Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/100332

An Improved Agglomerative Clustering Algorithm for Outlier Detection

Krishnamoorthy R[1,*] *and Sreedhar Kumar S*[2]

[1] Department of CSE, Anna University Chennai, BIT Campus, Trichy-620024,Tamilnadu, India
[2] Departments of CSE, Don Bosco Institute of Technology, Bangalore-560074, Karnataka, India

Received: 28 Oct. 2015, Revised: 29 Jan. 2016, Accepted: 1 Feb. 2016
Published online: 1 May 2016

Abstract: In this paper, three new techniques namely improved Limited Iteration Agglomerative Clustering (iLIAC), Global Outlier Validation (GOV) and Effective Cluster Validation Method (ECVM) are proposed. The proposed work aims to automatically separate the outliers (irrelevant or error data) and normal clusters over the large dataset through the process of identifying the maximum number of highly relative clusters with good accuracy. The first proposed technique iLIAC works with a new threshold (optimum merge cost) that aims to limit the number of iterations, and it automatically identifies the maximum number of highly relative clusters and outliers over the large dataset with higher accuracy and fewer misclassification errors and less computational time. The second technique GOV evaluates the global outliers around the result, and the last technique ECVM measures the purity (intra-cluster similarity) and impurity (intra-cluster dissimilarity) over the result of the iLIAC technique. Experimental results show that the proposed iLIAC technique is quicker and better to separate the normal clusters and outliers over the large dataset with good accuracy than the existing techniques.

Keywords: improved Limited Iteration Agglomerative Clustering (iLIAC), Outlier, Global Outlier Validation (GOV), Effective Cluster Validation Method (ECVM), Intra Cluster Similarity and Intra Cluster Dissimilarity

1 Introduction

Generally, outlier is an observation point that is at a distance from other observations. The outlier points can indicate faulty data, erroneous procedures, experimental errors and systematic errors. The outliers in the observation set can directly affect the accuracy of data analysis process such as classification, clustering, decision tree learning, statistical measures, and standard deviation and asymmetric. The inclusion or exclusion of outliers in an analysis depends upon the purpose of data mining. Sometimes removing or replacing outliers have improved the accuracy of the resulting cluster or patterns [1].

Statistical outlier detection methods are reported in [2] which targets the distribution of data, parameters and types and also the number of expected outliers. In [3] reported a Local Outlier Factor (LOF) based outlier detection approach. This approach used to identify the outliers based on the density of local neighborhood relying on the local outlier factor (LOF) of each point, which depends on the local density of its neighborhood. Bin-mei Liang [4] reported a hierarchical clustering based global outlier detection method for finding the outliers over the unsupervised clustering tree by top down approach. In [5] reported an automatic Partition Around Medoids (PAM) clustering algorithm that used to identifies the outliers over the large dataset.Some of the popular traditional clustering techniques namely DBSCAN, CHAMELEON, CLARANS, ROCK, CURE and BIRCH are reported to find the patterns or clusters over the dataset while also finding outliers in the dataset. They are optimized for clustering rather than outlier detection [7]. George Kollios et al.[8] reported a density-biased sampling technique for speed-up the clustring and outlier operations over the large multidimensional datasets. They suggested this technique is great flexibility and improved accuracy of the results over simple random sampling. In [9] reported a different algorithm namely DBSCAN, it makes use of two external parameters, the minimum number of points in the neighborhood of a point and the radius that defines this neighborhood. Choosing the appropriate parameters, it is then possible to identify the objects located in the high and low density regions. Neighboring objects in the high

* Corresponding author e-mail: rkrish07@yahoo.com

density region define clusters. The advantage of the DBSCAN is that it consumes lesser computational and time complexities. George Karypis et al.[10] reported a method called CHAMELEON. They suggested that the CHAMELEON finds the normal clusters and outliers over the dataset through a two-phase algorithm. In the first phase, it uses a graph partitioned algorithm to cluster the data items into several relatively small sub-clusters and in the second phase, it finds the genuine clusters through repeatedly merging these sub-clusters. Zengyou-He et al. [11] reported a frequent item-set based outlier detection mechanism. In this approach, the outliers are separated through the Frequent Pattern Outlier Factor (FPOF). In [12] reported an algorithm namely FindCBLOF that used to discover the outliers over the data set. Another approach called CLARNS was motivated through PAM and CLARA techniques [13]. This CLARANS approach is used to identify the distinct clusters over the data set based on the randomized search. The authors suggested that the CLARANS could produce better clustering result with higher accuracy than PAM and CLARA techniques.

Sudipto Guha et al. [16]reported a robust hierarchical clustering algorithm called ROCK that employs links and not distances while merging clusters. The authors have suggested that the methods were naturally extending to non-metric similarity measures and that the relevance in situations where a domain expert or similarity tables is the only source of knowledge. Dutta M et al. [17] suggested drawback in the DBSCAN in which the entire clustering result accuracy is based on two external parameters and also reported a technique namely QROCK, that computes the clusters by determining the connected components of the graph. This method is very efficient in obtaining the clusters and giving a drastic reduction to the computing time of the ROCK. In [18] reported a clustering technique called CURE that is more robust to outliers and identifies clusters having non-spherical shapes and wide variances in size. The authors have suggested that the CURE achieves through representing each cluster and then shrinking them toward the center of the cluster by a specified fraction. Basically, the CURE employed a combination of random sampling and partitioning, a random sample drawn from the data set is first partitioned, and each partition is partially clustered. The partial clusters are then clustered in a second pass to yield the desired clusters. Zhang et al.[19] reported a technique namely Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH). They suggested that the BIRCH is faster and better suitable to process the large data set with noise, and it can produce higher quality clustering result with the available memory resource.

Seung Kim et al. [20] reported a method called fast outlier detection that was reducing the Local Outlier Factor (LOF) computation time. In [21] reported a method namely Spatial Local Outlier Measure (SLOM) that captures both spatial autocorrelation and spatial non-constant variance. The author suggested that the SLOM method is sharper to discern local spatial outliers that are usually missed by global techniques. Another author was reported a statistical approach to the problem of inlier-based outlier detection for finding outliers in the test set based on the training set consisting only of inliers [22]. Vijaya et al. [23] reported a two level clustering algorithm namely Leaders-Subleaders. This approach is used to identify the subgroups in the each cluster. Xiaohui Liu et al. [24], the authors had suggested a cautious approach to outlier analysis in that only those outliers most likely to be noisy are eliminated. This approach to knowledge-based outlier analysis is a useful extension to existing work in both statistical and computing communities on outlier detection.

Yang P and Zhu Q S [25] reported a method for finding the key attribute subset in dataset that starts with seeking all outliers on the full attribute set, and then searches through all outlying attribute subsets for these points. Later it was able to determine the key attribute subset in accordance with the similarity between outlying partitions. Yong Zhang et al. [26] reported two algorithms namely Local Distribution Based Outlier Detection (LDBOD) and LDBOD+ for outlier detection. These approaches could detect the local outliers from the viewpoint of local distribution that is characterized with three measurements local average distance,local density and local asymmetry degree. They noticed two drawbacks in this approach: 1) not applicable for non-continuous features, and 2) higher computational complexity for handling large scale with high dimensional dataset.

