academicJournals

Vol. 9(4), pp. 30-41, December 2017
DOI: 10.5897/JETR2017.0628
Article Number: 46C8B8967119
ISSN 2006-9790

http://www.academicjournals.org/JETR

Journal of Engineering and Technology Research

Full Length Research

An improved frequency based agglomerative clustering algorithm for detecting distinct clusters on two dimensional dataset

Madheswaran M.[1]* and Sreedhar Kumar S.[2]

[1]Department of Electronics and Communication Engineering (ECE), Mahendra Engineering College, Mallasamudram-637503, Tamilnadu, India.
[2]Department of Computer Science and Engineering (CSE), KS School of Engineering and Management, Bangalore-560062, India.

Received 12 July, 2017; Accepted 11 October, 2017

In this study, a frequency based Dynamic Automatic Agglomerative Clustering (DAAC) is developed and presented. The DAAC scheme aims to automatically identify the appropriate number of divergent clusters over the two dimensional dataset based on count of distinct representative objects with higher intra thickness and lesser intra separation. The Distinct Representative Object Count (DROC) is introduced to automatically trace the count of distinct representative objects based on frequency of object occurrences. It also identifies the distinct number of highly comparative clusters based on the count of distinct representative objects through sequence of merging process. Experimental result shows that the DAAC is suitable for instinctively identifying the K distinct clusters over the different two dimensional datasets with higher intra thickness and lesser intra separation than existing techniques.

Key words: Dynamic automatic agglomerative clustering, clusters, intra thickness, intra separation, distinct representative object count.

INTRODUCTION

Agglomerative hierarchical clustering is an unsupervised clustering technique to cluster the dataset into a hierarchical tree structure form through a sequence of merging based on similarity metrics (Han and Kamber, 2006). In recent years, this clustering approach is applied to Machine Learning, Pattern Recognition, Data Mining, Text Mining, Spatial Data Base Application, Web Application, Dig Data, Image Analysis, Information Retrieval and Bioinformatics (Douglass et al., 1992; Martin et al., 2000; Cadez et al., 2001; Fogs et al., 2001). In general, the agglomerative hierarchical clustering scheme is classified into divisive and agglomerative categories (Pakhira, 2009; Jain, 2010; Jain et al., 1999; Frigui and Krishnapuram, 1997). The divisive method continuously divides the dataset into smaller clusters until each cluster consists of a single object.

*Corresponding author. E-mail: madheswaran.dr@gmail.com.

The agglomerative technique starts with n clusters, each containing exactly one data object. Afterward, it follows a series of merging operations that ultimately forces all clusters into the same single cluster.

The limitation in the existing agglomerative clustering techniques is the identification of the predetermined number of distinct clusters over the large dataset and the entire result quality is based on the number of clusters which is predetermined by user. In this paper, a Dynamic Automatic Agglomerative Clustering (DAAC) is proposed to automatically identify appropriate number of discrete clusters in the two dimensional dataset based on count of distinct representative objects in the dataset without user input.

RELATED WORK

Here, literatures related to the present clustering scheme are presented. Some of the popular traditional agglomerative clustering techniques UPGMA, WARDS, SLINK, CLINK and PNN were designed to identify the distinct number of clusters over the dataset based on similarity measures. A simple agglomerative hierarchical clustering scheme called Unweighted Pair Group Method with Arithmetic Mean (UPGMA) was reported by Murtagh (1984). This method constructs a rooted tree to reflect the structure present in a pair wise similarity matrix. At each step, the nearest two clusters are combined into a higher level cluster. The distance between any two clusters is taken to be the average of all distances between pairs of, that is, the mean distance between elements of each cluster.

Fionn and Legendre (2014), reported a general agglomerative clustering technique with minimum variance method. In this method, each step finds a pair of clusters that can lead to minimum increase in total within-cluster variance after merging. This increase is weighted square distance between cluster centers. Another technique namely Ward *p* was reported by De Amorim (2015), as an improved version of Ward's method. This method uses subspace feature weighting to take into consideration the different degrees of relevance of each feature. Sibson (1973) reported a single linkage (SLINK) method for grouping clusters in bottom-up fashion, which at each step combines two clusters that enclose the closest pair of objects not yet belonging to the same cluster as each other.

Defays (1977) reported an agglomerative clustering technique complete linkage (CLINK) method. In this method, initially, each object is considered to be a cluster of its own and the clusters are serially combined into larger clusters until all objects end up within the same cluster. At each step, two clusters that are separated by the shortest distance are combined. Franti et al. (2000) reported a fast and memory efficient implementation of the exact Pair-wise Nearest Neighbor (PNN) technique. It is claimed that this technique could improve the results with reduced memory and computational complexity of exact PNN technique. The fast agglomerative clustering using k-nearest neighbor graph scheme was reported by Chih-Tang et al. (2010). This scheme is intended to reduce the number of distance calculation and time complexity for identifying the distinct number of clusters in the dataset.

Recently, some popular agglomerative clustering techniques called DKNNA, KnA, NNB, etc., identify the distinct number of clusters over the dataset and reduce the computational complexity. Lai and Tsung-Jen (2011) presented a hierarchical clustering technique called Dynamic K-Nearest Neighbor Algorithm (DKNNA). This scheme is used to identify the distinct number of clusters based on k-nearest neighbor graph to reduce the number of distance calculations and time complexity. The advantage of this approach is that it is faster and simultaneously produces better clustering result than Double Linked Algorithm (DLA) and Fast Pair-wise Nearest Neighbor (FPNN) techniques. Qi et al. (2015) reported an agglomerative hierarchical clustering to construct a cluster hierarchy based on a group of centroids. It followed a group of centroids instead of raw data points to build cluster hierarchies, where centroid was indicated as a group of adjacent points in the data space. The authors claimed that this approach reduced the computational cost without compromising clustering performance.

