A Novel Heuristic for Evolutionary Clustering

Pranav. A. Nerurkar
Research Scholar
Dept. of Computer Engineering & IT
Veermata Jijabai Technological Institute
Mumbai - 400019, Maharashtra
Email: pranav.nerurkar@upscfever.com

Sunil. G. Bhirud
Professor
Dept. of Computer Engineering & IT
Veermata Jijabai Technological Institute
Mumbai - 400019, Maharashtra
Email: sgbhirud@vjti.org.in

***Abstract*—Clustering is considered a challenging problem of data mining due to its unsupervised nature. The literature is inundated with algorithms and concepts related to determining the most suitable clustering structure in data. These techniques have a mathematical model of a cluster and attempt to obtain a result that shall represent this model as closely as possible. However as the problem of clustering is NP hard such strategies have disadvantages such as converging to local optima or suffering from the curse of dimensionality. In such scenario, meta heuristics could be more suitable strategies. Such techniques utilizes biologically inspired techniques such as swarm intelligence, evolution etc. to traverse the search space. Due to their inherent parallel nature, they are most robust towards converging to a local optima. The objective (cost) function used by such meta heuristics is responsible for guiding the agents of the swarm towards the best solution. Hence it should be designed to achieve trade-off between multiple objectives and constraints and at the same time produce relevant clustering. In this paper, a cost function is proposed (PSO-2) to produce compact well separated clusters by using the concept of intracluster and intercluster distances. Experiments have been performed on artificial benchmark datasets where performance of the particle swarm optimizer using the proposed cost function is evaluated against other evolutionary and non evolutionary algorithms. The clustering structures produced by the methods have been evaluated using distance based and internal cluster validation metrics to demonstrate that the performance of PSO-2 is comparable to other techniques.**

***Index Terms*—Data clustering, Applications of evolutionary algorithms, Genetic algorithm.**

I. Introduction

Clustering involves creating coarse grained descriptions of data. This is done with the main purpose of highlighting a set of autonomous regions in the data known as clusters. Clustering has found wide applicability in multiple fields ranging from web mining and information retrieval through customer segmentation and bio-informatics [1]. Regardless of the domain to which it is applied or the algorithm used, the goal of clustering is uniform i.e. to maximize homogeneity within cluster and heterogeneity between clusters [2]. Clustering algorithms produce partitions of data which may be hard (nodes have unique memberships to clusters), soft (nodes have memberships to multiple clusters) or fuzzy (nodes have varying degrees of memberships to every cluster). Thus, a hard partition of a dataset X = $x_1, x_2, x_3, x_4 ... x_n$ where x_i is a data point which stands for a n-dimensional feature vector, is a collection $C = C_1, C_2, ..., C_k$ of non null clusters such that $C_i \cap C_j = \phi$ for $i \neq j$. Relaxing the mutual dis-junction condition to allow $C_i \cap C_j \neq \phi$ for $i \neq j$ creates overlapping partitions [3].

Clustering is considered the most challenging task of machine learning due to the unavailability of spatial distribution of the data in terms of its clustering tendency [1] [2] [3]. Furthermore, the requirement of dealing with various types of attributes (binary, categorical, continuous, discrete), their conditions (complete or missing) and scales (nominal, ratio, interval, ordinal) has to be factored in while deciding the approach of clustering. In addition to this, the lack of information about the orientation of the clusters, their numbers, shapes, densities and volumes makes it difficult in selecting a particular clustering technique and in evaluating the results obtained by it [16] [17]. From the optimization perspective, Clustering is considered as a NP-hard grouping problem and the literature consists of a wide range of objective functions, approximation algorithms and heuristics for solving it [1].

Each approach has its own bias and comes with certain advantages and disadvantages to a given analysis or application scenario. Popular partition based clustering algorithms are K-means, PAM, CLARA, CLARANS. These algorithms have a objective function which is non-convex and hence the results might be locally optimal. Hierarchical algorithms too exist like AGNES, DIANA, BIRCH, ROCK and CHAMALEON etc. but these are computationally expensive. One inherent weakness of hierarchical clustering is that clusters formed in an iteration cannot be undone in subsequent iterations. DBSCAN, OPTICS are approaches that define clusters as dense regions separated by less dense regions. DBSCAN though is popular but the prior specification on ϵ and $MinPts$ makes it sensitive to parameter tuning. In real life experiments, various types of noise are introduced at different experimental stages during data collection, due to which the performance of these algorithms are less than ideal [1] [3]. To effectively handle noisy data, we need an algorithm that is able to overcome noise. These traditional algorithms are based on gradient based methods and are difficulty to extend to multi-objective problems as their basic design

doesn't allow the consideration of multiple solutions [4] [5]. Population based methods such as evolutionary algorithms have an inherent advantage here [7] [8].

