Data Prediction and Optimized Clustering For MPSO and CSO Based Clustering

S.S.Saravana Kumar
Asst professor Dept of Information Technology, Kovai Kalaimagal College of Arts and Science Narasipuram Coimbatore, India.

G.Divya
Department of Computer Science, Kovai Kalaimagal College of Arts and Science, Narasipuram Coimbatore, India.

Abstract – Cat Swarm optimization (CSO) is one in every of the new heuristic optimization algorithms that supported swarm intelligence. Previous analysis shows that this algorithmic program has higher performance compared to the opposite heuristic optimization algorithms: Particle Swarm optimization (PSO) and weighted-PSO within the cases of perform diminution. During this analysis a brand new CSO algorithmic program for bunch drawback is projected. The new CSO bunch algorithmic program was tested on four totally different datasets. The modification is created on the CSO formula to get higher results. Then, the accuracy level of projected algorithmic program was compared to those of K-means and PSO bunch. The modification of CSO formula will improve the performance of CSO bunch. The comparison indicates that CSO bunch may be thought-about as a sufficiently correct bunch technique.

Index Terms – CSO, PSO, Optimization, K-means.

1. INTRODUCTION

1.1. Ant Colony Optimization

The Ant Colony Optimization (ACO) algorithm is a more met heuristic which is a grouping of the distributed environments, positive feedback systems, and systematic greedy approaches to find an optimal solution value for combinatorial optimization problems on lung cancer. The Ant Colony Optimization algorithm is mainly inspired by the various types of experiments & treatment plans run by Goss et al. [19] which using a grouping the real ants in the real environments. They studied and observed the behaviors of those real ants and suggest that the real ants were having capability to choose and select the shortest path between their shelter and food products resource, in the existence of alternate paths between the two. The above Searching for food products resource is possible through an indirect communications known as stigmergy amongst of the ants. When ants are travelling for the food Resources, ants deposit a new type of chemical substances, called pheromone.

When they arrive at a closing point; ants make a probability on choices, biased by the intensity of Pheromone they smell. This behavior has an autocatalytic effect because of the very fact that An ant choosing a path will increase the probability that the corresponding path will be chosen Again by other ants in the next move of the. After finishing the search ant's returns back,

The Probability on choosing the same path is higher because of increasing pheromone quantity. So the pheromone will released on the chosen way, it provides the new way to the ants. We can say that, all ants will select the shortest path. Figure 1 shows the behavior of ants in a double bridge experiment [20]. If we analyze the case then we observed that because of the same pheromone laying the shortest Path will be taken. It will be starts with the first ants which arrive at the food source are those that took the two shortest branches of the path. After approaching the

Food destination point these ants start. Ants return trip, was more pheromone is present on the shortest branch is the possibility for choosing the shortest one than the one on the Long Branch. This ant behavior was first formulated and arranged as Ant System (AS). Based on the AS algorithm, the Ant Colony Optimization (ACO) algorithm was proposed [22]. In ACO algorithm, the optimization problem can be expressed as a formulated graph G = (C; L), where C is the setoff components of the problem which is given, and L is the set of main possible connections or transitions among the element values of C. The proposed solution is mainly represented in terms of feasible paths on the given graph G, with respect to a given constraints and predicate

Cancer therapies require classification of cancers to target specific cancers with specific treatments. Thus the improvements in cancer therapies have been linked to improvements in cancer classification. To enhance the efficiency of the treatment it is important to identify the specific markers that are to be targeted in a treatment. Apart from enhancing the efficiency of treatment, targeting of specific markers allows for the minimization of toxicity resulting from the treatment. The advent of micro array technologies has greatly aided in the identification of specific genes through the measurement of gene expression data. Current micro array technologies allow for the measurement of thousands of gene expression levels from a single sample

simultaneously [9]. However in almost all the cases the number of samples considered is far less than the number of genes measured. This often causes the problem of over fitting during classification. Thus it is important to reduce the dimensionality of the sample before using it for classification [3]. Moreover from a diagnostics perspective it is important to isolate the specific genes so that a specific diagnostic setup and treatment setup may be developed to predict, classify and treat such a cancer. This also helps in reducing the cost of treatment [1].

