Jurnal Teknologi

Full Paper

KERNEL-BASED EXPONENTIAL GREY WOLF OPTIMIZER FOR RAPID CENTROID ESTIMATION IN DATA CLUSTERING

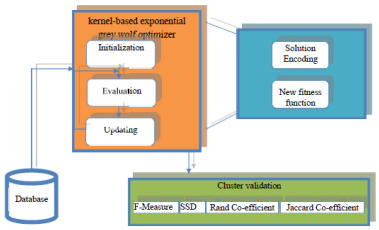
Amolkumar Narayan Jadhav*, Gomathi N.

Vel-Tech Dr. RR & Dr. SR Technical University, Chennai, India

*Corresponding author
amolkumarnj@gmail.com

Article history
Received
1 April 2016
Received in revised form
12 July 2016
Accepted
18 October 2016

Graphical abstract



Abstract

Clustering finds variety of application in a wide range of disciplines because it is mostly helpful for grouping of similar data objects together. Due to the wide applicability, different algorithms have been presented in the literature for segmenting large multidimensional data into discernible representative clusters. Accordingly, in this paper, Kernel-based exponential grey wolf optimizer (KEGWO) is developed for rapid centroid estimation in data clustering. Here, KEGWO is newly proposed to search the cluster centroids with a new objective evaluation which considered two parameters called logarithmic kernel function and distance difference between two top clusters. Based on the new objective function and the modified KEGWO algorithm, centroids are encoded as position vectors and the optimal location is found for the final clustering. The proposed KEGWO algorithm is evaluated with banknote authentication Data Set, iris dataset and wine dataset using four metrics such as, Mean Square Error, F-measure, Rand co-efficient and jaccord coefficient. From the outcome, we proved that the proposed KEGWO algorithm outperformed the existing algorithms.

Keywords: Clustering, data partitioning, kernel, grey wolf optimizer (GWO), optimization, centroid estimation, f-measure

1.0 INTRODUCTION

The rapid development of computer and database technologies leads to the accumulation of data and it exceeds the data processing capability of human. Many applications such as Engineering data management, scientific data management, government administration, business management and others used millions of databases [21]. The inexpensive database system is easily available and so the databases are increasing in a large number. The vital intention of this huge data collection is utilizing this information to attain several benefits, by means of finding the previously unrevealed patterns in data that directs the decision making process. Information extraction from the databases is a major topic under data mining. Data mining has a set of functional modules for several tasks including characterization, association, classification, cluster analysis, and evolution. Clustering can be defined as the unsupervised classification of patterns into groups. Clustering groups a set of objects into different subsets, such that similar objects are grouped into a single cluster. The objects in a cluster will be very similar to each other [22]. The various applications of clustering are information retrieval, image segmentation, web pages grouping, sequence analysis, market segmentation, scientific and engineering analysis and human genetic clustering.

The two major categories of clustering are hierarchical clustering and partitional clustering [11]. A hierarchy of clusters is generated by hierarchical clustering approach in which each cluster is nested within the cluster at a higher level of the hierarchy. A one-level (unnested) partitioning of the data points is created by partitional clustering techniques. If the desired number of clusters is represented by K, then the partitional approach usually find all K clusters at once. These partitional clustering methods are again divided into two categories namely hard and fuzzy [9]. In hard

clustering method, each object is assigned to a single group and in fuzzy method, membership degrees between objects and the different groups of the dataset are introduced [10]. For hard clustering, K-means is the most popular algorithm, which finds the complexity in finding the optimal clusters sue to its iterative nature. This approach has a problem that it finds only the suboptimal solution [4]. Further, this approach considers only the similarities among the objects in a cluster by minimizing the dispersions of the cluster. Also, during the process of minimizing, it deals with all the features equally. But in real applications, different features have different discriminative capabilities [1].

Recently, applying evolutionary algorithms or swarm intelligence to optimal clustering appears to be a common choice on solving difficult clustering problem [4]. As an example, initially, the optimization algorithm called, Genetic algorithm (GA) [12] is applied for clustering and then, Particle Swarm Optimization (PSO) algorithm [13], Artificial Bee Colony [14], Bacterial Foraging Optimization [15], Simulated Annealing [20], Differential Evolution Algorithm [16], Evolutionary algorithm [17] and Firefly [18] are consequently applied for clustering. Later, hybrid algorithms did the clustering process on the datasets to make use of the advantages of both the algorithms taken for hybridization. Here, GA and PSO are combined for the clustering task [19]. The present evolution is using the effective objective function to find the optimal clustering results using optimization-based centroid estimation or hybridizing the optimization algorithm for fast estimation of finding cluster results.

1.1 Literature Review

Literature presents different algorithms for data clustering using optimization algorithms like, particle swarm optimization, genetic algorithm and firefly algorithm. In [2], cuckoo search algorithm was utilized for data clustering which utilizes the kernel-based objective function and [3] utilized memetic algorithm which utilizes the adaptive niching strategy. Yuwono *et al.* [6] developed the clustering process using PSO algorithm which estimates the centroids very rapidly. Then, two algorithms are hybridized to obtain the effective results as like [4, 5, 7, 8]. Accordingly, Tvrdík *et al.* [4] hybridized the differential evolution with k-means algorithm and Kuo *et al.* [5] hybridized the kernel clustering with bee colony optimization. Similarly, Parker et al. [7] hybridized the single-pass fuzzy c-means (SPFCM) and progressive sampling. Telmo *et al.* [8] have developed a hybrid approach by combining the Fuzzy C-means (FCM) with improved PSO. Then, different kernel functions are applied to perform clustering task in recent years. In [25], kernel function-based clustering is applied for gene selection and FCM is integrated with in kernel-based clustering in [26] and automatic weighting based on kernel functions was done for clustering in [27]. Also, multiple kernels are integrated to perform FCM clustering in [28]. Table 1 presents the recent works related to clustering and the major advantages with disadvantages.