Another approach, Nearest Neighbor Boundary (NNB) to reduce the time and space complexity of standard agglomerative hierarchical clustering based on nearest neighbor search was designed by Wei et al. (2015). First, it divided the dataset into independent subsets and then groups the closest data points together among each of the individual subset based on nearest neighbor search. Afterward, it joins the closest subsets based on nearest data points in the boundary between the subsets. The authors declared that the merit of their method was that it consumed lower space and computational complexity for grouping the nearest data points. Lin and Chen (2005) reported a two phase clustering algorithm called Cohesion-based Self Merging (CSM). The first phase, it partitioned the input dataset into several small sub-clusters and in the second phase, it continuously merged the sub-clusters based on cohesion in a hierarchical way. This CSM approach is claimed to be robust and possesses excellent tolerance to outlier in various datasets. The detail of the DAAC algorithm is presented in the next section.

PROPOSED APPROACH

Here, a detail of the DAAC approach is presented. It consists of two stages DROC and clustering. In the Distinct Representative Object Count (DROC) stage, the approach traces the count of distinct representative objects over the input dataset based on occurrence of each individual object in the dataset. In the clustering stage, it partitions the input dataset into maximum number of discrete

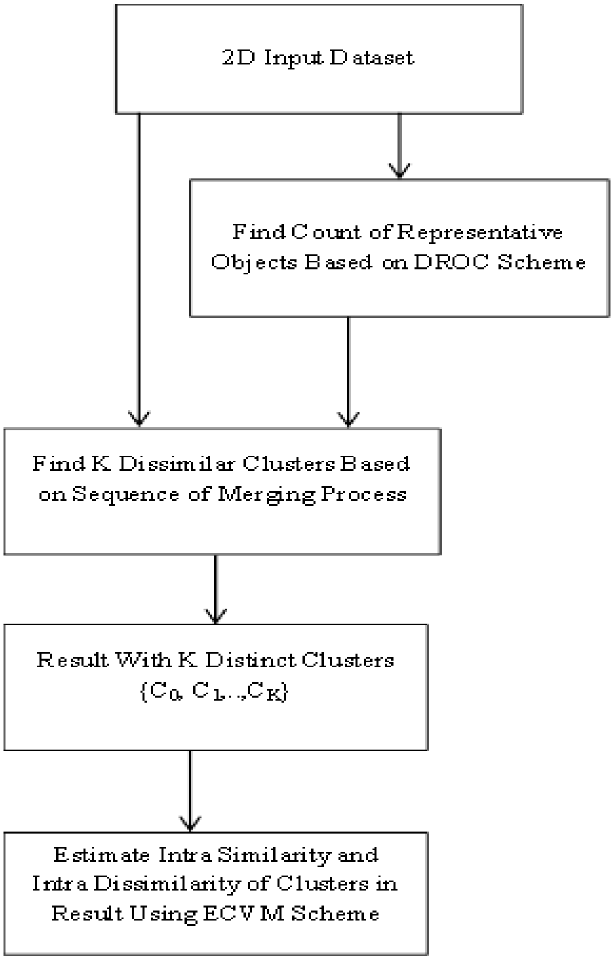

Figure 1. Functional diagram of proposed approach.

clusters based on count of distinct representative objects. The stages are involved in the DAAC approach as shown in Figure 1.

DROC Stage

This stage aims to trace the count of distinct representative objects over the two dimensional dataset. It consists of three steps. In the first step, it represents each of the object in the dataset $X = x_i$ for $i = 1,2,.., n$ with D features $f = 0,1,.., D$ into single dimensional $\overline{X} = \bar{x}_i$ based on a statistical mean operation and is defined in Equation 1 as:

$$\overline{X}_i = \left\{ \sum_{i=1}^{n} \frac{1}{D} \sum_{f=1}^{D} x_{if} \;\middle|\; \forall x_{if} \in x_i,\ \forall x_i \in X \right\} \quad (1)$$

where x_{if} represents the f^{th} feature in i^{th} object that belongs to the input dataset X. In the second step, the proposed DROC scheme measures the count of each object occurrence $COO(X_i)$ in dataset $\overline{X} = \bar{x}_i$, for $i = 0,.., n$ and is defined in Equation 2 as:

$$COO(\overline{X}_i) = \sum_{j=i+1}^{n} \left|\bar{x}_i - \bar{x}_j\right| \;\Bigg|\; \forall \bar{x}_i, \bar{x}_j \in \overline{X},\ where \begin{cases} 1 & \left|\bar{x}_i - \bar{x}_j\right| < T \\ 0 & \left|\bar{x}_i - \bar{x}_j\right| > T \end{cases} \quad (2)$$

where $\bar{x}_i$ and $\bar{x}_j$ represent i^{th} and j^{th} object that belong to the input dataset X , n denotes the size of $\overline{X}$ and T is the external parameter (threshold) which predetermined by user which used to limit the dissimilarity difference between i^{th} and j^{th} objects. If the difference of i^{th} and j^{th} objects is lesser than T , it means the j^{th} object is similar to i^{th} object that belongs to the dataset $\overline{X}$. The predetermined value of T could contrast based on dataset nature. Final step, it estimates the count of distinct representative objects over the dataset $\overline{X}$ based on maximum occurrence of objects in dataset X and is defined in Equation 3 as:

$$K = \left\{ \sum_{i=1}^{n} COO_i \;\middle|\; \forall COO_i \in COO, \begin{cases} 1 & COO_i >= MO \\ 0 & COO_i < MO \end{cases} \right\} \quad (3)$$

Here, COO_i denotes the count of occurrence of i^{th} object in X and MO represents the maximum occurrence threshold that limits the count of K distinct representative objects with maximum occurrence over the X . For instance, if the MO is too small, a large numbers of clusters are generated as the final result. On the other hand, if the MO is too large, only lesser numbers of clusters are generated.

