An efficient method to improve the clustering performance for high dimensional data by Principal Component Analysis and modified K-means

Tajunisha[1] and Saravanan[2]

[1]Department of Computer Science, Sri Ramakrishna College of Arts and Science (W), Coimbatore, Tamilnadu, India
tajkani@gmail.com
[2]Department of Computer Application, Dr. N.G.P. Institute of Technology, Coimbatore, Tamilnadu, India
tvsaran@hotmail.com

ABSTRACT :

Clustering analysis is one of the main analytical methods in data mining. K-means is the most popular and partition based clustering algorithm. But it is computationally expensive and the quality of resulting clusters heavily depends on the selection of initial centroid and the dimension of the data. Several methods have been proposed in the literature for improving performance of the k-means clustering algorithm. Principal Component Analysis (PCA) is an important approach to unsupervised dimensionality reduction technique. This paper proposed a method to make the algorithm more effective and efficient by using PCA and modified k-means. In this paper, we have used Principal Component Analysis as a first phase to find the initial centroid for k-means and for dimension reduction and k-means method is modified by using heuristics approach to reduce the number of distance calculation to assign the data-point to cluster. By comparing the results of original and new approach, it was found that the results obtained are more effective, easy to understand and above all, the time taken to process the data was substantially reduced.

KEYWORDS:

k-means, principal component analysis, dimension reduction

1. INTRODUCTION

Data mining is a convenient way of extracting patterns, which represents knowledge implicitly stored in large data sets and focuses on issues relating to their feasibility, usefulness, effectiveness and scalability. It can be viewed as an essential step in the process of knowledge discovery. Data are normally preprocessed through data cleaning, data integration, data selection, and data transformation and prepared for the mining task. Data mining can be performed on various types of databases and information repositories, but the kind of patterns to be found are specified by various data mining functionalities like class description, association, correlation analysis, classification, prediction, cluster analysis etc.

Clustering is a way that classifies the raw data reasonably and searches the hidden patterns that may exist in datasets. It is a process of grouping data objects into disjoint clusters so that the

DOI: 10.5121/ijdms.2011.3113

data in the same cluster are similar, and data belonging to different cluster are differ. Many algorithms have been developed for clustering. A clustering algorithm typically considers all features of the data in an attempt to learn as much as possible about the objects. However, with high dimensional data, many features are redundant or irrelevant. The redundant features are of no help for clustering; even worse, the irrelevant features may hurt the clustering results by hiding clusters in noises. There are many approaches to address this problem. The simplest approach is dimension reduction techniques including principal component analysis (PCA) and random projection. In these methods, dimension reduction is carried out as a preprocessing step.

K-means is a numerical, unsupervised, non-deterministic, iterative method. It is simple and very fast, so in many practical applications, the method is proved to be a very effective way that can produce good clustering results. The standard k-means algorithm [10, 14] is effective in producing clusters for many practical applications. But the computational complexity of the original k-means algorithm is very high in high dimensional data. Different methods have been proposed [4] by combining PCA with k-means for high dimensional data. But the accuracy of the k-means clusters heavily depending on the random choice of initial centroids.

If the initial partitions are not chosen carefully, the computation will run the chance of converging to a local minimum rather than the global minimum solution. The initialization step is therefore very important. To combat this problem it might be a good idea to run the algorithm several times with different initializations. If the results converge to the same partition then it is likely that a global minimum has been reached. This, however, has the drawback of being very time consuming and computationally expensive.

In this work, initial centers are determined using PCA and k-means method is modified by using heuristic approach to assign the data-point to cluster. This paper deals with the method for improving the accuracy and efficiency by reducing dimension and initialize the cluster for modified k-means using PCA.

2. K-MEANS CLUSTERING ALGORITHM

K-means is a prototype-based, simple partitional clustering technique which attempts to find a user-specified k number of clusters. These clusters are represented by their centroids. A cluster centroid is typically the mean of the points in the cluster. This algorithm is simple to implement and run, relatively fast, easy to adapt, and common in practice. The algorithm consist of two separate phases: the first phase is to select k centers randomly, where the value of k is fixed in advance. The next phase is to assign each data object to the nearest center. Euclidean distance is generally considered to determine the distance between each data object and the cluster centers. When all the data objects are included in some clusters, recalculating the average of the clusters. This iterative process continues repeatedly until the criterion function becomes minimum. The k-means algorithm works as follows:

a) Randomly select k data object from dataset D as initial cluster centers.

b) Repeat

   a. Calculate the distance between each data object di(1<=i<=n) and all k cluster centers cj(1<=j<=n) and assign data object di to the nearest cluster.

   b. For each cluster j(1<=j<=k), recalculate the cluster center.

   c. Until no changing in the center of clusters.