In [28] focused cluster validity measure with outlier detection and cluster merging algorithms for the Support Vector Clustering (SVC). They reported through these three parameters that the SVC algorithm is capable of identifying the ideal cluster number with compact and smooth arbitrary shaped cluster contours and increased robustness to outliers and noises. In [29] reported an outlier detection method based on clustering analysis. It detects outliers over the suspicious outlier set, and puts non-outlier into a cluster which has a similar characteristic with it. They suggested that the outlier may lead to wrong analysis and hence to a wrong prediction, which in turn resulted in making a wrong decision. Many of the authors [30,31,32] have suggested several drawbacks over the traditional hierarchical agglomerative clustering technique (HAC): 1) higher space and computation complexity for clustering the large dataset 2) validation method is inefficient and inaccurate for evaluating the clustering result 3) difficulties in finding the optimum number of clusters over the single clustering tree.

The above clustering techniques have failed to automatically separate the normal clusters and outliers over the large dataset. In order to overcome the above drawbacks, in this paper, three new techniques namely improved Limited Iteration Agglomerative Clustering (iLIAC), Global Outlier Validation (GOV) and Effective Cluster Validation Method (ECVM) are proposed. The first technique iLIAC works with a new threshold

(optimum merge cost) that aims to limit the number of iterations and it automatically identifies the maximum number of highly relative clusters and outliers over the large dataset with higher accuracy and fewer misclassification errors and less computational time. The GOV is able to evaluate the global outliers over the resulting cluster. The ECVM is better suitable in computation of the intra-cluster similarity and intra-cluster dissimilarity over the resulting cluster. Experimental results show that the proposed iLIAC is quicker and better to identify the perfect number normal clusters and outliers over the large dataset with good accuracy than the existing techniques.

This paper is organized as follows: section 2 and section 3 deals with the proposed threshold and GOV technique respectively. Section 4 contains the details of the proposed iLIAC algorithm. Proposed ECVM method is discussed in section 5 and experimental results are presented in section 6. Conclusions and future research scope are drawn in section 7.

2 Proposed Threshold Technique

In this section, the detailed description of the proposed threshold technique is presented. The threshold is a semi-supervised technique that computes the optimal merge cost to limit the number of iterations [15]. The proposed threshold is an optimal merge cost that helps to find the exact iteration to end the clustering process and then automatically produces the optimum number of clusters over the given object set with good accuracy. Generally, the optimum merge cost is the major component that can directly affect the quality of the cluster. For example, if the optimal merge cost is too small, then large numbers of clusters are generated. On the other hand, if the optimal merge cost is too large, then a very few clusters are generated on the final clustering result. The proposed threshold method that computes the optimum merge cost (OMC) is defined in the equation (1) as:

$$OMC = |(SD(X) - VA(X))| \tag{1}$$

where, $SD(X)$ denotes the standard deviation of input object set X is defined in equation (2) as:

$$SD(X) = \{((\frac{1}{n}\sum_{i=0}^{n}(X_i - \bar{X})^2)^{\frac{1}{2}} | \forall X_i \in X\} \tag{2}$$

where, X_i represents the i^{th} object that belongs to the input object set X and $\bar{X}$ denotes the mean of the input object set X with n objects for $i = 0, 1, .., n$ and is defined in equation (3) as:

$$\bar{X} = \{\frac{\sum_{i=0}^{n} X_i}{n} | \forall X_i \in X\} \tag{3}$$

where, $VA(X)$ is the variance of the input object set X and is defined in equation (4) as:

$$VA(X) = \{((\frac{1}{n}\sum_{i=0}^{n}(X_i - \bar{X})^2)^{\frac{1}{2}})^{\frac{1}{2}} | \forall X_i \in X\} \tag{4}$$

where, X_i represents the i^{th} object that belongs to the input object set X and n denotes the size of the input object set. Corresponding to the optimum merge cost(OMC) the proposed technique iLIAC can identify the maximum number of highly relative clusters over the input object set X.

3 Proposed Global Outlier Validations

The outlier validation is a type of measure that is capable of identifying the outliers over the cluster set. The proposed method GOV is aimed to evaluate the global outliers over the resulting cluster of proposed technique iLIAC. This method consists of two steps. The first step, is that it measures the degree of the each individual cluster $D(C_i)$ over the resulting cluster R_C, where $D(C_i)$ represents the degree of the i^{th} cluster that belongs to the resulting cluster and is defined in equation (5) as:

$$D(C_i) = \{\sum_{i=1}^{k}\sum_{j=1}^{N_i} C_{ij} | \forall C_{ij} \in C_i, \forall C_i \in R_C\} \tag{5}$$

where, C_{ij} represents the count of the j^{th} object in i^{th} cluster that belongs to the resulting cluster R_C, C_i denotes the i^{th} cluster in the resulting cluster and N_i is describing the number of objects in i^{th} cluster. In the second step, it verifies the cluster is normal or outlier based on the degree of cluster. If the degree of the cluster is equal to one then it confirms that the particular i^{th} cluster is an outlier otherwise it marks that the particular i^{th} cluster is normal. Fig 1 shows an example of the proposed outlier validation.

Fig 1 contains five clusters namely C_1, C_2, C_3, C_4 and C_5. It is clear that the size of each cluster varies from one another among the clusters in the cluster set. The proposed method GOV is tested over each individual cluster indicated in the Fig 1, and it has identified two outliers (C_4 and C_5) and three normal clusters (C_1, C_2 and C_3) respectively based on degree of clusters are defined as D(C_1)=7, D(C_2)=8, D(C_3)=6, D(C_4)=1 and D(C_5)=1. The above result shows that the proposed method GOV is simple and easy to evaluate the outliers and normal clusters over the resulting cluster.

4 Proposed iLIAC technique

From the literature survey, many authors have clearly noticed the drawbacks over the traditional Hierarchical

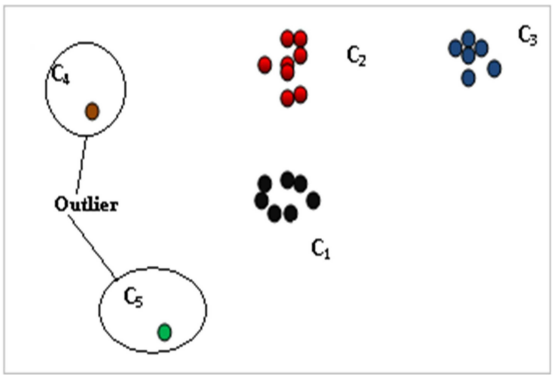

Fig. 1: An example of normal clusters and outliers

Agglomerative Clustering (HAC) technique: (1) failed to automatically separate the optimum number of clusters and outliers (2) higher space and computation complexity (3) consumes $(n-1)$ iterations, where n denotes the size of the dataset (4) producing singleton clustering tree (5) need cluster validation technique to trace the optimum number of clusters over the singleton clustering tree, and (6) producing result with lesser accuracy and higher misclassification error. In order to overcome the above drawbacks, in this section, a new technique called improved Limited Iteration Agglomerative Clustering (iLIAC) is present. The iLIAC automatically separates the normal clusters and outliers over the large dataset through the process of identifying the maximum number highly relative clusters based on optimum merge cost. The iLIAC technique consists of three stages viz. (1) threshold stage, (2) clustering stage, and (3) outlier validation stage. In the threshold stage, it computes the optimum merge cost (OMC) over the input object set X through the equation (1) and the input object set X is defined as $X = \{X_0, X_1, .., X_n\}$, where X_1 is representing the i^{th} object that belongs to the input object set X and n denotes the number of objects in the input object set X. In the clustering stage, it starts with the each individual object in the input object set $X = \{X_0, X_1, .., X_n\}$ as individual cluster. In the beginning, the proposed technique constructs the upper triangular distance matrix Ud_{ij} for input object set X, and subsequently it identifies the closest clusters pair (X_i, X_j) with a minimum merge cost $\triangle d$ over the matrix Ud_{ij} and is defined in equation (6) as:

$$\triangle d = \min\{d(X_i, X_j) | \forall (X_i, X_j) \in Ud_{ij}\} \quad (6)$$

where Ud_{ij} is the upper triangular distance matrix for n cluster and which is defined in equation (7) as:

$$Ud_{ij} = \{d(X_i, X_j) | 0 \leq i \leq (n-1) \forall j > i, j \leq (n-1)\} \quad (7)$$

and $d(X_i, X_j)$ is the Euclidean distance betweeni^{th} and j^{th} clusters that belongs to the input cluster set X is defined in equation (8) as:

$$d(X_i, X_j) = \{(\sum_{r=1}^{d} (X_{ir} - X_{jr})^2)^{\frac{1}{2}} | \forall X_{ir} \in X_i, \forall X_{jr} \in X_j\} \quad (8)$$

