ENHANCING THE PERFORMANCE OF NEAREST NEIGHBOR CLUSTRING ALGORITHM

Sudhir Singh
Deptt of Computer Science & Applications Maharshi Dayanand University, Rohtak, India

Nasib Singh Gill
Deptt of Computer Science & Applications Maharshi Dayanand University, Rohtak, India

ABSTRACT
The Nearest Neighbor Clustering is the common model in data mining. However, some models are better than the others due to the types of data, time complexity, and space requirement. This paper describes the performance of Nearest Neighbor Clustering and Proposed Clustering Algorithm based on the time and space complexity. Experimental studies show that Proposed Clustering Algorithm outperformed as compare to Nearest Neighbor Clustering. This paper also defines the chain effect in Nearest Neighbor Clustering Algorithm and best threshold value for dataset. For this for we have used Max Hospital Diabetic Patient Dataset.

Keywords: Clustering, Nearest Neighbor, Threshold, Outlier, Square Error, Chain effect.

1. INTRODUCTION
Clustering groups data instances into subsets in such a manner that similar instances are grouped together, while different instances belong to different groups [1] [6]. The instances are thereby organized into an efficient representation that characterizes the population being sampled. Formally, the clustering structure is represented as a set of subsets C = C_1, C2,…….C_k of S, such that:

$$S = \bigcup_{i=1}^{k} C_i$$

and $C_i \cap C_j = \emptyset$ for $i \neq j$. Consequently, any instance in S belongs to exactly one and only one subset.
The Nearest Neighbor Clustering (Single Link Algorithm) method has been shown to be effective in producing good clustering results for many practical applications but computationally very expensive especially for large datasets[2][3]. We propose a novel algorithm for implementing the Nearest Neighbor Clustering method. Our algorithm produces the better clustering results to the Nearest Neighbor Clustering algorithm on the basis of Square Error.
The rest of this paper is organized as follows. We review previously proposed Nearest Neighbor Clustering algorithm in Section 2. We present our algorithm in Section 4, time complexity of algorithms in Section 5, we describe the experimental results in Section 6 and we conclude with Section 7.

2. NEAREST NEIGHBOR CLUSTERING ALGORITHM

Input: // Set of n items to cluster
D= { d_1,d_2,d_3,……,d_n }
// Adjacency Matrix of the order of n × n
A[n][n]= distance between each pair of data items.
Output: // K is set of subset of D as final clusters
K= { k_1,k_2,k_3,……,k_p }
Algorithm:
Incremental k-means (D, A)

1. Let k=1
2. k_k={d_k}
3. K={k_k}
4. Assign some constant value to T_{th}
5. For i=2 to n do
6. Determine d_m in some cluster k_i in K such that distance (M) between d_m and d_i is minimum.(1<=j<=k)
7. If (M<=T_{th}) then //T_{th} – threshold limit for max. distance allowed.
8. k_j=k_jU d_i
9. Else k=k+1
10. k_k=d_i
11. K=K U k_k

2.1. LIMITATTIONS OF NEAREST NEIGHBOR CLUSTERING ALGORITHM

A critical look at the available literature indicates the following shortcomings are in the existing Nearest Neighbor Clustering Algorithm [4] [2].

1. In Nearest Neighbor Clustering, adjacency matrix stores the distance between each data object. Preprocessing cost of making an adjacency matrix of order n*n requires time and space.
2. Clusters are spherical.
3. It takes more time to cluster than proposed work.
4. Nearest Neighbor Clustering Algorithm is sensitive to “Chain Effect” or ‘Chaining’.

3. CHAIN EFFECT

The minimum distance between two observations a ∈ A and b ∈ *B* defines what is known as “Nearest Neighbor Technique” or “single linkage”. Weak point of Nearest Neighbor Technique is that it is sensitive to the ‘chain effect’ (or chaining) [4]. If two widely separated clusters are linked by a chain of individuals who are close to each other, they may be grouped together.

Let us consider a hypothetical data set which is chain from object point (1, 8) to point (12, 6). As shown in Fig 1.

