ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

PROJECTIVE CLUSTERING FOR OUTLIER DETECTION IN HIGH DIMENSIONAL DATASET

R.Parimaladevi[1], C.Kavitha[2]

M.E(CSE) Second Year, Erode Sengunthar Engineering College, Erode[1]

Assistant Professor/SL GR-1, CSE Department, Erode Sengunthar Engineering College, Erode[2]

ABSTRACT: Clustering the case of non-axis-aligned subspaces and detection of outliers is a major challenge due to the curse of dimensionality. To solve this problem, the proposed implementation is extension to traditional clustering and finds subsets of the dimensions of a data space .In this project, a probability model is proposed to describe in hidden views and the detection of possible selection of relevant views. A projective clustering is proposed for Outlier Detection in High Dimensional Dataset that discovers the detection of possible outliers and non-axis –aligned subspaces in a data set and to build a robust initial condition for the clustering algorithm. Improving the parameters in the connection between L∞ corsets and sensitivity that is made in Lemma and improve clustering in the case of non-axis-aligned subspaces and detection of outliers in datasets. The suitability of the proposal demonstrated is done with synthetic data set and some widely used real-world data set.

Keywords: Clustering, high dimensions, projective clustering, probability model.

I. INTRODUCTION

DATA clustering has a wide range of applications and has been studied extensively in the statistics, data mining, and database communities. Many algorithms have been proposed in the area of clustering [1], [2]. One popular group of such algorithms, the model-based methods, have sparked wide interest because of their additional advantages, which give them the capacity to describe the underlying structures of populations in the data. In model-based methods, data are thought of as originating from various possible sources, which are typically, modeled by Gaussian mixture. The goal is to identify the generating mixture of Gaussians, that is, the nature of each Gaussian source, with its mean and covariance. Examples include the classical k-means and its variants. The goal of clustering is to group a given set of data points into clusters that capture some notion of similarity between the data points in each cluster. Data is represented by a number of features, not all of which are useful for comparing individual data points.

A. Basic Concepts of Subspace Clustering

A subspace clustering is a collection of subspace clusters. The first2 subspace clustering algorithm CLIQUE was published in 1998 and was soon followed by many related methods [1, 2, 3, 9, 13, 14, 15, 16, 19, 20, 26, 33, 39, 41, 44, 49, 51, 55, 63, 66, 67]. The algorithms have been applied for instance to clustering gene expression data: it is often the case that a group of genes behaves similarly only in a subset of experiments (i.e. in a subspace) [15, 16, 20, 26, 55, 63, 66]. Reviews of some of the existing subspace clustering algorithms can be found in [47, 48, and 68]. 1Other names that have been used for the same or a closely

In high dimensional spaces1, traditional clustering methods suffer from the curse of dimensionality, which is why their application is often preceded by feature selection and extraction. For instance, a practitioner might apply Principal Component Analysis (PCA) to project the data onto a low-dimensional subspace before trying to cluster the data points. However, it is sometimes unrealistic to assume that all clusters of points lie in the same subspace of the data space. Subspace clustering methods address this issue by assigning a distinct subspace to each group of data points.

Before proceeding, let us introduce our notation. The data matrix X consists of elements xij 2 R, where i 2 {1, 2, . . . ,m} and j 2 {1, 2, . . . , p}. We denote the m rows by {r1, r2, . . . , rm}, where ri = (xi1, xi2, . . . , xip), and the p columns by {c1, c2, . . . , cp}, where ci = (x1i, x2i, . . . , xmi)T . We will often refer to the rows as data points and to the columns as attributes. A clus- ter Ci _ {r1, r2, . . . , rm} is a subset of the data points. A clustering2 C is a partitioning of the set of m data points into clusters C1,C2, . . . ,CK of sizes m1,m2, . . . ,mK.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

Two examples of a subspace cluster. Traditional clustering algorithms will not find clear clusters in these two-dimensional data sets. However, if the points are projected onto an appropriate one-dimensional subspace, a compact cluster emerges in each case. In (a), the appropriate subspace is the x-axis, but in (b), the subspace is not aligned along the axes. a 2 R it holds that vi + vj 2 F and that avi 2 F. A subspace cluster is a pair (R,F), where R _ {r1, r2, . . . , rm} is a subset of the rows and F is a subspace of Rp. A subspace clustering S = {S1, S2, . . . , SK} is a collection of subspace clusters.

In the two-dimensional data sets, no clustering structure is visible, but if the data is projected onto a 1-dimensional subspace, a compact cluster emerges. However, in general, using feature extraction/selection before clustering is not enough to find the subspace clusters. As an example, Fig. 2.2 illustrates that PCA is of no use in solving the subspace clustering problem. The three-dimensional data set contains three subspace clusters with orthogonal 1-dimensional s.