To achieve the above objectives various feature selection methods in combination with various Classification tools have been used [1, 3, 4, 15, 16, 17, 18]. Some of the prominent methods that have been used to classify data from micro-arrays have been k-nearest neighbors (KNN), nearest centroid, linear discriminate analysis (LDA), neural networks (NN) and support vector machines (SVM). For selection of subsets of genes, feature selection methods such as t-test, Principal component analysis (PCA), individual gene selection, pair-wise gene selection, non-parametric scoring and now recently evolutionary computing algorithms such as genetic algorithms (GA's) and particle swarm optimization (PSO) are being applied [9].

Currently the gene selection methods may be classified into two categories. One, a filter approach wherein each gene is considered independently evaluated according to specific criteria and then ranked accordingly based on its score. The top ranked genes are considered for while evaluating the classification accuracy of the classifier. Prominent approaches in this category include t-test filtering, SNR filtering, PCA etc.[9]. Second is the wrapper approach wherein, selection of a subset of genes and classification is performed in the same process. A subset of genes are considered and evaluated based on the classifier's performance. This process is carried out recursively till the desired classification accuracy is obtained. Evolutionary Algorithms like PSO [15, 16, 18], GA [16] have been used in conjunction with the SVMs.

Guymon et. al demonstrated a recursive feature elimination method (RFE) –SVM [3] based Wrapper. Though the wrapper approach results in the improvement of classification accuracy, a Major problem is the computational cost associated while using the wrapper. It is important to use algorithms that traverse the search space efficiently with reduced computational costs. Thus PSO is used here which when compared to GAs or RFE, is simpler, faster and converges to an Optimum quickly. However PSO has certain drawbacks like converging to a local optimum, Reduction in convergence rate while approaching optimum etc. To this end a novel discrete PSOSVM is proposed that not only avoids local optima but also converges to a global optimum quickly and demonstrates enhanced classification accuracy.

Clustering is an important technique for discovering the inherent structure in any given pattern set without any prior knowledge. The clustering result should possess two properties: (1) homogeneity within the clusters, that is, the objects belonging to the same cluster should be as similar as possible, and (2) heterogeneity between the clusters, that is, the objects belonging to different clusters should be as different as possible. Clustering analysis has been applied in many fields such as machine learning, pattern recognition, and statistics (Pedrycz, 2005). Many clustering approaches have been reported which can be classified into two categories: hierarchical and partitional (Omran et al., 2007).

In this article, we focus our attention on partitional clustering. Some clustering techniques are available in the literature. Among them, *k*-means algorithm, a typical iterative hill-climbing method, is popular. However, the major drawbacks of the *k*-means algorithm are that it often gets stuck at local minima and its result is largely dependent on the choice of initial cluster centers (Selim and Ismail, 1984). *K*-harmonic means algorithm, another center-based clustering method, is proposed by Zhang et al. (1999) and modified by Hammerly and Elkan (2002) to solve the problem of initialization of the *k*-means algorithm. It is demonstrated that the *k*-harmonic means algorithm is essentially insensitive to the initialization of cluster centers. However, it tends to converge to local optima in some cases. In order to overcome the shortcomings of the *k*-means algorithm, researchers designed some improved clustering methods (Pedrycz, 2005; Omran et al., 2007). Recently, researchers employed met heuristic techniques such as genetic algorithms, simulated annealing, particle swarm optimization, and tabu search to deal with the clustering problem so as to achieve the optimal or near-optimal solution within a specified number of iterations.