Table 1 Literature review

Authors	Contribution	Advantages	Disadvantages
Huang *et al*. [1]	Integrating intracluster compactness and intercluster separation with k-means	Robust algorithm and balanced the intracluster compactness and intercluster separation	k-means is much sensitive to initial cluster assignment
Binu [2]	Cuckoo search with kernel-based objective function	capability of changing the condition for various complex task	Traditional cuckoo search algorithm is sensitive to exploitation and exploration problem
Sheng *et al*. [3]	Adaptive Niching Based Memetic Algorithm	capable to locate suitable clustering solutions with the accurate number of clusters	Much computational effort for high dimensional data
Tvrdík *et al*. [4]	Hybridization of Differential evolution with k-means	more reliable and more efficient, especially in difficult tasks	Re-cluster assignment of k-means have a chance of getting local minimum solutions
Kuo *et al*. [5]	Hybridization of kernel clustering with bee colony optimization	kernel function increases clustering capability	Random assignment to scout bee may have the chance of getting saturated results
Yuwono *et al*. [6]	Rapid centroid estimation using PSO algorithm	potentially helpful for obtaining solutions in large datasets of high dimensionality.	Fitness function (sum of squared distance) is found directly from the data space which is not much differentiable
Parker *et al*. [7]	Hybridization of single-pass fuzzy c-means (SPFCM) and progressive sampling	Scalable algorithm	Very sensitive to sample size and selection of sample data
Telmo *et al*. [8]	Hybridization of FCM with improved PSO	provide better balance between exploration and exploitation, avoiding falling into local minima quickly	Fixing level of cluster fuzziness is very challenging one to get the better results for different valued data

1.2 Problem Definition

Let assume that the input database D contains n data objects and every data object is represented with m attribute values. For example, database D is represented as, $D=\{D_1,D_2,\ldots,D_n\}$; $1\le i\le n$ and every data object within the database is indicated as, $D_i=\{d_{i1},d_{i2},\ldots,d_{im}\}$; $1\le j\le m$. The challenge here is to perform the clustering over the input database to split data objects into k clusters. The clustering over the input database can be signified as the identification of k centroids which are represented as, $C=\{C_1,C_2,\ldots,C_k\};1\le l\le k$. Here, every centroid is represented with m attribute values as like, $C_l=\{c_{l1},c_{l2},\ldots,c_{lm}\}$.

1.3 Challenges

Clustering finds a challenge of searching the optimal centroids which should be optimum to divide the data into k number of partition. So, clustering problem can be formulated as optimal searching problem. It can be stated that k number of centroids should be found out from the data space provided for the input data. Recently, the clustering searching problem is solved in [6] using particle swarm clustering (PSC). In PSC, the centroid estimation was done using the position updating formula developed by them. Again, the evaluation of every centroid is done using the sum of squared distance. When analyzing the PSC algorithm, these are challenges are identified to further extend the work.

i) PSC has the possibility to converge in local optimal solutions (or) clusters due to random assignment of weights.

ii) The particle position updation does not include the data characteristics to initiate the cluster centroids so this may become complex because of wide data distribution, time series characteristics and high dimension

iii) It aims to find the global centroids throughout the process, rather than focusing on initialization part.

iv) The termination strategy has not made the converging procedure to be aware of the quality improvement of centroids.

v) As per [2], the algorithmic effectiveness is decided by objective function but this work [6] utilizes the Sum of Squared Distance (SSD) as objective function even though a lot of improved objective functions are presented in the literature.

vi) Also, data space-based objective function affect the convergence performance based on the characteristics of datasets such as, range of values, dimension, image and data type (integer or floating point).

The above mentioned challenges are solved in the proposed EGWO clustering. The first challenge of convergence to the local optimum is avoided by the proposed EGWO clustering process as the GWO algorithm [24] target the global solution instead of local solution. The second challenge is effectively handled by the proposed EGWO algorithm because the proposed algorithm utilises the exponential function which avoid the data bias effectively. According to the GWO [24], the random assignment of initial solution is not much affected the final results. Also, fourth and fifth challenges are solved using the kernel based objective function.

In this paper, a clustering method is developed for optimal clustering by alleviating the drawbacks discussed in Table 1 which shows the different drawbacks associated with recent clustering methods. Here, instead of particle swarm clustering, grey wolf optimizer [24] is utilized after modifying the searching behavior with exponential model. Thus, a new algorithm, called Exponential Grey Wolf Optimizer (EGWO) is newly proposed to search the cluster centroids. Again, a new objective evaluation is proposed to evaluate centroid estimation behavior using kernel-based distance measure instead of Sum of Squared Distance (SSD). In the proposed clustering procedure, input data is read out with user input k value. Then, the proposed EGWO algorithm is applied with new fitness function to find the cluster centroids. The paper is organized as follows: section 2 presents the kernel-based exponential grey wolf optimizer for rapid centroid estimation. Section 3 discusses the experimental results and section 4 provides the conclusion of the paper.

2.0 METHODOLOGY

This section presents the proposed kernel-based exponential grey wolf optimizer for rapid centroid estimation in data clustering. In the proposed centroid estimation, the existing GWO algorithm is modified with the exponential function to identify the position of wolf. Also, new objective function is devised to evaluate the position using two parameters called, logarithmic kernel function and distance difference between two top clusters. Based on the new objective function and the modified EGWO algorithm, centroids are encoded as position vectors and the optimal location is found for the final clustering. The major advantages of the proposed EGWO algorithm is that i) It is not sensitive to initial cluster assignment, ii) GWO handles effectively the exploitation and exploration problem, iii) it avoided the chance of getting local minimum solutions, iv) It utilizes the kernel function for finding the fitness function which can easily differentiate the data points.

