Chapter 21

Alternative Clustering Analysis: A Review

James Bailey

Department of Computing and Information Systems
The University of Melbourne
Melbourne, Australia
baileyj@unimelb.edu.au

21.1 Introduction 533
21.2 Technical Preliminaries 535
21.3 Multiple Clustering Analysis using Alternative Clusterings 536
21.3.1 Alternative Clustering Algorithms: A Taxonomy 536
21.3.2 Unguided Generation 537
21.3.2.1 Naive 537
21.3.2.2 Meta Clustering 537
21.3.2.3 Eigenvectors of the Laplacian Matrix 537
21.3.2.4 Decorrelated k-means and Convolutional EM . 538
21.3.2.5 CAMI 538
21.3.3 Guided Generation with Constraints 539
21.3.3.1 COALA 539
21.3.3.2 Constrained Optimization Approach 539
21.3.3.3 MAXIMUS 540
21.3.4 Orthogonal Transformation Approaches 540
21.3.4.1 Orthogonal Views 541
21.3.4.2 ADFT 541
21.3.5 Information Theoretic 542
21.3.5.1 Conditional Information Bottleneck (CIB) 542
21.3.5.2 Conditional Ensemble Clustering 542
21.3.5.3 NACI 542
21.3.5.4 mSC 543
21.4 Connections to Multiview Clustering and Subspace Clustering 543
21.5 Future Research Issues 544
21.6 Summary 545
Bibliography 545

21.1 Introduction

Clustering is one of the most fundamental and important techniques in knowledge discovery and is used in a wide variety of fields, ranging from biomedicine and information retrieval, to financial analysis and Web mining. Clustering analysis provides a way to automatically identify patterns and

relationships in complex data, to form hypotheses about its structure, and to make predictions for sub classes of objects.

There exist a large variety of knowledge discovery workflows that rely on clustering. In its basic form, clustering analysis is used to explore a complex dataset, by automatically identifying object groupings. Given an input dataset for analysis, a clustering algorithm, such as k-means, can be executed, producing a clustering as output. This output clustering consists of a set of clusters which partition the objects in the dataset.

It is well known that the process of clustering is subjective and the clustering that is output is strongly dependent on the nature of the specific clustering algorithm chosen. Indeed, vastly different outputs may be possible if one changes the clustering algorithm, or varies the parameters input to a fixed algorithm. Alternative outputs are also possible according to different pre-processing methods for the input, such as when applying a feature selection step.

This inherent subjectiveness and instability of clustering is widely recognised and has provided impetus to the emerging area of multiple clustering analysis. The philosophy here, is that making the assumption that only a singe clustering exists for a dataset is too strict. Instead, one should expect that multiple *alternative* clusterings are reasonable for a dataset. Each one of these alternatives corresponds to a different grouping of the objects and reflects a different perspective, view, or hypothesis about the nature of the data.

Why might multiple clusterings be reasonable for the same dataset ? Firstly, the data being analysed could be very complex, containing many features, which may be of different types. Different combinations of these features (or subspaces) may provide natural alternative perspectives of the data. The data might also consist of many instances, meaning a diversity of possible sub populations, resulting in many possible views. Secondly, the data could be temporal in nature and evolving over time. As the data evolves, concept drift can occur, meaning that different groupings of the data become stronger or weaker. Thirdly, the data may be spatial in nature, meaning that the different perspectives have a spatial origin. Fourthly, the data objects may be diverse, due to datasets being merged or information having been integrated from multiple sources (e.g. a large clinical cohort study that pools data from multiple sites). Again, this may mean that multiple perspectives of the data are necessary and natural for knowledge discovery, rather than relying on just a single perspective.

Given that multiple clusterings or views of the data are possible, it is therefore also important to consider why they may be important for a user. Firstly, clustering analysis is frequently exploratory in nature. A user often does not know what behaviour they are looking for. What they need, is to navigate through and assess multiple alternatives, so they can evaluate different options. Conversely, the user may have a strong hypothesis (clustering) in mind and desire to verify that no other strong hypotheses are supported by the data. Secondly, users themselves can differ widely in their requirements and expectations. It is therefore unlikely that a single clustering will be appropriate for all users. Thirdly, a common scenario in data mining studies is that the investigation focuses on a new or novel clustering algorithm and it is necessary to test the flexibility and limits of this proposed algorithm, to assess how many alternatives it is able to identify.

Due to these reasons, the area of multiple clustering analysis has been attracting considerable attention. Indeed several recent workshops have been devoted to the topic [25, 26, 27, 34, 24]. The literature in the area is also growing fast, with a number of algorithms proposed, which particularly focus on the problem of generating alternative clusterings, that are each of high quality and also dissimilar to one another [14, 15, 13, 5, 29, 37, 2, 3, 8, 7, 30, 31, 21, 10, 9].

The focus of this chapter is to review algorithms for generating alternative clusterings, which is one of the prime tasks in the field of multiple clustering analysis. We also highlight connections to the areas of multiview clustering and subspace clustering, which are distinct, yet closely related. In multiview clustering, the aim is to learn a single clustering using multiple sources (representations) of the data [4, 19, 35, 6, 22, 16]. These sources usually contain the same set of objects, but with different features. In subspace clustering, the aim is to discover different subspaces, where each sub-

space contains a good cluster (as opposed to clustering). (See Chapter AAA for more on subspace clustering methods).