Cat swarm optimization (CSO), a recent met heuristic technique firstly reported by Chu and Tsai (2007), models the behavior of cats to solve the optimization problem. In this article, we employ cat swarm optimization to deal with the clustering problem, develop *k*-means improvement based seeking mode, and design simulated annealing selection based tracing mode. As a result, a new clustering method is proposed called *K*-means improvement and Simulated Annealing selection based cat swarm optimization clustering (KSACSOC). On one hand, *k*-means improvement fine-tunes the object distribution among different clusters so as to enhance the convergence of the KSACSOC algorithm, and on the other hand, simulated annealing selection accepts bad solutions probabilistically so as to strengthen the exploration of the unvisited solution space. In this paper, our aim is to introduce cat swarm optimization to deal with the clustering problem, explore its applicability to clustering analysis, and to hybridize cat swarm optimization with *k*-means algorithm and simulated annealing so as to combine the advantages of each one of them and evolve the proper clustering of data sets. To our best knowledge, this is the first reported study that reflects on the usage of the combination of cat swarm optimization, *k*-means

algorithm, and simulated annealing in clustering analysis. Experimental results on two artificial and six real life data sets are given to illustrate that the KSACSOC algorithms can provide better objective function values and higher success rates than *k*-means algorithm, a simulated annealing clustering method, and a particle swarm optimization clustering method.

2. LITERATURE REVIEW

In 2011, Shyi-Ching Liang et al. [28] suggest Cataloging rule is the most common representation of the rule in data mining. It is based on controlled learning process which causes rules from drill data set. The main goal of the cataloging rule mining is the prediction of the predefined class based on the collection. Based on ACO procedure, Ant-Miner solved the arrangement law problematic. According to the author, Ant-Miner shows good presentation in many dataset. In this research paper author future, an extension of Ant-Miner is proposed to integrate the concept of parallel dispensation and alliance. In this paper intercommunication is provided via pheromone among ants is a critical part in ant colony optimization's pointed device. The algorithm design in such a way, with a slight adjustment in this part which removes the parallel pointed capability. Based on Ant-Miner, they propose an addition that modifies the algorithm design to incorporate parallel processing. The pheromone trail deposited by ants during the searching technique affected each other. With the help of pheromone, ants can have better decision making while searching. They provide a possible direction for researches toward the grouping rule problem.

Data mining is the extraction of hidden predictive information from large databases are powerful technology with great potential that helps to focus on the most important information in data warehouses. Modern medicine generates a great deal of information stored in the medical database. Extracting useful knowledge and providing scientific decision-making for the diagnosis and treatment of disease from the database increasingly becomes necessary. Data mining in medicine can deal with this problem. It can also improve the management level of Hospital information and promote the development of telemedicine and community medicine. Because the medical information is in nature of redundancy, multiattribution. Incompletion and closely related with time, medical data mining differs from other one.

In 2012, M. H. Mehta et al. observed that in engineering field, many problems are hard to solve in some definite interval of time. These problems known as "combinatorial optimization problems" are of the category NP. These problems are easy to solve in some polynomial time when input size is small but as input size grows problems become toughest to solve in some definite interval of time. Long known conventional methods are not able to solve the problems and thus proper heuristics is necessary. Evolutionary algorithms based on behaviors of different animals and species have been invented and studied for this purpose. Particle swarm optimization is a new evolutionary approach that copies behavior of swarm in nature. However, neither traditional genetic algorithms nor particle swarm optimization alone has been completely successful for solving combinatorial optimization problems. So the authors present a hybrid algorithm in which strengths of both algorithms are merged and performance of proposed algorithm is compared with simple genetic algorithm.

In 2012, Priyanka Dhasal et al. proposed a feature sampling technique of image classification. Their sampling technique optimized the feature selection process and reduced the unclassified region in multi-class classification. For the process of optimization they used ant colony optimization algorithm for the proper selection of feature sub set selection Support Vector Machines are designed for binary classification. When dealing with several classes, as in object recognition and image classification, one needs an appropriate multi class method. They also discuss about the possibilities which include: Modify the design of the SVM, as in order to incorporate the multi-class learning directly in the quadratic solving algorithm. Combine several binary classifiers: "One-against- One" (OAO) applies pair wise comparisons between classes, while "One-against-All" (OAA) compares a given class with all the others put together. OAO and OAA classification based on SVM technique is efficient process, but this SVM based feature selection generate result on the unclassified of data. When the scale of data set increases the complexity of preprocessing is also increases, it is difficult to reduce noise and outlier of data set.