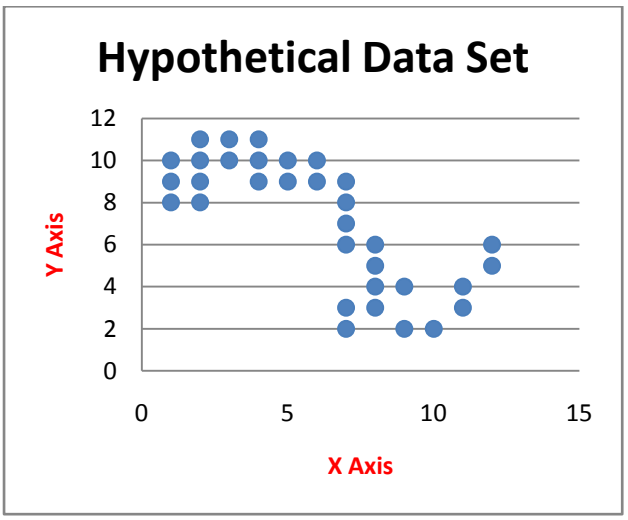

Figure1. Hypothetical Data Set

When we apply Nearest Neighbor Clustering Algorithm for creating cluster then whole the objects/elements grouped into single cluster showing chain effect.

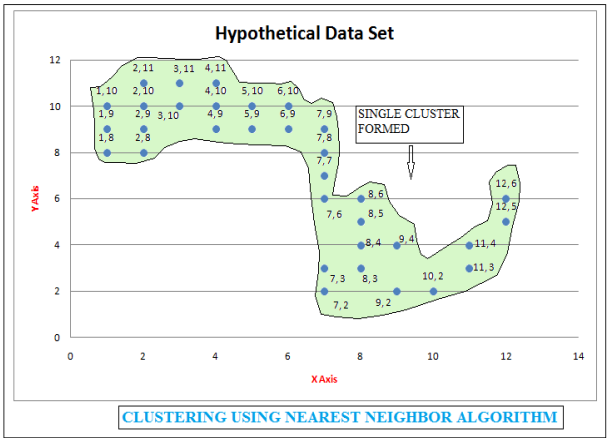

Figure2. Hypothetical data set showing chain effect using Nearest Neighbor Clustering

When we apply Proposed Algorithm then data objects are divided into seven clusters as shown below in Figure 3.

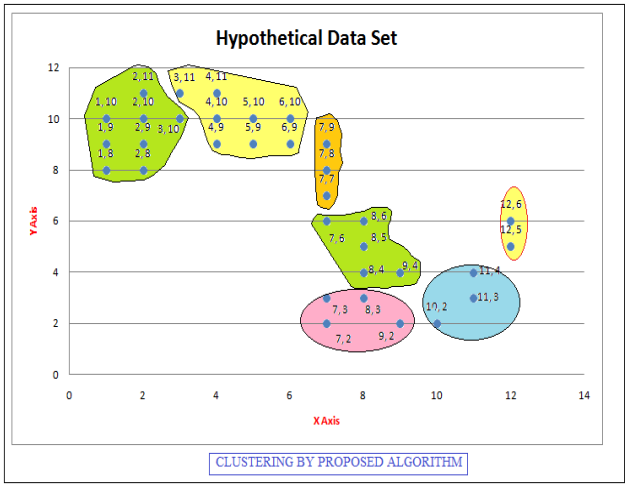

Figure3. Cluster formed using Proposed Algorithm

Table1 Comparison of Square Error of algorithm's.

Algorithm Used	Threshold	Min.No. of ob. Inside cluster	Square Error	No. of Obj. as Outlier.	No. of cluster formed
Proposed Algo.	2	2	39.04	0	7
Nearest Neighbor	2	2	602.08	0	1

Table1 show the calculated error of the Nearest Neighbor Clustering Algorithm and Proposed Clustering Algorithm, it is obvious that the Square-error and No. of Cluster formed obtained by Proposed Clustering Algorithm (39.04, 7) is better than the calculated Squared-error by Nearest Neighbor Clustering (602.08, 1).

4. PROPOSED CLUSTERING ALGORITHM

Input: // A set D of n objects to cluster. A threshold value Tth.

D= { $d_1,d_2,d_3,\ldots\ldots\ldots\ldots\ldots,d_n$ }, T_{th}

Output: // A set K of k subsets of D as final clusters and a set C of centroids of these clusters.