In the context of solving NP-hard optimization problems, evolutionary algorithms are considered to be particularly effective in obtaining near optimal solutions in reasonable time [9] [10]. Evolutionary algorithms are meta-heuristics based on optimization of a fitness function that guides the process of evolutionary search. Computational advantages are also present as the algorithms can be parallelized leading to increased coverage and possibly faster convergence [6]. However, evolutionary algorithms have their owns drawbacks as different features of these algorithms have to chosen in advance such as encoding scheme for the data points, choice of operators for crossover and mutation and a fitness (cost) function. Fitness functions are basically validity criteria and hence it is better to use multiple criteria in order to avoid the drawbacks of individual ones. One strategy for this is to assign weights to individual criteria and then optimize, but this approach only works when the multiple criteria are commensurable. In this paper, the fitness function proposed computes cost using a variation of intra cluster distance and inter cluster distance. This variation allows computation to be performed in linear time. The conventional definitions of these distances requires $O(n^2)$ computations. Hence a reduction in computations is achieved leading to faster execution.

The rest of this paper is organized as follows. In Section II, an overview of the existing clustering algorithms used for identification of clusters in data is presented, these include both evolutionary and non evolutionary approaches. In Section III, the mathematical model of the proposed evolutionary algorithm and its intuition is described. In Section IV, the algorithm is evaluated and compared with existing clustering algorithms using benchmark artificial datasets. Experimental results are presented to highlight the suitability of the proposed technique. Section V consists of the summary of the work along with the conclusion.

II. Related Work

Clustering algorithms follow evolutionary and non evolutionary strategies and a selection of the best known algorithms [3] of both categories are described below.

A. Traditional Clustering Algorithms

Clustering techniques belonging to the k-partitioning family minimize the objective criterion known as variance or sum of squared distances $SSD = \sum_{k=1}^{k} \sum_{x_i \epsilon c_k} \|x_i - c_k\|^2$ and hence the detected clustering structures correspond to minimum variance. K-Means is the popular algorithm of this category with a time complexity of $O(kN)$ and hence is scalable. Hierarchical clustering algorithms have a fully unsupervised approach but time complexity is higher i.e. $O(n^3)$ and $O(n^2 logn)$ for priority queue implementations. This makes scalability an issue on larger datasets. Model based clustering algorithms assume that clusters are generated by probability distributions whose parameters such as mean and covariance matrix have to be estimated using maximum likelihood estimation. Eqn. 1 is the prior probability that denotes the percentage of instances that came from a cluster c. Eqn. 2 gives the mean i.e. expected value of attribute j from cluster c. Eqn. 3 gives the covariance matrix denoting the covariance of attributes j, k in cluster c.

$$P(c) = \frac{1}{n} \sum_{i=1}^{n} P(c|\vec{x_i}) \tag{1}$$

$$\mu_{c,j} = \sum_{i=1}^{n} \Big(\frac{P(c|\vec{x_i})}{nP(c)} \Big) x_{i,j} \tag{2}$$

$$\sum_{c} {}_{j,k} = \sum_{i=1}^{n} \Big(\frac{P(c|\vec{x_i})}{nP(c)} \Big) (x_{i,j} - \mu_{c,j})(x_{i,k} - \mu_{c,k}) \tag{3}$$

$$P(c|\vec{x_i}) = \frac{P(\vec{x_i}|c)P(c))}{\sum_{i=1}^{k} P(\vec{x_i}|c)P(c)} \tag{4}$$

$$P(\vec{x_i}|c) = \frac{1}{\sqrt{2\pi \sum_c}} \exp\Big(-\frac{1}{2} (\vec{x_i} - \vec{\mu_c})^T \sum_{c}^{-1} (\vec{x_i} - \vec{\mu_c}) \Big) \tag{5}$$