In 2011, Yao Liu et al. implement a classifier using DPSO with new rule pruning procedure for detecting lung cancer and breast cancer, which are the most common cancer for men and women. Experiment shows the new pruning method further improves the classification accuracy, and the new approach is effective in making cancer prediction.

All researchers have aim to develop such a system which predict and detect the cancer in its early stages. Also tried to improve the accuracy of the Early Prediction and Detection system by preprocessing, segmentation feature extraction and classification techniques of extracted database. The major contributions of the research are summarized below.

3. METHODOLOGY

3.1. PSO Algorithm

Particle swarm optimization (PSO) is a stochastic global optimization technique developed by Beernaert and Kennedy in 1995 based on social behavior of birds [2]. In PSO a set of particles or solutions traverse the search space with a velocity based on their own experience and the experience of their neighbors. During each round of traversal, the velocity, thereby the position of the particle are updated based on the above two parameters. This process is repeated till an optimal solution is

obtained. According to the original PSO the particle velocity and position are updated according to the following equations.

$$v_k^{[n+1} = v_k^{[n} + c_1 r_1 (pbest_k^{[n} - p_k^{[n}) + c_2 r_2 (gbest^{[n} - p_k^{[n}) \quad (1)$$

$$x_k^{[n+1} = x_k^{[n} + v_k^{[n+1} \quad (2)$$

where v k i n and p k i n are the velocity and position of kth particle in ith dimension during nth iteration, pbest is the best position experience by the particle upto that iteration and gbest is the best position experience by all particles upto that iteration. The best positions of a particle are evaluated according to a fitness function.c1, c2 are called acceleration constants usually equal to 2 and r1 and r2 are random numbers uniformly distributed in (0, 1). Thus these constants are a measure of inertia experienced by the particle. The PSO developed by Eberhart and Kennedy is suited for continuous optimization problems.

The current problem requires a discrete version of the PSO as the features here are genes which are discrete entities. To address this problem Q.Shen [15] developed a discrete version of PSO and applied it to gene selection. Each particle contains n number of features wherein each feature or position is assigned 0 or 1. An assignment of 1 corresponds to the selection of the feature and an assignment of 0 corresponds to its rejection. In Shen's approach velocity of a particle in a dimension for a given iteration is generated randomly between 0 and 1. Thereby position of each particle is updated according to the following rules,

$$0 \le v_k^{[n} < 0.4;\ p_k^{[n}(new) = p_k^{[n}(old)$$

$$0.4 \le v_k^{[n} < 0.6;\ p_k^{[n}(new) = pbest_k^{[n}$$

$$0.6 \le v_k^{[n} < 1;\ p_k^{[n}(new) = gbest^{[n}$$

Yu et. al [18] also followed the same update rules as suggested by Shen. However to avoid converging to a local optimum they used a variable to store continuous unchangeable values of particle best values. If a particle has the same number of particle best values consecutively for a fixed number of times, the particle best was set to zero. This was done to allow the particles to escape local optima. Alba et. al used geometric particle swarm optimization which applied a 3- parent mask based crossover to move the particle [17].

The current approach however uses update rules for particles that differ from the ones used above. It uses a linear combination of current position, particle best position and global best position to determine the next position of a particle. Each particle position is a vector whose features are binary valued. For example (1, 0,1, 1,1,0,0…..1) is a position vector of the particle where 1 represents selection of the corresponding gene and 0 represents rejection. The subsequent position vector is determined by a linear combination of three vectors, the particle's current position vector, best position vector of the particle and the best position vector among all Particles.

$$x_k^{[n+1} = w_1 x_k^{[n} + w_2 pbest_k^{[n} + w_3 gbest^{[n}$$

Particle Swarm Optimization was first proposed by Kennedy and Eberhart in 1995 [13]. PSO is a population based evolutionary algorithm inspired in the social behavior of bird flocking or fish schooling. In the description of PSO, the swarm is made up of a certain number of particles (similar to population of individuals in EAs). At each iteration, all the particles move in the problem space to find the global optima. Each particle has a current position vector and a Velocity vector for directing its movement.