II. NON-AXIS-ALIGNED SUBSPACE CLUSTERINGS

A non-axis-aligned subspace cluster S is a pair (R,W), where R _ {r1, r2, . . . , rm} is a subset of the rows and W is a collection of vectors {w1,w2, . . . ,wD}, where wi 2 Rp. The vectors in W form a basis for an arbitrary subspace of the original p-dimensional data space. We use W also to denote this subspace. Naturally, an axis-aligned subspace cluster is a special case of a non-axisaligned subspace cluster. In the case of an axis-aligned subspace cluster, W is a subset of the original basis vectors {e1, e2, . . . , ep}, where e1 = (1 0 0 . . . 0),e2 = (0 1 0 0 . . . 0), etc.

A non-axis-aligned subspace clustering S is a collection {S1, S2, . . . , SK} of K non-axis aligned subspace clusters. The algorithms ORCLUS, KSM , and 4C produce these kinds of clusterings. Non-axis-aligned subspace clustering is a generalization of feature extraction; instead of defining a single set of features for the whole data.

A. Meta-Clustering

Meta-clustering refers to investigating the structure of a set of clusterings. Meta-clustering discards the idea of trying to derive a single good clustering for a data set; instead, it is acknowledged that the data can be well represented in several different, complementary ways. For instance, assume that a given data set has been clustered several times by different algorithms. A metaclusterer might now observe that these clusterings form two tight groups of clusterings, and give the user a representative of each of these groups, instead of a single 'best' clustering.

There are various ways to produce different clusterings for a data set: we could use different algorithms, a single algorithm with various parameter values and initializations, change metrics, use various dimensionality reduction schemes, or sample the data. Meta-clustering may be used to investigate whether some of these clusterings form tight groups, whether some of the clusterings are outliers, whether the effect of the parameter values is strong or weak, etc. For instance, it has been empirically shown by means of metaclustering that only a small number of clustering algorithms is enough to represent a large number of clustering criteria.

B. Measures Based on Counting Point Pairs

An important class of criteria for comparing clusterings is based on counting the pairs of points on which two clusterings agree/disagree. Each pair of data points falls in one of the four categories labeled as N11, N10, N01, and N00. The category N11 contains the pairs of points that are in the same cluster both in C and in C0. The category N10 contains the pairs of points that are in the same cluster in C but not in C0. The definitions of N01 and N00 are similar.

All four Algorithm

1. Hungarian method.

Input: A cost matrix M of size K × K0.

Output: A modified cost matrix M0 of size max(K,K0)×max(K,K0) in which

it is possible to find one or more permutations of the rows/columns such that

the total cost (the sum of the diagonal elements) becomes zero. The same permutations

(excluding the extra rows/columns) are optimal also for the original

cost matrix M.

1. Make the matrix square by adding rows or columns of zeroes if necessary.

The matrix is now of size max(K,K0) × max(K,K0).

2. Subtract the row minimum from the entries of each row. Each row now

has at least one zero.

3. Subtract the column minimum from the entries of each column. Each

row and each column now has at least one zero.

4. Select rows and columns across which you draw lines, in such a way that

all the zeroes are covered and that no more lines have been drawn than

necessary.3

5. (i) If the number of the lines is max(K,K0), return the modified cost

matrix.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

(ii) If the number of the lines is smaller than max(K,K0), go to step 6.

6. Find the smallest element which is not covered by any of the lines. Then

subtract it from each entry which is not covered by the lines and add it

to each entry which is covered by both a vertical and a horizontal line.Go back to step 4.

III. A NEAR-LINEAR ALGORITHM FOR PROJECTIVE CLUSTERING INTEGER POINTS

A near-linear algorithm for integer (j; k) projective clustering in the L1 sense when the dimension is part of the input. Recall that in this problem we are given a set P of n points in Rm and integers j _ 1, k _ 0, and the goal is to _nd j k-subspaces so that the sum of the distances of each point in P to the nearest subspace is minimized; the point coordinates are integers of magnitude polynomial in m and n. Our randomized algorithm , for any parameter " > 0, runs in time O(mn polylog(mn)) and outputs a solution that with constant probability is within (1 + ") of the optimal solution.

A. Probability Model

It is important to note that the Gaussian mixture is a fundamental hypothesis that many model-based clustering algorithms make regarding the data distribution model. In this case, data points are thought of as originating from various possible sources, and the data from each particular source is modeled by a Gaussian.