2.1 EGWO: A Modified Algorithm for Optimization

This section presents the proposed EGWO algorithm for clustering of input database. The proposed EGWO algorithm is developed by modifying the existing GWO

algorithm with exponential weighted function. GWO [24] is one of the recent optimization method developed based on the hunting behaviour of the grey wolves. Here, gray wolves are categorized into four categories such as, alpha, beta, omega and delta which are the search agents for hunting. The major advantages of EGWO algorithm is given as below. The social hierarchy assists EGWO to save the best solutions obtained so far over the course of iteration. The encircling mechanism defines a circle-shaped neighborhood around the solutions which can be extended to higher dimensions as a hyper-sphere. The random parameters A and F assist candidate solutions to have hyper-spheres with different random radii. The hunting method allows candidate solutions to locate the probable position of the prey. Exploration and exploitation are guaranteed by the adaptive values of a and A. The adaptive values of parameters a and A allow GWO to smoothly transition between exploration and exploitation. The main phases of the GWO algorithm contains three steps such as, i) tracking, chasing, and approaching the prey, ii) Pursuing, encircling, and harassing the prey until it stops moving and , iii) Attack towards the prey. These three phases are mathematically modelled for the search optimization problems.

This work aims to adapt the GWO algorithm for clustering as the main objective is to estimate or discovery of centroids for the given number of clusters. The existing GWO algorithm update the position of each search agent based on the alpha, bête and delta agents without assigning the numerical weights. From the definition given in [24], we understand that alpha α is the first best search agent, beta β is the second best search agent and delta δ is the third best agent. But, these best agents are then utilized to generate new positions by assigning equal importance but the top best agent should have more weightage in the updating formulae. We consider this problem in the GWO algorithm and the solution is given using exponential function.

The proposed EGWO algorithm is performed using four important steps.

Initialization: The grey wolf population is initialized with a position of q wolves. The elements in the population will be within the lower and upper bound. The population is represented as, $P = \{P_1, P_2, \ldots, P_q\}$. Also, the coefficient vectors such as, A and F are initialized with and the component a. Here, a is linearly decreasing from 2 to 0 over the course of iterations.

Fitness: Once the initialization is performed, the fitness is computed for all the search agents using the fitness function. Based on the fitness function, search agents are categorized into three categories such as, alpha, bête, and delta. Alpha α is the search agent having the best fitness, beta β is a search agent having the second best fitness and the delta δ is the searching segment having third best fitness function. Grey wolves mostly search according to the position of the alpha, beta, and delta. They diverge from each other to search for prey and converge to attack prey. Search agents to update their position based on the location of the alpha, beta, and delta and attack towards the prey.

Updating of search agents: Grey wolves have the ability to recognize the location of prey and encircle them. The hunt is usually guided by the alpha. Every search agents are then updated their position based on alpha, beta and delta. The beta and delta might also participate in hunting occasionally. However, in an abstract search space we have no idea about the location of the optimum (prey). In order to mathematically simulate the hunting behavior of grey wolves, we suppose that the alpha (best candidate solution) beta, and delta have better knowledge about the potential location of prey. Therefore, we save the first three best solutions obtained so far and oblige the other search agents (including the omegas) to update their positions according to the position of the best search agents. The updating of the population can be done using the following equation in GWO.

$$P(t+1) = \frac{P_x + P_y + P_z}{3} \quad (1)$$

Here, the updating is performed using the same weight age for the three top search agents. In order to give dynamic and differential weights for alpha, beta and delta-based updating, the proposed EGWO have given the weight parameters as like below.

$$P(t+1) = \frac{W_x * P_x + W_y * P_y + W_z * P_z}{3} \quad (2)$$

Where,

$$W_x = \frac{1}{1+\exp(P_x)}; W_y = \frac{1}{1+\exp(P_y)}; W_z = \frac{1}{1+\exp(P_z)} \quad (3)$$

From the above equation, we understand that weightage parameter utilizes the exponential function with the parameters of P_x , P_y and P_z . The weightage parameters are usually ranging from 0 to 1. The higher value shows the maximum weightage and the lower value show the less weightage. The parameters of P_x , P_y and P_z are computed using the following equation.

$$P_x = P_\alpha - A_1.B_\alpha ; P_y = P_\beta - A_2.B_\beta ; P_z = P_\delta - A_3.B_\delta \quad (4)$$

From the above equation, we understand that every parameters are updated based on the alpha, beta and delta agents along with the positions of the current iteration.

$$B_\alpha = F_1.P_\alpha - P(t);\ B_\beta = F_2.P_\beta - P(t)\ ;\ B_\delta = F_3.P_\delta - P(t) \quad (5)$$

The values of A and F are computed utilizing the component a which is linearly decreasing from 2 to 0 over the course of iterations and r_1 & r_2 are the random numbers ranging in between o to 1.

Once the population is updated using the above mathematical equation, fitness is computed and the alpha, bête and delta are updated based on the best fitness. Again, the values of a , A and F are also updated based on the following equation.

$$A = 2 * a * r_1 - a \quad (6)$$

$$F = 2 * r_2 \quad (7)$$

Termination: The above process is repeated until the number of iteration is greater than the user given threshold t and the search agent having the best fitness or alpha is taken as the final output from the algorithm. The pseudo code of the EGWO algorithm is given in Figure 1.

