An Enhanced Equilin Multi-Clustering Algorithm for Constructing Numeric Clusters

Joy Christy[1]. A, Hari Ganesh.S[2]

[1]Research Scholar, Bishop Heber College, Trichy-620017, India.
[2]Assistant Professor, HH The Rajah's College, Pudukottai-622001, India.

Abstract

Clustering techniques are often applied in data analytics for interpreting the similarities within data objects over large datasets. Despite the existence of many clustering algorithm in the literature such as connectivity, centroid, distribution, density etc, the factor that constitutes a cluster are different from one another. However, the success of clustering depends upon the maximization of intra-cluster similarity and inter-cluster dissimilarity. The significant implication of clustering algorithms in many real-world applications emerges the proposal of newer algorithms. As a consequence, in this paper, a novel effort is made to generate clusters from a different aspect of grouping data objects using multidimensional geographical linear equation called Equilin Clustering. The technique incorporates the standard linear equation and the method of percentage split for clustering numerical data. The results show that the performance of Equilin Clustering yields better cluster results with reduced complexity over time and number of iterations.

Keywords: linear equation; clustering; percentage split; Euclidean distance;

I. INTRODUCTION

Nowadays, the availability of data almost in all domains is found to be enormous and often contains meaningful hidden facts. Data Mining (DM) refers to the attainment of meaningful knowledge from huge volume of data by analyzing them with varieties of tools and techniques. Clustering is one among the popular techniques of DM used to perform similarity based grouping of categorical, ordinal and ratio scaled variables.

The process of clustering groups the data into classes of clusters, so as to classify the objects in a cluster are identical and dissimilar with the objects of other clusters. Dissimilarities are assessed with the values of attributes that represent an object, usually through distance measures. Clustering is said to be unsupervised learning as it does not rely on predefined class labels which is indeed essential for classification. The researches over data mining techniques have been focusing on proposing scalable and effective methods for clustering multidimensional, complex shaped mixed types of data. Clustering can majorly be categorized into methods of connectivity, distribution, density and centroid based.

As an addition to the traditional clustering methods, in this study we are trying to bring in a new dimension of geographical shapes based clustering by applying the equations of certain geometrical shapes over data objects. Hence, in this paper, we have made an attempt to relate equation of line to form numeric clusters. The proposed method called Equlin clustering groups complex shaped numerical data using linear equation with the method of percentage split.

This paper is organized as follows: section II describes review of existing literature, section III contains the background of the study, section IV elucidates the methodology of the proposed work, section V discusses the analysis of the results and finally section VI concludes the benefits and future enhancement of the proposed work.

II. BACKGROUND

The recent enhancements on the traditional algorithms has been studied and presented in this section

A. Connectivity based clustering

Bouguettaya et al. [1] have presented an agglomerative hierarchical method for exempting the process of feature extraction or selection during cluster analysis. The proposed KnA approach integrates K-means and agglomerative approaches to reduce the overall computational cost. The process first applies K-means to the individual data objects to generate k clusters called middle level clusters and then embeds the Single Linkage Method and Group Average Linkage on the centroids of those clusters to build cluster hierarchy. The distance between two middle-levels is then measured to analyze the distribution between the clusters. The authors claim that the proposed method is highly correlates between the trees that are produced by the centroids with the data points.

Pourjabbar et al. [2] have implemented Fuzzy Divisive Hierarchical Clustering (FDHC) and Fuzzy Hierarchical Cross-Clustering (FHCC) to investigate the source of contamination by analyzing pore water samples and leachates from Ordovician-Silurian slate samples near an abandoned uranium mine in Germany. FDHC uses cluster centers and Euclidean distance function to compute the membership degree of the data into the cluster center. FHCC produces a fuzzy partition of the compositions as well as the fuzzy partition of the samples to form clusters. From the results the authors have proven that the fuzzy algorithms have identified the most influencing characteristic parameters of sample clusters for better analysis.

