Examination And Study of K-Means Clustering Algorithm

Nagarajan. S[1], A. Srinivasan[2]
[1, 2] Dept of Computer Science and Engineering
[1, 2] Viswam Engineering College Madanapalle

Abstract- *Investigation of this paper depicts the conduct of K-implies calculation. Through this paper we have attempt to conquer the impediments of K-implies calculation by proposed calculation. Fundamentally genuine K-mean calculation takes parcel of time when it is applied on a huge data set. That is the reason the proposed bunching idea comes into picture to give fast and productive grouping method on huge informational index. In this paper execution assessment is finished proposed calculation utilizing Max Hospital Diabetic Patient Dataset.*

Keywords- Clustering, K-means, Threshold, outlier, Square Error.

I. INTRODUCTION

Bunching is the method involved with dividing or gathering a given arrangement of examples into disjoint groups. This is done to such an extent that examples in a similar group are indistinguishable and designs having a place with two unique bunches are unique. Grouping has been a broadly concentrated on issue in an assortment of utilization areas. A few calculations have been proposed in the writing for bunching: CLARA, CLARANS, Focusing Techniques, P-CLUSTER, DBSCAN and BIRCH. The k-implies technique has been demonstrated to be compelling in creating great grouping results for some useful applications. Nonetheless, an immediate calculation of k-implies technique requires time corresponding to the result of number of examples and number of groups per emphasis. This is computationally pricey particularly for enormous datasets. We propose a clever calculation for executing the k-implies strategy. Our calculation delivers something very similar or tantamount (because of the adjust mistakes) grouping results to the immediate k-implies calculation. It has essentially prevalent execution than the immediate k-implies calculation by and large.

K-MEANS CLUSTERING

K-means algorithm is one of the partitioning based clustering algorithms. The general objective is to obtain the fixed number of partitions/clusters that minimize the sum of squared Euclidean distances between objects and cluster centroids.

Let

$$X = \{x | i = 1,2, \dots \dots, n\}$$

be a data set with n objects, k is the number of clusters, m_j is the centroid of cluster cj where *j=1,2,………,k.* Then the algorithm finds the distance between a data object and a centroid by using the following Euclidean distance formula .

The Euclidean distance between two points/objects/items in a dataset, defined by point X and point Y is defined by Equation below .

$$(|X_1 - Y_1|^2 + |X_2 - Y_2|^2 + \dots . . + |X_{N-1} - Y_{N-1}|^2 + |X_N - Y_N|^2)1/2$$

OR Euclidean distance formula=$\sqrt{\sum}|x_i\text{-}m_j|^2$ where **X** represents is the first data point, **Y** is the second data point, **N** is the number of characteristics or attributes in data mining terminology.

Starting from an initial distribution of cluster centers in data space, each object is assigned to the cluster with closest center, after which each center itself is updated as the center of mass of all objects belonging to that particular cluster. The procedure is repeated until convergence.

K-MEANS ALGORITHM

INPUT: //Set of n items to cluster
D= {d1, d2, d3,… dn}
// No. of cluster (temporary cluster) randomly chosen i.e. k
// So below, K is set of subset of D as temporary cluster and C is set of centroids of those clusters.

K= {k1, k2, k3,… kk},

C= {c1, c2, c3,…ck}
Where k1= {d1}, k2= {d2}, k3 = {d3}…kk={dk}
And c1=d1, c2=d2, c3=d3…ck=dk,

// here k<=n

Output: // K is set of subset of D as final cluster and C is set of centroids of these cluster.
K= {k1, k2, k3… kk},
C= {c1, c2, c3,… ck}

Algorithm:

```
1. K-means (D, K, C)
2. Arbitrarily choose k objects from D as the initial cluster centers.
3. Repeat
4. (re) assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster.
5. Update the cluster means, i.e., calculate the mean value of the objects for each cluster.
6. Until no change.
```

II. LIMITATTIONS OF K-MEANS CLUSTERING ALGORITHM

A critical look at the available literature indicates the following shortcomings are in the existing K-means clustering algorithms .

1. In partitioning based K-means clustering algorithms, the number of clusters (k) needs to be determined beforehand.
2. The algorithm is sensitive to an initial seed selection (starting cluster centroids). Due to selection of initial centroid points it is susceptible to a local optimum and may miss the global optimum. It may converge to suboptimal solutions. This means suboptimal classification may be found, requiring multiple runs with different initial conditions. The selection of spurious data points as a center may lead to no data points in the class, with the outcome that the center cannot be updated.
3. It can model only a spherical shape of clusters. Thus the non convex shape of clusters cannot be modeled in center based clustering.
4. It is sensitive to outliers since a small amount of outliers can substantially influence the mean value.
5. Due to the nature of iteration scheme in producing the clustering result, it begins at starting cluster centroids and iteratively updates these centroids to decrease the square error. But it is not confirmed how many time it iterates which is not relevant for bigger data set. It may take a huge number of iterations to converge. Such number of iterations cannot be determined beforehand and may change from run to run. Result may be bad with high dimensional data.