Where $X_{ir}$and X_{jr} represent the r^{th} feature that belong to the respective i^{th} and j^{th} clusters and d describes the number of features in the cluster. Once, that the closest clusters pair $d(X_i, X_j)$ is merged into a single cluster $\acute{X}_{ij}$with minimum merge cost and later it updates the merged cluster $\acute{X}_{ij}$ by average function that is defined in equation (9) as:

$$X_i = \{\frac{X_i + X_j}{2} | X_i \in X, X_j \in X\} \quad (9)$$

where, X_i and X_j represents the respective i^{th} and j^{th} clusters that belong to the input cluster set X . Next, it deletes the j^{th} cluster in the input cluster set X and then reduces the input cluster set size to $(n-1)$. The above process is repeated until the minimum merge cost of the cluster pair $\triangle d$ exceeds the optimum merge cost (OMC) and finally it produces a maximum number of K highly relative clusters or multi-ton clusters in the resulting cluster R_C over the input object set X and is defined as $R_C = \{C_1, C_2, .., C_K\}$, where C_1 denotes the i^{th} cluster that belongs to the resulting cluster R_C and K represents the number of clusters in the resulting cluster R_C for $i = 1, 2, .., K$. In the outlier validation stage, it measures the degree of each individual cluster $D(C_i)$ over the resulting cluster R_C and subsequently it identifies the each individual cluster that is normal or outlier based on its degree of the cluster. The proposed iLIAC algorithm is described as follows:

Algorithm
Threshold Stage:
Input: $X = \{X_0, X_1, .., X_n\}$
Output: (OMC)
Begin

1.Initialize the input object set X
2.Compute the standard deviation SD of the input object set X using equation (2)
3.Compute the variance VA of input object set X using equation (4)
4.Calculate the merge cost (OMC) using equation (1) Based on the results of equations (2) and (4)

End
Clustering Stage:
Input: $X = \{X_0, X_1, .., X_n\}$ and Optimum Merge Cost (OMC)
Output: $R_C = \{C_1, C_2, .., C_K\}$
Begin

1.Assume each individual object as a cluster in the input object set $X = \{X_0, X_1, .., X_n\}$
2.Construct the upper triangular distance matrix Ud_{ij} for input object set X

3.Find the closest clusters pair (X_i, X_j) with minimum merges cost $\triangle d$ over Ud_{ij} using equation (7).
4.Compare the selected clusters pair(X_i, X_j) with minimum merge cost $\triangle d$ to optimum merge cost (OMC) : If $(\triangle d < OMC)$ then step 5 Else step 10
5.Merge the selected clusters pair (X_i, X_j) into single Cluster $(X_i, X_j) -> \acute{X_{ij}}$
6.Update merged cluster $\acute{X_{ij}}$ by equation (9)
7.Delete j^{th} cluster in the input cluster set X
8.Reduce the input cluster set size by $(n-1)$
9.Repeat the steps 2 to10 until the condition is unsatisfied in the step 4.
10.Stop the clustering process.

End
Outlier Validation Stage:
Input: $R_C = \{C_1, C_2, .., C_K\}$
Begin

1.Measure the degree of each individual cluster in the resulting cluster R_C with equation (5).
2.Identify the normal cluster and outlier following conditions:
(a)If the degree of the cluster $D(C_i)$ is equal to one, then mark the i^{th} cluster is outlier.
(b)If the degree of the cluster $D(C_i)$ is greater than one, then mark the i^{th} cluster is normal.

End

5 Complexity Analysis

In this section an analysis of computational complexity of the proposed technique iLIAC is presented. The proposed iLIAC technique consumes $O(\frac{n(n-1)}{2})$ time to construct the upper triangular distance matrix Ud_{ij}, where n denotes the number of clusters in the input cluster set X. An iteration time $O(\frac{n(n-1)}{2})$ is required for linear search of the closest clusters pair (X_i, X_j) with minimum merge cost $\triangle d$ on the matrix Ud_{ij} for $i = 0, 1, .., n$, $j = i+1, .., n-1$ and $j > i < n-1$, where i and j are represent the i^{th} and j^{th} clusters respectively. In the merging process, it requires $O(1)$ time to merge the selected closest clusters pair $(X_i, X_j) -> \acute{X_{ij}}$. The updating process requires $O(1)$ time to eliminate the j^{th} cluster on the input cluster set X. Therefore, the overall time complexity of the proposed technique iLIAC is $O((\frac{n(n-1)}{2}) + 1 + 1)$ for $(n-k)$ iterations where k represents the number of iterations reduced and n denotes the size of the input cluster set X. As a whole, the proposed technique reduces the space complexity from $O(n^2)$ to $O(\frac{n(n-1)}{2})$ and computational complexity from$O(n^3)$ to $O((\frac{n(n-1)}{2}) + 1 + 1)$ compared to the existing agglomerative clustering technique.

6 Proposed Cluster Validation Technique

Generally, the cluster validation is a type of quality measure that calculates the accuracy and misclassification errors around the clustering result. Many authors have reported drawbacks in the existing cluster validation methods in [15, 26, 30, 32]. The authors suggested that the existing cluster validity measures are inefficient and inaccurate for evaluating the resulting cluster of the large dataset with noise or outliers and also it consumes higher time and computation complexity. To overcome these drawbacks, in this section, a new method namely Effective Cluster Validation Method (ECVM) is proposed. This method aims to measure the intra-cluster similarity (purity) and intra-cluster dissimilarity (impurity) over each individual cluster in the resulting cluster of the unsupervised clustering technique. The ECVM consists of two measures namely Purity Measure (PM) and Impurity Measure (IM) which are described in the given below subsections.