1	**Algorithm: EGWO**
2	**Input:** D →Input database
3	k →Number of cluster
4	**Output:**
5	X_α → Best search agent
7	**Begin**
8	**Initialization** of grey wolf population P
9	**Initialize** a, A and F
10	**While** $t <$*MaxIteration*
11	**Find** fitness of each search agent
12	**For** each search agent
13	**Find** P_x, P_y and P_z
14	**Find** W_x, W_y and W_z
15	**Update** $P(t+1)$
16	**EndFor**
17	**Update** a, A and F
18	**Update** X_α, X_β and X_δ
19	$t = t + 1$
20	**EndWhile**
21	**Return** X_α
22	**End**

Figure1 Algorithmic description of the EGWO algorithm

2.2 Encoding Partitioning for KEGWO

Encoding of clustering result is important when the optimization algorithm is adapted to perform clustering task. Ultimately, the clustering task is a process of searching the centroids from the data space to optimally split the data objects. So, clustering task can be represented by a vector of centroids which have the length of $k * m$. Based on this representation, the proposed KEGWO algorithm can easily found out the centroids from the data space by minimizing the objective function. For example, the original database having n data objects with m dimensional feature vector is represented in Figure 2. If the encoding partitioning of individuals can be represented with the elements size of $k * m$. From the Figure 2, we understand that individual is k set of centroids represented as , $\{c_{l1}, c_{l2}, \ldots, c_{lk}\}$. The advantage of this encoding mechanism is that it is simple representation of clustering task and the dimension of the solution for the clustering problem is small than the other representation methods.

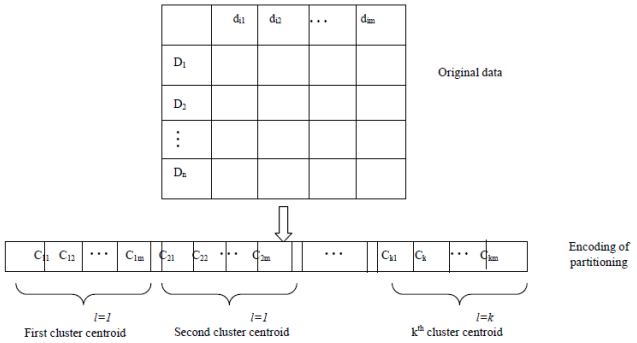


Figure 2 Sample encoding mechanism for KEGWO algorithm

2.3 Designing of Fitness Function

The fitness function to evaluate the clustering task is explained in this section. The fitness evaluation of the clustering task is newly proposed by including logarithmic function and the distance difference between two top clusters. The most common objective to evaluate the clustering task is to minimize the summation of the minimum distance posed by the relevant clusters. This can be represented as follows:

$$F = \sum_{p=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} b_{il} * G_{jl} \quad (8)$$

$$b_{il} = \begin{Bmatrix} 1 \; if \; ith \; data \; belongs \; to \; lth \; cluster \\ 0 \; Otherwise \end{Bmatrix} \quad (9)$$

$$G_{jl} = \left\| x_{ij} - c_{lj} \right\|_2 \quad (10)$$

This common function is modified here by including the logarithm function and distance difference between two top clusters. Accordingly, the objective function formulated in this proposed work is given as follows:

$$F = \sum_{p=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} f_{il} * J_{jl} * H_{ijl} \quad (11)$$

$$f_{il} = \begin{Bmatrix} 1 \; if \; ith \; data \; belongs \; to \; lth \; cluster \\ 0 \; Otherwise \end{Bmatrix} \quad (12)$$

The distance value is given to logarithmic function and it is divided from original distance. Then, the difference is taken from the unity to minimize the objective function. The advantage of using kernel distance is that, i) it can easily control the amount of outliers allowed, ii) Nonlinearity of data can be easily avoidable using kernel based distance.

$$J_{jl} = 1 - \left(\frac{\log \left\| x_{ij} - c_{lj} \right\|_2}{\left\| x_{ij} - c_{lj} \right\|_2} \right) \quad (13)$$

Along with the minimum distance, the additional parameter called, distance difference between two neighbor clusters is found out and it divided from the maximum distance. This difference should be high for better clustering task. So, the difference of the resultant and the unity is found out to minimize the objective. Then, this objective is also multiplied with the distance parameters to find the objective value for the clustering results.

$$H_{ijl} = 1 - \left(\frac{\left\| x_{ij} - c^{ne}{}_{lj} \right\|_2 - \left\| x_{ij} - c_{lj} \right\|_2}{\left\| x_{ij} - c^{ne}{}_{lj} \right\|_2} \right) \quad (14)$$

Where, $\left\| x_{ij} - c_{lj} \right\|_2$ is Euclidean distance between the data object x_{ij} and centroid c_{lj}. $\left\| x_{ij} - c^{ne}{}_{lj} \right\|_2$ is Euclidean distance between the data object x_{ij} and centroid $C^{ne}{}_{lj}$ which is the second best centroid for the input data objects, x_{ij}

2.4 Optimal Clustering using KEGWO

The proposed KEGWO and the fitness function are utilized to perform data clustering task by optimally finding the cluster centroids.

Step 1: Preprocessing: In the first step, input dataset is read out and the class attribute is separated from the original data space. Then, missing data values are replaced with mean value to further make the dataset for doing the clustering task.

Step 2: Searching cluster centroids using KEGWO algorithm: Once the dataset is prepared for the clustering task, the preprocessed data and number of cluster required is given as the input for the KEGWO algorithm. At first, individuals are randomly initialized within the search space. Then, the fitness function is used to evaluate the individuals and the updating of search agents is performed continuously until the maximum number of iterations is reached. Out of all the iterations, the best individuals having the minimum fitness function is selected.

Step 3: Perform clustering: Once we identify the centroids from the best individual, partitioning is performed by finding the distance among the centroids and data objects. The centroid having the minimum distance for the input data objects is identified and the corresponding data objects are grouped together.

3.0 RESULTS AND DISCUSSION

This section presents experimental validation of the clustering algorithms. In order to handle with, evaluation metrics and dataset taken for the validation of the clustering algorithms is explained with full description. Then, detailed experimental results are given with graphs and the corresponding discussion is given in this section.