Jeon et al. [3] have proposed a parallelization scheme to reduce the cost of executing Hierarchical Agglomerative Clustering (HAC) on large data. This scheme is introduced for multi-threaded shared-memory machines based on the concept of Nearest-Neighbor (NN) chains. The algorithm grows multiple chains simultaneously by allocating the threads into two groups of managing NN chains and updating distance information. The NN has provided more parallelization opportunities as it considers the merging of multiple pairs by growing one chain per pair. The authors have claimed that the theoretical analysis of the approach provides the optimal ratio for partitioning the thread which bounds on parallelization.

B. Centroid based Clustering

Prabha et al. [4] have proposed an Improved PSO Based K-Means Clustering by integrating the merits of Particle Swarm Optimization and normalized K-Means clustering algorithms. In the proposed PSO based K-Means algorithm the individual position of the solution particles p are randomly selected along with the velocity in the solution space. The data are normalized using max-min procedure to all the attributes of the input dataset. PSO process is embedded with every step of K-Means procedure to compute the fitness function to select the pbest cluster centroid by comparing fitness values of every particle. The authors have concluded that the application of normalization before clustering obtained better clusters.

Tzortzis et al. [5] have introduced a method that integrates MinMax and K-means algorithms, to assign weights to the clusters with respect to their variance and optimized a weighted version of the k-Means. Weights are trained iteratively along with the cluster assignments. The proposed weighing scheme limited the emergence of large variance clusters and allowed high quality solutions to be systematically uncovered, irrespective of the initialization. Experiments verified the effectiveness of the approach and its robustness over bad initializations, as it compared favorably to both k-Means and other methods from the literature that consider the k-Means initialization problem.

Liao et al. [6] introduced a sample-based hierarchical adaptive K-means (SHAKM) clustering algorithm for large scale video retrieval. To handle large databases efficiently, In SHAKM a multilevel random sampling strategy was employed. Furthermore, SHAKM utilized the adaptive K-means clustering algorithm to determine the correct number of clusters and to construct an unbalanced cluster tree. Furthermore, SHAKM used the fast labeling scheme to assign each pattern in the dataset to the closest cluster. To evaluate the proposed method, several datasets were used to illustrate its effectiveness. The results show that SHAKM was fast and effective on very large datasets. Furthermore, the results demonstrated that the proposed method could be used efficiently and successfully for a project on content-based video copy detection.

C. Distribution based Clustering

Chamroukhi et al. [7] have proposed a model-based EM algorithm for clustering curves in combination with regression mixtures. The approach is designed to address the issues faced by EM algorithm such as initialization problem and selection of optimal number of clusters by allowing an appropriate mixture model. In their proposed work the data are assumed to be curves than vectors reduced a dimension which leads to fitting of regression mixture model. The model optimizes penalized observed data log-likelihood using the entropy of hidden structure for generating the curves.

Yu et al. [8] have presented Spatial –EM algorithm for finite mixture learning procedures. Median based location and rank-based scatter estimators that replaces sample mean and covariance in each M step to enhance the stability and robustness of the algorithm. Spatial-EM is applied to both supervised and unsupervised learning scenarios for robust clustering and outlier detection methods. The authors have claimed that their approach is simple, superior and easy to implement than existing methods such as K-median, X-EM and SVM.

Liu et al. [9] have introduced a scheme called Improved EM algorithm for clustering gene expression data based on multivariate elliptical contoured mixture models to solve the problem of over-reliance on the initialization. The algorithm adds and deletes the initial value for the classical EM algorithm and the number of clusters can be treated as a known parameter to infer with the QAIC criterion. The authors have claimed that their scheme outperforms the Gaussian mixture models.