K= {k_1, k_2, $k_3,\ldots\ldots\ldots\ldots\ldots$, k_k,},

C= {c_1, c_2, $c_3,\ldots\ldots\ldots\ldots\ldots,c_k$}

Algorithm:

Proposed cluster algorithm (D, Tth)

1. Let k=1
2. // Randomly choose a object from D, let it be p
   k_1= {p}
3. K={k_1}
4. c_1=p
5. C={c_1}
6. Assign a constant value to Tth
7. for l=2 to n do
8. Choose next random point from D other than already chose points let it be q.
9. Determine m, distance between q and centroid c_m(1<=m<=k) in C such that distance is minimum using eq. (1).
10. If (distance<=T_{th}) then
11. k_m=k_m union q
12. Calculate new mean (centroid cm) for cluster k_m using eq. (2).
13. Else k=k+1
14. k_k={q}
15. K=K union {k_k}
16. c_k=q
17. C=C union {c_k}

4.1. Advantages of Proposed Clustering

Having looked at the available literature indicates the following advantages can be found in proposed clustering over Nearest Neighbor Clustering Algorithm.

1. In Nearest Neighbor Clustering adjacency matrix stores the distance between each data object. Preprocessing cost of making an adjacency matrix of order n×n requires time and space but in proposed clustering algorithm there is no such preprocessing and less memory space is required.
2. Nearest Neighbor Clustering takes more time to cluster than proposed clustering algorithm.
3. Chain effect can be removed with the help of Proposed Algorithm.

5. TIME COMPEXITY

Time taken by an algorithm depends on the input data set. Clustering a thousand data objects takes longer time than clustering one object. Moreover Nearest Neighbor algorithm and proposed algorithm takes different amounts of time to cluster same data objects. In general, the time taken by an algorithm grows with the size of input, so it is traditional to describe the running time of program as a function of size of its input. To do so, there is need to define the terms "Running Time" and "Size of Input" more carefully. Most natural measure is the number of objects in the input. In this analysis number of objects is represented by n. Running time of an algorithm on a particular input is the number of primitive operations or "steps" executed. It is convenient to define the notion of steps so that it is as machine –independent as possible. A constant amount of time is required to execute each line of algorithm. One line may take different amount of time than another line, but it is assumed that each execution of ith line takes time m_i where m_i is a constant. In the following discussion, expression for running time of both algorithms evolves from a messy formula that uses all the statement costs m_i to a much simpler notation that concise and more easily manipulated. This simpler notation makes it easy to determine whether one algorithm is more efficient than another.

5.1 Time Complexity of Nearest Neighbor Clustering

In Nearest Neighbor Clustering, number of times each statement rune is known [5]. 1st, 2nd, 3rd, and 4th statement runs one time only with cost m1, m2, m3, m4 respectively. Next statement, for i=2, 3,……, n where n is number of data objects, runs n times with cost m5. 6th statement for each n, scans each object in each cluster with cost m6. To understand running time for this statement, let there are k cluster and in each cluster there are s objects. So running time of this statement, for each n and for each k is s+1. 7th statement runs n-1 times. Rest of statement is part of if-then-else body. Let if –then part body run for r times with cost m8 and then else part runs for n-1-r times with cost m9, m10, m11.

Running time for algorithm is the sum of running time for each statement executed i.e.

$T(n)=m1*q+m2*q+m3*q+m4*q+m5*n+m6*_{i=2}\sum^{n}{}_{j=1}\sum^{k}(s+1)+m7*(n-1)+m8*r+m9*(n-1-r)+m10*(n-1-r)+m11*(n-1-r).$

$T(n)==m1+m2+m3+m4+(m5+m7+m10+m11)*n-(m7+m10+m11)+(m8-m9-m10-m11)*r+m6*_{i=2}\sum^{n}{}_{j=1}\sum^{k}(s+1).$

For **worst case** it will be O(nks).
For **best case** O(nks).
For **average case** O(nks).