$$v_i^{k+1} = \omega \cdot v_i^k + \varphi_1 \cdot rnd_1 \cdot (pBest_i - x_i^k) + \varphi_2 \cdot rnd_2 \cdot (g_i - x_i^k)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1}$$

Equations 2 and 3 describe the velocity and position update of a given particle i at a certain iteration k. Equation 2 calculates a new velocity vi for each particle (potential solution) based on its previous velocity, the particle's location at which the best fitness so far has been found pBesti,And the population global (or local neighborhood, in the neighborhood version of the algorithm) location at which the best fitness so far has been achieved gi. Individual and social weight is represented by means of '1 and '2 factors respectively. Finally, rnd1 and rnd2 are random numbers in range {0, 1}, and! Represents the inertia weight factor. Equation 3 updates each particle's position xi in solution space.

In this version, the location of each particle i is represented as vector xi = hxi1, xi2, ..., xenia taking each bit xij (with j in {1,N}) binary values 0 or 1. The key issue of the GPSO is the concept of particle movement. In this approach, instead of the notion of velocity added to the position, a*three-parent mask-based crossover* (3PMBCX) operator is applied to each particle in order to "move" it. According to the definition of 3PMBCX [14], given three parents a, b and c in {0, 1}n, generate randomly a crossover mask of length n with symbols from the alphabet {a, b, c}. Build the offspring filling each element with the bit from the parent appearing in the crossover mask at the position.

The pseudo code of the GPSO algorithm for Hamming spaces is illustrated in Algorithm 1. For a given particle i, three parents take part in the 3PMBCX operator (line 13): the current position xi, the social best position gi and the historical best position found hi (of this particle). The weight values w1, w2 and w3 indicate for each element in the crossover mask the probability of having values from the parents xi, gi or hi

respectively. These weight values associated to each parent represent the *inertia* value of the current position (w1), the *social* influence of the global/local best position (w2) and the *individual* influence of the historical best position found (w3). A constriction of the geometric crossover forces w1, w2 and w3 to be non-negative and add up to one.

In summary, the GPSO developed in this study operates as follows: In a first phase of the pseudo code, the initialization of particles is carried out by means of the Swarm Initialization () function (Line 1). This special initialization method (used also in our GA approach) was adapted to gene selection as follows. The swarm (population) was divided into four subsets of particles (chromosomes) initialized in different ways depending on the number of features in each particle. That is, 10% of particles were initialized with N (prefixed value) selected genes (1s) located randomly. Another 20% of particles were initialized with 2N genes, 30% with 3N genes and finally, the rest of particles (40%) were initialized randomly and 50% of the genes were turned on. In these experiments N will be equal to 4. In,

1: $S \leftarrow SwarmInitialization()$
2: **while** not stop condition **do**
3: **for** each particle x_i of the swarm S **do**
4: evaluate(x_i)
5: **if** $fitness(x_i)$ is better than $fitness(h_i)$ **then**
6: $h_i \leftarrow x_i$
7: **end if**
8: **if** $fitness(h_i)$ is better than $fitness(g_i)$ **then**
9: $g_i \leftarrow h_i$
10: **end if**
11: **end for**
12: **for** each particle x_i of the swarm S **do**
13: $x_i \leftarrow 3PMBCX((x_i, w_1), (g_i, w_2), (h_i, w_3))$
14: mutate(x_i)
15: **end for**
16: **end while**
17: **Output:** best solution found

3.1.1. Neural Networks (NN):

An artificial neural network is a mathematical model based on biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation.

Working of Neural Network:

Create neural network

Train neural network

Test targets

Cancer samples classified as cancerous

Cancer samples classified as normal

Normal samples classified as normal

Normal samples classified as cancerous

Classification matrix in percentage.