The most widely used convergence criteria for the k-means algorithm is minimizing the SSE.

$$SSE = \sum_{j=1}^{k} \sum_{x_i \in c_j} \left\| x_i - \mu_j \right\|^2 \quad \text{Where } \mu_j = \frac{1}{n_j} \sum_{x_i \in c_j} x_i$$

Denotes the mean of cluster c_j and n_j denotes the no. of instances in c_j.

The k-means algorithm always converges to a local minimum. The particular local minimum found depends on the starting cluster centroids. The k-means algorithm updates cluster centroids till local minimum is found. Before the k-means algorithm converges, distance and centroid calculations are done while loops are executed a number of times, say l, where the positive integer l is known as the number of k-means iterations. The precise value of l varies depending on the initial starting cluster centroids even on the same dataset. So the computational complexity of the algorithm is O(nkl), where n is the total number of objects in the dataset, k is the required number of clusters and l is the number of iterations. The time complexity for the high dimensional data set is O(nmkl) where m is the number of dimensions.

3. PRINCIPAL COMPONENT ANALYSIS

As a preprocessing stage of data mining and machine learning, dimension reduction not only decreases computational complexity, but also significantly improves the accuracy of the learned models from large data sets. PCA [11] is a classical multivariate data analysis method that is useful in linear feature extraction. Without class labels it can compress the most information in the original data space into a few new features, i.e., principal components. Handling high dimensional data using clustering techniques obviously a difficult task in terms of higher number of variables involved. In order to improve the efficiency, the noisy and outlier data may be removed and minimize the execution time and we have to reduce the no. of variables in the original data set

The central idea of PCA is to reduce the dimensionality of the data set consisting of a large number of variables. It is a statistical technique for determining key variables in a high dimensional data set that explain the differences in the observations and can be used to simplify the analysis and visualization of high dimensional data set.

Principal Component

A data set x_i (i= 1,…,n) is summarized as a linear combination of ortho-normal vectors called principal components, which is shown in the Figure 1.

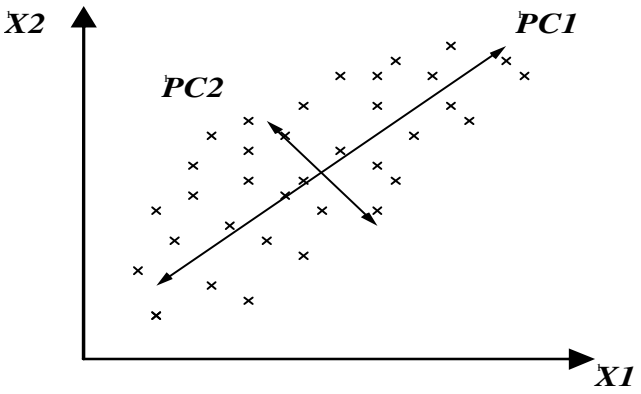

Figure 1. The principal components

The first principal component is an axis in the direction of maximum variance.
The steps involved in PCA are
Step1: Obtain the input matrix Table

Step2: Subtract the mean

Step3: Calculate the covariance matrix

Step4: Calculate the eigenvectors and eigenvalues of the covariance matrix

Step5: Choosing components and forming a feature vector

Step6: deriving the new data set.

The eigenvectors with the highest eigenvalue is the principal component of the data set. In general, once eigenvectors are found from the covariance matrix, the next step is to order them by eigenvalue, highest to lowest. To reduce the dimensions, the first d (no. of principal components) eigenvectors are selected. The final data has only d dimensions.