3.1 Experimental Set Up

1) Dataset description: Three datasets such as, banknote authentication Data Set, iris dataset and wine dataset are taken from UCI machine learning repository [23]. Banknote authentication dataset (D1): This data was collected from images that were acquired from genuine and forged banknote-like specimens. An industrial camera frequently utilized for print inspection was utilized for digitization of the image taken in the size of 400x 400 pixels. In order to extract features from the images, wavelet transformation was used. The attributes taken are variance (continuous), skewness (continuous), curtosis (continuous) from the wavelet image and entropy of image (continuous) and class (integer). Iris (D2): Iris is one of the popular databases widely used in pattern recognition literature. This data consists of 50 instances for every class. This data totally contains 150 data instance with 3 classes which indicates the type of iris plant. Also, the total number of considered attributes is four with one class attribute. Wine (D3): Wine is a popular databases widely used in data clustering. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

2) Evaluation metrics: We utilized four performance metrics such as, Mean Square Error (MSE), F-measure, Rand co-efficient and jaccord coefficient. The definitions are given as follows:

$$MSE = \sum_{p=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} b_{il} * G_{jl} \quad (15)$$

$$b_{il} = \begin{Bmatrix} 1 \ if \ ith data belongs \ to \ lth \ cluster \\ 0 \ \ Otherwise \end{Bmatrix} \quad (16)$$

$$G_{jl} = \left\| x_{ij} - c_{lj} \right\|_2 \quad (17)$$

Let C the set of clusters to be evaluated, L the set of categories (reference distribution) and n the number of clustered items. Then, *F-measure* is computed as follows:

$$F\_measure = \sum_{i} \frac{|L_i|}{n} \max_j \{F(L_i, C_j)\} \quad (18)$$

$$F(L_i, C_j) = \frac{2 * Precision(L_i, C_j) * Recall(L_i, C_j)}{precision(L_i, C_j) + Recall(L_i, C_j)} \quad (19)$$

$$Precision(L_i, C_j) = \frac{|C_i \cap L_j|}{C_i} \quad (20)$$

$$Recall(L_i, C_j) = Precision(C_j, L_i) \quad (21)$$

Rand co-efficient, $RC = (SS + DD)/(SS + SD + DS + DD)$ (22)

Jaccard co-efficient, $JC = (SS)/(SS + SD + DS)$ (23)

Here, SS, SD, DS, DD represent the number of possible pairs of data points where,

SS: both the data points belong to the same cluster and same group.

SD: both the data points belong to the same cluster but different groups.

DS: both the data points belong to different clusters but same group.

DD: both the data points belong to different clusters and different groups.

3) Parameters fixed: The clustering algorithms are written in MATLAB programming (Version: R2014a) and the results are taken after running with a system of

having 2.2GHz Intel (R) Core (TM) i5 CPU with 4GB RAM. For EGWO algorithm, the parameters are fixed as like, $q = 10$ and $t = 50$ by trial and error method by manually varying q from 1 to 100 and t from 1 to 100.

3.2 Experimental Results

Figure 3 shows the clustered results of three datasets along with its original data visualization. The bank data is directly visualized in Figure 3(a) and the clustering output of the proposed EGWO is also shown in figure 3(b). The iris data is visualized in Figure 3(c) and the clustering results are shown in Figure 3(d). From the Figure 3(b), the dataset are clearly partitioned without much overlapping among the clusters. Also, iris data is also separated into compact way without much overlapping among the clusters even though the number of cluster is large. The wine data is visualized in Figure 3(e) and the clustering results are shown in Figure 3(f). The results shown in Figure 3(f) are plotted using only 2 dimensional data space so it is not clear about the groups.

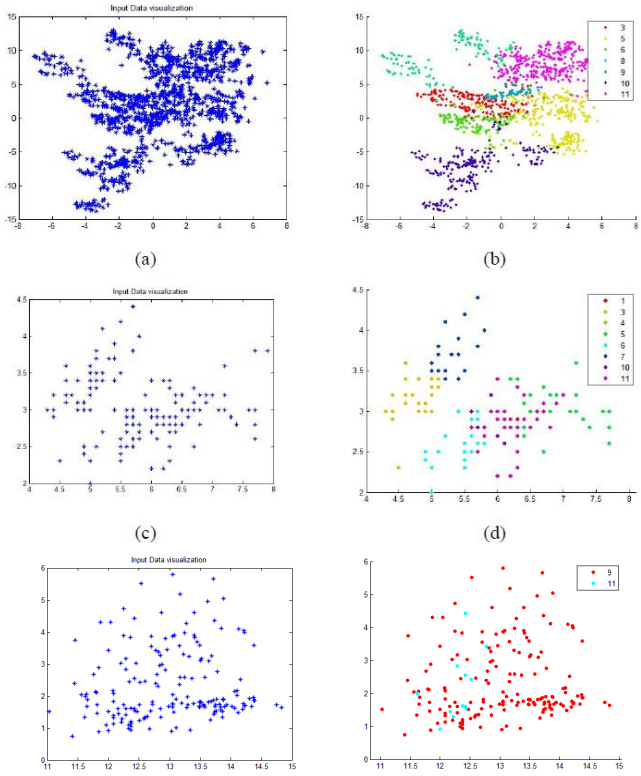
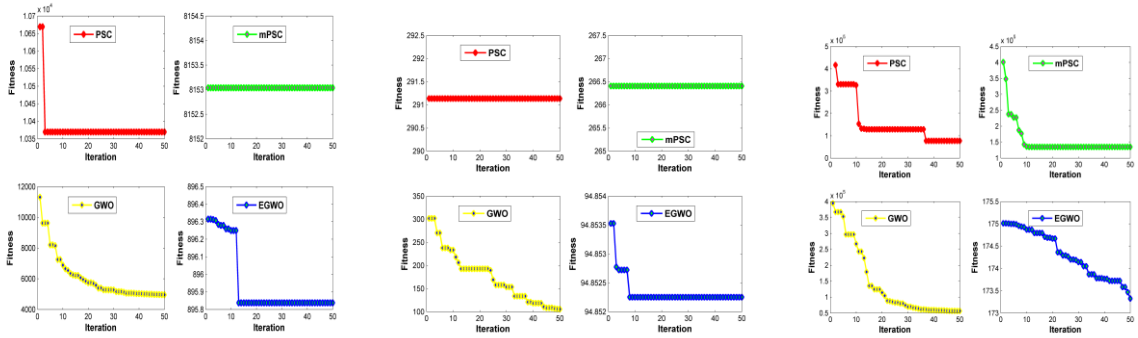