An outline of the rest of this chapter is as follows. In Section 21.2, we present necessary terminology and definitions. In Section 21.3.1, we present a taxonomy of alternative clustering techniques and discuss different dimensions of evaluation. In Sections 21.3.2, 21.3.3, 21.3.4 and 21.3.5, we review specific approaches for alternative clustering. In Section 21.4, we compare the areas of multi view clustering and subspace clustering to alternative clustering and identify similarities and dissimilarities. In Sections 21.5 and 21.6 we outline future directions and conclude.

21.2 Technical Preliminaries

Let D be a dataset containing N objects $o_1,\ldots,o_N$ and using n features $F_1,\ldots,F_n$. A (hard) clustering[1] C is a partition of the objects in D into k clusters $\{c_1,\ldots,c_k\}$, where each cluster is a set of objects and $c_i \cap c_j = \emptyset$. Let the universe of all clusterings of D be denoted as $\mathbb{C}_D$. We will also use the notation C^i to refer to cluster c_i of clustering C.

The quality of a clustering may be measured using a function $Qual : \mathbb{C}_D \rightarrow [0,1]$ where higher values indicate higher quality. A large range of quality measures have been defined, with some well known examples being the Dunn Index [12], the David Bouldin Index [11] and the Silhouette Width [33].

Let $C_1 = \{c_1,\ldots,c_k\}$ and $C_2 = \{c'_1,\ldots,c'_{k'}\}$ be two clusterings of D. The similarity between C_1 and C_2 may be measured using a function $Sim : \mathbb{C}_D \times \mathbb{C}_D \rightarrow [0,1]$ where higher values indicate higher similarity. For measuring similarity, there are a number of possible measures, including the Rand Index [32], Adjusted Rand Index [20], Jaccard Index [17], Normalized Mutual Information [23] and Adjusted Mutual Information [36]. Measurement of similarity between clusterings is important, since it provides insight for the user into the relationship between them. When managing multiple clusterings, assessment of similarity may allow removal of redundant clusterings, selection of interesting clusterings, or increased understanding about clustering evolution. It is also a key step when exploring the convergence properties of a clustering algorithm or assessing its output compared to an expert generated clustering.

Given the large range of measures that can be 'plugged in' for measuring quality and similarity, appropriate choices are often be made in an application dependent way. We will shortly describe the issues involved in generating alternative clusterings and the different dimensions along which the existing algorithms may be compared. A general description of the task is as follows:

Definition 21.2.1 *Generalized Alternative Clustering: Given a (possibly empty) collection of clusterings $K = \{C_1,\ldots,C_m\}$ provided as background knowledge (either $K = \emptyset$, or $K \neq \emptyset$ and $m \geq 1$), generate j alternative clustering(s) $O = \{C_{m+1},\ldots,C_{m+j}\}$, such that i) $\sum_{i=m+1}^{m+j} Qual(C_i)$ is maximized and $\sum_{i,j\in[1,m+j]} sim(C_i,C_j)$ is minimized.*

The task here corresponds to generating a set of new (alternative) clusterings, where each individually is of high quality and also the pairwise similarity between the clusterings is low (the clusterings are distinctive). Three common cases are:

- $|K| = 1$ and $|O| = 1$: *singular alternative clustering*
- $K = \emptyset$ and $|O| = 2$: clusterings in O are generated in parallel: *simultaneous alternative clustering*

[1] It is also possible to use fuzzy clusterings as the basis for development, but the literature on alternative fuzzy clustering is less mature and we concentrate on the hard case.

- $|K| > 1$ and $|O| = 1$: *sequential alternative clustering*

In the next section, we consider the ways in which the behaviour of alternative clustering algorithms can be explained and specified.

21.3 Multiple Clustering Analysis using Alternative Clusterings

In this section, we will first review the different dimensions that may be used for assessing the behaviour of alternative clustering algorithms. We then describe in detail the different approaches, broken down according to style of technique.

21.3.1 Alternative Clustering Algorithms: A Taxonomy

Alternative clustering algorithms (ACAs) may be characterized in a range of different ways. We review the different options in turn.

Format of the input: The input to an ACA consists of the dataset to be clustered, which may be represented as feature valued instances, or by a similarity matrix for all pairs of objects. A technique might additionally require features to be either continuous or discrete. In addition to these, the input may optionally include *background knowledge*, which is a single existing clustering or a collection of two or more existing clusterings that are already available. It is not specified where the background knowledge comes from, it might come from the application of a standard clustering algorithm, or from user insights.

Format of the output: The output can consist of a single alternative clustering or two or more alternative clusterings. Some algorithms may place constraints on the number of clusters in each clustering (e.g. must be equal), or may require that the number of clusters in the output matches with the number of clusters in the clustering that is input as background knowledge. Also, the output may either consist of an entire (alternative) clustering, or a *partial* alternative clustering. The latter is useful if the user only wishes to change some characteristics (clusters) of the clustering(s) being used as background knowledge, while keeping other characteristics the same.

Style of output generation: If more than one alternative clustering can be output, is each alternative generated one at a time in a greedy fashion (*sequential generation*) or are all alternatives generated in parallel (*simultaneous generation*) ? The latter may produce a more globally optimal solution. However, the former may be more realistic when one or more existing clusterings exist. It might also identify some strong clusterings which would be missed by a simultaneous generation technique.