Clustering stage

In the clustering stage, it first, computes the upper triangular distance matrix Ud_{ij} for input cluster set $X = x_i$ for $i = 1,2,.., n$ through Euclidean distance metric as calculated by

$$Ud_{ij} = \left\{ \begin{array}{l} d(x_i, x_j) \\ i = 0,1,.., n \\ j = i+1,.., n \end{array} \;\middle|\; \forall x_i, x_j \in x \right\} \quad (4)$$

where n denotes the number of clusters in the input cluster set X and $d(x_i, x_j)$ is the Euclidean distance between i^{th} and j^{th} clusters in the cluster set X for $i = 1,.., n$ and are computed as:

$$d(x_i, x_j) = \left\{ \left(\sum_{f=0}^{D} \left(x_{if} - x_{jf}\right)^2 \right)^{1/2} \right\} \quad (5)$$

Here, x_{il} denotes the f^{th} feature in the i^{th} cluster that belongs to the cluster set X and D represents the number of features in cluster $x_i = x_{il}$ for $f = 1,2,..,D$. Next, the proposed scheme traces the closest clusters pair (x_i, x_j) with minimum merging cost φ on the upper triangular distance matrix Ud_{ij} and is then computed as follows:

$$\varphi = \underset{\substack{i=0,1,2,..,n,\\ j=i+1,..,n}}{Min} \{d(x_i, x_j) \mid \forall\, d(x_i, x_j) \in Ud_{ij}, \forall x_i, x_j \in X\} \qquad (6)$$

Equation 6 finds the closest clusters pair (x_i, x_j) with minimum merge cost φ and then compares the number of clusters not exceeding the count of representative objects as described earlier. If the number of clusters i does not exceed the K, then the closest cluster pair (x_i, x_j) is merged into a single cluster x_{ij} which subsequently computes the centroid over the new cluster x_i using Equation 7 and is defined as:

$$x_{if} = \left\{ \sum_{f=1}^{D} \frac{1}{2}\left(x_{if} + x_{jf}\right) \middle| \forall x_{if} \in x_i,\, x_{jf} \in x_j \right\} \qquad (7)$$

Next, updates the merged cluster x_i status into respective c_i through $c_i \cup c_j \rightarrow c_i$, where c_i denotes the status of the i^{th} cluster and subsequently it modifies the size of merged cluster x_i by $N_i \cup N_j \rightarrow N_i$, where N_i and N_j represent the number of related objects in i^{th} and j^{th} clusters, respectively. After, it deletes the j^{th} cluster in the input cluster set X including its status C_j and size N_j respectively and reduces the input cluster set size by one. This process is repeated until the size of the cluster set is equal to K and afterward the results with K district clusters are denoted as $\{c_1, c_2,..,c_K\}$. This stage involved in the proposed DAAC technique is presented as an algorithm hereunder.

Algorithm

Input: Dataset X containing n objects $x_0,.x_1,..,x_n$ with D features and Threshold MO

Output: Generate K Distinct Clusters $\{c_1, c_2,..,c_K\}$

Begin

(1) Represent each object in dataset $X = x_i$ into single value $\bar{X} = \bar{x}_i$ using Equation 1

(2) Measure the count of occurrence of each individual object $COO(\bar{X}_i)$ in $\bar{X} = \bar{x}_i$ for $i = 0,1,2,..,n$ as described in Equation 2

(3) Identify representative objects in X based on count of object occurrences $COO(\bar{X}_i)$ and threshold MO as described in Equation 3.

(4) Count (sum) the distinct representative objects in X using Equation 3 and obtain the count in K

(5) Consider each object as an individual cluster in the input dataset $X = x_i$ for $i = 1,2,..,n$

(6) Compute the upper triangular matrix Ud_{ij} as given in Equation 4.

(7) Find the adjoining clusters pairs (x_i, x_j) with lowest merge cost φ over Ud_{ij} as given in Equation 6

(8) Merge the closest cluster pairs (x_i, x_j) as a single cluster x_{ij}

(9) Update the newly merged cluster x_{ij} into x_i as described in the clustering stage

(10) Update the status of newly merged cluster x_i in c_i by $c_i \cup c_j \rightarrow c_i$

(11) Update the size of newly merged cluster by $N_i \cup N_j \rightarrow N_i$

(12) Delete j^{th} cluster, cluster status (c_j) and its size (N_j) respectively.

(13) Reduce dataset X size by one.

(14) Repeat steps 6 to 13 until the size of the cluster set n is equal to K

(15) Obtain the final clustering result in C

End

Complexity analysis

Complexity analysis discusses in detail the computational complexity of the proposed approach. The first stage in the proposed approach requires time $O(n-K)$ to search the count of K distinct representative objects over the input dataset $X = x_i$ for $i = 1,2,..,n$, where n represents the size of the dataset X and K is the count of distinct number of representative objects in dataset X. The second stage in the proposed approach, consumes time $O(n^2)$ to trace the K distinct clusters $C = c_l$ for $l = 1,2,..K$ in dataset X, where C denotes the resulting cluster. Overall, the proposed approach requires time $O((n-K) + n^2)$ to partitions the input dataset X into maximum K distinct highly relative clusters.