D. *Density Based Clustering*

Ghanbarpour et al. [10] have proposed EXDBSCAN algorithm to cover multi-density datasets as an addition of detecting clusters with various densities by getting a single parameter (Minpts) from the user. The proposed method applies greedy technique for expanding the cluster density. The method starts by choosing a random initial point p. If p is found to the core object, the cluster extension is performed with the same initial E, else the value of E is added by ΔE and p is examined again and again to find the core. The authors have claimed that their work detects various cluster densities and outliers better than DBSCAN.

Though, GbDBSCAN algorithm overcomes the issues of DBSCAN Algorithm such as parameter sensitivity and inability to process large databases, it treats the total number of points in a grid as the grid dense which depresses the accuracy of clustering. Therefore, Zhang et al. [11] have presented CGDBSCAN that implicates migration-coefficient for the optimal selection of parameter Eps and SP-tree query index to improve cluster accuracy. The experiment has claimed that the algorithm has better comprehensive performance.

Kim et al.[12] have presented DBCURE, a density-based clustering to generate clusters with varying densities that are suitable for parallelizing the algorithm with MapReduce. The first step of the algorithm chooses an arbitrary core point, which is an unvisited point in D, as a seed and inserts it to the seed set S. The second step of the algorithm retrieves all points that are density-reachable from the seed set S by extracting a point p from S and inserts its neighborhood of every core point p belongs to S until S becomes empty. The authors have also extended their work by developing DBCURE-MR, a parallel DBCURE using MapReduce that finds multiple clusters together in parallel which estimates the neighborhood covariance matrices, computes ellipsoidal neighborhoods, discovers the core clusters and merges the core clusters. The authors have claimed that the results of DBCURE-MR is able to find correct clusters with varying densities and scales up well with MapReduce framework.

E. *Spherical Clustering*

Duwairi et al. [13] have presented an approach for initializing the spherical K-means based which is based on the calculation of well distributed space across the input space. The authors have proposed a new approach for calculating vector’s directional variance that is used as a measure of cluster’s compactness. The proposed approach is compared with the traditional K-means algorithm over three distinct measures such as intra cluster similarity, cluster compactness and time convergence. The authors have proved that their approach is outperformed the traditional K-means algorithm with respect to cluster compactness and intra cluster similarity. The authors have also confirmed that the time convergence of initialized K-means is faster than the random K-means for small number of clusters.

Fahim et al. [14] have proposed a method of shifting the center of the large cluster toward small cluster at the end of K-means algorithm for re-computing the membership of small cluster points. The authors have suggested that their proposed method could be extensively used in the datasets that contains spherical shaped clusters with large difference in their sizes. The authors have demonstrated that their approach improved the quality of clusters. The author also evidences that their proposed algorithm produce the same result as K-means when the centers of the smaller clusters lie out of them, because in this situation the clusters seem to have very small difference between their radius.

Hill et al. [15] have compared the species distributions with cluster centroids of spherical K-means clustering using the cosine similarity measure. The authors have created an R program called clustaspec based on spherical K-means clustering that is started by being agglomerative and continues with a second phase in which the smallest clusters are systematically removed and their species is distributed to larger ones. The authors have suggested that the spherical K-means algorithm is a powerful clustering method for measuring the similarity between clusters.

III. PROPOSED METHODOLOGY

As the spherical clustering encompasses the equation of sphere to cluster data objects, the proposed Equilin clustering uses the standard linear equation to form clusters of similar data. The construction of Equilin clustering consists of two major steps, where the first step computes the additive multiplication of data objects with respect to mean of attributes which creates a single representation for each data object. The second step analyzes the min-max boundaries for each cluster through Percentage Split Distribution (PSD).

A. Standard Linear Equation

The standard form of linear equation with two variables x and y is denoted as Ax+By=C, where A, B and C represents constants and x and y represents variables. In the proposed approach, constants A and B are considered as mean of attributes and x and y and the distance between data objects are computed by multiplying the mean with all data objects for generating an individual representation of data object, called C as follows:

$$C_{i=1}^{n} = \forall_{i=1}^{n} Ax_i + By_i \tag{1}$$

Where *i* represents the current data objects. Value C_i acts as an important factor for breaking objects into clusters as compared to boundaries that are set through PSD.