Style of Technique: This describes the overall process of alternative clustering generation. One class of techniques is *unguided generation*, where no background knowledge is used for generation of alternatives. The other class of techniques is *guided generation*, where background knowledge is used as input and explicit effort is made to ensure dissimilarity between new clusterings and existing ones specified in the background knowledge. Guided generation techniques can be further broken down, according to whether the technique i) relies on the use of inferred constraints to generate alternatives (*constraint based*), or ii) operates by generating feature spaces which are orthogonal to the existing feature space and to each other (*orthogonal feature space transformation*) . Such a transformation means the feature space of the alternative clustering(s) is different from that of the clustering(s) in the background knowledge, or iii) uses an objective function based on information theoretic criteria, in order to optimize quality and dissimilarity characteristics (*information theoretic*).

Style of clustering algorithm: Is the method tied to a specific technique (e.g. *k*-means, hierarchical or expectation maximization) ? Or can any clustering algorithm be plugged in ? The latter is typically possible for the orthogonal feature space transformation style, where once an orthogonal feature space is discovered, any clustering algorithm can be used to generate the alternative. This increases flexibility and means an appropriate clustering algorithm can be chosen according to the dataset and desired cluster characteristics.

Parameter requirements: Apart from the number of clusterings output and the number of clusters in each, are there any other parameters that must be specified to use the technique ? Many techniques rely on the use of a regularization/tradeoff parameter, which is used to tune the relative weightings of quality and dissimilarity criteria in the objective function.

We now proceed with a description of the different approaches, grouped according to the style of the technique.

21.3.2 Unguided Generation

Unguided generation techniques do not employ any background knowledge for generating the alternative clustering(s). There are a variety of approaches in this category, ranging from the simple to the sophisticated.

21.3.2.1 Naive

The most basic technique is the naive method. Using this technique, alternative clusterings are obtained by i) running a clustering algorithm multiple times, using different parameters each time, or ii) running different clustering algorithms, or iii) a combination of i) and ii). The advantage of such an approach is that it is straightforward to implement and any clustering algorithm(s) may be used The principal disadvantage is that its behavior can be quite random and there is a risk of generating alternative clusterings that are very similar to one another. Background knowledge is also not taken into account. Due to the issue of redundancy, post processing is required to filter out clusterings having a high amount of overlap. Naive generation is a very common technique employed by users who are not familiar with alternative clustering. It is also often used in conjunction with consensus clustering, where the alternatives are combined into a single clustering using a voting strategy.

21.3.2.2 Meta Clustering

An extension of the naive technique is the approach of meta clustering [5]. Similar to naive, meta clustering does not make use of any background knowledge to generate alternatives. Instead, it adopts a more principled approach to achieve dissimilarity. In particular, k-means is repeatedly used with i) random choices of initial centroids and ii) different attribute weightings in the distance functions, according to a Zipf distribution. After generation of alternative clusterings in this fashion, meta clustering then treats the output clusterings as objects themselves and clusters them using a cluster difference distance function. This yields a meta level perspective for the clusterings, which can be explored by the user. Whilst the generation strategy here is more sophisticated than the naive one, it still does not explicitly ensure that the output clusterings will be dissimilar, but only that they have a reasonable chance to be dissimilar. Also, the random use of centroids in k-means again may result in duplicates. Furthermore, the use of differently weighted distance functions, while it increases the chance of dissimilarity between alternatives, may have an impact on the quality of the clusterings, since some weightings may produce unnatural output clusterings. For these reasons, like the naive technique, the results are likely to need postprocessing to reduce redundancy. Again though, like naive, meta clustering has the advantage of being simple and clean to implement.

21.3.2.3 Eigenvectors of the Laplacian Matrix

Dasgupta et al [9] show that alternative clusterings can be found by looking at different eigenvectors of the Laplacian matrix. The input is a similarity matrix S and no background knowledge is used. There is the strong requirement that each alternative clustering output is constrained to have two clusters. The approach is a spectral one, where the objects are represented as a graph, with edges between nodes indicating pairwise similarities and a partition is generated using a normalized cut criterion. Let $D_{i,i} = \sum_j S_{i,j}$ and the Laplacian matrix L is $L = D^{-1/2}(D-S)D^{-1/2}$. The first alternative clustering is found by applying 2-means to the objects represented by e_2, the eigenvector corresponding to the second smallest eigenvalue of L. The mth alternative clustering is produced by applying 2 means to the objects represented by the $m+1$th eigenvector of L. The dissimilarity objective is achieved by the orthogonality of the different eigenvectors. Quality is achieved by the 2-means algorithm, but the second and later alternatives will be "suboptimal", compared to the first, since the optimality decreases as m increases. The approach has the advantage of being simple to implement, but the limitation of two clusters in each clustering is restrictive.

21.3.2.4 Decorrelated k-means and Convolutional EM

The approach by Jain et al [21] is a simultaneous one for generating two clusterings C_1 and C_2, without using any background knowledge. Supposing each of the output clusterings has k_1 and k_2 clusters respectively, then they are generated in a decorrelated fashion using k-means style. The objective function has the form:

$$\sum_{i=1}^{k_1}\sum_{x\in C_1^i}||x-\mu^i||^2+\sum_{j=1}^{k_2}\sum_{x\in C_2^j}||x-\nu^j||^2+\lambda\sum_{i,j}(\beta_j^T\mu^i)^2+\lambda\sum_{i,j}(\alpha_i^T\nu^j)^2$$

where λ is a parameter used for regularization, μ^i and ν^j are the representative vectors of clusters C_1^i and C_2^j respectively and α_i and β_j are the mean vectors of C_1^i and C_2^j respectively. The initial two terms correspond to k means type error terms, whilst the second two terms ensure dissimilarity (decorrelation) between the two clusterings. The objective function can be extended to generate more than 2 clusterings, by including an extra k means type error term for each new clustering and including a pairwise dissimilarity term for each possible pair of clusterings. An iterative approach is used to minimize the objective function. The regularization parameter λ is set empirically and it is also possible to extend the objective to a kernelized version, to handle non linearities. A disadvantage is that the representative vectors in the decorrelated k-means algorithm do not have a natural interpretation for the user.