5.2 Time complexity of Proposed Clustering Algorithm.

In proposed clustering algorithm, like Nearest Neighbor Clustering, number of times each statement runs is known. 1st, 2nd, 3rd, 4th, 5th and 6th statement runs one time only with cost m1, m2, m3, m4, m5 and m6 respectively. Next statement for i=2, 3,……..n, runs n times with cost m7 where n is number of data objects. 8th statement finds next random object to cluster. 9th statement scans centroid of each cluster with cost m9. So it runs k+1 times where k is number of clusters. Rest of statement is part of if-then-else body which runs for n-1 times.

Let if-then part body runs for r times with cost m11, m12 and then else part body runs for n-1-r times with cost m13, m14, m15, m16 and m17.

Running time algorithm is the sum of running time for each statement executed i.e.

$T(n)=m1*1+ m2*1+ m3*1+ m4*1+ m5*1 +m6*1+ m7*n+ m8*q+ m9*i=2\sum n (k+1)+ m10*(n-1)+m11*r+ m12*r+ m13*(n-1-r)+ m14*(n-1-r)+ m15*(n-1-r)+ m16*(n-1-r)+ m17*(n-1-r).$

$T(n)=m1+ m2+ m3+ m4+ m5 +m6+ (m7+ m10+ m13+m14+m15)*n-(m10+m13+ m14+ m15+ m16+ m17)+(m11+ m12- m13- m14- m15- m16- m17)*r+m9* i=2\sum n (k+1)+m8*q.$

For **worst case** let p increases with increase in i then

$$i=2\sum n (k+1) =2+3\text{……..}n$$

$$=n*(n+1)/2-1$$

So $T(n)=m1+ m2+ m3+ m4+ m5 +m6+ (m7+ m10+ m13+m14+m15)*n-(m10+m13+ m14+ m15+ m16+ m17)+(m11+ m12- m13- m14- m15- m16- m17)*r+m9* n*(n+1)/2-1+m8*q.$

$T(n)=O(n2)$

For **best case** let p=1 for 2<=i<=n

then $i=2\sum n (k+1)=2*n$

$T(n)=m1+ m2+ m3+ m4+ m5 +m6+ (m7+ m10+ m13+m14+m15)*n-(m10+m13+ m14+ m15+ m16+ m17)+(m11+ m12- m13- m14- m15- m16- m17)*r+m9* 2*n+m8*q.$

$T(n)=O(n)$

For **average case** it will be $O(n^i)$ for 1<=i<=2.

Table2 Comparison of algorithm's running time

Name of algorithm	Worst case	Average case	Best case
Nearest Neighbor Clustering	*O(nks)*	*O(nks)*	*O(nks)*
Proposed Algorithm	$O(n^2)$	$O(n^i)$ where 1<=i<=2	*O(n)*

6. EXPERIMENTAL RESULT

The implementation of proposed algorithm and Nearest Neighbor Clustering Algorithm are done 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 3: Experimental Result obtained by Nearest Neighbor Clustering Algorithm.

TESTCASES	THERSHOLD	MIN. NO. OF OBJ. INSIDE CLUSTER	SQUARE ERROR*100	NO. OF OBJ. AS OUTLIER	NO. OF CLUSTER FORMED	SQ.ERR.* NO. OF OUTLIERS
1	5	2	5.97	9	8	53.73
	6	2	**6.63**	**6**	9	**39.78**
	7	2	15.3	4	7	61.2
	8	2	44.36	2	6	88.72
	9	2	60.61	2	5	121.22
	10	2	60.61	2	5	121.22
2	5	3	5.97	9	8	53.73
	6	3	**6.57**	**8**	8	**52.56**
	7	3	15.3	4	7	61.2
	8	3	44.36	2	6	88.72
	9	3	60.61	2	5	121.22
	10	3	60.61	2	5	121.22
3	5	4	5.84	12	7	70.08
	6	4	**6.44**	11	**7**	**70.84**
	7	4	15.17	7	6	106.19
	8	4	44.23	5	5	221.15
	9	4	60.47	8	3	483.76
	10	4	60.47	8	3	483.76