Cluster validation

Cluster validation presents the result of Dynamic Automatic Agglomerative Clustering scheme validated based on Effective Cluster Validation Method (ECVM) scheme reported by Krishnamoorthy and Sreedhar (2016). The ECVM scheme is slightly modified and to estimate the intra tightness and intra separation among the objects with D features within each individual cluster in cluster set of DAAC scheme. It contains two measures: Intra Cluster Similarity and Intra Cluster Dissimilarity that are described subsequently.

Intra cluster similarity measure

This measure computes the intra similarity among the each individual cluster in the cluster set $C = c_l$ of DAAC approach for $l = 0,1,..,K$. This method is expressed in the equation.

$$ICS(C) = \left\{ \frac{1}{K} \sum_{l=0}^{K} It_l \;\middle|\; \forall It_l \in It \right\} \tag{8}$$

where K represents the distinct clusters in C for $l = 1,..,K$, It_l is the intra tightness measures of l^{th} individual cluster in C and is defined in Equation 9:

$$It_l = \left\{ \begin{array}{l} \left(\left(\frac{1}{N_l} \sum_{j=0}^{N_l} \sum_{f=0}^{D} \left| c_{ljf} - \alpha_{lf} \right| \right) \times 100 \right) \forall\ \alpha_{lf} \in \alpha_l, c_{ljf} \in c_{lj}, \\ where \begin{cases} 1 & \left| c_{ljf} - \alpha_{lf} \right| \le T_1 \\ 0 & \left| c_{ljf} - \alpha_{lf} \right| > T_1 \end{cases}, \begin{cases} 1 & \sum_{f=0}^{D} \left| c_{ljf} - \alpha_{lf} \right| \ge T_2 \\ 0 & \sum_{f=0}^{D} \left| c_{ljf} - \alpha_{lf} \right| > T_2 \end{cases} \end{array} \right\} \tag{9}$$

Here, c_{ljf} denotes f^{th} value in j^{th} object that belongs to the l^{th} cluster in C, N_l represents the size of l^{th} cluster in cluster set C, T_1 is the similarity distance threshold which predetermined by user based on cluster set and it is used to trace higher closeness features among the j^{th} object of l^{th} cluster and centroid point of l^{th} cluster, T_2 is the predetermined similarity distance threshold used to identify higher similarity objects in l^{th} cluster in the cluster set C and α_l is the centroid point of l^{th} cluster and is expressed in Equation 10:

$$\alpha_l = \left\{ \frac{1}{N_l} \sum_{j=0}^{N_l} c_{lj} \right\} \tag{10}$$

Intra cluster dissimilarity measure

It intentions to calculate the intra separation among the each individual cluster in the cluster set of DAAC approach. This measure is defined in Equation 11.

$$ICD(C) = \left\{ \frac{1}{K} \sum_{i=0}^{K} Is_l \;\middle|\; \forall Is_l \in Is \right\} \tag{11}$$

where N represents the number of clusters in the cluster set C for $l = 1,..,K$, Is_l denotes s the intra separation measure of l^{th} individual cluster in C and is defined in Equation 12:

$$IS_l = \left\{ \begin{array}{l} \left(\left(\frac{1}{N_l} \sum_{j=0}^{N_l} \sum_{f=0}^{d} \left| c_{ljf} - \alpha_{lf} \right| \right) \times 100 \right) \forall\ \alpha_{lf} \in \alpha_l, c_{ljf} \in c_{lj}, \\ \begin{cases} 0 & \left| c_{ljf} - \alpha_{lf} \right| \le T_1 \\ 1 & \left| c_{ljf} - \alpha_{lf} \right| > T_1 \end{cases}, \begin{cases} 1 & \sum_{f=0}^{d} \left| c_{ljf} - \alpha_{lf} \right| < T_2 \\ 0 & \sum_{f=0}^{d} \left| c_{ljf} - \alpha_{lf} \right| \ge T_2 \end{cases} \end{array} \right\} \tag{12}$$

here, c_{ljf} denotes f^{th} value in j^{th} object that belongs to the l^{th} cluster in C, T_1 is the predetermined dissimilarity threshold used to identify higher divergence features among the j^{th} object of l^{th} cluster and centroid point of l^{th} cluster, and T_2 is the predetermined dissimilarity threshold used to detect higher contrast objects in l^{th} cluster in the cluster set C.

RESULTS AND DISCUSSION

The DAAC scheme experimented with more than 100 2D UCI datasets of different sizes is presented here. Among these 100 2D UCI datasets, a subset of nine sample benchmark datasets (http://www.archive.ics.uci.edu/ml/), viz. White-Wine, Image-Seg, Heart-Diseases, Red-Wine, WBDC, Wisconsin, Iris and Wine including its size and dimensional are presented in Table 1.

The DROC method traces count of distinct representative objects of nine datasets with three different MO's as 5, 10, and 15, respectively and the computed results are obtained in Table 2. For the MO value of 5, that the DROC is identified K distinct representative objects over the nine UCI datasets of 43, 13, 17, 19, 23, 12, 5, and 12, respectively and the results are presented in Table 2. Similarly, the DROC found count of K distinct objects of MO's values 10 and 15 in same UCI datasets as 40, 6, 8, 17, 17, 9, 5, 7 and 36, 3, 7, 16, 10, 7, 2, 2.

Then the clustering process is followed and partitions the datasets into K discrete clusters based on sequence of merging process with distance metric. The DAAC clustering scheme has produced three different clustering results on nine UCI datasets based on count of representative objects of three MO's {5, 10, 15} which are obtained in Table 2. The results of DAAC scheme with three MO's are incorporated in Table 3.