In a companion proposal to decorrelated k-means, the work in [21] also outlines a convolutional EM algorithm, where it is assumed that the data can be modeled as the sum of two mixtures of distributions, each of which is associated with a clustering. One clustering has k_1 clusters, the other has k_2 clusters. Then, since the distribution of the sum of two independent random variables is the convolution of the distributions, the data is modeled as being sampled from a convolution of two mixtures. This then leads to the problem of learning a convolution of mixture distributions, using an expectation maximization method to determine the distributions' parameters. The technique is again simultaneous and can be kernelized.

21.3.2.5 CAMI

The CAMI [7] algorithm is designed to discover two alternative clusterings at the same time using the original data space. Formulating the clustering problem under mixture models, CAMI optimizes a dual-objective function in which the log-likelihood (accounting for clustering quality) is maximized, while the mutual information between two mixture models (accounting for the dis-

tinction between two clusterings) is minimized. The objective function of CAMI can be written as

$$L(\Theta,D) = L(\Theta^1;D) + L(\Theta^2;D) - \eta \sum_{i,j} p(C_1^i,C_2^j) \log \frac{p(C_1^i,C_2^j)}{p(C_1^i)p(C_2^j)}$$

The first two terms correspond to the likelihood of each of the two clusterings that will be simultaneously discovered and Θ^1 and Θ^2 are their parameters. The third term corresponds to the dissimilarity between the clusterings C_1 and C_2 as measured by mutual information. The η is a regularization parameter used to trade off dissimilarity and quality (and which can be specified by the user). Using Gaussian mixture models, an EM approach can be used to optimize the objective function.

21.3.3 Guided Generation with Constraints

The next class of techniques uses constraints to guide the generation of one or more alternative clusterings. The type of constraints and the way they are used distinguishes each of the methods.

21.3.3.1 COALA

The COALA method takes as input a similarity matrix and a single existing clustering as background knowledge. It uses an hierarchical algorithm. Using the existing clustering, a set of "cannot-link" constraints are generated, one for each pair of objects in the same cluster. Intuitively, it is less desirable for objects in these pairs to again be together in the same cluster of the alternative clustering. A hierarchical clustering approach is then used. At each iteration, COALA finds two candidate pairs of clusters for a possible merge, one denoted as $(q1,q2)$, called a qualitative pair and the other denoted as $(o1,o2)$, called a dissimilar pair. The qualitative pair is the one with the minimum distance over all the pairs of clusters (ensuring the highest quality clusters when merged). The dissimilar pair has the minimum distance over all the pairs of clusters that also satisfy the cannot-link constraints (these pairs may be the same). COALA will select just one of these pairs to merge. Given a tradeoff factor parameter ω, if $\frac{d(q_1,q_2)}{d(o_1,o_2)} \geq \omega$ then the pair (o_1,o_2) is merged. Otherwise, the pair (q_1,q_2) is merged. By varying the value of ω, different behaviours can be achieved.

COALA is a simple and intuitive technique and has been used as a baseline method for comparing against in a range of papers. A limitation of COALA is that it is specifically tied to a hierarchical clustering algorithm. It also was not formulated for the case of generating multiple alternative clusterings. However, it is easy to conceive generalizations in which multiple clusterings are used as background knowledge, yielding a larger set of cannot-link constraints for generating the alternative clustering.

21.3.3.2 Constrained Optimization Approach

The approach of Qi and Davidson in [31] uses constraints in a different way. It takes the original dataset $X = \{x_1,\ldots,x_n\}$ and transforms it to a new dataset $Y = \{y_1,\ldots,y_n\}$ where $Y = DX$ and D is a transformation matrix representing a distance metric. Any clustering algorithm can then be applied to the new dataset to generate an alternative to the original clustering.

The objective function is formulated as a constrained optimization task

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

where $B = D^T D$ and $||.||_B$ is the Mahalanobis distance using matrix B and $B \succeq 0$ signifies that B is required to be positive semi-definite.

The original dataset X follows probability density function $p_x(x)$ and dataset Y follows probability density function $p_y(y)$. D_{KL} signifies the KL divergence between two distributions and $a \geq 1$ is a tradeoff parameter, with larger values ensuring higher dissimilarity of the alternative clustering.

The first part of the objective aims to ensure that the transformed data preserves the characteristics of the original data (with the KL distance being zero when they are identical). The second part of the objective ensures dissimilarity and discourages the original clusters being found, by requiring each object in the new data space to be closer to the cluster centers of the cluster that it was not originally part of. To achieve a closed form solution to the objective function, a mixture model of multivariate Gaussian distributions can be assumed, having the same covariance matrix.

