ISSN:0975-9646

Discovery of Preliminary Centroids Using Improved K- Means Clustering Algorithm

N. Koteswara Rao, G. Sridhar Reddy

Department of Computer Science,
Vignan university, Vadlamudi, 522213
Guntur, India.

***Abstract*—The emergence of modern technology has enforced to collect the scientific data in a large quantity and those data are getting amassed in different databases. An organized analysis of data is very essential to obtain useful information from swiftly growing data repositories. Cluster analysis is one of the major data mining methods and the k-means clustering algorithm is widely used for many practical applications. But the original k-means algorithm is computationally expensive and the quality of the resulting clusters substantially relies on the choice of initial centroids. Fast and high quality clustering is one of the most important tasks in the modern era of information processing wherein people rely heavily on search engines. With the huge amount of available data and with an aim to creating better quality clusters, scores of algorithms having quality-complexity trade-offs have been proposed. However, the k-means algorithm proposed during late 1970's still enjoys a respectable position in the list of clustering algorithms. It is considered to be one of the most fundamental algorithms of data mining. It is basically an iterative algorithm. In each iteration, it requires finding the distance between each data object and centroid of each cluster. Considering the hugeness of modern databases, this task in itself snowballs into a tedious task. This paper proposes an improvement on the classic k-means algorithm to produce more accurate clusters. The proposed algorithm comprises of a *O(n logn)* heuristic method, based on sorting and partitioning the input data, for finding the initial centroids in accordance with the data distribution. Experimental results show that the proposed algorithm produces better clusters in less computation time.**

Keywords: **Clustering-means, time complexity, centroid, data sets.**

I. INTRODUCTION

As a result of modern methods for scientific data collection, huge quantities of data are getting accumulated at various databases. Such data banks are growing so rapidly that it is practically difficult to extract useful information from them by using conventional database techniques. Effective and efficient algorithms for data mining are necessary to unravel implicit information from huge databases.

Cluster analysis [1] is one of the major data analysis methods which helps to identify the natural grouping in a set of data items. Clustering is the process of partitioning a given set of objects into disjoint clusters. This is done in such a way that objects in the same cluster are similar while objects belonging to different clusters differ considerably, with respect to their attributes.

The k-means algorithm [1, 4, 5, 6, 7] is effective in producing clusters for many practical applications in emerging areas like Bioinformatics [2, 3]. But the computational complexity of the original k-means algorithm is very high. Moreover, this algorithm results in different types of clusters depending on the random choice of initial centroids. This paper deals with a heuristic method based on sorting and partitioning the input data for finding better initial centroids, thereby improving the accuracy of the k-means algorithm.

II. THE K-MEANS CLUSTERING ALGORITHM

The k-means clustering algorithm consists of two separate phases: the first phase is to define *k* centroids, one for each cluster. The next phase is to take each point belonging to the given data set and associate it to the nearest centroid. When all the points are included in some clusters, the first phase is completed and an early grouping is done. At this point we need to recalculate the new centroids, as the inclusion of new points may lead to a change in the cluster centroids. Once we find *k* new centroids, a new binding is to be created between the same data points and the nearest new centroid, generating a loop. As a result of this loop, the *k* centroids may change their position in a step by step manner. Eventually, a situation will be reached where the centroids do not move anymore. This signals the convergence of clustering. Pseudo code for the k-means clustering algorithm is listed as Algorithm 1 [4].

Algorithm 1: The k-means clustering algorithm

Input:

$D = \{d_1, d_2, ..., d_n\}$ //set of *n* data items.
k // Number of desired clusters

Output:

A set of *k* clusters.

Steps:

1. Arbitrarily choose *k* data-items from D as initial centroids;
2. **Repeat**
   - 2.1 Assign each data item *di* to the cluster which has the closest centroid;
   - 2.2 Calculate the new mean of each cluster;

Until convergence criterion is met.