Next, the three different results of these nine sample UCI datasets are validated based on ECVM scheme. Initially, the intra closeness (It_l) and intra separation (Is_l) are computed among the each individual cluster in the results of UCI datasets in percentage as expressed in Equations 8 and 11. The estimated measures of these three clustering results of DAAC scheme with three MO's

Table 1. Description of sample UCI datasets.

UCI dataset	Dataset size	Number of features
White-Wine	4898	12
Image-seg	210	19
Heart-Diseases	297	13
Red-Wine	1599	12
WBDC	569	30
Wisconsin	699	10
Iris	150	04
Wine	178	13

Table 2. DROC scheme tested on UCI dataset with different MO's.

UCI Datasets	Identified distinct representative objects with various MO's		
	MO=5	MO=10	MO=15
White-Wine	43	40	36
Image-seg	13	06	03
Heart-Diseases	17	08	07
Red-Wine	19	17	16
WBDC	23	17	10
Wisconsin	12	09	07
Iris	05	05	02
Wine	12	07	02

Table 3. DAAC scheme tested on UCI dataset with different MO's.

UCI Datasets	Number of clusters identified on dataset		
	DAAC with (MO=5)	DAAC with (MO=10)	DAAC with (MO=15)
White-Wine	43	40	36
Image-seg	13	06	03
Heart-Diseases	17	08	07
Red-Wine	19	17	16
WBDC	23	17	10
Wisconsin	12	09	07
Iris	05	05	02
Wine	12	07	02

are presented in Tables 4 to 6, respectively.

Thereafter, the overall intra closeness measure (ICS) in percentage is estimated over the three different results of nine UCI datasets as 99.0, 91.7, 93.9, 96.9, 92.5, 92.5, 99.92, 82.46; 98.79, 82.22, 84.28, 95.76, 90.33, 85.09, 99.72, 66.70 and 98.79, 77.06, 73.10, 93.19, 86.75, 80.72, 88.0, 53.67, respectively. The estimated results are incorporated in Tables 7 to 9, respectively. Similarly, the overall intra separation (ICD) in percentage is calculated on three different results of sample eight UCI datasets Image_Seg, Wine, Red_Wine, White_Wine, WBDC, Wisconsin, Heart-Diseases and Iris as 0.99, 8.20, 6.009, 3.02, 17.49, 7.34, 0.27, 17.53; 1.20, 17.7, 15.71, 4.2, 9.66, 14.90, 0.27, 33.29 and 1.20, 22.93, 26.89, 6.80, 14.24, 19.27, 12.0, 46.32, respectively. The estimated measures of these three clustering results of eight UCI datasets are presented in Tables 7, 8, and 9.

The experiments are conducted for nine UCI sample datasets with different MO's values and the validation results are obtained with the proposed DAAC scheme as illustrated in Figure 2a, b and c respectively. It is clearly

Table 4. Result of intra cluster validation obtained with ECVM scheme on results of DAAC scheme with (MO=5).

UCI dataset	Intra cluster validation of DAAC with (MO=5) (%)	
	Intra tightness Measure (It_l)	Intra separation measure (Is_l)
White-Wine	95.75,96.83,92.30,100,98.48,94.42,98.4375,100,100,100,97.58,98.79,99.55,88.88,96.09,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100	4.24,3.16,7.69,0.0,1.51,5.57,1.5625,0.0,0.0,0.0,2.41,1.209,0.44,11.11,3.90625,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
Image-seg	100,93.33,100,100,0.0,100,100,100,100,100,100,100,100	0.0,6.66,0.0,0.0,100,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
Heart-Diseases	79.66,100,75.92,85.0,73.91,83.33,100,100,100,100,100,100,100,100,100,100,100	20.33,0.0,24.07,15.0,26.08,16.66,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
Red-Wine	87.5,98.36,93.24,87.21,100,100,98.24,86.20,91.66,100,100,100,100,100,100,100,100,100,100	12.5,1.63,6.75,12.78,0.0,0.0,1.75,13.79,8.333333333333332,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
WBDC	100,100,2.31,100,100,100,57.14,100,100,100,100,100,100,100,100,90.0,87.14,90.90,100,100,100,100,100	0.0,0.0,97.68,0.0,0.0,0.0,42.85,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,10.0,12.85,9.09,0.0,0.0,0.0,0.0,0.0
Wisconsin	77.89,100,42.30,91.66,100,100,100,100,100,100,100,100	22.,0.0,57.69,8.33,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
Iris	100,100,98.61,100,100	0.0,0.0,1.38,0.0,0.0
Wine	53.33,83.33,100,60.60,92.30,100,85.71,100,61.53,70.37,82.35,100	46.66,16.66,0.0,39.39,7.69,0.0,14.28,0.0,38.46,29.62,17.64,0.0
White-Wine	95.75,96.83,92.30,100,98.48,94.42,98.4375,100,100,100,97.58,98.79,99.55,88.88,96.09,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100	4.24,3.16,7.69,0.0,1.51,5.57,1.5625,0.0,0.0,0.0,2.41,1.209,0.44,11.11,3.90625,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0

indicated in Tables 3, 4 and 5, respectively that the DAAC scheme automatically identified the K distinct highly relative clusters with higher ICS and lower ICD based on K distinct representative objects without user input. It is demonstrated in the experimental results, that the MO acts as a major key element in the proposed clustering scheme and directly affects the performance of the proposed scheme.

Comparison with existing schemes

Here, the result of the DAAC approach is compared to existing schemes DKNNA (Lai and Tsung-Jen, 2011) and KnA (Qi et al., 2015). For comparison purposes, these existing schemes are implemented and tested over the same seven UCI datasets.