Figure 3 Visualization of results

3.3 Analysis the Performance of Search Algorithm

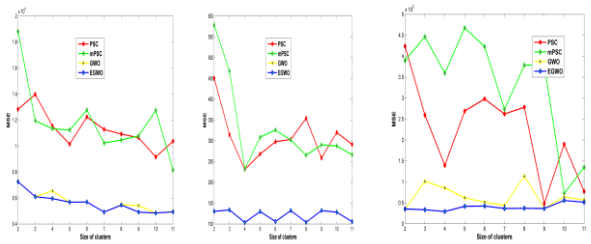
This section analyzes the performance of the search algorithm through convergence graph. Figure 4(a) shows the convergence graph of PSC, Modified particle Swarm Clustering (mPSC), EWO and EGWO in bank data. From the Figure 4(a), we understand that the convergence is steeply decreasing for the proposed algorithm. In the PSC, the performance is decreased from 10669 to 10369 for 50 iterations but the performance of mPSC is constant for all the 50 iterations. For GWO, the fitness is decreased from 11307 to 4953 for 50 iterations. These three algorithms utilized the sum of squared distance as fitness function. But, the proposed algorithm utilized the proposed fitness function as objective criteria to evaluate the clustering algorithm. Here, fitness function is decreased from 896.31 to 895.8. Figure 4(b) shows the convergence graph of PSC, mPSC, EWO and EGWO in iris data. Here, the performance is constant from first to 50 iterations without showing the fitness deviation. But, GWO and EGWO show the performance variation from first to last iterations. The fitness value of EGWO is decreased from 94.8535 to 94.8523 for the 50 iterations. For the GWO algorithm, the fitness is decreased from 301.8220 to 105.4527. This analysis on both the datasets ensured that the proposed EGWO shows the better performance in terms of convergence graph compared with other existing algorithms. Figure 4(c) shows the convergence graph of PSC, mPSC, EWO and EGWO in wine data. It shows that the better fitness reached by the proposed EGWO is 173.22 which is lesser than the existing methods.



(a) (b) (c)

Figure 4 Convergence graph

Figure 5 shows the performance analysis of the clustering algorithms through the sum of squared distance (MSE). The size of clusters is varied from 2 to 11 and the best values among the 50 iterations are used to plot the graphs. Figure 5(a) shows the MSE of bank data. From the figure, we understand that the proposed EGWO is better than the other existing algorithm by showing the minimum value. For some cases, EGWO and GWO behaved similarly by reaching the same value. When the number of cluster is equal to two, GWO and EGWO obtained the same value of 7247 but the existing mPSC obtained the value of 18812. The lowest performance is achieved by the EGWO is 4840. The Figure clearly indicates that PSC and mPSC behaved almost similar and GWO and EQWO behaved almost similar. This ensured the extension and their root algorithms show the similar performance with little deviation. Figure 5(b) shows the performance analysis of the proposed algorithm and existing algorithms in three datasets using MSE. From the graph, we understand that the better performance of 103.28 is reached by the proposed EGWO algorithm when the cluster size is fixed to four. Here, the existing PSC, mPSC and GWO algorithm obtained the value of 230.88, 231.82, 103.28. Figure 5(d) shows the MSE of wine data. From the Figure, we understand that the proposed EGWO is better than the other existing algorithm by showing the minimum value. The lowest performance is achieved by the EGWO is 33350. This ensured the extension and their root algorithms show the similar performance with little deviation.

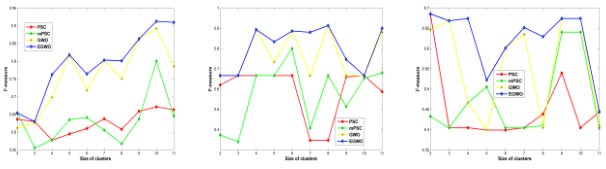


(a) (b) (c)

Figure 5 Sum of squared distance

3.4 Analysis the Performance of Clustering

This section presents the analysis of the clustering algorithm using three different metrics like f-measure, jaccord coefficient and rand coefficient in both the datasets. Figure 6(a) shows the performance of four clustering algorithms on bank dataset. When the cluster size is fixed to 10, the PSC, mPSC, GWO and EGWO algorithm obtained the value of 66.33%, 64.5%, 78.57% and 90.96%. Also, when the cluster size is fixed to two, the proposed EGWO obtained the higher value of 65.38% when compared with other existing algorithms. For some cases, GWO and EGWO show the similar performance in terms of f-measure. This graph ensures that the proposed EGWO outperformed the existing algorithms. Figure 6(b) shows the performance analysis of the clustering algorithms on iris data through f-measure. The size of clusters is varied from 2 to 11 and the best values among the 50 iterations are used to plot the graphs. From the figure, we understand that the proposed EGWO is better than the other existing algorithm by showing the higher value. For some cases, EGWO and GWO behaved similarly by reaching the same value. Figure 6(d) shows the performance of four clustering algorithms on wine dataset. When the cluster size is fixed to 10, the PSC, mPSC, GWO and EGWO algorithm obtained the value of 54.49%, 64.61%, 43.26% and 68.54%.