3.2. Proposed Methods

3.2.1. Cat Swarm Optimization (CSO)

Chu et al. [3] divided CSO algorithm into two sub models based on two of the major behavioral traits of cats. These are termed "seeking mode" and" tracing mode". In CSO, we first decide how many cats we would like to use in the iteration, then we apply the cats into CSO to solve the problems. Every cat has its own position composed of D dimensions, velocities for each dimension, a fitness value, which represents the accommodation of the cat to the fitness Function, and a flag to identify whether the cat is in seeking mode or tracing mode. The final solution would be the best position of one of the cats. The CSO keeps the best solution until it reaches the end of the iterations.

A. Seeking Mode

This sub model is used to model the cat during a period of resting but being alert- looking around its environment for its next move. Seeking mode has four essential factors, which are designed as follows: seeking memory pool (SMP), seeking range of the selected dimension (SRD), counts of dimension to change (CDC) and self position consideration (SPC).Seeking mode according to Chu et al. [3] is described below.

Step 1: Make j copies of the present position of cat k, where j = SMP. If the value of SPC is true, let j = (SMP − 1), then retain the present position as one of the candidates.

Step 2: For each copy, according to CDC, randomly plus or minus SRD percents the present values and replace the old ones.

Step 3: Calculate the fitness values (FS) of all candidate points.

Step 4: If all FS are not exactly equal, calculate the selecting probability of each candidate point by equation (1), otherwise set all the selecting probability of each candidate point be 1.

Step 5: Randomly pick the point to move to from the candidate points, and replace the position of cat k,

$$P_i = \frac{|SSE_i - SSE_{max}|}{SSE_{max} - SSE_{min}}, \quad 0 < i < j$$

If the goal of the fitness function is to find the minimum solution, FSb = FSmax, otherwise FSb = FSmin.

B. Tracing Mode

Tracing mode is the sub-model for modeling the case of the cat in tracing targets. The action of tracing mode according to Chu et al. [2] can be described as follows:

Step 1: Update the velocities for every dimension (vk,d) according to equation (2).

Step 2: Check if the velocities are in the range of maximum velocity. In case the new velocity is over-range, it is set equal to the limit.

Step 3: Update the position of catk according to (3).

$$v_{k,d}=v_{k,d}+r_1\times c_1\times(x_{best,d}-x_{k,d})$$

Where *best d x* , is the position of the cat, who has the best fitness value; *xk,d* is the position of catk, c1 is a constant and r1 is a random value in the range of [0, 1].

$$x_{k,d}=x_{k,d}+v_{k,d}$$

C. Core Description of CSO

To combine these two modes into the algorithm, we define a mixture ratio (MR) which dictates the joining of seeking mode with tracing mode.CSO Clustering proposed in this research generally consists of two main parts, clustering data and searching for the best cluster center using CSO algorithm. The inputs for clustering CSO will be the population of data that are going to be clustered, number of cluster k, and number of copy (will be used in seeking mode). Steps of clustering CSO are described below.

Step 4.1: Seeking mode

Seeking mode is intended to look for points in an area around the cluster center which have possibilities resulting a more optimal fitness value.

There are three parameters need to be defined. SMP (seeking memory pool), SRD (seeking range of the selected dimension), and SPC (self-position considering). SMP represents how many copy will a cluster center has. SRD declares the mutative ratio, with a value between [0,1]. SPC is a Boolean random value {0, 1}.Seeking mode starts with making SMP copy of the present cluster center position. Then defining *j* value. j value represents how many copy of cluster center *i* that will experience mutation. If the value of SPC = 1, let j = (SMP –1) then retain the present sition as one of the candidates. The next step will be calculating the mutative value that is (*SRD* x *cluster center*). This step will give (*SMP* x *k*) candidates of cluster center as the output. For every cluster center candidates do step 2 and 3. After we get the SSE value calculate the selecting probability of each candidate point by

(1) Based on their SSE value.