B. Neural Network Based Clustering

Unsupervised learning using neural networks has two main models which are Kohenon's Self Organizing Maps and Grossberg's Adaptive Resonance Theory. ART represents a family of neural networks in which the current input vector is matched to the category prototype vector. If the input vector doesn't match with the prototype then a new prototype is selected. This means that other prototypes are not affected by new input. In ART systems, there are two layers: $L1$ has the comparison field where input patterns and expectations are compared. Layer 2 $L2$ is the competition layer where winner takes all learning strategy is implemented. Connections from Layer 1 to Layer 2 perform clustering operations and each row of weights is a prototype pattern. Layer 2 to Layer 1 connections perform pattern recall (expectation). The vigilance parameter is a user defined gain control mechanism that controls the degree of similarity required for patterns to be assigned to the same cluster.

Kohenon's Self Organizing Map are neural networks that map their weights (without any target vector) to conform to the given input data with the goal of representing multidimensional data in a lower dimension. SOM's have a two layered architecture with input nodes connected to every computational nodes to form a lattice pattern. Weight vectors $W_{ij} = W_{ij1}, W_{ij2}, ..., W_{ijn}$ are of the same dimensions as the input vectors V. The SOM algorithm has following steps:

- Each nodes weights are initialised randomly.
- A vector is chosen at random from the set of input vectors and presented to the training data.
- Every nodes is compared to the input pattern to determine which nodes weights are closest to the input pattern. This winning node is the Best Matching Unit (codebook vector). Closeness measure is euclidean distance.
- Once the BMU is identified, codebook vectors in the neighbourhood of the BMU are updated. The update rule for the codebook vectors in the neighbourhood is given in Eqn.6 is the BMU, h_{ci} is a non increasing neighbourhood function.
- The neighbourhood function is given in Eqn. 7

Where,

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \tag{6}$$

$$h_{ci}(t) = \alpha(t).exp(-\frac{||r_i - r_c||^2}{2\sigma(t)^2}) \tag{7}$$

C. Evolutionary Clustering Algorithms

Particle swarm optimization is an intelligent optimization algorithm which belongs to a class of optimization algorithms called Meta Heuristics. It is based on the paradigm of "swarm intelligence" and is inspired by social behavioral animals like birds or fish proposed by J. Kennedy *et al* in 2007 and modified in 2011 by M. Clerc *et al* [11][12][13]. Swarm intelligence systems have simple agents that interact locally with each other and with the environment (without any central control as to how individual agents should behave). Local interaction between the agents however lead to emergence of a complex global behavior. In PSO, solutions are assumed as points in an n-dimensional space. The swarm particles $\overrightarrow{x_i}$ are solutions that are randomly initialized in the search space X and these have to traverse it in search of the global optima with a velocity v_i.

Differential evolution [14] is a stochastic direct search method that minimizes the objective (cost) function by using the difference of solution vectors to create new candidate solutions. NP D-dimensional parameter vectors $x_{i,G}i = 1, 2, ..., NP$ are used as a population for each generation G. DE generates new parameter vectors by adding the weighted difference between two population vectors to a third vector. The parameters of this new vector also called the "mutated vector" are mixed with the parameters of another predetermined vector, the target vector, to obtain the "trial vector". If the trial vector results in a lower cost than the target vector it replaces the target vector in the subsequent generation. At every stage of the process each population vector has to serve once as the target vector so that NP competitions take place in one generation.

Genetic algorithms are a class of computational models inspired by biological evolution and selection. The mechanism followed by these models is to encode a potential solution to a problem on a chromosome like data structure and apply recombination operators to preserve critical information. Problem dependent parts are selection of suitable objective function and problem encoding. The optimization problems are often non linear and so it is not possible to treat each parameter as an independent variable which can be solved in isolation from others [15].

III. Mathematical Model

The mathematical model of a meta heuristic algorithm has structured sections with each section having a logical meaning. The sections in the meta heuristic in this section are Problem definition, Parameters definition, Initialization and the Iterative procedure. The meta heuristic follows the same structure as the standard PSO implemented in [12] [13].