6.1 Purity Measure

The proposed PM is a simple and effective quality measure that aims to measure the purity or intra-cluster similarity around the each individual cluster in the resulting cluster of the iLIAC. It consists of three steps. In the first step, it computes the centroid β_i of the each individual cluster that belongs to the resulting cluster R_C for $i = 1, 2, .., K$ and is defined in equation (10) as:

$$\beta_i = \{\frac{1}{N_i}\sum_{i=1}^{K}\sum_{j=1}^{N_i} C_{ij} | \forall C_{ij} \in C_i, \forall C_i \in R_C\} \quad (10)$$

where, C_{ij} denotes the j^{th} object belong in the i^{th} cluster in the resulting cluster R_C and C_i is the i^{th} cluster with N objects. In the second step, it measures the purity or intra cluster similarity P_i over the each individual cluster through its centroid β_i for $i = 1, 2, .., K$ and is defined in equation (11) as:

$$P_i = \{(\frac{1}{N_i}\sum_{i=1}^{K}\sum_{j=1}^{N_i} |C_{ij} - \beta_i|) \times 100 | \forall C_{ij} \in C_i, \forall C_i \in R_C\}, \quad and \begin{Bmatrix} 1 & |C_{ij} - \beta_i| \leq T \\ 0 & |C_{ij} - \beta_i| > T \end{Bmatrix} \quad (11)$$

where, β_i represents the i^{th} cluster centroid belongs to the resulting cluster R_C , N_i denotes the number of objects in the i^{th} cluster and T is the threshold which limit the similarity level between β_i and j^{th} object that belongs to

the i^{th} cluster and respectively its value would varies based on input object set behavior. In the final step, it computes the accuracy $A(R_C)$ or overall intra cluster similarity over the resulting cluster R_C through the intra cluster similarity of the each individual cluster that is defined in equation (12) as:

$$A(R_C) = \frac{1}{K}\sum_{i=1}^{N} P_i | \forall P_i \in P \tag{12}$$

Where, P_i denotes the purity or intra cluster similarity of the i^{th} cluster that belongs to the resulting cluster R_C and K represents the total number of clusters in the resulting cluster R_C for $i = 1,2,..,K$.

6.2 Impurity Measure

The proposed IM is a type of impurity measure that aims to measure the intra-cluster dissimilarity over each individual cluster in the resulting cluster. It measures the impurity of the resulting cluster R_C in two steps. In the first step, it computes the impurity or intra-cluster dissimilarity P_i of each individual cluster that belongs to the resulting cluster R_C for $i = 1,2,..,K$ and is defined in equation (13) as:

$$IP_i = \left\{ \left(\frac{1}{N_i}\sum_{i=1}^{K}\sum_{j=1}^{N_i} |C_{ij} - \beta_i|\right) \times 100 | \forall C_{ij} \in C_i, \forall C_i \in R_C \right\}, \quad and \left\{ \begin{array}{l} 1 \, |C_{ij} - \beta_i| > T \\ 0 \, |C_{ij} - \beta_i| \leq T \end{array} \right\} \tag{13}$$

where, β_i represents the i^{th} cluster centroid that belongs to the resulting cluster R_C and N_i denotes the number of objects in the i^{th} cluster. In the second step, it calculates the overall misclassification error or intra-cluster dissimilarity of the resulting cluster R_C through the intra cluster dissimilarity of each individual cluster and is defined in equation (14) as:

$$M_E(R_C) = \frac{1}{K}\sum_{i=1}^{K} IP_i | \forall IP_i \in IP \tag{14}$$

where, IP_i denotes the impurity of the i^{th} cluster belongs in the resulting cluster R_C and K represents the total number of clusters in the resulting cluster R_C. The experimental results of the proposed cluster validation measures PM and IM are discussed in the section given below.

7 Experiment and Results

In this section, the extensive performance analysis of the proposed technique over the existing algorithms namely k-means, AHC, DBSCAN, CHAMELEON, and CURE are presented. For the experimental purpose, Sample Object Set (SOS) which contains 27 human height data that is collected from our laboratory and house are constructed. The SOS contains six normal clusters and three outliers as indicated in Table 1. Fig 2 illustrates the scatter graph of the SOS shown in Table 1. The iLIAC is tested over the SOS with single dimensional feature and is described in the next subsection.

Table 1: Sample Object Set (SOS).

Ob.Id	Objects	Ob.Id	Objects
1	6.0	2	5.9
3	5.6	4	5.5
5	5.9	6	5.6
7	5	8	5.9
9	6	10	6.3
11	5.2	12	6.1
13	5.8	14	5.8
15	4	16	2.8
17	3.2	18	4.1
19	1.2	20	2.7
21	6.1	22	2.7
23	2.8	24	8.2
25	5.3	26	5
27	6		

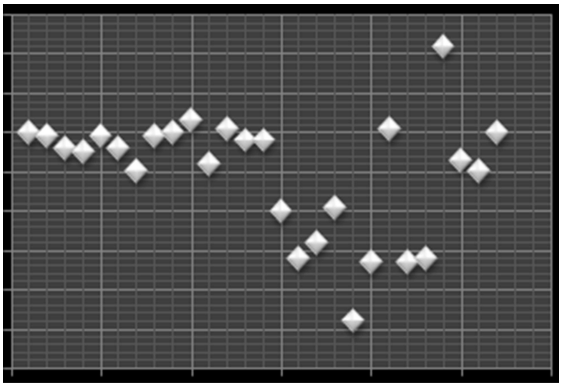

Fig. 2: Scatter Graph of the SOS shown in Table 1

7.1 iLIAC

In this subsection, the proposed technique iLIAC is tested around the SOS with n objects shown in Table 1. Table 3 shows the experimental result of iLIAC that is tested over the SOS indicated in Table 1. Table 2 shows the iLIAC that has taken (18) iterations to partition the SOS into nine highly relative clusters and is indicated in Table 3. At the every iteration, it finds the two highly relative objects or clusters that are merged together. Hence, the optimal merge cost is very smaller which is calculated using the

proposed equation (1). Based on the optimal merge cost , the iLIAC partitions the SOS into nine high relative clusters with fewer numbers of iterations and the respective results are obtained in Table 3. From the Table 3 it is clearly noticed that the resulting cluster of the proposed iLIAC is containing nine clusters such as C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8 and C_9 and each cluster contains highly similar objects with itself. Fig 3, illustrates that the experimental result of the proposed iLIAC technique tested over the SOS indicated in Table 1.

It is evidenced from the Table 4, that the outliers and normal clusters are perfectly evaluated over resulting cluster of the proposed iLIAC technique through the GOV. The GOV is properly evaluated the each individual cluster in the resulting cluster through the degree of cluster, and subsequently it has been identified that five normal clusters and three outliers over the resulting cluster are obtained and indicated in Table 4. Table 5 shows the purity or intra cluster similarity measures over the each individual cluster in the resulting cluster of the iLIAC that has been tested around the SOS indicated in Table 1 and the respective result is obtained in Table 3. It is clearly noticed from the Table 5, that the proposed method PM is accurately measured the intra cluster similarity over each individual cluster that belongs to the resulting cluster . The overall resulting accuracy is calculated through the intra cluster similarity or purity of each individual cluster in the resulting cluster and the results obtained is indicated in the Table 5.

Table 5, illustrates result of the intra-cluster dissimilarity measures over the each individual cluster in the resulting cluster of iLIAC as indicated in Table 3. According to the experimental result, IM measure is perfectly calculated the intra-cluster dissimilarity or impurity over the each individual cluster in the result of the iLIAC technique and the measured result as obtained in Table 5. The overall misclassification error has calculated over the result of the iLIAC through the intra cluster dissimilarity of each individual cluster in resulting cluster. According to the experiment results that the proposed technique iLIAC is better suitable to automatically separate the highly relative clusters and outliers over the dataset than the existing techniques that is described in the following subsection.