Pick the new cluster center from the candidate points by using Roulette Wheel Selection method. Candidate with the biggest P value will have the biggest opportunity to be chosen. Figure 1 shows the algorithm for Seeking Mode in CSO-clustering 1. Define the parameter of seeking mode (SMP, SRD, and SPC)

2. For i = 1 to k (number of cluster center), do Copy cluster center (i) position as many as SMP

Determine j value Calculate the shifting value (SRD*cluster center (i))

3. For m = 1 to SMP, do randomly plus or minus cluster centers with shifting value.

/the output will be (SMP x k) cluster center candidates/

4. Calculate the distance, grouping data into clusters, and calculate SSE

5. Choose a candidate to be the new cluster center r oulette wheel selection Figure 1. Algorithm for Seeking Mode - CSO Clustering

Step 4.2: Updating SSE and cluster center

The value of SSE obtained from seeking mode then compared with the previous value of SSE, if seeking SSE < earlier SSE then the cluster center resulting from seeking will become the new cluster center. Conversely, if the value of seeking SSE ≥ earlier SSE, use the previous cluster center.

Step 4.3: Tracing Mode

Tracing mode is intended to shift the point so it will be concentrated to a better position with a more optimal fitness value. Tracing mode starts with updating velocity value, using (2) where xbest is mean value in a *cluster*. Then, updating the position of cluster center by adding it with the velocity value, according to (3). For each cluster center, do step 2 and 3. The output will be SSE value and best cluster for each data.

Figure 2 shows the Tracing Mode algorithm for CSO Clustering 1. For i = 1 to k, do

Update velocity (i) Update position (i), get the new cluster center (i)

2. Calculate the distance, grouping data into clusters, and calculate SSE Figure 2. Process Chart of Tracing Mode - CSO Clustering

Step 4.4: Repeat step 4.2 for tracing SSE and cluster center

The value of SSE obtained from tracing mode then compared with the previous value of SSE, if tracing SSE < earlier SSE then the cluster center resulting from tracing will become the new cluster center. Conversely, if the value of tracing SSE ≥earlier SSE, use the previous cluster center.

Step 5: Repeat step 4 until it reach the stopping criteria.

Shows the complete algorithm for CSO-Clustering until stopping criteria is met do 1 to 8

1. Define the population of data, number of cluster (k), and number of copy

2. Choose k data as initial cluster center

3. Grouping data into cluster by their closeness, and calculate SSE

4. Initialize CSO parameter

5. Enter seeking mode

6. Compare seeking SSE with earlier SSE. if seeking SSE < earlier SSE use new cluster center. Conversely, use the previous cluster center

7. Enter tracing mode

8. Compare tracing SSE with earlier SSE. if tracing SSE < earlier SSE use new cluster center. Conversely, use the previous cluster center

9. Get SSE, cluster center, and best cluster of each data.

After seeking mode and tracing mode are performed, cats are reassigned between these two modes. Here, we randomly select some cats into the tracing mode according to mixture ratio, and then set the others into the seeking mode. The reassignment of cats is described as follows. mr R .

Step 1: Given the population after seeking mode and tracing mode, set. 1xi

Step 2: Cat is randomly assigned into seeking mode or tracing mode according to mixture ratio. i X mr R,

4. RESULTS

S.No	Algorithm	Accuracy	Time period
1	ACO	89.5	3.61
2	MPSO	92.4	2.52
3	MCSO	95.2	1.4