B. Percentage Split Distribution (PSD)

In PSD the entire values of C are treated as the values that range between zero to cent percentage. The minimum and the maximum values of C are assigned to 0% and 100% respectively. The boundaries of the clusters are computed using the formula

$$PSD = \left(\left(\max(C) - \min(C)\right) * percentage\right) + min(C) \tag{2}$$

Where, max(C) is refers to the maximum value of C and min(C) refers to the minimum value of C and percentage refers the percentage split value for a cluster. Once, the boundaries for clusters have been set, PSD entails an IF-THEN association rule to discretize percentage splits in accordance with the number of clusters. The number of iterations required for Equilin clustering is the total number clusters.

C. Pseudo Code:

```
1. calculate the  mean for all attributes $a_1$ to $a_n$ as $m_1$ to $m_n$
2. multiply the mean values with each data object to draw a single representation of data objects
   $C_{i=1}^{n} = \forall_{i=1}^{n} m_1 x_i + m_2 y_i + \cdots + m_n z_i$
3. determine the boundaries of clusters by computing the percentage split distribution
4. Assign data objects to the clusters using if-then association
```

Fig. 1. Equilin Clustering – Pseudo Code

IV. ILLUSTRATION

As an illustration an interval scaled data samples with ten data objects have been taken to demonstrate the step-by-step process of Equilin clustering with different number of clusters. The data samples are described with two attributes namely x and y. Table 1. Describes the sample dataset.

Table 1. Data Sample for Equilin Demonstration

x	**y**	**A*x**	**B*y**	**C=A *x+ B*y**
1	1	5.5	5.5	**11**
2	2	11	11	**22**
3	3	16.5	16.5	**33**
4	4	22	22	**44**
5	5	27.5	27.5	**55**
6	6	33	33	**66**
7	7	38.5	38.5	**77**
8	8	44	44	**88**
9	9	49.5	49.5	**99**
10	10	55	55	**110**
A=Mean(m) =5.5	**B=Mean(n) =5.5**			

The mean of attributes x and y are computed and multiplied with each data objects x_1,y_1…x_n,y_n for deriving C_1..C_n. After deriving the single representation of data objects, the percentage split distribution is then computed by taking up the minimum and maximum values from C. Let us assume that the number of clusters be 2. From table.1 it is been observed that:

Min(C) =11

Max(C) =110

As it is discussed before, PSD can be viewed as IF-THEN association depending on the number of clusters, supposing, if the number of cluster is two means the percentage split should be (50%, 50%) i.e. (100%/2). The values are then applied on to equ.2 to derive the boundaries for clusters as

$$\begin{aligned} PSD &= \left((110-11) * \left(\frac{50}{100}\right)\right) + 11 \\ &= \left((99) * \left(\frac{1}{2}\right)\right) + 11 \\ &= (49.5) + 11 \\ &= 60.5 \end{aligned}$$

The boundary that split the cluster is into two is 60.5 (ie). Hence, the association is, If C value of data object is <= 60.5 is in the first cluster, else the object is grouped the second cluster. Table 2 shows the cluster assignments of the given dataset.

if -then association (2 clusters)

```
if(c>60.5)
"assign cluster 2"
else
"assign cluster 1"
```

Table 2. Equilin Cluster Assignment for two clusters

x	y	C=A*x+ B*y	Cluster Assignment
1	1	11<60.5 (True)	Cluster 1
2	2	22<60.5 (True)	Cluster 1
3	3	33<60.5 (True)	Cluster 1
4	4	44<60.5 (True)	Cluster 1
5	5	55<60.5 (True)	Cluster 1
6	6	66<60.5 (False)	Cluster 2
7	7	77<60.5 (False)	Cluster 2
8	8	88<60.5 (False)	Cluster 2
9	9	99<60.5 (False)	Cluster 2
10	10	110<60.5 (False)	Cluster 2
A=Mean(m) =5.5	B=Mean(n)= 5.5		