Table 2: Step by step result of proposed technique is tested over SOS indicated in Table 1

Iterations (I)	Merged clusters pair for iteration	Minimum merge cost($\triangle d$)	Number of object	Number of link
1	(6,6)	0	2	1
2	(6, 9),6)	0	3	2
3	(5.9,5.9)	0	2	1
4	((5.9,5.9),5.9)	0	3	2
5	(5.6,5.6)	0	2	1
6	(5,5)	0	2	1
7	(5.8,5.8)	0	2	1
8	(2.8,2.8)	0	2	1
9	(2.7,2.7)	0	2	1
10	(((6,6),6),((5.9, 5.9),5.9)))	0.1	6	5
11	((5.6,5.6),5.5)	0.1	3	2
12	(5.2,5.3)	0.1	2	1
13	(4, 4.1)	0.1	2	1
14	((2.8,2.8),(2.7, 2.7))	0.1	4	3
15	(6.1,6.1)	0.1	2	1
16	((((6,6),6),((5.9, 5.9), 8)), (5.8,5.8))	0.15	8	7
17	((5, 5),(5.2,5.3))	0.25	4	3
18	(6.3,(6.1,6.1))	0.25	3	2

Table 3: Result of iLIAC technique that tested around the SOS

R_C	Status of the each cluster in (R_C)
C_1	((((6, 6),6),((5.9, 5.9), 5.9)), (5.8, 5.8))
C_2	((5.6, 5.6), 5.5)
C_3	((5,5), (5.2, 5.3))
C_4	(6.3, (6.1, 6.1))
C_5	(4, 4.1)
C_6	((2.8, 2.8), (2.7, 2.7))
C_7	3.2
C_8	1.2
C_9	8.2

Table 4: (GOV) measures over the result of proposed iLIAC technique

Resulting Cluster (R_c)	Status of the clusters in (R_c)	Degree of cluster ($D(C_i)$)	Remark of clusters in (R_c)
C_1	((((6, 6),6),((5.9, 5.9), 5.9)), (5.8, 5.8))	8	NC
C_2	((5.6, 5.6), 5.5)	3	NC
C_3	((5,5), (5.2, 5.3))	4	NC
C_4	(6.3, (6.1, 6.1))	3	NC
C_5	(4, 4.1)	2	NC
C_6	((2.8, 2.8), (2.7, 2.7))	4	NC
C_7	3.2	1	Outlier
C_8	1.2	1	Outlier
C_9	8.2	1	Outlier

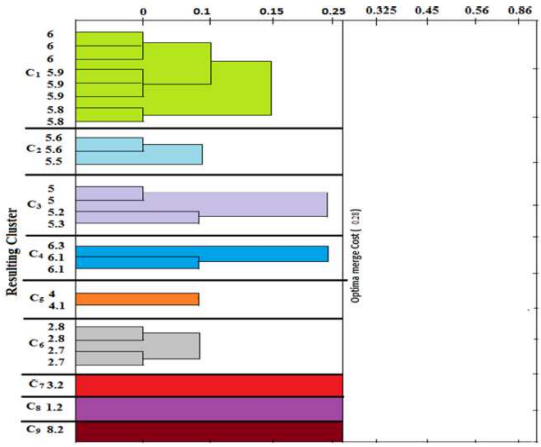

Fig. 3: Result of proposed technique iLIAC tested over the SOS indicated in Table 1

Table 5: Performance evaluation of proposed technique iLIAC

R_c	Status of the clusters in (R_c)	Performance Measures			
		P_i with (T=0.17) (%)	IP_i with (T=0.17) (%)	$A(R_C)$ (%)	$M_E(R_C)$ (%)
C_1	((((6, 6),6),((5.9, 5.9), 5.9)), (5.8, 5.8))	100.0	0.0		
C_2	((5.6, 5.6), 5.5)	100.0	0.0		
C_3	((5,5), (5.2, 5.3))	66.00	33.33		
C_4	(6.3, (6.1, 6.1))	100.0	0.0	96.22	3.703
C_5	(4, 4.1)	100.0	0.0		
C_6	((2.8, 2.8), (2.7, 2.7))	100.0	0.0		
C_7	3.2	100.0	0.0		
C_8	1.2	100.0	0.0		
C_9	8.2	100.0	0.0		

7.2 Comparison of the proposed technique over the traditional techniques

In this subsection a comparative study of the proposed technique iLIAC with traditional techniques like AHC [6, 27], k-means[7], DBSCAN [9], CHAMELEON [10] and CURE [18] are given. Firstly, we implemented the above traditional techniques and tested the same SOS that is used in the proposed technique iLIAC. Through our experiments it is found that the k-means technique has partitioned the SOS into three clusters with low accuracy (67.00) and high misclassification error (33.00) as indicated in Table 6. Also it is found that the quality of the partitioning result is based on the k centroid, where k denotes the number of centroid values that belongs to the actual input object set, and it is unsuitable to the large object set with higher dimensional. Fig 4, shows the result of the k-means technique that is tested over the SOS which indicated in Table 1.

The traditional agglomerative hierarchical clustering technique has been clustered around the SOS into a singleton cluster in the form of hierarchical tree structure through the sequence of merging operation, and the tested result is obtained in Table 6. From the experiment result, we have identified many drawbacks over the AHC technique: 1) it failed to automatically separate the optimum number of clusters and outliers, 2) higher computational and time complexity for merging two clusters with many objects, 3) consumption (n-1) iterations where n denotes the number of objects, and 4) it produces low accuracy resulting cluster with high misclassification error. Fig 5 and Fig 6, shows the experiment results of the AHC technique that is tested over the SOS shown in Table 1.

The DBSCAN has produced good results compared to k-means and AHC techniques with lesser time complexity as shown in Table 6. Fig 7 illustrates the experiment result of the DBSCAN that is tested over SOS indicated in Table 1. Based on the experimental results, we suggested that the DBSCAN is suitable for identifying the outliers and normal clusters over large object set. The main drawback in the DBSCAN is that the entire result quality is based on two internal parameters.

Similarly, the CHAMELEON technique is tested over the same SOS that is used in previous techniques. The CHAMELEON has identified seven clusters over the SOS and the result is illustrated in the Fig 8. It is from Table 6, that the CHAMELEON produces a better result with higher accuracy compared DBSCAN, AHC and k-means techniques. Finally, we tested the CURE technique over the same SOS that is used in previous techniques such that iLIAC, AHC, k-means, DBSCAN and CHAMELEON. The CURE technique has been identified eight highly relative clusters over the SOS and the results are illustrated in the Fig 9. It is evidenced from the Table 6 that the CURE technique is better suitable for finding the outliers and normal clusters with higher accuracy and lesser misclassification errors compared to DBSCAN and CHAMELEON. Through the experiment, we have identified the main drawback over the CURE technique is that it has two level partitioning procedures.

Figures 4 to 9 illustrates the overall experimental results of existing techniques that are tested over the same SOS as shown in Table 1. Fig 10 illustrates results of the GOV method evaluated for outliers and normal clusters over the results of the proposed and existing techniques presented in Figures 3 to 9. Similarly, the overall intra cluster similarity (accuracy) and intra cluster dissimilarity (misclassification) are measured over the results of the proposed and existing techniques through the ECVM measure is illustrated in Fig 11. The above experimental results prove that the proposed technique iLIAC separates normal clusters and outliers over the SOS with higher intra cluster similarity (accuracy) and lesser intra cluster dissimilarity (misclassification) compared to the existing techniques namely AHC, k-means, DBSCAN, CHAMELEON and CURE.