The main objective of applying PCA on original data before clustering is to obtain accurate results so that the researchers can do analysis in better way. Secondly, minimize the running time of a system because time taken to process the data is a significant one. Normally it takes more time when the number of attributes of a data set is large and sometimes this dataset not supported by all the clustering techniques hence the number of attributes are directly proportional to processing time. In this paper, PCA is used to reduce the dimension of the data. This is achieved by transforming to a new set of variables (Principal Components) which are uncorrelated and, which are ordered so that the first few retain the most of the variant present in all of the original variables. The first Principal Component is selected to find the initial centroid for the clustering process.

4. EXISTING METHODS

There is no commonly accepted or standard “best” way to determine either the no. of clusters or the initial starting point values. The resulting set of clusters, both their number and their centroids, depends on the specified choice of initial starting point values. Two simple approaches to cluster initialization are either to select the initial values randomly or to choose the first k samples of the data points. As an alternative, different sets of initial values are chosen and set, which is closest to optimal, is chosen. However, testing different initial sets are considered

impracticable criteria, especially for large number of clusters. Therefore different methods have been proposed in literature [6].

In [1], Adam said, global k-means method [13] provide a way of determining good initial cluster centers for the k-means algorithm without having to use random initialization. The experiment result has shown clustering performance under global k-means to be as good as or better than using random initialization. But the execution times for the global k-means are much greater than random, due to the need to compute the initial cluster centers.

In 2006, a new way of choosing the initial centers was proposed in [2]. The idea is to select centers in a way that they are already initially close large quantities of points. This seeding method gives out considerable improvements in the final-error of k-means. In 2007, a new method was proposed in [5]. This method was compared with CCIA [11]. In 2009, AIM-K-Means (Automatic initialization of means) has been proposed [17] to overcome the problem of initial mean generation. Fang Yuan et al. [8] proposed a systematic method for finding the initial centroids. The centroids obtained by this method are consistent with the distribution of data. Hence it produced clusters with better accuracy, compared to the original k-means algorithm. However, Yuan's method does not suggest any improvement to the time complexity of the k-means algorithm.

Fahim A. M. et al. [6] proposed an efficient method for assigning data points to clusters. The original k-means algorithm is computationally very expensive because each iteration computes the distances between data points and all the centroids. Fahim's approach makes use of two distance functions for this purpose- one similar to k-means algorithm and another one based on a heuristics to reduce the number of distance calculations. But this method presumes that the initial centroids are determined randomly, as in the case of the original k-means algorithm. Hence there is no guarantee for the accuracy of the final clusters. In 2009, Fahim A M et al. [7] proposed a method to select a good initial solution by partitioning dataset into blocks and applying k-means to each block. But here the time complexity is slightly more. Though the above algorithms can help finding good initial centers for some extent, they are quite complex and some use the k-means algorithm as part of their algorithms, which still need to use the random method for cluster center initialization.

Nazeer et al. (2009) proposed [16] an enhanced k-means, which combines a systematic method for finding initial centroids and an efficient way of assigning data point to cluster. Similarly, Xu et al. (2009) specify a novel initialization scheme [20] to select initial cluster centers based on reverse nearest neighbor search. But all the above methods do not work well for high dimensional data sets. In our previous work [18], the new approach was proposed to find the initial centroid using PCA and we compared the results with existing methods. In [19], we have used this method for iris dataset and we have compared the results with other initialization method. This new method was outperformed with better accuracy and less running time than the existing methods. In this paper, we have applied our proposed method for wine, glass and image segmentation dataset. To improve the efficiency of our method we have used heuristics approach to reduce the number of distance calculation in the standard k-means algorithm

5. PROPOSED METHOD

The proposed method that performs data partitioning with Principal component. It partitions the given data set into k sets. The median of each set can be used as good initial cluster centers and then assign each data points to its nearest cluster centroid. The Proposed model is illustrated in Figure 2.