For generating three clusters, the percentage split distribution is (33.3%, 66.6%.99.9%). Hence, the percentage split boundaries are to be calculated for all three distributions.

$$\begin{aligned} PSD &= \left((110-11) * \left(\frac{33.3}{100}\%\right)\right) + 11 \\ &= (32.967) + 11 \\ &= 43.967 \end{aligned}$$

$$\begin{aligned} PSD &= \left((110-11) * \left(\frac{66.6}{100}\%\right)\right) + 11 \\ &= (65.934) + 11 \\ &= 76.934 \end{aligned}$$

$$\begin{aligned} PSD &= \left((110-11) * \left(\frac{99.9}{100}\%\right)\right) + 11 \\ &= (98.901) + 11 \\ &= 109.901 \end{aligned}$$

if-then association(3 clusters)

```
if(c>109.901)
"assign as outlier"
else if(c>76.934 && c<=109.901)
" assign cluster 3"
else if(c>43.967 && c<=76.934)
"assign cluster 2"
else
"assign cluster 1"
```

Table 3. Equilin Cluster Assignment for 3 clusters

x	y	C=A*x+ B*y	Cluster Assignment
1	1	11	Cluster 1
2	2	22	Cluster 1
3	3	33	Cluster 1
4	4	44	Cluster 2
5	5	55	Cluster 2
6	6	66	Cluster 2
7	7	77	Cluster 3
8	8	88	Cluster 3
9	9	99	Cluster 3
10	10	110	outlier
A=Mean (m)=5.5	B=Mean(n) =5.5		

As the tenth object of the sample dataset exceeds the boundary of third cluster ie. 110>109.901 it is said to be an outlier for three clusters. The visual assignments of Equilin Clustering with respect to two and three clusters are shown in Fig.2.