B. Chenet al.: model-based method for projectiveclustering

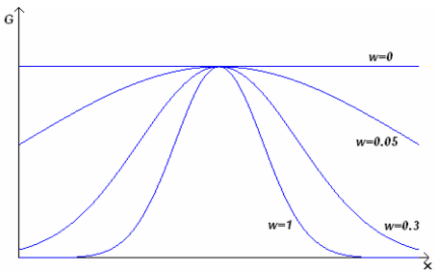

Fig.1.**Model-based method for projective clustering**

Changes in probability density with different weighting values

C. j-flat Fitting Using Lemma Concepts:

In this section, we consider the j-at fitting problem. We first introduce the concept of shape kernel and then use it to derive PTAS for the j-at fitting problem. To solve the j-at setting problem, one way is to use the concept of kernel set introduced by Agarwal et al. in. For a set P of Rd points, its kernel set is a new set of Rd points of size O (1 (__)(d□1)=2) which can be constructed through an _-net inside a unit sphere, where _ is a measure of the fatness of P. Kernel set captures the structure and extent of P and is rather powerful for solving many problems. Despite the obvious advantages provided by kernel set, there are also some issues when used for solving the RPC problem, which leads us to adopt a different structure called shape kernel. One issue is that the value of _ could be large for some point sets.

Although as pointed out in [1], it can be reduced by using some linear transform on the point set. However, this seems to be difficult to extend to the case of k _ 2 (i.e., multiple j-ats as in the RPC problem), as there may not exist a single linear transform for all j-ats. Another issue is that kernel set maintains more than succinct information for RPC. For RPC, it is actually succinct to maintain a small set of points which jointly approximate the mean of the original point set. One consequence of the redundant information in the kernel set is that its size could still be relatively large, making it di_cult to further improve the total running time of kernel set based algorithms.

To resolve the aforementioned issues, we use a different strategy to construct the kernel.

IV. AVERAGE VPC OF THE THREE FUZZY CLUSTERING ALGORITHMS

Average FScore of the algorithms, with increment of variances on the relevant dimensions algorithms choose their initial cluster centers via some random selection methods, and thus the clustering results may vary depending on the initialization. Figs. 3 and 4 show the average results of the algorithms on these data sets, in terms of VPC and FScore, respectively. Detailed clustering results on the data set with s ¼ 8, which is the most difficult case of the seven data sets (as shown in Table 2), are illustrated in Table 3. The values in the max columns correspond to the best results of the algorithms, and the average results are reported in the format average _ 1 standard deviation in the table. Figs. 3 and 4 show that outlier is able to achieve high quality overall results, especially when the clusters overlap considerably, whereas FCM, Fuzzy-FWKM, and EWKM perform poorly, and the other algorithms encounter difficulties when the cluster overlapping becomes significant, i.e., when s > 6. Examining these results in more detail, we can see that the values of VPC yielded by FCM and Fuzzy-FWKM are close to 1 K , which indicates that these two algorithms tend to assign each point to all the clusters with approximately equal membership degrees. This is due to the fact that FCM measures the similarity

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

between data points by considering all features of a data set. With the high-dimensional data used in the experiments, such a similarity measurement in the entire data space would be less meaningful due to the empty space phenomenon . Fuzzy-FWKM employs a feature weighting mechanism in the clustering process; however, each dimension is assigned the same weight for different clusters in this algorithm.

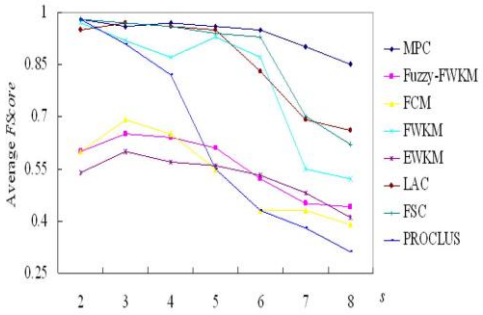

Fig 2. **Average vpc of the three fuzzy clustering algorithms**

Our result is a near-linear algorithm for integer (j; k) projective clustering in the L1 sense when the dimension is part of the input. Recall that in this problem we are given a set P of n points in Rm and integers j _ 1, k _ 0, and the goal is to _nd j k-subspaces so that the sum of the distances of each point in P to the nearest subspace is minimized; the point coordinates are integers of magnitude polynomial in m and n. Our randomized algorithm, for any parameter " > 0, runs in time O(mn polylog(mn)) and outputs a solution that with constant probability is within (1 + ") of the optimal solution. To obtain this result, we observe that in a fairly general sense, shape setting problems that have small coresets in the L1 setting also have small coresets in the L1 setting. Using this observation, and the coreset construction of for the L1 setting in axed dimension, we are able to obtain a small coreset for the L1 setting in axed dimension. To solve the problem when the dimension is part of the input, we use a known dimension reduction result.