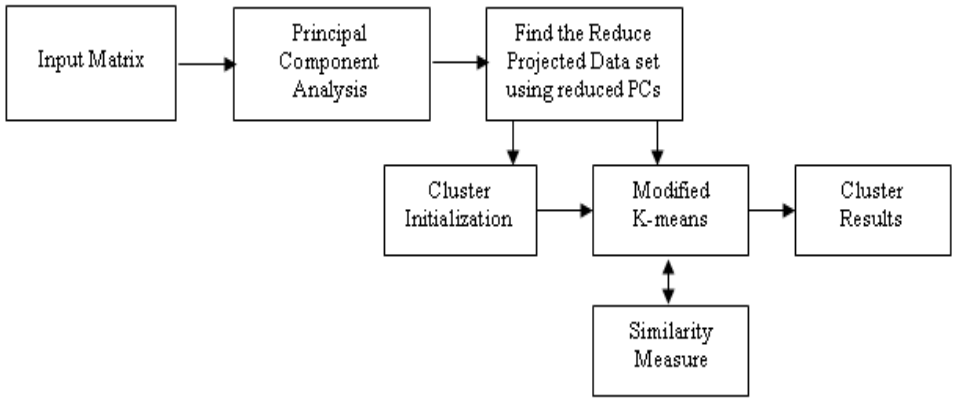

Figure.2 Proposed Model

In this method Following are the steps of the proposed algorithm.

Algorithm 1: The proposed method
Steps:

1. Reduce the dimension of the data into d dimension and determine the initial centroid of the clusters by using Algorithm 2.
2. Assign each data point to the appropriate clusters by using Algorithm 3.

In the above said algorithm the data dimensions are reduced and the initial centroids are determined systematically so as to produce clusters with better accuracy.

Algorithm 2: Dimension reduction and finding the initial centroid using PCA.
Steps:

1. Reduce the D dimension of the N data using Principal Component Analysis (PCA) and prepare another N data with d dimensions (d<D).
2. The Principal components are ordered by the amount of variance.
3. Choose the first principal component as the principal axis for partitioning and sort it in ascending order.
4. Divide the Set into k subsets where k is the number of clusters.
5. Find the median of each subset.
6. Use the corresponding data points for each median to initialize the cluster centers.

The initial centroids of the clusters are given as input to Algorithm 3. It starts by forming the initial clusters based on the relative distance of each data-point from the initial centroids. The Euclidean distance is used for determining the closeness of each data point to the cluster centroids. For each data-point, the cluster to which it is assigned and its distance from the centroid of the nearest cluster are noted. For each cluster, the centroids are recalculated by taking the mean of the values of its data-points. The procedure is almost similar to the original k-means algorithm except that the initial centroids are computed systematically.

The next stage is an iterative process which makes use of a heuristic method to improve the efficiency. During the iteration, the data-points may get redistributed to different clusters. The method involves keeping track of the distance between each data-point and the centroid of its present nearest cluster. At the beginning of the iteration, the distance of each data-point from the

new centroid of its present nearest cluster is determined. If this distance is less than or equal to the previous nearest distance, that is an indication that the data point stays in that cluster itself and there is no need to compute its distance from other centroids. This result in the saving of time required to compute the distances to k-1 cluster centroids. On the other hand, if the new centroid of the present nearest cluster is more distant from the data-point than its previous centroid, there is a chance for the data-point getting included in another nearer cluster. In that case, it is required to determine the distance of the data-point from all the cluster centroids. This method improves the efficiency by reducing the number of computations.

Algorithm 3: Assigning data-points to clusters

Steps:

1. Compute the distance of each data-point x_i (1<=i<=n) to all the centroids c_j(1<=j<=k) using Euclidean distance formula..
2. For each data object xi, find the closest centroid cj and assign x_i to the cluster with nearest centroid c_j and store them in array Cluster[ ] and the Dist[ ] separately.
   Set Cluster[i] = j, j is the label of nearest cluster.
   Set Dist[i]= d(xi, cj), d(xi, cj) is the nearest Euclidean distance to the closest center.
3. For each cluster j (1<=j<=k), recalculate the centroids;
4. Repeat
5. for each data-point
   5.1 Compute its distance from the centroid of the present nearest cluster
   5.2 If this distance is less than or equal to the previous nearest distance, the data-point stays in the cluster
   Else
   For every centroid c_j
   Compute the distance of each data object to all the centre
   Assign the data-point x_i to the cluster with nearest centroid c_j
6. For each cluster j (1<=j<=k), recalculate the centroids;

Until the convergence criteria is met.