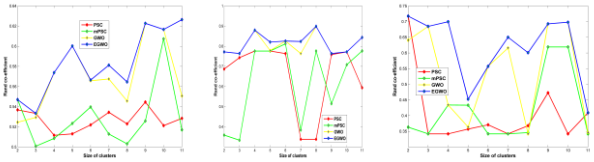


(a) (b) (c)

Figure 6 F-measure

Figure 7 shows the performance of the clustering algorithms on bank, iris and wine data in rand coefficient. From the Figure 7(a), we understand that the proposed EGWO shows either better performance or similar performance while compared with GWO

algorithm. The proposed EGWO obtained the value of 54.70% for the cluster size of two. The better performance of 62.64% is achieved when the cluster size is equal to ten by the proposed EGWO but the existing PSC, mPSC and GWO obtained the values of 52.84%, 51.70% and 55.06%. Overall, the proposed algorithm outperformed the existing PSC and mPSC algorithms by showing the better performance. Figure 7(b) shows the performance of four clustering algorithms on iris dataset. When the cluster size is fixed to 11, the PSC, mPSC, GWO and EGWO algorithm obtained the value of 59.38%, 77.66%, 84.36% and 84.36%. Also, when the cluster size is fixed to two, the proposed EGWO obtained the higher value of 77.19% when compared with other existing algorithms. For some cases, GWO and EGWO show the similar performance in terms of rand coefficient. Figure 7(c) shows the performance analysis of the proposed EGWO with the existing algorithms on wine data. This graph ensures that the proposed EGWO outperformed the existing algorithms



(a) (b) (c)

Figure 7 Rand coefficient

Figure 8 shows the performance analysis of the proposed algorithm and existing algorithms in three datasets using jaccord coefficient. From the graph, we understand that the better performance of 49.94% is reached by the proposed EGWO algorithm when the cluster size is fixed to four. Here, the existing PSC, mPSC and GWO algorithm obtained the value of 49.94%, 48.97%, 33.10%. From the Figure 8(b), we understand that the proposed EGWO shows either better performance or similar performance while compared with GWO algorithm. The proposed EGWO obtained the value of 54.70% for the cluster size of two. The better performance of 73.70% is achieved when the cluster size is equal to eight by the proposed EGWO but the existing PSC, mPSC and GWO obtained the values of 32.61%, 59.51% and 73.70%. Figure 8(c) shows the performance analysis on wine dataset. The figure shows that the proposed EGWO obtained the maximum value of 50.4%. Overall, the proposed algorithm outperformed the existing PSC and mPSC algorithms by showing the better performance.

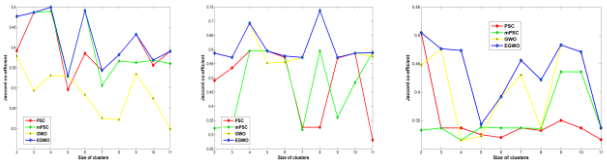


Figure 8 Jaccord coefficient

3.5 Discussion

Table 2 shows the summary of the best performance of all the four clustering algorithms in dataset 1, dataset 2 and dataset 3. This table is derived by finding the best performance of the algorithm for the different values of cluster size. For dataset 1, the proposed EGWO algorithm obtained the value of 91.25% but the existing algorithms like PSC and mPSC obtained values of 67.13% and 80.03%. In terms of rand coefficient, the proposed EGWO algorithm obtained the value 62.64% but the existing GWO algorithm obtained the value of 62.27%. Similarly the performance metrics of jaccord for the PSC, mPSC, GWO and EGWO are 49.94%, 49.10%, 37.89% and 49.94%. The best MSE reached by the proposed algorithm is 4840 which is less as compared with PSC and mPSC algorithms. Similarly, the summary of performance metrics for the dataset 2 is given in Table 2. For the dataset 3, the proposed EGWO obtained the maximum F-measure of 68.5% as compared with other algorithms. Also, the maximum rand coefficient reached by the proposed EGWO algorithm is 71.7%. Here, the proposed EGWO and its root algorithm called, GWO showed the similar performance in all the metrics considered. These two algorithms outperformed the existing algorithm in f-measure, rand-coefficient and jaccord coefficient and MSE.

Table 2 Best performance of algorithms

	Dataset 1				***Dataset 2***				***Dataset 3***			
	F-measure	Rand coefficient	Jaccord coefficient	MSE	F-measure	Rand coefficient	Jaccord coefficient	MSE	F-measure	Rand coefficient	Jaccord coefficient	MSE
PSC	67.13	54.46	49.94	9152	66.67	77.63	59.51	230.88	64.04	61.9	33.8	35320
mPSC	80.03	60.73	49.10	8153	80.0	81.21	59.51	231.8	66.85	68.4	47.6	72010
GWO	89.21	62.27	37.89	4840	91.33	89.88	73.70	103.2	67.4	69.9	47.3	47760
EGWO	91.25	62.64	49.94	4840	91.33	89.88	73.70	103.2	68.5	71.7	50.4	29030

4.0 CONCLUSION

In this paper, we have presented a new algorithm, called (KEGWO) to search the cluster centroids within the data space. Here, the existing GWO algorithm is modified with the exponential function to identify the position of wolf. The evaluation of the position vectors is done with new objective function which is proposed including logarithmic kernel function and distance difference between two top clusters along with minimum distance. The proposed KEGWO and the fitness function are utilized to perform data clustering task by optimally finding the cluster centroids. Finally, three dataset such as, banknote authentication data, iris data and wine data are utilized to perform the experimental evaluation of the proposed algorithm using MSE, F-measure, Rand co-efficient and jaccord coefficient. The performance of the proposed EGWO algorithm is analyzed with respect to the effectiveness of search algorithm and clustering task by comparing with existing algorithms such as PSC, mPSC and GWO. From the results, we proved that the proposed clustering algorithm obtained the maximum F-measure of 91.33%. In future, this algorithm can be extended with multi-objective search for better task of clustering.