The existing schemes are tested over the seven UCI datasets and subsequently these results are incorporated in Table 10. Similarly, the performance measures intra cluster similarity and intra cluster dissimilarity are estimated over the results of existing schemes based on ECVM technique. The estimated results are incorporated in the Tables 11 and 12. The overall performance measures shown in Tables 10 to 12 reveal that the DAAC scheme has produced better results with higher intra cluster similarity, lower intra cluster dissimilarity and limited number of iterations compared to existing techniques DKNNA and KnA. Based on the experimental results and performance measures, it is found that the existing techniques identified predetermined number of distinct clusters. The technique DKNNA is identified M number of dissimilar clusters around the dataset, where M is the number of clusters which determined by user. Similarly, the KnA scheme follows two predetermined parameters k and M respectively, where k is the number of predetermined distinct partitions. The comparison results reveal that the DAAC scheme

Table 5. Intra Cluster Validation obtained with ECVM scheme on result of DAAC scheme with (MO=10).

UCI dataset	Intra cluster validation of DAAC with (MO=10) (%)	
	Intra tightness measure (It_l)	**Intra separation measure** (Is_l)
White-Wine	95.75,96.83,92.30,94.74,98.48,94.42,98.43,100,100,97.58,98.79,99.55,88.88,96.09375,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100	4.24,3.16,7.69,5.25,1.51,5.57,1.5625,0.0,0.0,2.41,1.20,0.44,11.11,3.90625,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
Image-seg	100,93.3,100,100,0.0,100	0.0,6.66,0.0,0.0,100,0.0
Heart-Diseases	40.0,54.248366013071895,100,100,80.0,100,100,100	60.0,45.751633986928105,0.0,0.0,20.0,0.0,0.0,0
Red-Wine	66.63,98.36,87.21,100,99.59,98.24,86.20,91.66,100,100,100,100,100,100,100,100,100	33.36,1.63,12.78,0.0,0.406,1.75,13.79,8.33,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
WBDC	100,100,1.36,100,100,100,57.14,100,100,100,100,100,100,100,100,90.0,87.14	0.0,0.0,98.63,0.0,0.0,0.0,42.85,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,10.0,12.85
Wisconsin	77.89,61.11,26.82,100,100,100,100,100,100	22.10,38.88,73.17,0.0,0.0,0.0,0.0,0.0,0.0
Iris	100,100,98.61,100,100	0.0,0.0,1.38,0.0,0.0
Wine	6.38,60.60,85.71,61.53,70.37,82.35,100	93.61,39.39,14.28,38.,29.62,17.64,0.0
White-Wine	95.75,96.83,92.30,94.74,98.48,94.42,98.43,100,100,97.58,98.79,99.55,88.88,96.09375,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100	4.24,3.16,7.69,5.25,1.51,5.57,1.5625,0.0,0.0,2.41,1.20,0.44,11.11,3.90625,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0

Table 6. Result of intra cluster validation obtained with ECVM scheme on results of DAAC Scheme with (M0=15).

UCI dataset	Intra cluster validation of DAAC with (MO=15) (%)	
	Intra tightness measure (It_l)	**Intra separation measure** (Is_l)
White-Wine	95.53,97.55,98.01,95.88,95.90,96.74,99.09,100,100,100,99.56,100,91.48,100,100,100,100,97.14,100,100,100,100,96.77,100,100,100,100,100,100,100,100,100,100,100,100,100	4.46,2.44,1.98,4.11,4.09,3.25,0.90,0.0,0.0,0.0,0.43,0.0,8.51,0.0,0.0,0.0,0.0,2.85,0.0,0.0,0.0,0.0,3.22,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
Image-seg	100,93.3,100	0.0,6.666,0.0
Heart-Diseases	40.0,54.248366013071895,37.5,80.0,100,100,100	60.0,45.751633986928105,62.5,20.0,0.0,0.0,0
Red-Wine	66.63,98.36,87.21,100,99.59,61.44,86.20,91.66,100,100,100,100,100,100,100,100	33.36,1.63,12.78,0.0,0.40,38.55,13.79,8.33,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
WBDC	100,100,0.44,100,100,100,57.14,100,100,100	0.0,0.0,99.55,0.0,0.0,0.0,42.85,0.0,0.0,0
Wisconsin	77.14,61.11,26.82,100.0,100.0,100.0,100.0	22.85,38.88,73.17,0.0,0.0,0.0,0.0
Iris	100.0,76.0	0.0,24.0
Wine	7.34,100.0	92.65,0.0
White-Wine	95.53,97.55,98.01,95.88,95.90,96.74,99.09,100,100,100,99.56,100,91.48,100,100,100,100,97.14,100,100,100,100,96.77,100,100,100,100,100,100,100,100,100,100,100,100,100	4.46,2.44,1.98,4.11,4.09,3.25,0.90,0.0,0.0,0.0,0.43,0.0,8.51,0.0,0.0,0.0,0.0,2.85,0.0,0.0,0.0,0.0,3.22,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0

Table 7. Performance measures of the result of DAAC scheme when (MO=5).

UCI dataset	Number of clusters	Result of cluster validation (%)	
		ICS (C)	ICD (C)
White-Wine	43	99.0	0.99
Image-seg	13	91.7	8.20
Heart-Diseases	17	93.9	6.009
Red-Wine	19	96.97	3.025
WBDC	23	92.5	7.49
Wisconsin	12	92.65	7.34
Iris	05	99.72	0.27
Wine	12	82.46	17.53

Table 8. Performance measures of the result of DAAC scheme when (MO=10).