Some advantages of this approach are that i) any clustering algorithm may be used to generate the alternative clustering, once the new dataset is obtained, ii) the approach extends naturally for discovering a partial alternative clustering. Users may specify properties of the original clustering they wish to keep (i.e. some original clusters or groups of objects should remain the same) and then solve the objective function with the intention of only finding some alternative clusters to add to the desired original clusters. A limitation of the approach is that it is somewhat unclear what kind of properties of the original dataset X get preserved in the new dataset Y, due to the generality of the KL-distance function.

21.3.3.3 MAXIMUS

Work by Bae and Bailey in [3] describes an algorithm known as MAXIMUS for discovering multiple alternative clusterings in a sequential manner. The MAXIMUS algorithm calculates the maximum dissimilarity between any currently available clusterings and a potential target alternative solution, by forming an integer programming model. The objective of this integer programming model is to maximize the distance between the density profiles of the known clusterings, versus the unknown target alternative clustering. It then uses the output of the model to generate an alternative clustering.

MAXIMUS is based on the use of a clustering similarity function known as *ADCO*, which can compare clusterings according to their spatial characteristics. At a high level, the *ADCO* measure constructs a spatial histogram for each cluster and represents a clustering as a vector containing the spatial histogram counts for the clusters. The two clusterings can then be compared using vector operations. Intuitively, the output of *ADCO* is a containment judgement between a clustering C_1 and a clustering C_2, expressed as "How much of clustering C_2 is contained in clustering C_1 ?", or "What percentage of clustering C_2 is contained in clustering C_1 ?".

Using the *ADCO* measure, one may generate a spatial template to ensure that a single alternative clustering has maximal (average) dissimilarity from the input background clusterings. This template describes how many objects must be present in bins within one-dimensional projections of the feature space. Using the template, a constrained k-means algorithm is used to derive a clustering for each bin. Next, consensus clustering is then used to combine the clusterings from all the bins into a single clustering. Thus, the quality of the alternative clustering is achieved by the use of k-means and consensus clustering. The dissimilarity objective is achieved by using the integer programming model and the *ADCO* measure to obtain a spatial template which can be expected to have very high dissimilarity from the background knowledge clusterings.

In order to use MAXIMUS, it is necessary to specify the binning strategy for representing the density profile (10 bins equi density is recommended as a default). Unlike some other algorithms, MAXIMUS does not require the user to specify a regularization parameter to trade off between the quality and dissimilarity objectives.

21.3.4 Orthogonal Transformation Approaches

Our next class of approaches approach the task of alternative clustering from a feature space perspective. Using an existing clustering as background knowledge, this style of approach constructs a new feature space which is 'orthogonal' to the data space that is characterised by the existing clustering. Once this orthogonal feature space is generated, any clustering algorithm can be used in this space to generate an alternative clustering. Thus, the objectives of quality and dissimilarity are decoupled, with the former being tied to the use of the chosen clustering algorithm and the latter being tied to the characteristics of the orthogonal space that gets generated. Overall, these approaches have an appealing mathematical formulation based on linear algebra. They are also relatively efficient. A limitation is that the orthogonality requirement may be too strict for some datasets and it is not always clear how it trades off against the quality of the clustering.

21.3.4.1 Orthogonal Views

Work by Cui et al in [37] presents two approaches that can generate multiple alternative clusterings, in a sequential manner. Each alternative clustering is determined by subsets of features of the data set, which are best described by the clustering. Given a clustering C_1, a subset of features is found that are well represented in C_1 and then another set of features is found, which are orthogonal to the first subset. Their first approach carries out a transformation as follows: Each data object x_i from cluster j is projected onto its cluster center μ_j and then a residue is found by projection onto an orthogonal subspace:

$$x_i^{new} = (I - \frac{\mu_j \mu_j^T}{\mu_j^T \mu_j}) x_i$$

One then clusters the data in this orthogonal subspace to obtain an alternative clustering. The method may be executed iteratively to generate multiple alternative clusterings. A version where the input is a soft (fuzzy) clustering is also outlined.

In the second approach, a feature subspace F_2 that is a good representation for the clustering C_1 is first found using principal component analysis on the mean vectors of C_1. The data X is then projected to a subspace that is orthogonal to F_2 and a clustering algorithm applied to the new data X^{new} to generate an alternative clustering C_2. Specifically

$$X^{new} = ((I - F_2(F_2^T F_2)^{-1} F_2^T)X$$

Again, the method can be applied iteratively to generate further alternative clusterings.

21.3.4.2 ADFT

Work By Davidson and Qi [10] describes the ADFT approach to finding an alternative clustering, using a set of instance level constraints. This approach is also a transformation approach like that of [37]. However, instead of characterizing the background knowledge clustering C_1 according to mean vectors or a feature subset, it is characterized using instance must-link and cannot-link constraints and then a distance function D_{C_1} is learnt using these constraints. This distance function can be decomposed using singular value decomposition into $D_{C_1} = HSA$, where H is the hanger matrix, S is the stretcher matrix and A is aligner matrix.

Once the characteristic distance function D_{C_1} has been learnt, an alternative distance function can be computed and is equal to $HS^{-1}A$. This alternative distance function is then employed to generate a new dataset $X^{new} = (HS^{-1}A)X$. The alternative clustering is then found by applying any clustering algorithm on X^{new}.

This method has an advantage over the approach of [37], since it can be applied in situations where the dimensionality of the dataset is smaller than the number of clusters (as is common for spatial data).