III. RELATED WORK

Several attempts were made by researchers to improve the accuracy and efficiency of the k-means algorithm [8, 9, 10]. A variant of the k-means algorithm is the k-modes [9, 11] method which replaces the means of clusters with modes. Like the k-means method, the k-modes algorithm also produces locally optimal solutions which are dependent on the selection of the initial modes. The k-prototypes algorithm [9] integrates the k -means and k-modes processes for clustering the data. In this method, the dissimilarity measure is defined by taking into account both numeric and categorical attributes.

Fang Yuan et al. [10] 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 efficiency of the k-means algorithm.

Fahim A M et al. [8] 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 the 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.

Abdul Nazeer and Sebastian proposed an algorithm [12] comprising of separate methods for accomplishing the two phases of clustering. Though this algorithm produced clusters with better accuracy and efficiency compared to k-means, it takes $O(n^2)$ time for finding the initial centroids.

IV. PROPOSED ALGORITHM

In the improved algorithm discussed in this paper, a novel heuristic method is proposed to determine the initial centroids of the clusters. The basic idea of this algorithm is to determine the initial centroids of the clusters in a heuristic manner, so as to ensure that the centroids are chosen in accordance with the distribution of data. The method involves sorting the input data set and partition the sorted data set into 'k' number of sets where 'k' is the number of clusters to be formed. Mean values of each of these sets are taken as the initial centroids.

More often, we may have to deal with multi-dimensional data values. Each data point *di* may contain multiple attributes such as *di1, di2,…..dim*, where *m* is the number of attributes or columns in each data value. In such cases we first determine the column with maximum range [6], where range is the difference between the maximum and the minimum element in the column. Initially, from the multi-dimensional data values, we determine the maximum and minimum element of each column and compute the range of values for each column as the difference between the maximum and minimum values. Then we identify the attribute (column) having maximum range. The entire set of data values are then sorted in non-decreasing order, using the Heap Sort algorithm [13], based on the attribute with maximum range. The sorted list of data points are then divided into 'k' equal sets. Finally, the arithmetic means of each of these 'k' sets are computed. These means become the initial centroids of the clusters to be formed.

After determining the initial centroids as described above, the data points are assigned to various clusters by using the same method used in the second phase of the original k-means algorithm. Each data point *di* is assigned to the cluster having the closest centroid. Euclidean distance is used as the measure for determining the distance between the data points and the centroids. The proposed algorithm is outlined below as Algorithm 2.

Algorithm 2: Proposed Algorithm for Clustering

Input:
D = {d1, d2,.....,dn} // set of *n* data items.
k // Number of desired clusters.

Output:
A set of *k* clusters.

Steps:
1. For each column of the data set, determine the *range* as the difference between the maximum and the minimum element;
2. Identify the column having the maximum *range*;
3. Sort the entire data set in non-decreasing order based on the column having the maximum *range*;
4. Partition the sorted data set into 'k' equal parts;
5. Determine the arithmetic mean of each part obtained in Step 4 as *c1, c2,….ck*; Take these mean values as the initial centroids.
6. **Repeat**
   - 6.2 Assign each data item *di* to the cluster which has the closest centroid;
   - 6.3 Calculate new mean of each cluster;

Until convergence criterion is met.

Unlike the original k-means algorithm in which the initial centroids are selected randomly, the proposed algorithm determines the initial centroids in a more meaningful way, in accordance with the distribution of data. Consequently, the algorithm converges much faster than the original k-means algorithm. Moreover, since the method for determining the initial centroids is based on the technique of sorting, this phase requires less time compared to other similar approaches available in the literature [10, 12].

V. TIME COMPLEXITY ANALYSIS

In the original k-means algorithm, the initial centroids are selected quite randomly. As a result, the centroids are re-calculated many times before the algorithm converges and the data points are assigned to their nearest centroids. Since complete redistribution of data points takes place according to the new centroids, this procedure takes time *O(nkl)* where *n* is the number of data-points, *k* is the number of clusters and *l* is the number of iterations.

For the Enhanced algorithms discussed in [10] and [12], the

first phase of determining the initial centroids takes $O(n^2)$ time even though it produces better results compared to the original k-means algorithm.