References

[1] Huang, X., Ye,Y. and Zhang, H. 2014. Extensions of Kmeans-Type Algorithms: A New Clustering Framework by Integrating Intracluster Compactness and Intercluster Separation. *IEEE Transactions On Neural Networks And Learning Systems*. 25(8): 1433-1446.

[2] Binu, D. 2015. Cluster Analysis Using Optimization Algorithms With Newly Designed Objective Functions. *Expert Systems with Applications*. 42(14): 5848-5859.

[3] Sheng, W., Chen, S., Fairhurst, M., Xiao, G. and Mao, J. 2014. Multilocal Search and Adaptive Niching Based Memetic Algorithm with a Consensus Criterion for Data Clustering. *IEEE Transactions On Evolutionary Computation*. 18(5): 721-741.

[4] Tvrdík, J. and Krivy, I. 2015. Hybrid Differential Evolution Algorithm For Optimal Clustering. *Applied Soft Computing*. 35: 502-512.

[5] Kuo, R. J., Huang, Y. D., Lin, C. C., Wu, Y. H. and Zulvia, F. E. 2014. Automatic Kernel Clustering with Bee Colony Optimization Algorithm. *Information Sciences*. 283: 107-122.

[6] Yuwono, M., Su, S. W., Moulton, B. D. and Nguyen, H. T. 2014. Data Clustering Using Variants of Rapid Centroid Estimation, *IEEE Transactions On Evolutionary Computation*. 18(3): 366-377.

[7] Parker, J. K. and Hall, L. O. 2014. Accelerating Fuzzy-C Means Using an Estimated Subsample Size. *IEEE Transactions On Fuzzy Systems*. 22(5): 1229-12445.

[8] Filho, T. M. S., Pimentel, B. A., Souza, R. M. C. R. and Oliveira, A. L. I. 2015. Hybrid Methods For Fuzzy Clustering Based On Fuzzy C-Means And Improved Particle Swarm Optimization. *Expert Systems with Applications*. 42(17-18): 6315-6328.

[9] Xu, R., Wunsch, D. I. 2005. Survey Of Clustering Algorithms. *IEEE Transactions on Neural Networks*. 16(3): 645-678.

[10] Pal, N., Pal, K., Keller, J. and Bezdek, J. 2005. A Possibilistic Fuzzy C-Means Clustering Algorithm. *IEEE Transactions on Fuzzy Systems.* 13(4): 517-530.

[11] Jain, A. K. 2010. Data Clustering: 50 Years Beyondk-Means. *Pattern Recognit. Lett*. 31(8): 651-666.

[12] Mualik, U. and Bandyopadhyay, S. 2002. Genetic Algorithm Based Clustering Technique. *Pattern Recognition*. 33: 1455-1465.

[13] Premalatha, K. and Natarajan, A. M. 2008. A New Approach For Data Clustering Based On PSO With Local Search. *Computer and Information Science*. 1(4).

[14] Zhang, C., Ouyang, D. and Ning, J. 2010. An Artificial Bee Colony Approach For Clustering. *Expert Systems with Applications*. 37: 4761-4767.

[15] Wan, M., Li, L., Xiao, J., Wang, C. and Yang, Y. 2012. Data Clustering Using Bacterial Foraging Optimization. *Journal of Intelligent Information Systems*. 38(2): 321-341.

[16] Das, S., Abraham, A. and Konar, A. 2008. Automatic Clustering Using An Improved Differential Evolution Algorithm. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*. 38(1): 218-237.

[17] Castellanos-Garzón, J. A. and Diaz, F. 2013. An Evolutionary Computational Model Applied To Cluster Analysis Of DNA Microarray Data. *Expert Systems with Applications*. 40(7): 2575-2591.

[18] Senthilnath, J., Omkar, S. N. and Mani, V. 2011. Clustering Using Firefly Algorithm: Performance Stud. *Swarm and Evolutionary Computation*. 1: 164-171.

[19] Kuo, R. J., Syu, Y. J., Chen, Z. Y. and Tien, F. C. 2012. Integration Of Particle Swarm Optimization And Genetic Algorithm For Dynamic Clustering. *Journal of Information Sciences*. 19: 124-140.

[20] Selim, S. Z. and Alsultan, K. 1991. A Simulated Annealing Algorithm For The Clustering Problem. *Pattern Recognition*. 10(24): 1003-1008.

[21] Berkhin, P. 2002. *Survey of Clustering Data Mining Techniques*. Grouping Multidimensional Data, Springer-Verlag. 25-71.

[22] Yasodha, M., and Mohanraj, M. 2011. Clustering Algorithms for Biological Data - A Survey Approach. *CiiT Journal Of Data Mining And Knowledge Engineering*. 3(3).

[23] Datasets from http://archive.ics.uci.edu/ml/.

[24] Mirjalili, S., Mirjalili, S. M. and Lewis, A. 2014. Grey Wolf Optimizer. *Advances in Engineering Software*. 69: 46-61.

[25] Chen, H., Zhang, Y. and Gutman, I. 2016. A Kernel-Based Clustering Method For Gene Selection With Gene Expression Data. *Journal of Biomedical Informatics*. 62: 12-20.

[26] Ding, Y. and Xian F. 2016. Kernel-Based Fuzzy C-Means Clustering Algorithm Based On Genetic Algorithm. *Neurocomputing*. 188: 233-238.

[27] Ferreira, M. R. P., Carvalho, F. A. T. and Simões, E. C. 2016. Kernel-based Hard Clustering Methods With Kernelization Of The Metric And Automatic Weighting Of The Variables. *Pattern Recognition*. 51: 310-321.

[28] Nguyen, D. D., Ngo, L. T., Pham, L. T. and Pedrycz, W. 2015. Towards Hybrid Clustering Approach To Data Classification: Multiple Kernels Based Interval-Valued Fuzzy C-Means Algorithms. *Fuzzy Sets and Systems*. 279: 17-39.