Table 4: Experimental Result obtained by Proposed Algorithm

TESTCASES	THERSHOLD	MIN. NO. OF OBJ. INSIDE CLUSTER	SQUARE ERROR*100	NO. OF OBJ. AS OUTLIER	NO. OF CLUSTER FORMED	SQ.ERR.*NO. OF OUTLIERS
1	5	2	2.7	14	12	37.8
	6	2	4.29	10	11	42.9
	7	2	**5.33**	**7**	11	**37.31**
	8	2	7.9	6	10	47.4
	9	2	10.57	5	9	52.85
	10	2	12.41	4	9	49.64
2	5	3	2.42	24	7	58.08
	6	3	4.15	14	9	58.1
	7	3	**5.2**	**9**	10	**46.8**
	8	3	7.57	10	8	75.7
	9	3	10.23	7	8	71.61
	10	3	12.07	6	8	72.42
3	5	4	2.58	28	6	72.24
	6	4	3.72	20	7	74.4
	7	4	**4.49**	**18**	7	**80.82**
	8	4	7.31	13	7	95.03
	9	4	9.97	10	7	99.7
	10	4	11.81	9	7	106.29

Above table shows eighteen test cases (3×6) which are observed by taking minimum number of object in a cluster as 2, 3 and 4. Keeping "**Min. No. of Object inside Cluster**" as constant we have change the threshold value starting from 5 to 10. For each threshold value we have obtained different values of Square Error, No. of Objects as Outliers and No. of Cluster Formed as shown in table 3 for Nearest Neighbor Clustering and table 4 for Proposed Clustering Algorithm.

7. DETERMINING THE THRESHOLD VALUE

Both the clustering algorithms require the threshold value pre-set by the user. It is well-known that this parameter affects the performance of the algorithm significantly. This poses a serious question as to which threshold should be chosen when prior knowledge regarding the clusters is unavailable. Most of the criteria have been used to decide the threshold value one of these is that threshold value should be equal to the mean value of distance between pair of object.

In above tables in column No. 7 we have given the product of Square Error and No. of Object as Outlier, there is huge variation in values obtained in both the algorithm. For a cluster Square Error value and No. of object as Outlier should

be minimum. From above observation we concluded that minimum value of product of Square Error and No. of Object as Outlier in column 7 will give the best possible threshold value for the data set.

Best possible threshold value for Nearest Neighbor Clustering Algorithm when Min. No. of Object inside cluster is taken as 2 is **"6"** which give product of Square Error and No. of Object as Outlier =39.78 which is minimum of all the values.

Best possible threshold value for Proposed Clustering Algorithm when Min. No. of Object inside cluster is taken as 2 is **"7"** which give product of Square Error and No. of Object as Outlier =37.31 which is minimum of all the values.

Similarly when we take min. No. of object inside cluster as 3 then best possible threshold value for Nearest Neighbor and Proposed Clustering algorithm are comes out "6" and "7" respectively.

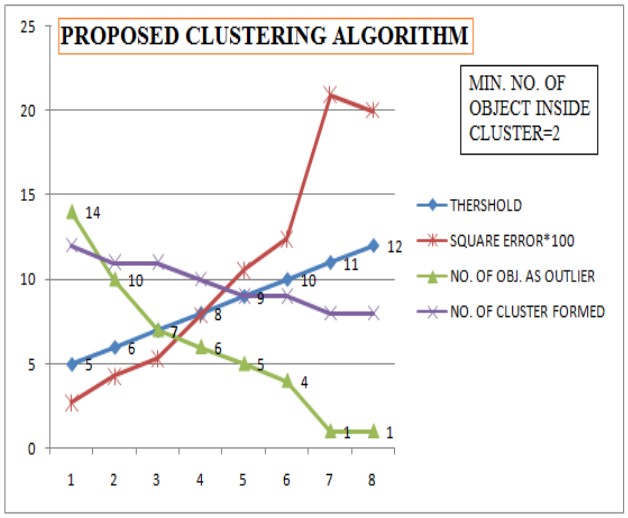

Figure4: Graph representing test case1 using proposed Algorithm.