V. EXPERIMENTAL RESULTS

Six clustering algorithms, OUTLIER, PROCLUS, EWKM, LAC, FSC, and FWKM, were tested on the real world data sets. Since FCM and Fuzzy-FWKM are not projective clustering algorithms, we left them out of this set of experiments. The performances of the algorithms were measured in terms of FScore. Table 5 illustrates the clustering results returned by each algorithm on its 20 clustering processes. The two figures in each cell represent the maximal and average FScore values. The latter is in the format average _ 1 standard deviation. The best clustering results are marked in bold typeface. From the table, we can see that outlier is able to achieve high-quality results on all the data sets. On the two UCI data sets, which have collections, such as cash and sex of Email1431, advertisement and unsolicited for Ling-Spam, and play and hot for Enron- Spam.

A. Proposed method

Clustering the case of non-axis-aligned subspaces and detection of outliers is a major challenge due to the curse of dimensionality. To solve this problem, the proposed implementation is extension to traditional clustering and finds subsets of the dimensions of a data space .In this project, a probability model is proposed to describe in hidden views and the detection of possible selection of relevant views.

In this paper, we first discussed the problem of providing a probability model to describe projected clusters in high dimensional data. The experiments were conducted on synthetic data sets, UCI data sets, and email corpora widely used in real-world applications and the results show the effectiveness of outlier.

There are many directions that are clearly of interest for future exploration. One avenue of further study is to extend outlier to the case of non-axis-aligned subspaces. Another interesting extension would be for the detection of possible outliers in a data set. Our future efforts will also be directed toward developing techniques to build a robust initial condition for the clustering algorithm.

VI. CONCLUSION

In this paper, we first discussed the problem of providing a probability model to describe projected clusters in high dimensional data. This problem becomes difficult due to the sparsity of high-dimensional data and the fact that only a small number of the dimensions may be considered in the clustering process. We proposed an extended Gaussian model which meets the general requirements of projective clustering well. We also derived an objective clustering criterion based on the model, allowing the use of a k-means type paradigm. By mathematical derivations, we obtained computational expressions for calculating the optimal values of the parameters automatically, and proposed a fuzzy clustering algorithm named outlier. The experiments were conducted on synthetic data sets, UCI data sets, and email corpora widely used in real-world applications and the results show the effectiveness of outlier. Outlier to the case of non-axis-aligned subspaces. Another interesting extension would be for the detection of possible outliers in a data set. Our future efforts will also be directed toward developing techniques to build a robust initial

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 9, November 2012

condition for the clustering algorithm. the Ling-Spam data set, the maximal FScore of LAC was large (0.98), but the average FScore was only 0.88 and the standard deviation reached 0.10. In general, outlier is more robust than the other algorithms. This can be explained by the observation of Jain et al. that fuzzy clustering is usually better than hard clustering at avoiding local minima.

The experiments show that outlier is suitable for clustering real-world data, especially for e-mail documents. To confirm the suitability of our algorithm for document clustering, the capability of outlier in identifying the keywords of document categories is analyzed below. From the subspaces of resulting clusters, we can obtain the relevant dimensions that represent important keywords by sorting the dimension weights in descending order. Table 6 gives some examples for the three e-mail data sets used in the experiments. From the table, we can see outlier has identified important keywords that indicate the spam category

REFERENCES

[1] R.K. Agarwal and N.H. Mustafa, "K-Means Projective Clustering," Proc. ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems (PODS), pp. 155-165, 2004

[2] M.J. Zaki and M. Peters, "Clicks: Mining Subspace Clusters in Categorical Data via Kpartite Maximal Cliques," Proc. Int'l Conf. Data Eng. (ICDE), pp. 355-356, 2005.

[3] M. Dutta, A.K. Mahanta, and A.K. Pujari, "QROCK: A Quick Version of the ROCK Algorithm for Clustering of Categorical Data," Pattern Recognition Letters, vol. 26, pp. 2364-2373, 2005.

[4] K. Jain and R. C. Dubes. "*Algorithms for Clustering Data*." Prentice Hall, 1988.

T. Zahn. Graph-theoretic methods for detecting and describing gestalt clusters. *IEEE Transactions on Computing*,20(31):68–86, 1971.

BIOGRAPHY

Parimaladevi.R received the bachelor's degree in Computer Science from Bharathiar University, Coimbatore in 1998 and Master's Degree in Computer Application from Periyar University, Salem in 2007, M.Phil from Vinayaka Mission University, Salem in 2009. His research interests include pattern recognition, data mining, image processing, statistical analysis, and fuzzy logic.

Kavitha.C received the bachelor's degree in Computer Science from Bharathiar University, Coimbatore in 1996 and Master's Degree in Computer Application from Bharathidasan University, Trichy in 1999, M.Phil from MS University, Tirunelveli in 2003 and ME (Computer Science) in 2005. His research interests include pattern recognition, data mining, image processing, statistical analysis, and fuzzy logic.