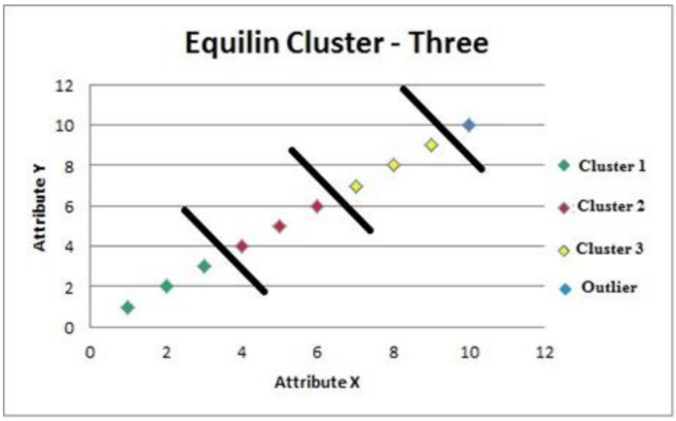


Fig. 2. a). Visualization of Two Clusters

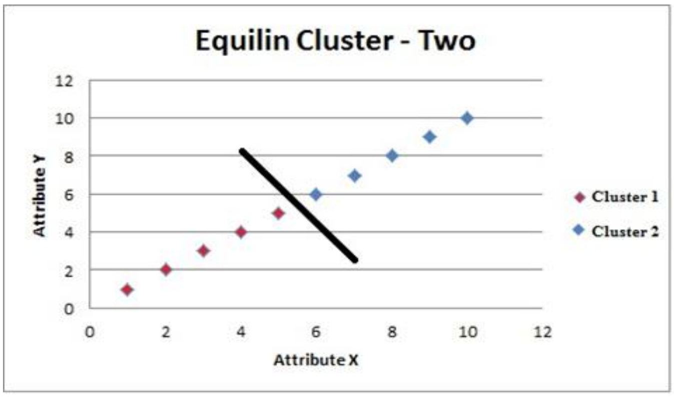


Fig.2 b). Visualization three clusters

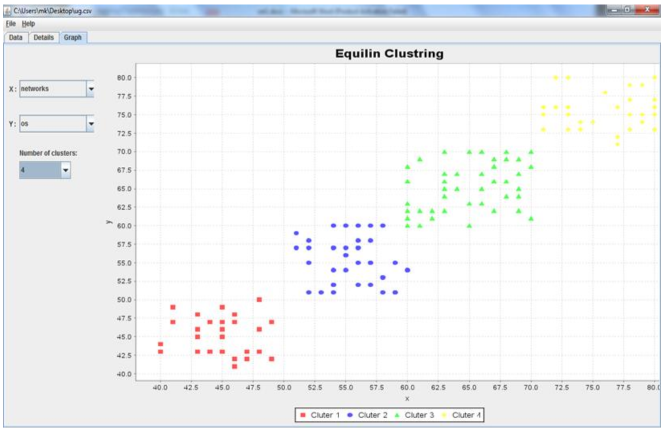
V. EXPERIMENTAL STUDY

A software tool is developed using java programming to evaluate the performance of equilin clustering. For our experimentation, we have collected the semester mark list of students studying final year undergraduate program in computer science discipline at Bishop Heber College. The collected data are categorized into two datasets namely, UG and PG datasets. UG dataset consists of 150 instances with six attributes.

Table.4 UG dataset Description

Attribute Name	Type	Range
Computer Networks	Numeric	40-100
RDBMS	Numeric	40-100
Operating Systems	Numeric	40-100
Mircoprocessor	Numeric	40-100
Data Structures	Numeric	40-100
ASP	Numeric	40-100

The attribute names represent the fifth semester subjects of the curriculum, where the minimum and maximum value ranges between 40 to 100. Choosing Input button from the file menu of the tool prompts the user to select the location of the dataset. Once the dataset is successfully loaded, the data grid of the software tool displays the dataset which implies its readiness for execution. As a default attribute an instance number is added to all data objects to enhance cluster visualization. The next step of the execution is data clustering, which is done by clicking the graph tab of the software tool. The graph tab consists of the specifications of number of clusters, x and y axis representations. User may adjust x and y axis specifications for visualizing the cluster assignments with different set of attributes. Fig. shows the cluster assignments of UG dataset. The objects are visualized through the attributes networks and os and the number of clusters is given as four.



X: networks

Y: os

Number of clusters: 4

Fig.3. Equilin Cluster Assignments

The percentage split of Equilin clustering for four clusters is compared with the traditional clustering algorithms and shown in table. The result of the percentage split of Equlin clustering is as equal to hierarchical and EM clustering algorithms and yields same clustering results.

Table. 5 Comparison of Percentage Split Distribution: Equilin Vs Traditional Clustering Algorithm

Cluster Number	Equilin	Simple K-means	Hierarchical	EM
Cluster 0	20%	20%	20%	20%
Cluster 1	20%	47%	20%	20%
Cluster 2	33%	17%	33%	33%
Cluster 3	27%	15%	27%	27%

A. *Benefits of Equilin Clustering*

- Easy to implement: The implementation of Equilin clustering is simple and requires less computational cost
- Number of Iterations: Unlike other traditional algorithm, Equilin clustering is an iteration free algorithm as it uses if then association for clustering similar data
- Speed: As Equilin clustering is an iteration free algorithm, the time taken to build clusters is very less when compared to other iterative based clustering algorithm
- Accuracy: Equilin clustering works best for interval based datasets where the intervals between data objects are well defined and quantified.

VI. CONCLUSION

This paper tries to uncover the new dimensions of clustering based on geometrical shapes based equations. As a result, we have proposed a novel Equilin clustering method that works on the notion of linear equation with if-then association. At the outset, Equilin clustering is a less computational, high-speed clustering algorithm that builds well defined interval based numeric clusters with greater accuracy. This algorithm is highly suggested for the applications that encompasses quantifiable differences between data objects. Though the extremity of the algorithm is high, it suffers from two main issues, such as creation of empty clusters and detection of outliers. In future, this algorithm may be enhanced to overcome these issues.