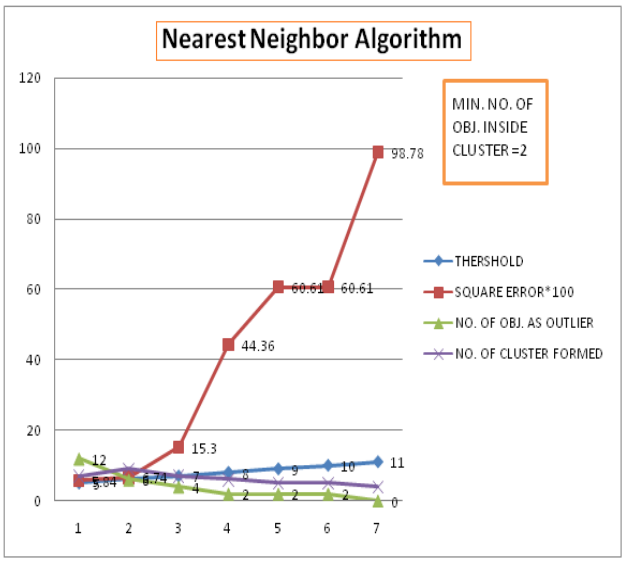

Figure5: Graph representing test case1 using Nearest Neighbor Algorithm.

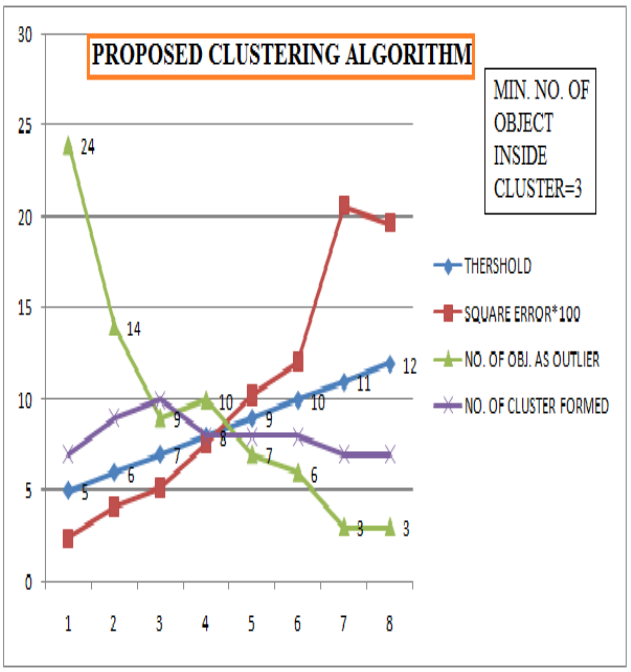

Figure6: Graph representing test case2 using proposed Algorithm.

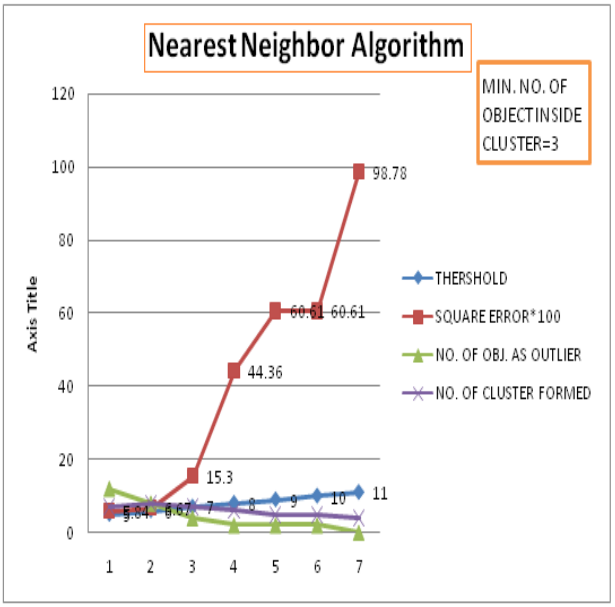

Figure7: Representing test case2 using Nearest Neighbor Algorithm.

Above graph shows that

1. As threshold value increases Square Error increases. 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 increase the threshold value number of cluster form monotonically decreasing.

3. As we increase the threshold value number of object as Outlier monotonically increases.
4. From the graph of Nearest Neighbor Clustering Algorithm we find that with small increment in threshold value we get high increment in Square Error which shows that compactness of cluster decreases very rapidly with increment in threshold value.
5. In case of Nearest Neighbor Clustering Square Error value comes out very high as compare to Proposed Algorithm.