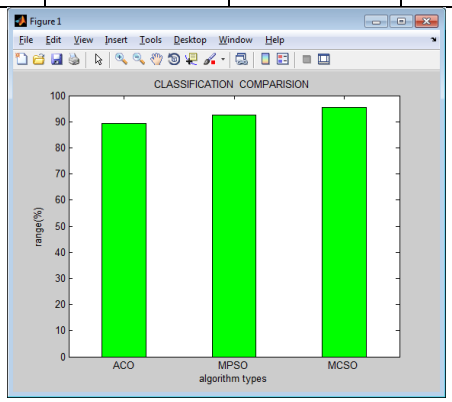

Figure 1 show Accuracy comparison

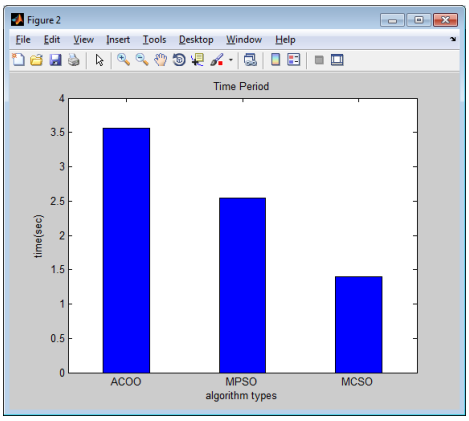

Figure 2 show Time period comparison

5. CONCLUSIONS

We have conferred CSO algorithmic program for bunch and testing on 3 datasets. From our study we are able to conclude that the accuracy level of CSO bunch has no correlation with variety of iteration during a vary fifty – one hundred fifty. Modification in speed change formula, that is that the addition of inertia variable (W) as a speed number, can be thought-about to induce a much better robust an improved} and more correct cluster, although there square measure increasing on computer hardware time. It takes additional CPU time to try to to the CSO bunch than k-means and PSO clustering, it's as a result of CSO bunch includes a longer and more sophisticated algorithmic program. CSO bunch has higher accuracy on bunch information with little variety of clusters, conversely not higher than the opposite technique on bunch large data. CSO bunch is thought-about as sufficiently accurate bunch technique; however it takes longer time to try to to the computation. Developing the analysis targeted on creating the result of CSO bunch less random, for instance by doing a hybrid and applying CSO bunch algorithmic program for bunch real information from the important drawback.

REFERENCES

[1] Lv T., Huang S., Zhang X., and Wang Z,Combining Multiple Clustering Methods Based on Core Group. Proceedings of the Second International Conference on Semantics, Knowledge and Grid (SKG'06), pp: 29-29,2006.

[2] Nock R., and Nielsen F., On Weighting Clustering. IEEE Transactions and Pattern Analysis and Machine Intelligence, 28(8): 1223-1235, 2006.

[3] Xu R., and Wunsch D., Survey of clustering algorithms. IEEE Trans. Neural Networks, 16 (3): 645-678, 2005.

[4] MacQueen J., Some methods for classification and analysis of multivariate observations. Proc.5th Berkeley Symp. Math. Stat. and Prob, pp:281-97, 1967.

[5] Kanungo T., Mount D.M., Netanyahu N., Piatko C., Silverman R., and Wu A.Y., An efficient kmeans clustering algorithm: Analysis and implementation. IEEE Trans. Pattern Analysis and Machine Intelligence, 24 (7): 881-892,2002.

[6] Pelleg D., and Moore A., Accelerating exact kmeans algorithm with geometric reasoning. Proceedings of the fifth ACM SIGKDD International

Conference on Knowledge Discovery and Data Mining, New York, pp.727-734, 1999.
[7] Sproull R., Refinements to Nearest-Neighbor Searching in K-Dimensional Trees.Algorithmica, 6: 579-589, 1991.
[8] Bentley J., Multidimensional Binary Search Trees Used for Associative Searching.Commun. ACM, 18 (9): 509-517, 1975.
[9] Friedman J., Bentley J., and Finkel R., An Algorithm for Finding Best Matches in Logarithmic Expected Time. ACM Trans. Math.Soft. 3 (2): 209-226, 1977.
[10] Elkan, C., Using the Triangle Inequality to Accelerate k-Means. Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), pp. 609-616, 2003.
[11] Hernandez, C.A. et al."How to Choose the Training Data for Neural Network Medical Diagnosis Systems", ISA, pp. 283-290 (1993).
[12] I. Sluimer, A. Schilham, M. Prokop, and B. Ginneken, "Computer Analysis of Computed Tomography Scans of Lungs: A Survey, " IEEE Transactions on Medical Imaging, vol. 25, no. 4, 2006.