III. PROPOSED CLUSTERING ALGORITHM

Input: // A set D of n objects to cluster. A threshold value Tth.
D= {d1, d2, d3, , dn}, Tth

Output:// A set K of k subsets of D as final clusters and a set C of centroids of these clusters.
K= {k1, k2, k3,… kk},
C= {c1,c2,c3,… ck}

Algorithm:

```
Proposed cluster algorithm (D,Tth)
            1. Let k=1
// Randomly choose a object from D,
            2. let it be p  k1= {p}
            3. If (distance<=Tth) then
            4. km=km union q
            5. Calculate new mean (centroid cm) for cluster km using eq. (2).
            6. Else k=k+1
            7. kk={q}
            8. K=K union {kk}
            9. ck=q
            10. C=C union {ck}
```

IV. ADVANTAGES OF PROPOSED CLUSTERING

Having looked at the available literature indicates the following advantages can be found in proposed clustering over K-means clustering algorithm.

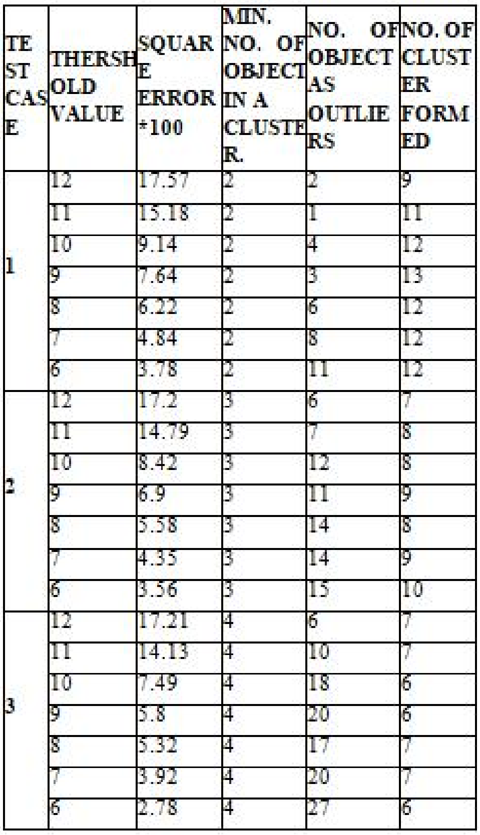
1. In K-means clustering algorithms, the number of clusters (k) needs to be determined beforehand but in proposed clustering algorithm it is not required. It generates number of clusters automatically.
2. K-means depends upon initial selection of cluster points, it is susceptible to a local optimum and may miss global optimum. Proposed clustering algorithm is employed to improve the chances of finding the global optimum.
3. K-means is sensitive to outliers since a small amount of outliers can substantially influence the mean value. In proposed clustering algorithm outliers can't influence the mean value. They can be easily identified and removed (if desired).
4. In K-means it is not confirmed that how many times it iterates but in proposed clustering it is known.
5. Data are stored in secondary memory and data objects are transferred to main memory one at a time for clustering. Only the cluster representations i.e. centroid are stored permanently in main memory to

alleviate space limitations thus space requirements of proposed algorithm is very small, necessary only for the centroids of clusters. In K- means memory space is more required to store each object permanently in memory along with centroids.

V. EXPERIMENTAL RESULT

The implementation of proposed algorithm is using Dot Net Visual Studio 2008 using language C# and backend Microsoft SQL Server 2008. We have evaluated our algorithm on Max hospital data set of diabetic patients. All the experimental results reported are on Intel Core i3 whose clock speed of processor is 3.0GHz and the memory size is 4 GB running on window7 home basic.

Table 2: Experimental Result obtained by Proposed Algorithm



TEST CASE	THERSHOLD VALUE	SQUARE ERROR *100	MIN. NO. OF OBJECT IN A CLUSTER.	NO. OF OBJECT AS OUTLIERS	NO. OF CLUSTER FORMED
1	12	17.57	2	2	9
	11	15.18	2	1	11
	10	9.14	2	4	12
	9	7.64	2	3	13
	8	6.22	2	6	12
	7	4.84	2	8	12
	6	3.78	2	11	12
2	12	17.2	3	6	7
	11	14.79	3	7	8
	10	8.42	3	12	8
	9	6.9	3	11	9
	8	5.58	3	14	8
	7	4.35	3	14	9
	6	3.56	3	15	10
3	12	17.21	4	6	7
	11	14.13	4	10	7
	10	7.49	4	18	6
	9	5.8	4	20	6
	8	5.32	4	17	7
	7	3.92	4	20	7
	6	2.78	4	27	6

Above table shows three test cases having minimum number of object in a cluster as 2,3 and 4, threshold value varies from 6 to 12 for each test case. On different –different threshold value we have obtained different values of square error, number of object as Outlier and number of cluster form.