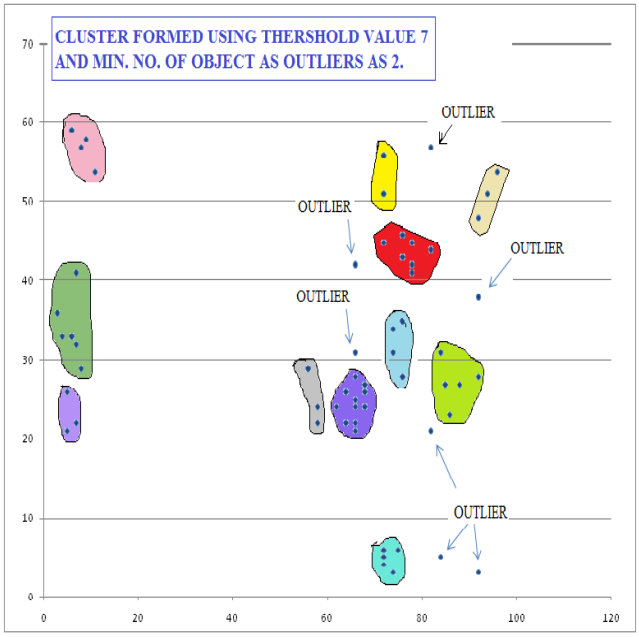

Figure 8: Cluster formed using Proposed Algorithm for test case 1 having threshold value 7.

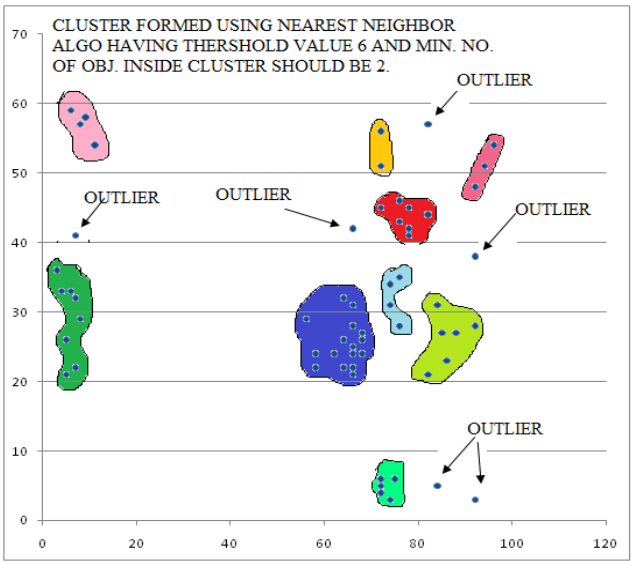

Figure 9: Cluster formed using Nearest Neighbor Algorithm for test case 1 having threshold value 6.

6. CONCLUSIONS

In this paper we presented a simple idea to enhance the efficiency of Nearest Neighbor Clustering. Experimental results demonstrated that our schemes can improve the execution of Nearest Neighbor Clustering, with no miss of clustering quality. This paper also explains the time complexity of Nearest Neighbor Clustering and our purposed algorithm. Proposed Algorithm can also remove the drawback of chain effect of Nearest Neighbor Clustering Algorithm. In last we have try to find out the best threshold value for data set.

7. REFERENCES

[1] Han, J. & Kamber, M. (2012). Data Mining: Concepts and Techniques. 3rd.ed. Boston: Morgan Kaufmann Publishers.
[2] Clustering Algorithms, Edie Rasmussen, Uni. Of Pittsburg,http://repository.cmu.edu/cgi/viewcontent.cgi?article=3484&context=compsci.
[3] G.K.Gupta, Introduction to Data Mining with Case Studies, 2nd Edition, PHI publication,4.7, Page No.189.
[4] Stephane Tuffery, Data Mining and Statistics for Decision Making, 2011, Wiley publication, page no. 261.
[5] Pang-Ning Tan, Michael Steinbach, Introduction to Data Mining, Pearson Education, page no.518.
[6] 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.
[7] 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.
[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 Evaluation of Incremental K-means Clustering Algorithm, IFRSA International Journal of Data Warehousing & Mining |Vol1|issue 1|Aug 2011.
[10] M H Dunham, "Data Mining: Introductory and Advanced Topics," Prentice Hall, 2002.
[11] R C Dubes, A K Jain, "Algorithms for Clustering Data," Prentice Hall, 1988.