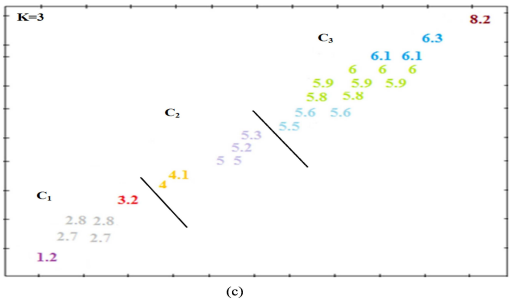

Fig. 4: Result of k-means

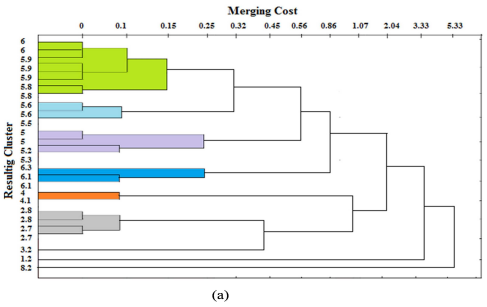

Fig. 5: Result of AHC (Single)

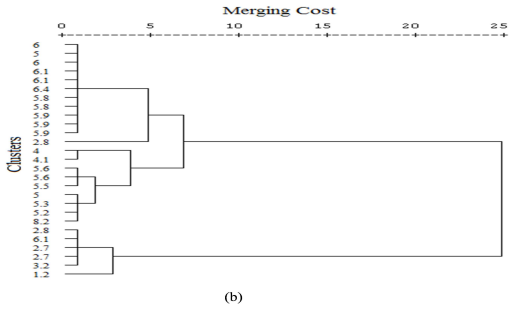

Fig. 6: Result of AHC (Average)

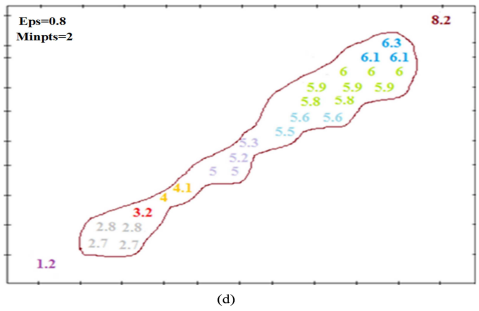

Fig. 7: Result of DBSCAN

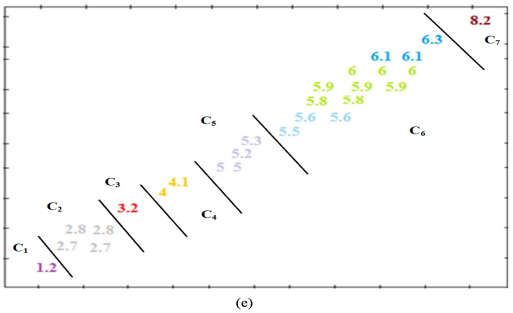

Fig. 8: Result of CHAMELEON

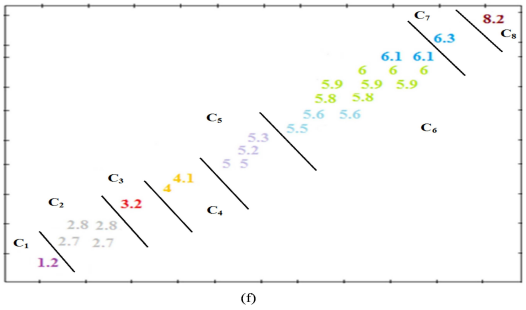

Fig. 9: Result of CURE

Table 6: Performance evaluation of iLIAC and comparison with six traditional techniques.

Techniques	Size of SOS	Number of Cluster in R_c	Number of (NC)	Number of outliers	$A(R_C)$ (%)	$M_E(R_C)$ (%)
AHC (sl)	27	01	01	Nil	66.66	33.34
AHC (al)	27	01	01	Nil	66.66	33.34
K-Mean	27	03	03	Nil	67.00	33.00
DBSCAN	27	06	04	02	91.00	9.00
CHAMELEON	27	06	03	03	92.38	7.62
CURE	27	08	04	04	93.75	6.25
iLIAC	27	09	06	03	96.22	3.70

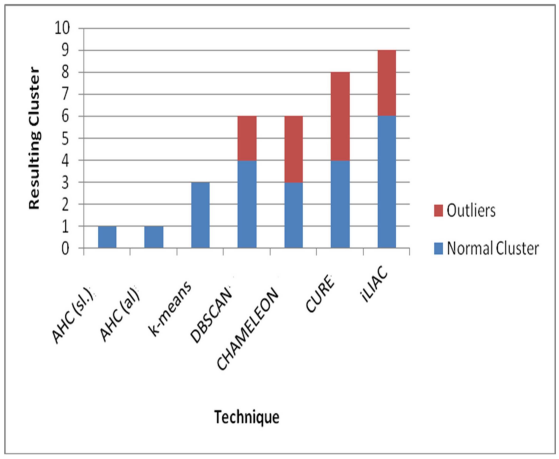

Fig. 10: GOV measures over the result of iLIAC and comparison with six traditional techniques

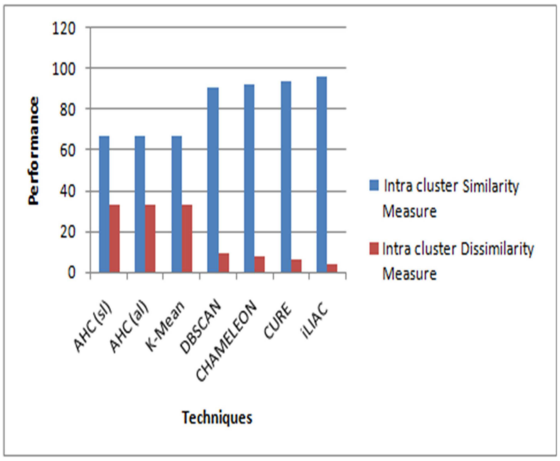

Fig. 11: ECVM measures over the results of the iLIAC and comparison with six traditional techniques

7.3 Experimentation with UCI object sets

The proposed and traditional techniques are tested over the six UCI object sets as indicated in Table 7. The six object sets are real-world instances which are taken from the UCI repository [14], including IMAGE SEG, IONOSPHERE, Red Wine, White Wine, WBDC and WISCONSIN. It is clearly indicated in Table 7 the UCI object sets are different from one another based on the number of instances and dimensionality. Table 8 shows the experimental results of the proposed and traditional techniques which are tested over the six bench mark UCI object sets are indicated in Table 7. It is clearly indicated in Table 8 that the iLIAC has identifies maximum number of highly relative clusters and outliers over the UCI object sets based on optimum merge costs 1.29, 0.228, 0.447, 1.39, 2.80, 1.1 respectively. The OMC is the major

Table 7: Description of UCI object sets.

UCI object set	Object set size (N)	Dimensionality
IMAGE SEG	210	19
IONOSPHERE	351	34
Red Wine	1599	12
White Wine	4898	12
WBDC	569	30
WISCONSIN	699	10

component of this work, as already discussed in the section 2. The experimental result clearly demonstrates that the proposed technique iLIAC is a superior performer than the CHAMELEON, DBSCAN, k-means and AHC (single and average), and it is slightly similar to CURE technique. Respectively the tested results are validated through our proposed methods GOV and ECVM. The GOV is properly evaluated or identified the normal clusters (NC) and outliers over the resulting clusters of the proposed and traditional techniques that are tested around the same UCI object sets, and the result is obtained in Table 9.