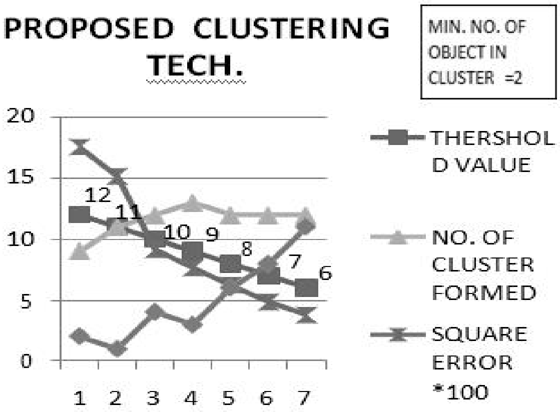


Figure 1: Graph representing test case1

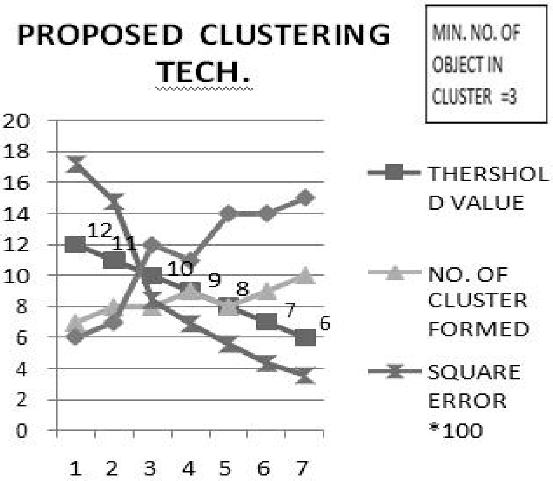


Figure 2: Graph representing test case2.

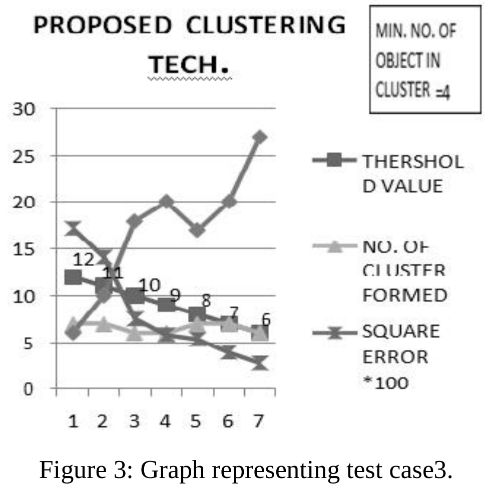


Figure 3: Graph representing test case3.

Above graph shows that

1. As threshold value decreases Square Error decreases. Lower the value of Square Error higher the compactness

of cluster and as separate as possible. Hence as we decrease the threshold value cluster quality increases.

2. As we decreases the threshold value number of cluster form increases.
3. As we decrease the threshold value number of object as Outlier increases.

VI. CONCLUSION

In this paper presented an algorithm for performing K-means clustering. Our experimental result demonstrated that our scheme can improve the direct K-means algorithm. This paper also explains the time complexity of K-means and our purposed algorithm. There are several improvements possible to the basic strategy presented in this paper. One approach will be to use the concept of Nearest Neighbor Clustering Algorithm to improve the compactness of clusters.

REFERENCES

[1] Han, J. &Kamber, M. (2012). Data Mining: Concepts and Techniques. 3rd.ed. Boston: Morgan Kaufmann Publishers.

[2] Sudhir Singh, Dr. Nasib Singh Gill,Comparative Study Of Different Data Mining Techniques : A Review, www. ijltemas.in, Volume II, Issue IV, APRIL 2013 IJLTEMAS ISSN 2278 – 2540.

[3] M. Ester, H. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proc. of the 2nd Int’l Conf. on Knowledge Discovery and Data Mining, August 1996**.**

[4] M. Ester, H. Kriegel, and X. Xu. Knowledge Discovery in Large Spatial Databases: Focusing Techniques for Efficient Class Identification. Proc. of the Fourth Int’l. Symposium on Large Spatial Databases, 1995.

[5] D. Judd, P. McKinley, and A. Jain. Large-Scale Parallel Data Clustering. Proc. Int’l Conference on Pattern Recognition, August 1996

[6] R. T. Ng and J. Han. Efficient and Effective Clustering Methods for Spatial Data Mining. Proc. of the 20th Int’l Conference on Very Large Databases, Santiago, Chile, pages 144–155, 1994

[7] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases Proc. of the 1996 ACM SIGMOD Int’l Conf. on Management of Data, Montreal, Canada, pages 103–114, June 1996

[8] Performance Evaluation of Incremental K-means Clustering Algorithm, Sanjay Chakraborty , N.K. Nagwani National Institute of Technology (NIT) Raipur, CG, India, IIJDWM, Journal homepage: www.ifrsa.org.

[9] PERFORMANCE ANALYSIS OF PARTITIONAL AND INCREMENTAL CLUSTERING, Seminar National Aplikasi Teknologi Informasi 2005 (SNATI 2005) ISBN: 979-756-061-6 Yogyakarta, 18 June 2005.

[10] Performance Evaluation of Incremental K-means Clustering Algorithm, IFRSA International Journal of Data Warehousing & Mining |Vol1|issue 1|Aug 2011.