21.3.5 Information Theoretic

Another approach to the generation of alternative clusterings is based on the use of objective functions using information theoretic principles. Such approaches are mathematically attractive and incorporate the use of mutual information (or similar) to measure the strength of correlations between clustering. Several algorithms fall into this category, beginning with the approach of Gondek and Hofmann [15], which was the first (to our knowledge) alternative clustering algorithm to be proposed.

21.3.5.1 Conditional Information Bottleneck (CIB)

The conditional information bottleneck approach (CIB) for alternative clustering was described in [14, 15]. This algorithm takes as input an existing clustering C_1 as background knowledge and sequentially generates a single alternative clustering C_2 by optimizing the objective function

$$\max_{C_2} \left(I(C_2;F|C_1) - \lambda_1 I(C_2;X) + \lambda_2 I(C_2;F)\right)$$

where F is the features, X is the objects and the existing clustering is C_1. The term $I(C_2;F|C_1)$ corresponds to the mutual information between the new alternative clustering being discovered and the features, given the pre-defined clustering. The term $I(C_2;X)$ corresponds to the mutual information between the desired alternative clustering and the objects (this is desired to be small, to avoid being overly confident about the groupings) and $I(C_2;F)$ corresponds to the mutual information between the desired alternative clustering and the features (we want this to be high). The symbols $\lambda_1 > 0$ and $\lambda_2 > 0$ are regularization parameters, used to trade off the different components of the objective function. The approach of [15] describes an alternating optimization scheme with deterministic annealing, which can be used for generating C_2 with this objective function. In practice, this style of approach has been found to behave particularly strongly for document datasets.

21.3.5.2 Conditional Ensemble Clustering

The CIB approach of [14, 15] was further extended in [13], which introduced the CondEns (Conditional Ensemble) alternative clustering algorithm.

CondEns operates in three stages. 1) Given the clustering $C_1 = \{c_1, \ldots, c_k\}$ as background knowledge, for each cluster c_i, a local clustering is generated using any clustering algorithm. This yields k local clusterings. 2) Each of the k local clusterings is extended into a global clustering, by assigning instances not already part of a local clustering, to one of its clusters. 3) The k global clusterings are then combined using a consensus technique based on the conditional information bottleneck, to yield a single alternative clustering.

Like the approach of [15], CondEns also performs well for text datasets. A limitation of CondEns is its guarantees about the dissimilarity of the alternative clustering are somewhat unclear, since the clusters in the original clustering c_1 may be quite similar amongst themselves. This means that the alternative clustering may in turn be similar to the background knowledge clustering.

21.3.5.3 NACI

The NACI algorithm was proposed by Dang and Bailey in [8] and targets scenarios where the borders between clusters in the alternative clustering may not be linearly separable.

At a high level, its objective function can be expressed as finding an alternative clustering C_2, given a clustering C_1 as background knowledge, according to

$$C_2 = \underset{C_2}{argmax}\{I(C_2;X) - \eta I(C_1;C_2)\}$$

where

$$I(C_1;C_2) = \sum_{C_1^i} \sum_{C_2^j} (p(C_1^i, C_2^k) - p(C_1^i)p(C_2^j))^2$$

where η is a regularization parameter, $p(\cdot,\cdot)$ is the probability density and the mutual information $I(C_1;C_2)$ is in fact a quadratic form of the mutual information, which has the advantage of being amenable to density estimation using a Parzen window technique with Gaussian Kernel. This objective can then be used as a component within a hierarchical clustering framework, to generate an alternative clustering. To use NACI, choices must be made for both the regularization parameter and the kernel parameter.

Another approach in the same spirit as NACI is that of minCEntropy [29], which instead of using a hierarchical algorithm with the quadratic mutual information, instead uses a k-means style algorithm with quadratic mutual mutual information. The style is again sequential and requires specification of a kernel width parameter and a tradeoff parameter.

21.3.5.4 mSC

The final method we mention in this section is the mSC alternative clustering approach outlined in [30]. This is a spectral approach which can simultaneously generate multiple alternative clusterings.

Rather than being based on mutual information, it uses the Hilbert-Schmidt Independence Criterion (HSIC) to assess the correlation between clusterings. Like mutual information, the HSIC is also able to recognize non linear dependencies. Specifically, the mSC technique embeds the HSIC measure within a spectral clustering framework. The objective is a dual function, where at each iteration, one term is fixed and the other term is optimized. The user is able to specify the number of alternative clusterings that are desired and the number of clusters in each.

21.4 Connections to Multiview Clustering and Subspace Clustering

We have thus far reviewed a range of techniques that can be used for generating alternative clusterings, which is a core component for multiple clustering analysis. We now mention the connections that exist between alternative clustering analysis and two other directions: multiview clustering and subspace clustering.

Multiview clustering is also concerned with multiple clusterings, but from a different angle. In multiview clustering, one is provided with multiple sources or representations of data (multiple views) and then wishes to learn a single clustering which is both consistent with and a good reflection of the multiple views. A prototypical example are Web pages, which may be modeled using features which describe the frequencies of words occurring in the page (View 1), or modeled using features which describe the links into the page (View 2), or modeled using features which describe the anchor text in the links going out from the page (View 3). It has been found that using the information in all views simultaneously, one can generate a better quality clustering than if just only using a single view obtained by merging the feature spaces. A particular benefit of multiview clustering is that using multiple views to produce a clustering can reduce the effect of noise within individual views. i.e. If one were to use a single view to derive a clustering, the presence of noise may corrupt the clusters and make the detection of cluster structure more difficult. Using multiple views to cluster, however, lessens the likelihood of noise within a view being dominant and instead emphasizes the commonalities between views and their contribution towards the overall cluster structure.

