or quality by adjusting the parameter a (see Table 2). Our approach can also find different interpretable clusterings in image segmentation applications (see Figures 3, 4 and 5). Finally, we presented the experiments where the desirable and undesirable clusters in the new clustering can be explicitly specified, and the results showed that our approach can find alternative clusterings while maintaining the expected clusters (See Table 3). Our future work will include kernelizing the current approach to find alternative clusterings by a non-linear transformation of the data.

Acknowledgments

The authors thank the anonymous reviewers for their excellent comments and the NSF for support of this work via GRANT IIS-0801528 CAREER: Knowledge Enhanced Clustering with Constraints.

7. REFERENCES

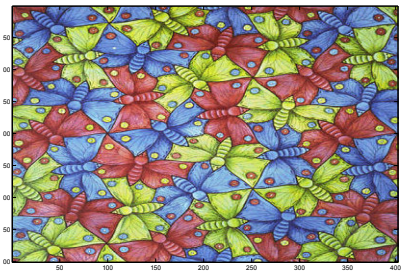
[1] A. Asuncion and D. Newman. UCI machine learning repository, 2007.

[2] E. Bae and J. Bailey. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *ICDM '06: Proceedings of the Sixth International Conference on Data Mining*, pages 53–62, 2006.

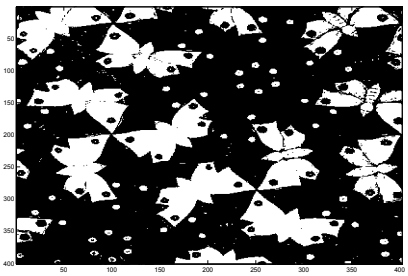
[3] T. Coleman, J. Saunderson, and A. Wirth. Spectral clustering with inconsistent advice. In *ICML '08: Proceedings of the*

Table 3: Results of specifying a cluster to keep and not to keep (results are averaged over ten random restarts of our approach).

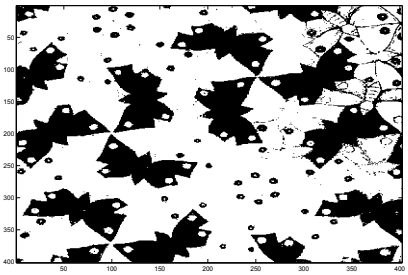
Approach	Data with Labels		This Approach (a=2)			
Measurement	DI (π)	VQE(π)	DI (π')	VQE(π')	JI	Hit Rate
Glass	0.21	911	0.38	757	0.31	0.73
ESL	0.38	1374	0.75	1122	0.31	0.74
Vehicle	0.56	$2.4*10^7$	0.57	$2.0*10^7$	0.31	0.99



(a) The original butterfly image



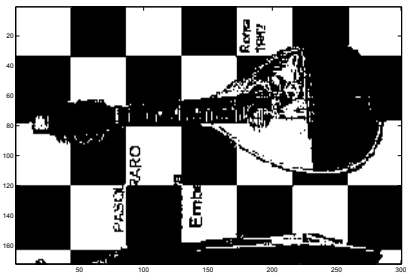
(b) π found by the spectral clustering



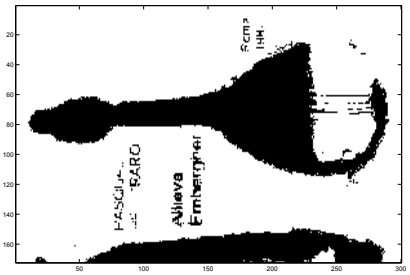
(c) π' found by our approach given π

Figure 4: Escher butterfly example of textures with alternative interpretations, $k = 2$.

(a) The original mandolin image



(b) π found by the spectral clustering



(c) π' found by our approach given π

Figure 5: Mandolin example, $k = 2$.

25th international conference on Machine learning, pages 152–159, 2008.

[4] Y. Cui, X. Z. Fern, and J. G. Dy. Non-redundant multi-view clustering via orthogonalization. In *ICDM '07: Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 133–142, 2007.

[5] I. Davidson and Z. Qi. Finding alternative clusterings using constraints. In *ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, 2008.

[6] I. Davidson and S. S. Ravi. The complexity of non-hierarchical clustering with instance and cluster level constraints. *Data Min. Knowl. Discov.*, 14(1):25–61, 2007.

[7] I. Davidson and S. S. Ravi. Intractability and clustering with constraints. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pages 201–208, 2007.

[8] I. Davidson, S. S. Ravi, and M. Ester. Efficient incremental constrained clustering. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 240–249, New York, NY, USA, 2007. ACM.

[9] D. Gondek and T. Hofmann. Non-redundant data clustering. In *ICDM '04: Proceedings of the Fourth IEEE International Conference on Data Mining*, pages 75–82, 2004.