It is clearly noticed from the Table 10 and Table 11, the ECVM is effectively measured the overall intra-cluster similarity (accuracy) and intra-cluster dissimilarity (misclassification error) over the resulting clusters of the proposed and traditional techniques that are tested around the same six UCI object sets as indicated in Table 9. The time complexity notations over the proposed and existing techniques are obtained in the Table 12. According to the experiment results, the proposed technique iLIAC is better suitable for automatically separating the highly relative clusters and outliers over the large object set with higher intra cluster similarity and lesser intra cluster dissimilarity than other existing techniques. Since, the proposed iLIAC is slower than the CURE, CHAMELEON, DBSCAN and k-means techniques, and at a same time it is faster than the traditional AHC (single and average) technique. Hence, the experiment result confirms that the proposed technique iLIAC is better suitable for separating the outliers and normal clusters over the large data set. All techniques are experimented on the Dell/ T4500 machine with 2 GB RAM and running windows7.

Table 8: Results of iLIAC that tested over six UCI object sets comparison with six traditional techniques .

Techniques	Result of UCI object sets					
	IMAGE SEG	IONOSPHERE	Red Wine	White Wine	WBDC	WISCONSIN
AHC (Single)	01	01	01	01	01	01
AHC (Average)	01	01	01	01	01	01
K-Mean	04	04	10	10	15	10
DBSCAN	11	03	16	32	32	26
CHAMELEON	17	04	22	37	39	28
CURE	24	06	30	44	44	31
iLIAC	25	05	30	44	43	32

Table 9: Global Outlier Validation (GOV) measure over the results of proposed iLIAC and comparison with six traditional techniques

Techniques	(GOV) measure over result of six UCI object sets											
	IMAGE SEG		IONOSPHERE		Red Wine		White Wine		WBDC		WISCONSIN	
	NC	Outlier	NC	Outlier	NC	Outlier	NC	Outlier	NC	Outlier	NC	Outlier
AHC (Single)	01	00	01	00	01	00	01	00	01	00	01	00
AHC (Average)	01	00	01	00	01	00	01	00	01	00	01	00
K-Mean	04	00	04	00	10	00	10	00	15	00	10	00
DBSCAN	10	01	03	00	15	01	30	02	28	04	26	00
CHAMELEON	15	02	03	01	20	02	34	03	30	09	28	00
CURE	20	04	04	02	25	05	39	05	34	10	30	01
iLIAC	22	03	04	01	26	04	39	05	34	09	32	00

Table 10: Overall intra-cluster similarity (purity) measure over the result of proposed iLIAC and comparison with six traditional techniques .

Techniques	Purity measures over result of six UCI object sets											
	IMAGE SEG		IONOSPHERE		Red Wine		White Wine		WBDC		WISCONSIN	
	R_C	$A(R_C)$	R_C	$A(R_C)$	R_C	$A(R_C)$	R_C	$A(R_C)$	R_C	$A(R_C)$	R_C	$A(R_C)$
AHC (Single)	01	60.00	01	61.82	01	62.16	01	60.82	01	40.42	01	70.24
AHC (Average)	01	60.00	01	61.82	01	62.16	01	60.82	01	40.42	01	70.24
K-Mean	04	72.5	04	90.00	10	92.89	10	85.64	15	77.56	10	92
DBSCAN	11	96.3	03	100.00	16	98.89	32	90.00	32	86.96	26	99
CHAMELEON	17	97.3	03	98.21	22	99.56	37	95.00	39	93.56	28	99.58
CURE	24	99.00	06	100.00	30	100.00	44	98.85	44	98.25	31	100
iLIAC	25	98.10	05	100.00	30	100.00	44	98.01	43	98.01	32	100

Table 11: Overall intra-cluster dissimilarity (impurity) measures over the results of the proposed iLIAC and comparison with six traditional techniques.

Techniques	Impurity measures over result of six UCI object sets											
	IMAGE SEG		IONOSPHERE		Red Wine		White Wine		WBDC		WISCONSIN	
	R_C	$M_E(R_C)$	R_C	$M_E(R_C)$	R_C	$M_E(R_C)$	R_C	$M_E(R_C)$	R_C	$M_E(R_C)$	R_C	$M_E(R_C)$
AHC (Single)	01	40.00	01	38.17	01	37.83	01	39.17	01	59.57	01	29.75
AHC (Average)	01	40.00	01	38.17	01	37.83	01	39.17	15	59.57	01	29.75
K-Mean	04	27.5	04	10	10	7.11	10	14.36	32	22.44	10	8.00
DBSCAN	11	3.64	03	0.0	16	1.10	32	10	39	13.03	26	1.00
CHAMELEON	17	2.70	03	1.11	22	0.44	37	5	44	6.44	28	0.42
CURE	24	1.00	06	0.0	30	0.00	44	1.15	43	1.78	31	0.0
iLIAC	25	1.98	05	0.0	30	0.00	44	1.98	43	1.98	32	0.0

Table 12: Computational complexity notations over the proposed iLIAC and comparison with six traditional techniques.

Techniques	Computation Complexity Notation
AHC (Single)	$O(n_3)$
AHC (Average)	$O(n_3)$
K-Mean	$O(n^{dk}+logn)$
DBSCAN	$O(nlogn)$
CHAMELEON	$O(nm+nlogn+m^2logm)$
CURE	$O(n_3+nmlogn)$
iLIAC	$O([\frac{n(n-1)}{2}]+1+1)$

8 Conclusion

In this paper, three new techniques namely improved Limited Iteration Agglomerative Clustering (iLIAC), Global Outlier Validation (GOV) and Effective Cluster Validation Method (ECVM) are proposed. The proposed work aims to automatically separate the outliers (irrelevant or error data) and normal clusters over large dataset through the process of identifying the maximum number of highly relative clusters. The first proposed technique iLIAC works with a new threshold (optimum merge cost) that aims to limit the number of iterations and it automatically identifies the maximum number of highly relative clusters and outliers over large data set with higher intra cluster similarity (accuracy) and fewer intra cluster dissimilarity (misclassification errors) and less computational time. The second technique GOV evaluates the global outliers around the result, and the third technique ECVM measures the intra cluster similarity (purity) and intra cluster dissimilarity (impurity) over the resulting cluster of the iLIAC. For the experimentation purpose, we had tested the proposed iLIAC and traditional techniques namely AHC, k-means, DBSCAN, CHAMELEON and CURE with the same six different UCI object sets. Experimental results shows that the proposed technique iLIAC is better suitable for automatically separating the highly relative clusters and outliers over the large object set with higher intra cluster similarity (accuracy) and lesser intra cluster dissimilarity (misclassification) than other existing techniques. Similarly, the validation results reveal that the proposed GOV and ECVM methods are simple and better suitable for evaluating the global outliers, and measuring the intra cluster similarity and dissimilarity over the results of the unsupervised clustering techniques iLIAC, AHC, k-means, DBSCAN, CHAMELEON and CURE. According to the experimental results, it is clear that the newly introduced clustering technique iLIAC is better suitable for identification of normal clusters and outliers over large data set. There are three drawbacks identified from the experimental results. First, the computational complexity is higher than k-means, DBSCAN, CHAMELEON and CURE. Second, the proposed GOV has failed to evaluate the local outliers and third, the proposed ECVM method has failed to measure the inter-cluster similarity over the resulting cluster. The future work involves in solving the above drawbacks and testing with large-scale and high dimensional real datasets.

References

[1] Hawkins D, Identification of Outliers, Chapman and Hall, 1996.

[2] Barnett V and Lewis T, Outliers in Statistical Data, John Willey and Sons, 1994.

[3] Breuning M M, Hans-Peter Kriegel, Ng R T, and Sander J, LOF: identify density based local outliers, ACM SIGMOD International Conference on Management of Data, pp. 93-104, 2000.

[4] Bin-mei Liang, A Hierarchical Clustering Based Global Outlier Detection Method, IEEE 5th International Conference on (BIC-TA), pp. 1213-1215, 2010.