Broadly speaking, there are two kinds of approaches for multiview clustering [22]. In the first approach (centralized), the multiple views are used in parallel to cluster the dataset [4, 38, 6]. In the second approach (distributed), a clustering is generated for each view independently and the clusterings are then later merged to produce a single clustering [22, 16].

In more detail for the centralised approach, Bickel and Scheffer [4] consider the setting of a dataset which has been generated by a mixture model and the objective is to determine the parameters for each of the components of the mixture. They develop both a multiview EM algorithm and a multiview k-means algorithm, which is based on an assumption of independence between views. They find that the multiview EM algorithm is able to optimize the agreement between the views and that it can achieve a significant improvement in performance compared to a single view version. They also evaluate an agglomerative multiview approach, but find that its results are not improved compared to a single view version. Zhou and Burges [38] consider a spectral clustering approach and propose an algorithm that generalizes the (single view) normalized cut to incorporate information from multiple views (graphs). The approach uses a random walk technique that traverses the vertices of both graphs, to derive a multiple graph cut which is good on average for both graphs. They find their approach consistently performs better than just using a single view. Chaudhuri et al [6] address the problem of clustering in high dimensions and how to discover a lower dimensional subspace, in which a standard clustering algorithm can then be applied. In their work, this lower dimensional subspace is found using the information from multiple views, where each view is composed of a mixture of distributions. A canonical correlation technique is used for subspace learning.

In more detail for the distributed approach, Long et al [22] propose a pattern based technique based on the use of a mapping function. After independently clustering each view, these clusterings are then combined into a single view using the mapping function. The objective function minimizes the averaged mapped distance of the views to the overall clustering, using an iterative algorithm. Greene and Cunningham [16] tackle the problem by proposing a matrix factorization approach. Specifically, a matrix is constructed that summarizes all the clusterings (one clustering per view). This matrix is then factorized (possibly with some approximation error) into the product of two non-negative matrices. The first contains information about the contribution of each cluster from the views to the overall, final clustering. The second matrix describes the membership of objects in the final clustering.

A key issue for multiview clustering is how to balance the relative contributions of the views and ensure that noisy views do not degrade the final result [35]. Another key issue for multiview clustering is how to handle application specific multiview integration. For example, the techniques needed to combine multiple views in a document domain, may be quite different from what is appropriate for combining multiple views in a protein (bioinformatics) domain. The multiview paradigm has also been extended to the discovery of subspaces, rather than aiming to produce only one, overall clustering [19].

Like alternative clustering, subspace clustering is also concerned with discovering multiple solutions. Here though, the principal aim is to discover multiple clusters, each hidden in a lower dimensional subspace, rather than discovering multiple clusterings. The motivation is that the dataset can contain features which are irrelevant to and confusing for clustering structure. Removing these features can make the clustering structure clearer and of better quality. Some well known examples include CLIQUE [1], MAFIA [28] and DENCLUE [18]. Issues for subspace clustering analysis include ensuring the dissimilarity between subspace clusters (which may otherwise have large overlap) and controlling the number of subspace clusters (which may be exponential in the number of features).

21.5 Future Research Issues

There are a number of issues that still remain to be explored in alternative clustering analysis.

1. Many good approaches have been proposed for generating alternative clusterings. These have tended to be evaluated on synthetic data, or small real-world data. Their degree of scalability is thus often untested. In order to extend the reach and applicability of alternative clustering, some more serious evaluation will be needed that is based on the use of very large datasets.
2. Discovery of alternative clusterings is intuitively reasonable. However, it will be important to identify application scenarios and compelling case studies where alternative clusterings have influence for a real application area. Good visualization tools for alternative clusterings could have potential impact here.
3. A number of alternative clustering methods are capable of generating more than one alternative. This raises the issue of how many alternatives is sufficient. Is this an issue which is user dependent, much like choosing the number of clusters, or are there more principled ways to evaluate the viability of alternatives ? Coupled with this issue, is the companion question of how many clusters should be included in each alternative clustering.
4. The traditional notion of a 'complete' alternative may sometimes be too strict. Instead, a user may sometimes desire partial alternatives, where the new clustering is similar in some respects, but different in other respects to the existing clustering(s). Work by Qi and Davidson [31] is a promising basis here.

21.6 Summary

We have reviewed the area of alternative clustering analysis. The impetus for the field has come from the complexity and heterogeneity of today's datasets. Users wish to obtain not only a single view or hypothesis of their data, but instead be presented with several alternatives.

We have seen that a number of approaches for alternative clustering exist, possessing considerable diversity in their technical details. At the core of each though, is the capability to generate new clusterings which achieve a balance between being novel and being different from clusterings that are already known.

The area has grown rapidly in the last few years and we believe it has a bright future. As the techniques become more widely known, the generation of alternative clusterings may become common place. This invites the following speculation "In the future, might every clustering be accompanied by an alternative clustering ?"

Bibliography

[1] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *International Conference on Management of Data*, pages 94–105, 1998.

[2] Eric Bae and James Bailey. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *Proc. of the International Conference on Data Mining (ICDM)*, pages 53–62, 2006.

[3] Eric Bae, James Bailey, and Guozhu Dong. A clustering comparison measure using density profiles and its application to the discovery of alternate clusterings. *Data Mining and Knowledge Discovery*, 21(3):427–471, 2010.