UCI dataset	Number of clusters	Result of cluster validation (%)	
		ICS(C)	ICD(C)
White-Wine	40	98.79	1.202
Image-seg	06	82.22	17.7
Heart-Diseases	08	84.28	15.71
Red-Wine	17	95.76	4.2
WBDC	17	90.33	9.66
Wisconsin	09	85.09	14.90
Iris	05	99.72	0.27
Wine	07	66.70	33.29

Table 9. Performance measures of the result of DAAC scheme when (MO=15).

UCI dataset	Number of clusters	Cluster validation (%)	
		ICS(C)	ICD(C)
White-Wine	36	98.79	1.202
Image-seg	03	77.06	22.93
Heart-Diseases	07	73.10	26.89
Red-Wine	16	93.19	6.80
WBDC	10	86.75	14.24
Wisconsin	07	80.72	19.27
Iris	02	88.0	12.0
Wine	02	53.67	46.32

produced much better results with higher intra cluster similarity, lower intra cluster dissimilarity and maximum number of clusters identified, compared to existing cluster techniques.

Conclusion

A simple two stage Dynamic Automatic Agglomerative Clustering scheme that could robotically produce clusters for two dimensional dataset is proposed in this paper. In the first stage, the DAAC scheme traces the count of distinct representative objects over the input dataset based on DROC method. In the second stage, a distance based clustering process instinctively partitions the input dataset into K discrete clusters based on count of distinct representative objects. The novelty of the DAAC is the automatic production of distinct number of dissimilar clusters, which is a contradiction to the existing schemes, where it is a user input. The DAAC can be better utilized as a pre-process to determine the maximum number of discrete clusters with higher intra similarity and be

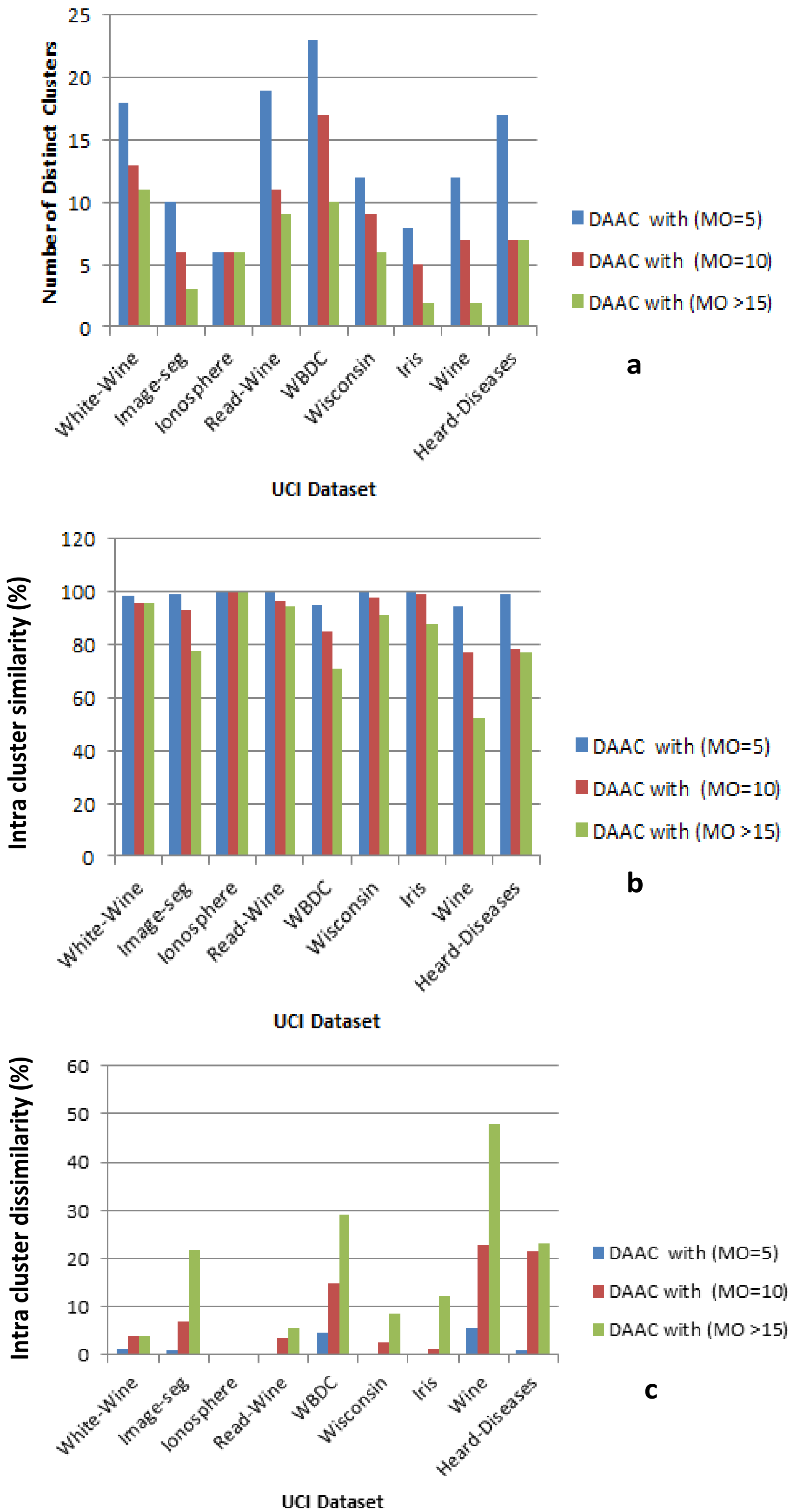

Figure 2. Comparison results of DAAC scheme with different MO's tested on UCI datasets. (a) Comparison of resulting clusters of DAAC with various MO. (b) Comparison of intra similarity measure on results of DAAC with various MO's. (c) Comparison of intra dissimilarity measure on results of DAAC with various MO's.