REFERENCES

[1]. Bouguettaya, Athman, Qi Yu, Xumin Liu, Xiangmin Zhou, and Andy Song. Efficient agglomerative hierarchical clustering.Expert

[2]. Systems with Applications 42, no. 5 (2015): 2785-2797.

[3]. Pourjabbar, A., C. Sârbu, K. Kostarelos, J. W. Einax, and G. Büchel. Fuzzy hierarchical cross-clustering of data from abandoned mine site contaminated with heavy metals. Computers & Geosciences 72 (2014): 122-133.

[4]. Jeon, Yongkweon, and Seokhyun Yoon. Multi-Threaded Hierarchical Clustering by Parallel Nearest-Neighbor Chaining. Prabha, K. Arun, and N. Karthikeyani Visalakshi. "Improved Particle Swarm Optimization Based K-Means Clustering." In Intelligent Computing Applications (ICICA), 2014 International Conference on, pp. 59-63. IEEE, 2014.

[5]. Tzortzis, Grigorios, and Aristidis Likas. "The MinMax k-means clustering algorithm." Pattern Recognition 47, no. 7 (2014): 2505-2516.

[6]. Liao, Kaiyang, Guizhong Liu, Li Xiao, and Chaoteng Liu. "A sample-based hierarchical adaptive K-means clustering method for large-scale video retrieval." Knowledge-Based Systems 49 (2013): 123-133.

[7]. Chamroukhi, Faicel. "Robust EM algorithm for model-based curve clustering." In Neural Networks (IJCNN), The 2013 International Joint Conference on, pp. 1-8. IEEE, 2013.

[8]. Yu, Kaiyuan, Xin Dang, Henery Bart, and Yuanfeng Chen. "Robust Model-based Learning via Spatial-EM Algorithm." (2015).

[9]. Liu, Zhe, Yu-qing Song, Cong-hua Xie, Feng Zhu, and Xiang Bao. "Clustering gene expression data analysis using an improved EM algorithm based on multivariate elliptical contoured mixture models." Optik-International Journal for Light and Electron Optics 125, no. 21 (2014): 6388-6394.

[10]. Ghanbarpour, Asieh, and Behrooz Minaei. "EXDBSCAN: An extension of DBSCAN to detect clusters in multi-density datasets." In Intelligent Systems (ICIS), 2014 Iranian Conference on, pp. 1-5. IEEE, 2014.

[11]. Zhang, Linmeng, Zhigao Xu, and Fengqi Si. "CGDBSCAN: DBSCAN Algorithm Based on Contribution and Grid." In Computational Intelligence and Design (ISCID), 2013 Sixth International Symposium on, vol. 2, pp. 368-371. IEEE, 2013.

[12]. Kim, Younghoon, Kyuseok Shim, Min-Soeng Kim, and June Sup Lee. "DBCURE-MR: an efficient density-based clustering algorithm for large data using MapReduce." Information Systems 42 (2014): 15-35.

[13]. Duwairi, Rehab, and Mohammed Abu-Rahmeh. "A novel approach for initializing the spherical K-means clustering algorithm." Simulation Modelling Practice and Theory 54 (2015): 49-63.

[14]. Fahim, A. M., G. Saake, A. M. Salem, F. A. Torkey, and M. A. Ramadan. "K-means for spherical clusters with large variance in sizes." Journal of World Academy of Science, Engineering and Technology (2008).

[15]. Hill, Mark, Colin A. Harrower, and Christopher D. Preston. "Spherical k-means clustering is good for interpreting multivariate species occurrence data." Methods in Ecology and Evolution 4, no. 6 (2013): 542-551.