[4] Steffen Bickel and Tobias Scheffer. Multi-view clustering. In *Proceedings of the 4th IEEE International Conference on Data Mining (ICDM 2004)*, pages 19–26, 2004.

[5] R. Caruana, M. Elhawary, N. Nguyen, and C. Smith. Meta clustering. In *Proc. of the International Conference on Data Mining (ICDM)*, pages 107–118, 2006.

[6] Kamalika Chaudhuri, Sham M. Kakade, Karen Livescu, and Karthik Sridharan. Multi-view clustering via canonical correlation analysis. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, page 17, 2009.

[7] Xuan Hong Dang and James Bailey. Generation of alternative clusterings using the CAMI approach. In *Proceedings of the SIAM International Conference on Data Mining, SDM 2010, April 29 - May 1, 2010, Columbus, Ohio*, pages 118–129, 2010.

[8] Xuan Hong Dang and James Bailey. A hierarchical information theoretic technique for the discovery of non linear alternative clusterings. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*, pages 573–582, 2010.

[9] Sajib Dasgupta and Vincent Ng. Mining clustering dimensions. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 263–270, 2010.

[10] Ian Davidson and Zijie Qi. Finding alternative clusterings using constraints. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy*, pages 773–778, 2008.

[11] David L. Davies and Donald W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:224–227, 1979.

[12] J. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1974.

[13] D. Gondek. Non-redundant clustering with conditional ensembles. In *Proc. of the International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 70–77, 2005.

[14] D. Gondek and T. Hofmann. Conditional information bottleneck clustering. In *3rd International Conference on Data Mining, Workshop on Clustering Large Data Sets*, pages 36–42, 2003.

[15] D. Gondek and T. Hofmann. Non-redundant data clustering. In *Proc. of International Conference on Data Mining (ICDM)*, pages 75–82, 2004.

[16] Derek Greene and Padraig Cunningham. A matrix factorization approach for integrating multiple data views. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD*, pages 423–438, 2009.

[17] L. Hamers, Y. Hemeryck, G. Herweyers, M. Janssen, H. Keters, R. Rousseau, and A. Vanhoutte. Similarity measures in scientometric research: the Jaccard index versus Salton's cosine formula. *Information Processing and Management*, 25(3):315–318, 1989.

[18] A. Hinneburg and D. Keim. An efficient approach to clustering in large multimedia databases with noise. In *International Conference on Knowledge Discovery and Data Mining*, pages 58–65, 1998.

[19] Ming Hua and Jian Pei. Clustering in applications with multiple data sources - a mutual subspace clustering approach. *Neurocomputing*, 92:133–144, 2012.

[20] L Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1):193–218, 1985.

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

[22] Bo Long, Philip S. Yu, and Zhongfei (Mark) Zhang. A general model for multiple view unsupervised learning. In *Proceedings of the SIAM International Conference on Data Mining, SDM*, pages 822–833, 2008.

[23] M. Meila. Comparing clusterings - an information based distance. *Journal of Multivariate Analysis*, 98(5):873=895, 2007.

[24] Emmanuel Muller, Stephan Gunnemann, Thomas Seidl, and Ines Farber. *Tutorial: Discovering Multiple Clustering Solutions Grouping Objects in Different Views of the Data*. ICDM 2010, SDM2011, ICDE 2012.

[25] *Proc. of the 1st International Workshop on Discovering, Summarizing and Using Multiple Clusterings (MultiClust). Held in conjunction with the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. July 25, 2010.

[26] *Proc. of the 2nd International Workshop on Discovering, Summarizing and Using Multiple Clusterings (MultiClust). Held in conjunction with the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*. 5 September, 2011.

[27] *Proc. of the 3rd International Workshop on Discovering, Summarizing and Using Multiple Clusterings (MultiClust). Held in conjunction with the 2012 SIAM International Conference on Data Mining (SDM)*. April 26, 2012.

[28] H. Nagesh, S. Goil, and A. Choudhary. Mafia: Efficient and scalable subspace clustering for very large data sets. technical report 9906-010, northwestern university, 1999.

[29] Xuan Vinh Nguyen and Julien Epps. minCEntropy: A novel information theoretic approach for the generation of alternative clusterings. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, pages 521–530, 2010.

[30] Donglin Niu, Jennifer G. Dy, and Michael I. Jordan. Multiple non-redundant spectral clustering views. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 831–838, 2010.

[31] Zijie Qi and Ian Davidson. A principled and flexible framework for finding alternative clusterings. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*, pages 717–726, 2009.

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

[33] P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[34] *Workshop on Multi-view data, High-dimensionality, External Knowledge: Striving for a Unified Approach to Clustering. Held in conjunction with the the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2012)*. May 29, 2012.

[35] Grigorios Tzortzis and C. L. Likas. Multiple view clustering using a weighted combination of exemplar-based mixture models. *IEEE Transactions on Neural Networks*, 21(12):1925–1938, 2010.

[36] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 9999:2837–2854, 2010.

[37] Xiaoli Fern Ying Cui and Jennifer Dy. Non-redundant multi-view clustering via orthogonalization. In *International Conference on Data Mining*, pages 133–142, 2007.

[38] Dengyong Zhou and Christopher J. C. Burges. Spectral clustering and transductive learning with multiple views. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007)*, pages 1159–1166, 2007.