[10] P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learning of disparate clusterings. In *SDM '08: Proceedings of the SIAM International Conference on Data Mining*, pages 858–869, 2008.

[11] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

APPENDIX

A. SOLUTION TO EQ.(1)

Here we derive the solution to the optimization problem in Eq.(1): $D = \widetilde{\Sigma}^{-\frac{1}{2}}$.

Since Y is transformed from X by the transformation $Y = DX$, we can find the connection between two density functions $p_x(x)$ and $p_y(y)$: $p_y(y) = \frac{p_x(x)}{|D|}$, where $|D|$ is the Jacobian determinant of transformation matrix D. Then we can rewrite the objective function in Eq.(1) as follows:

$$\begin{aligned} D_{KL}(p_y(y)||p_x(x)) &= \int_y p_y(y) \log \frac{p_y(y)}{p_x(x)}\, d(y) \\ &= \int_x \frac{p_x(x)}{|D|} \log \frac{1}{|D|}\, d(Dx) \\ &= \int_x p_x(x) \log \frac{1}{|D|}\, d(x) \end{aligned} \tag{18}$$

To minimize $D_{KL}(p_y(y)||p_x(x))$ is to maximize $\log|D|$. Therefore, Eq.(1) is equal to Eq.(19).

$$\max_{D \succeq 0} \ \log|D| \ \ s.t. \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 \leq \beta \tag{19}$$

Since B is a positive definite matrix where $B = D^T D$, and it can be rewritten as follows:

$$\max_{B \succeq 0} \ \log|B| \ \ s.t. \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 \leq \beta \tag{20}$$

Use the method of Lagrange multiplier,

$$\begin{aligned} &\log|B| - \gamma(\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1, x_i \notin C_j}^{k} \|(x_i - m_j)\|_B^2 - \beta) \\ =&\log|B| - \gamma(\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1, x_i \notin C_j}^{k} (x_i - m_j)^T B (x_i - m_j) - \beta) \\ =&\log|B| - \gamma(\frac{1}{n}(tr(B \sum_{i=1}^{n} \sum_{j=1, x_i \notin C_j}^{k} (x_i - m_j)(x_i - m_j)^T)) - \beta) \\ =&\log|B| - \frac{\gamma}{n}(tr(B\widetilde{\Sigma}) - \beta) \end{aligned} \tag{21}$$

Take the derivative of B to get Eq.(22) and let it be 0. The scaler $\frac{\gamma}{n}$ does not impact the transformation and can be removed. Then we get the solution $B = \widetilde{\Sigma}^{-1}$ i.e. $D = \widetilde{\Sigma}^{-\frac{1}{2}}$.

$$tr(B^{-1}) - \frac{\gamma}{n} tr(\widetilde{\Sigma}) \tag{22}$$

B. LIST OF NOTATIONS

The following notations are used in this paper.

Notation	Description
X	The given d-dimensional data set ($X \subseteq \mathbb{R}^d$)
x_i	The ith object in X
n	Number of objects in X
k	Number of clusters
π	The given clustering found in X
Y	The transformed data set by transformation $Y = DX$ ($Y \subseteq \mathbb{R}^d$)
y_i	The ith object in Y
π'	The new clustering found in Y
$p_x(x)$	Probability density function of X
$p_y(y)$	Probability density function of Y
D	The transformation matrix
B	$B = D^T D$
C_i	The ith cluster in π
n_i	Size of C_i
m_i	Centroid of C_i
$\widehat{C}_i$	The ith cluster in π'
$\widehat{n}_i$	Size of $\widehat{C}_i$
$\widehat{m}_i$	Projection of m_i in Y
I	Identity matrix
$\|\| \cdot \|\|_B$	Mahalanobis distance with weight matrix B
$f_i(y)$	Multivariate Gaussian distribution which $\widehat{C}_i$ follows
$\widehat{\Sigma}$	Covariance matrix of $f_i(y)$, ($1 \leq i \leq k$)
$D_{\text{KL}}(p_y\|\|p_x)$	Kullback-Leibler divergence between two distributions $p_y$and p_x
$\widetilde{\Sigma}$	Auxiliary matrix to derive D defined in Eq.(11)
a	Parameter to specify the trade-off between alternativeness and quality in Eq.(12)
C_Y	The set of clusters to be retained in Section 4
C_N	The set of clusters not to be retained in Section 4
$s(x_i, x_j)$	The constraint that x_i and x_j must be in the same cluster
S	$S = \{s(x_i, x_j), ...\}$ includes all the specified constraints in Section 4
$\widetilde{\Sigma}_1$	Auxiliary matrix to derive D defined in Eq.(14)
$\widetilde{\Sigma}_2$	Auxiliary matrix to derive D defined in Eq.(16)
$M \succeq 0$	Matrix M is positive semidefinite