Table 10. Comparison of result obtained with DAAC and existing schemes tested on UCI dataset.

UCI dataset	Number of clusters identified on datasets				
	DKNNA (Lai, 2011)	KnA (Qi, 2015)	DAAC scheme		
			MO>5	MO>10	MO>15
White-Wine	36	19	43	40	36
Wisconsin	09	06	12	09	07
Iris	02	03	05	05	02
Wine	07	06	12	07	02
Red_Wine	16	14	19	17	16
Heart-Diseases	07	06	17	08	07

Table 11. Comparison of Intra Similarity Measure Obtained with ECVM scheme on Results of DAAC and Existing Techniques.

UCI dataset	Measures of intra cluster similarity ICS (C) in (%)				
	DKNNA (Lai, 2011)	KnA (Qi, 2015)	DAAC Scheme		
			MO>5	MO>10	MO>15
White-Wine	98.99	95.2	99.0	98.79	98.99
Wisconsin	85.09	77.51	92.65	85.09	80.72
Iris	88.0	99.53	99.72	99.72	88.0
Wine	77.24	67.53	94.4	77.24	52.03
Red_Wine	94.25	92.25	100	96.24	93.25
Heart-Diseases	73.10	71.10	99.06	78.59	73.10

Table 12. Comparison results of Intra Dissimilarity Measure obtained with ECVM scheme on results of DAAC and existing techniques.

UCI dataset	Measures of intra cluster dissimilarity ICD (C) in (%)				
	DKNNA (Lai, 2011)	KnA (Qi, 2015)	DAAC Scheme		
			MO>5	MO>10	MO>15
White-Wine	1.007	4.21	0.99	1.202	1.007
Wisconsin	14.90	22.48	7.34	14.90	19.27
Iris	12.0	0.46	0.27	0.27	12.0
Wine	33.29	32.42	17.53	33.29	46.32
Red_Wine	6.80	7.80	3.02	4.2	6.80
Heart-Diseases	26.89	28.89	6.009	15.71	26.89

augmented compared to existing works with outstanding results.

CONFLICT OF INTERESTS

The authors have not declared any conflict of interests.

REFERENCES

Cadez I, Smyth P, Mannik H (2001). Probabilistic modeling of transactional data with applications to profiling, visualization and prediction, Proceedings of the Seventh ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. pp. 37-46.

Chih-Tang C, Lai JZC, Jeng MD (2010). Fast agglomerative clustering using information of k-nearest neighbors. Pattern Recogn. 43(12):3958-3968.

De Amorim RC (2015). Feature Relevance in Ward's Hierarchical Clustering Using the Lp Norm. J. Classif. 32(1):46-62.

Defays D (1977). An efficient algorithm for a complete link method. Comput. J. (British Computer Society) 20(4):364-366.

Douglass RC, David RK, Jan OP, John WT (1992). Scatter / Gather: A Cluster-based approach to Browsing Large Document Collections, Proceedings of the 15th annual international ACM SIGIR Conference on Research and Development in Information Retrieval pp. 318-329.

Fionn M, Legendre P (2014). Wards hierarchical agglomerative clustering method: which algorithms implement wards criterion. J.

Classif. 31(3):274-295.

Fogs A, Warg W, Zaane O (2001). A non-parametric approach to web log analysis, First SAMICDM Workshop on Web Mining, Chicago pp. 41-50.

Franti P, Kaukoranta T, Sen DF, Chang KS (2000). Fast and memory efficient implementation of the exact PNN, IEEE Trans. Image Process. 9(5):773-777.

Frigui H, Krishnapuram R (1997). Clustering by competitive agglomeration, Pattern Recogn. 30(7):1109-1119.

Han J, Kamber M (2006). Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, San Francisco, CA.

Jain AK (2010). Data clustering: 50 Years beyond K-means. Pattern Recogn.Lett. 31(8):651-666.

Jain AK, Murty MN, Flynn PJ (1999). Data Clustering: A Review, ACM Computer Surveys 31(3):264-323.

Krishnamoorthy K, Sreedhar KS (2016). An Improved Agglomerative Clustering Algorithm for Outlier Detection. Appl. Math. Inform. Sci. 10(3):1125-1138.

Lai JZC, Tsung-Jen H (2011). An agglomerative clustering algorithm using a dynamic k-nearest-neighbor list. Inform. Sci. 181(9):1722-1734.

Lin CR, Chen MS (2005). Combining partitional and hierarchical algorithms for robust and efficient data clustering with chesion self-merging, IEEE Trans. Knowl. Data Eng. 17(2):145-159.

Pakhira KAM (2009). Modified k- means Algorithm to avoid empty clusters. Int. J. Recent Trends Eng. 1:1-8.

Martin E, Alexander F, Hans-Peter K, Jörg S (2000). Spatial Data Mining: Database primitives, Algorithms and Efficient DBMS Support. Data Min. Knowl. Discov. 4(2-3):193-216.

Murtagh F (1984). Complexities of Hierarchic Clustering Algorithms: the state of the art. Comput. Stat. Q. 1:101-113.

Qi Y, Xumin L, Xiangmin Z, Andy S (2015). Efficient agglomerative hierarchical clustering. Expert Syst. Appl. 42(5):2785-2797.

Sibson R (1973). SLINK: an optimally efficient algorithm for the single-link cluster method. Comput. J. (British Computer Society) 16:30-34.

Wei Z, Gongxuan Z, Yongli W, Zhaomeng Z, Tao L (2015). NNB: An Efficient Nearest Neighbor Search Method for Hierarchical Clustering on Large Datasets. IEEE International Conference on Semantic Computing (ICSC). pp. 405-412.