In the proposed algorithm discussed in this paper, the step to find the maximum and minimum values of each column of the data set requires $O(n)$ time where n is the number of data items. The time required to find the maximum and minimum values of all the columns of the data set is $O(nm)$ where m is the number of attributes in each data item. The range of each column (difference between maximum and minimum values) can be determined in constant time and the time required to find the column with maximum range is $O(m)$ where m is the number of attributes in the data set. The next step of sorting the data items based on the column with the maximum range can be performed in $O(n \log n)$ time using Heap Sort. Time complexity for partitioning the n data items into k equal parts and finding the mean of each part is $O(n)$. Thus the overall time complexity for finding the initial centroids of a data set containing n elements is $O(n \log n)$, as m is much less than n.

The second phase of assigning data points to clusters is the same as that of the original k-means algorithm. The loop consisting of the assignment of data-points to the nearest clusters and the subsequent recalculation of centroids is executed repetitively until the convergence criterion is reached. This procedure takes time $O(nkl)$ where n is the number of data-points, k is the number of clusters and l is the number of iterations. Nevertheless, the algorithm converges in much less number of iterations as the initial centroids are computed in a strategic manner in tune with the data distribution. Thus the overall time complexity of the proposed algorithm is the maximum of O(n logn) and O(nkl), i.e. O(n (kl + logn)).

VI. EXPERIMENTAL RESULTS

For testing the accuracy and efficiency of the proposed algorithm, multivariate data sets with known clustering available at the UCI repository of machine learning databases [14] were used. The data sets used are E- Coli [15], Breast Cancer-Wisconsin [16] and Thyroid. The same sets of data are given as input to the standard k-means algorithm, the enhanced algorithm [8] and the proposed algorithm.

The original k-means and the enhanced k -means algorithms require the values of the initial centroids also as input, apart from the input data values and the value of k. The experiment is conducted for different sets of values of the initial centroids, which are selected randomly. For the proposed algorithm, the data values and the value of k are the only inputs required. The accuracy of clustering is determined by comparing the clusters obtained by the experiments with the pre- determined clusters already available in the UCI data set. The percentage accuracy and the time taken for each experiment are computed and tabulated. Performances of the algorithms for the different data sets are tabulated in Table I. Comparisons of the results obtained from the different algorithms are shown in Figures 1 to 3.

TABLE1. PERFORMANCE COMPARISON OF THE ALGORITHMS FOR DIFFERENT DATA SETS

Data sets	Algorithms					
	K-Means		Enhanced K-Means		Proposed Algorithm	
	Accuracy(%)	Time Taken (ms)	Accuracy(%)	Time Taken (ms)	Accuracy(%)	Time Taken (ms)
E-Coli	79.7	64	81.5	48	81.5	40
Breast cancer	96	68	96.2	56	96.2	42
Thyroid	75	60	82.3	56	86	52