[5] Dajiang Lei, Qingsheng Zhu, Jun Chen, Hai Lin and Peng Yang, Automatic PAM clustering algorithm for outlier detection, Journal of Software, 7(5), pp. 1045-1051, 2012.

[6] CaimingZhong, Duoqian Miao, and PasiFranti, Minimum Spanning tree based split-and-merge: A hierarchical clustering method, Information Sciences, vol. 181, pp. 3397-3410, 2011.

[7] Ham J, and Kamber M, Data Mining: Concepts and Techniques, 2nd edition, Kaufmann, 2006.

[8] George Kollios, Dimitrios Gunopulos, Nick Koudas, and Stefan Berchtold, Efficient Biased Sampling for Approximate Clustering and Outlier Detection in Large Data Sets, IEEE Transaction on Knowledge and Data Engineering 15(5), pp. 1170-1187, 2003.

[9] Martin Ester, Hans-peter Kriegel, Jorgs, Xiaowei Xu, A denisty-based algorithm for discovering cluster in large spatial data base with noise, 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96), pp. 226-231, 1996.

[10] George Karypis, Eui-Hong Han and Vipin Kumar, Chameleon: Hierarchical Clustering Using Dynamic Modeling, Computers, 32(8), pp. 68-75, 1999.

[11] He Zengyou, Xu Xiaofei, Hugang Zhexue Joshua, Deng Shengchun, FP-Outlier: frequent pattern based outlier Detection, Computer Science and Information Systems, 2(1), pp. 103-118, 2005.

[12] He Z Y, Xu X F, and Deng S C, Discovering Cluster Based Local Outliers, Pattern Recognition Letters, 24(9-10), pp. 1641-1650, 2003

[13] Ng R T and Jiawei Han, CLARANS: A Method for clustering objects for spatial data mining, IEEE Transactions on Knowledge and Data Engineering, 14(1), pp. 1003-1016, 2002.

[14] <http://www.ics.uci/mlearn/MLRepository.html>.

[15] Kristine Daniels, and Christophe Giraud-carrier, Learning the Threshold in Hierarchical Agglomerative Clustering, Proceedings of the 5th International Conference on Machine Learning and Applications: 270-276, 2006.

[16] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim, ROCK: A Robust Clustering Algorithm for Categorical Attributes, Information System 25(5), pp. 512-521, 2000.

[17] Dutta M, KakotiMahanta A, and Arun Pujari K, QROCK: A quick version of the algorithm for clustering of categorical data, Pattern Recognition Letter, vol.26, pp. 2364-2373, 2005.
[18] Sudipto Guha, Rajeev Rastogi and Kyuseok Shim, CURE: An Efficient Clustering Algorithm for Large Data bases, Information Systems 26, pp. 35-58, 2001.
[19] Zhang T, Ramakrishnan R and Livny M, BIRCH: an efficient data clustering method for very large databases, International Journal of Windom (Ed.) ACM SIGMOD international conference on management of data, pp. 103-114, 1996.
[20] Seung Kim, Nam Wook Cho, Bokyoung Kang, and Suk-Ho Kang, Fast outlier detection for very large log data, Expert Systems with Applications, 38(8), pp. 9587-9596, 2011.
[21] Sanjay Chawla. Pei Sun, SLOM: A New Measure for Local Spatial Outliers, Knowledge and Information Systems, 9(4), pp. 412-429, 2006.
[22] Shohei Hido, Yuta Tsuboi, Hisashi Kashima, Masahi Sugiyama and Takafumi Kanamori, Statistical outlier detection using direct density ratio estimation. Know. Inf. Syst., 26, pp. 309-336, 2011.
[23] Vijaya P A, NarasimhaMurty M, and Subramanian D K, Leader-Sub leaders: An Efficient Hierarchical Clustering Algorithm for Large data sets, Pattern Recognition Letter 25(4), pp. 505-513, 2004.
[24] Xiaohui Liu, Gongxian Cheng and John Wu X, Analyzing Outliers Cautiously. IEEE Transaction on Knowledge and Data Engineering, 14(2), pp. 432-437, 2002.
[25] Yang P, and Zhu Q S, Finding key attribute subset in dataset for outlier detection, Knowledge-Based Systems, 24(2), pp. 269-274, 2011.
[26] Yong Zhang Su Yang and Yuanyuan Wang, LDBOD: A novel local distribution based outlier detector, Pattern Recognition Letters, 29(7), pp. 967-976, 2008.
[27] William H E Day and Herbert Edelsbrunner, Efficient Algorithms for Agglomerative Hierarchical Clustering Methods, Journal of Classification, 1, pp. 7-24, 1984.
[28] Jeen-Shing Wang and Jen-Chieh Chiang, A Cluster Validity Measure with Outlier Detection for Support Vector Clustering, IEEE Transaction on System, Man and Cybernetics-Part B: Cybernetics, 38(1), pp. 78-89, 2008.
[29] Yue Zhang, Jie Liu and Hang Li, An Outlier Detection Algorithm Based on Clustering Analysis, International Conference on (PCSPA), pp. 1126-1128, 2010.
[30] Manoranjan Dash, Huan Liu, Peter Scheuerman and Kian Lee Tam, Fast hierarchical clustering and it?s validation. Data and Knowledge Engineering, 44(1), pp. 109-138, 2003.
[31] Rui XU and Donald Wunsch II C, Clustering. IEEE PRESS, 2009.
[32] Jianhua Yang and Ickjai Lee, Cluster validity through graph-based boundary analysis. International Conference on Information and Knowledge Engineering, pp. 204-210, 2004.

Krishnamoorthy R received the M.Sc. degree in Mathematics from Madurai Kamaraj University, Madurai, India in 1983 and the M.Tech. and Ph.D. degrees in Computer Science and Engineering from IIT , Kanpur, India in 1992 and IIT, Kharagpur, India in 1997 respectively. From 2001 to 2006 , he was worked Professor in the Computer Science and Engineering in Annamalai University, Tamilnadu, India. During 2006 to 2014 he worked as a Dean and Professor in the Computer Science and Engineering, Anna University Chennai, BIT Campus, Tiruchirappalli, India. Currently he is working as a Professor in the Department of Information Technology, Anna University Chennai, BIT Campus, Tiruchirappalli, India, since 2014. He was worked Principal Investigator in many funded Government research projects. He has published 29 International Journals, 48 International Conference and 7 Books. His current research interest includes Biometric, Computer Vision and Image Processing, Software Testing, Database, and Web Mining, Video Segmentation, Image Retrieval and Image Cryptography, Matching Process, Image Classification and Medical Image Process.

Sreedhar Kumar S received the B.E. degree in Computer Science and Engineering from Bharathidasan University, Tiruchirappalli, Tamilnadu, India in 2000 and the M.E. degree in Computer Science and Engineering from Annamallai University, Tamilnadu, India in 2006. He has 14 years experience in teaching and research, and currently Pursuing Ph.D., in Department of Computer Science and Engineering from Anna University Chennai, BIT Campus, Tiruchirappalli, Tamilnadu, India. Currently he is working as Associate Professor in the Department of Computer Science and Engineeringt, Don Bosco Institute of Technology, Bangalore, India, since 2013. He has published 5 International Journals, 5 International Conferences and 4 National Conferences. His current research includes Data Mining Concepts, Validation Techniques, Machine Learning, Biometric, Image Segmentation, Image Mining, Matching Process, Image Classification and Medical Image Process.