This algorithm requires two data structure Cluster [ ] and Dist[ ] to keep the some information in each iteration which is used in the next iteration. Array cluster [ ] is used for keep the label if the closest centre while data structure Dist [ ] stores the Euclidean distance of data object to the closest centre. The information in data structure allows this function to reduce the number of distance calculation required to assign each data object to the nearest cluster, and this method makes the improved k-means algorithm faster than the standard k-means algorithm.

6. EXPERIMENTAL RESULTS

We evaluated the proposed algorithm on the data sets from UCI machine learning repository [9]. We compared clustering results achieved by the k-means, PCA+k-means with random initialization and initial centers derived by the proposed algorithm.

TABLE 1. Dataset Description

Data Set	#Samples	#Dimensions	#Number of clusters
Iris	150	4	3
Wine	178	13	3
Glass	214	9	6
ImgSeg	2310	19	7

The above data sets are used for testing the accuracy and efficiency of the proposed method. The value of k is given in Table 1.

TABLE 2. PRINCIPAL COMPONENT ANALYSIS OF IRIS DATASET

Component	eigenvalue	Accumulation(%)
1	4.2248	92.46
2	0.2422	97.76
3	0.0785	99.48
4	0.0237	100.00

In this work the number of principal components can be decided by a contribution degree about total variance. Table 2 shows the results obtained by a principal component analysis of the Iris data. This shows that three principal components explained about 99.48% of all data. Therefore, there is hardly any loss of information along a dimension reduction.

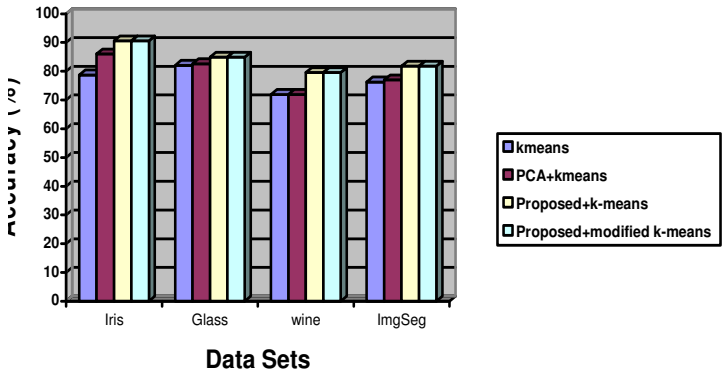

Figure 3. Accuracy on data sets: Iris, Glass, Wine and ImgSeg

Results presented in Figure 3 demonstrate that the proposed method provides better cluster accuracy with k-means and modified k-means than the existing methods. The clustering results of random initial center are the average results over 7 runs since each run gives different results. It shows the proposed algorithm performs much better than the random initialization algorithm. The experimental datasets show the effectiveness of our approach. This may be due to the initial

cluster centers generated by proposed algorithm are quite closed to the optimum solution and it also discover clusters in the low dimensional space to overcome the curse of dimensionality.

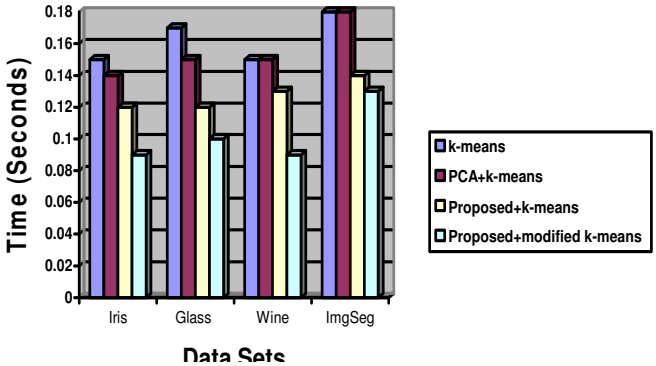

Figure 4. Execution time results on data sets: Iris, Glass, Wine and ImgSeg

In figure 4, we compare the CPU time (seconds) of the proposed method with the existing methods. The execution time of proposed algorithm was much less than the average execution time of k-means when used random initialization. Our proposed method with heuristics approach reduces the running time with the same accuracy. This approach improves the efficiency of our method.