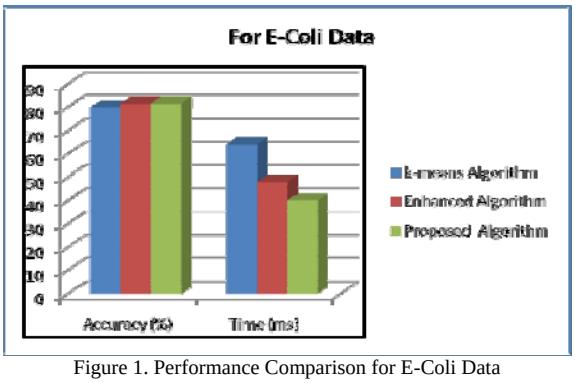

Figure 1. Performance Comparison for E-Coli Data

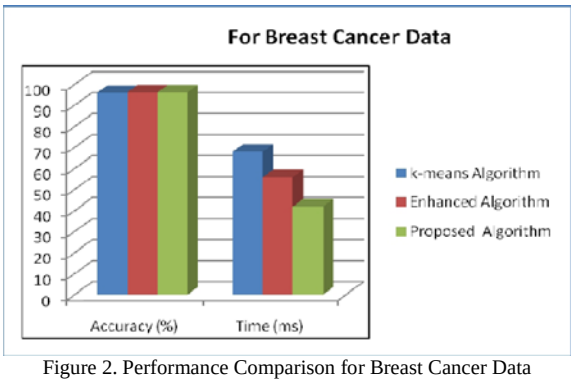

Figure 2. Performance Comparison for Breast Cancer Data

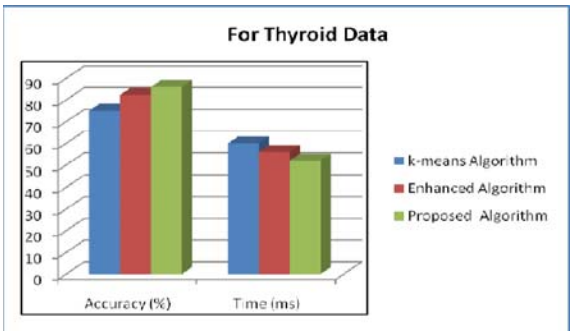

Figure 3. Performance Comparison for Thyroid Data

VII. CONCLUSION

The k-means algorithm is widely used for clustering large sets of data. But the standard algorithm do not always guarantee good results as the accuracy of the final clusters depend on the selection of initial centroids. Moreover, the computational complexity of the standard k-means algorithm is objectionably high owing to the need to reassign the data points a number of times, during every iteration of the loop. This paper presents an improved k-means algorithm using a *O(n logn)* heuristic method for finding the initial centroids. This method ensures that the initial centroids generated are in accordance with the distribution of the data. This results in clusters with better accuracy compared to the original k-means algorithm. Experimental results have shown that the proposed algorithm produces better clusters in less computation time compared to the original k-means algorithm.

A limitation of the proposed algorithm is that the value of *k*, the number of desired clusters, is still required to be given as an input. Evolving some statistical methods to compute the value of *k*, depending on the data distribution, is suggested for future research. More efficient methods for

Assigning data points to various clusters are also worth investigating.

REFERENCES

[1] Jiawei Han M. K, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, An Imprint of Elsevier, 2006.

[2] Amir Ben-Dor, Ron Shamir and Zohar Yakini, "Clustering Gene Expression Patterns," Journal of Computational Biology, 6(3/4): 281-297, 1999

[3] Daxin Jiang, Chum Tong and Aidong Zhang, "Cluster Analysis for Gene Expression Data," IEEE Transactions on Data and Knowledge Engineering, 16(11): 1370-1386, 2004.

[4] Margaret H. Dunham, Data Mining- Introductory and Advanced Concepts, Pearson Education, 2006.

[5] McQueen J, "Some methods for classification and analysis of multivariate observations," Proc. 5th Berkeley Symp. Math. Statist. Prob., (1):281–297, 1967.

[6] Pang-Ning Tan, Michael Steinback and Vipin Kumar, Introduction to Data Mining, Pearson Education, 2007.

[7] Stuart P. Lloyd, "Least squares quantization in pcm," IEEE Transactions on Information Theory, 28(2): 129-136.

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

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

[10] Yuan F, Meng Z. H, Zhang H. X and Dong C. R, "A New Algorithm to Get the Initial Centroids," Proc. of the 3rd International Conference on Machine Learning and Cybernetics, pages 26–29, August 2004.

[11] Chaturvedi J. C. A, Green P, "K-modes clustering," J. Classification, (18):35–55, 2001.

[12] Abdul Nazeer K A, Sebastian M P, "Improving the Accuracy and Efficiency of the k-means Clustering Algorithm," Proceedings of the International Conference on Data Mining and Knowledge Engineering, London, UK, 2009.

[13] T H Cormen, C E Leiserson, R L Rivest and C Stein, Introduction to Algorithms, Second Edition, MIT Press, 2001.

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

[15] Paul Horton and Kenta Nakai, "A Probablistic Classification System for Predicting the Cellular Localization Sites of Proteins," Intelligent Systems in Molecular Biology, 109-115.St. Louis, USA, 1996.

[16] O. L. Mangasarian and W. H. Wolberg, "Cancer diagnosis via linear programming," *SIAM News*, Volume 23, Number 5, pp 1 & 18, 1990.