7. CONCLUSION

The main objective of applying PCA on original data before clustering is to obtain accurate results. But the clustering results depend on the initialization of centroid. In this paper, we have proposed a new approach to initialize the centroid and reducing the dimension using principal component analysis to improve the accuracy of the cluster results and the standard k-means algorithm also modified to improve the efficiency by reducing the computation complexity of the algorithm. The experiment results show that the substantial improvement in running time and accuracy of the clustering results by reducing the dimension and initial centroid selection using PCA. Though the proposed method gave better quality results in all cases, over random initialization methods, still there is a limitation associated with this, i.e. the number of clusters (k) is required to be given as input. Evolving some statistical methods to compute the value of k, depending on the data distribution is suggested for future research. In the future, we plan to apply this method to microarray cancer datasets.

REFERENCES

[1] Adam schenker, mark last, horst bunke, Abraham kandel (2003): Comparison of two noval algorithm for clustering using web documents, WDA.

[2] Arthur D., vassilvitskii S.(2007): K-means++ the advantages of careful seeding, on discrete algorithms (SODA).

[3] Babu G. and Murty M. (1993): A near Optimal initial seed value selection in k-means algorithm using a genetic algorithm, Pattern Recognition Letters Vol.14,1993, PP, 763-769.

[4] Chris Ding and Xiaofeng He (2004) : k-means Clustering via Principal component Analysis, In Proceedings of the 21st international conference on Machine Learning, Banff, Canada.

[5] Deelers S. S. and Auwatanamongkol S. (2007): Enhancing K-Means Algorithm with initial cluster centers Derived from Data partitioning along the Data axis with the highest Variance, Proceedings of world Academy of Science, Engineering and Technology Volume 26, ISSN 1307-6884.

[6] Fahim A.M,Salem A.M, Torkey A and Ramadan M.A (2006) : An Efficient enchanced k-means clustering algorithm,Journal of Zhejiang University,10(7): 1626-1633,2006.

[7] Fahim A.M,Salem A.M, Torkey F. A., Saake G and Ramadan M.A (2009): An Efficient k-means with good initial starting points, Georgian Electronic Scientific Journal: Computer Science and Telecommunications, Vol.2, No. 19,pp. 47-57.

[8] Huang Z (1998): Extensions to the k-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery,(2):283-304.

[9] Ismail M. and Kamal M. (1989): Multidimensional data clustering utilization hybrid search strategies,Pattern Recognition Vol. 22(1),PP. 75-89.

[10] Jiawei Han M.K (2006): Data mining Concepts and Techniques, morgan Kaufmann publishers, An imprint of Elsevier.

[11] Jolliffe I.T. (2002): Principal Component Analysis, Springer, Second edition.

[12] Khan S.S,and Ahmad A., "Cluster Center Initialization for K-Means Clustering", pattern recognition Letter Volume 25, issue 11, 2004, PP 1293-1302.

[13] Likas A., Vlassis N. and Verbeek J.J (2003): The Global k-means Clustering algorithm, Pattern Recognition, Volume 36, Issue 2,pp. 451-461.

[14] Margaret H.Dunham (2006): Data Mining-Introductory and Advanced Concepts, Pearson Education.

[15] Merz C and Murphy P, UCI Repository of Machine Learning Databases, Available: ftp://ftp.ics.uci.edu/pub/machine-Learning-databases

[16] Nazeer K. A., Abdul and Sebastian M.P. (2009): Improving the accuracy and efficiency of the k-means clustering algorithm, Proceedings of the World Congress on Engineering,Vol. 1, pp. 308-312.

[17] Samarjeet Borah, Mrinal Kanti Ghose, "Performance Analysis of AIM-K-Means and K-means in Quality cluster generation," Journal of computing, Volume1, issue 1, December 2009.

[18] Tajunisha N., Saravanan V.,"An increased performance of clustering high dimensional data using Principal Component Analysis",Proceedings of the IEEE fist international conference on integrated intelligent computing pp 17-21,(2010).

[19] Tajunisha N., Saravanan V.,"An increased performance of clustering high dimensional data using Principal Component Analysis",Proceedings of the IEEE fist international conference on integrated intelligent computing pp 17-21,(2010).

[20] Xu Junling , Xu Baowen, Zhang Weifeng, Zhang Wei and Hou Jun (2009): Stable initialization scheme for k-means clustering, Wuhan University Journal of Natonal Sciences, Vol. 14, No. 1, pp. 24-28.