A. Problem Definition

1) Cost Function: The key component of the problem definition is the cost function which is to be optimized using the meta-heuristics. The aim is to achieve clusters well separated and compact. This is done using two distances WCD and ICD.

$$WCD = \sum_{i=1}^{n} \text{dist}(c_i, C) \tag{8}$$

$$ICD = \sum_{j=1}^{k} \sum_{i=1}^{n} \text{dist}(c_i, C_j) \tag{9}$$

Thus WCD is the distance of a point from its centroid summing over all points. ICD is the distance of point from other centroids summing over all points. WCD is to be minimised and ICD maximized, and this is the objective of PSO-2. To achieve efficiency the cost to be minimized is taken as $\frac{WCD}{ICD}$.

Algorithm 1: Cost function: PSO-2

Result: Cost of the cluster: z
1 m = coordinates of centroids;
2 X = coordinates of data points;
3 M = Calculate distance of each point from all centroids;
4 ind = Assign point to centroid closest to it;
5 WCD = Add distances of all points to assigned to a centroid;
6 ICD = M - WCD;
7 z = WCD / ICD;

The dissimilarity matrix isn't required to be calculated and the computations for WCD , ICD are completed in $O(n)$ computations.

2) Decision Variables: The decision variables are the k centroids of the data, where each centroid can be represented as a n dimensional vector $\overrightarrow{R^n}$. Thus the total decision variables are $n * k$. The range of the decision variables is limited to be between the maximum $VarMax$ and minimum value $VarMin$ of the points in the data, this also achieves the objective of restricting the search space.

B. Parameters of Meta heuristics

The parameters of the meta heuristics are maximum iterations allowed for the model, the size of the swarm, coefficients of inertia w and its damping ratio w_{damp} , personal acceleration coefficient c_1 (assigned to every particle) and the social acceleration coefficient c_2 (assigned to the entire swarm). In addition, limits are set to restrict the velocities of individual particles to the range $VelMin$ to $VelMax$. Constrictions coefficients phi_1, phi_2, phi, chi are also defined as per the Eqn.10 and Eqn.III-B. The values for these variables are set according to standard settings which may be modified.

$$phi = phi_1 + phi_2 \tag{10}$$

$$chi = \frac{2}{phi - 2 + \sqrt{phi^2 - 4 * phi}} \tag{11}$$

$$w = chi \tag{12}$$

$$c_1 = phi_1 * chi \tag{13}$$

$$c_2 = phi_2 * chi \tag{14}$$

C. Initialization and the Iterative Procedure

Each particle in the swarm has the components such as position x_{pos}, velocity x_{vel}, cost x_{cost}, best position x_{pbest} and best cost x_{cbest}. The Global best x_{gbest} is also defined to be $-\infty$ as the objective function is to be maximized. The positions of the particles are initialized using uniform distribution between $VarMax$ and $VarMin$. The velocities of the particles is initialized to zero. The values for the best position of a particle and the cost at the best position is currently set to the initial position and the cost at the initial position respectively.

The iterative procedure updates the values for the best position and best cost obtained so far by each particle in the swarm and also update the variable that records the global best cost and position obtained by the swarm together. The update rule for the particle velocity is given by Eqn. 15

$$x_{i+1}(vel) = w*x_i(vel)+c_1*(x_{pbest}-x_{pos})+c_2*(x_{gbest}-x_{pos}) \tag{15}$$

The velocity limits are applied as in Eqn: 16 and 17 to prevent the particle velocity from increasing or decreasing beyond the thresholds.

$$x_{i+1}(vel) = max(x_{i+1}(vel), VelMin) \tag{16}$$

$$x_{i+1}(vel) = min(x_{i+1}(vel), VelMax) \tag{17}$$

The position of the particle is updated by the Eqn 18

$$x_{i+1}(pos) = x_i(pos) * x_{i+1}(vel) \tag{18}$$

If the position of the particle is outside the thresholds $VarMax, VarMin$ it is updated by the rule in Eqn: 19 and 20

$$x_{i+1}(pos) = max(x_{i+1}(pos), VarMin) \tag{19}$$

$$x_{i+1}(pos) = min(x_{i+1}(pos), VarMax) \tag{20}$$

If the cost of the particle at its new position is better than the cost of the particle at its old position, then the values of personal best cost x_{cbest} and position x_{pbest} so far of the particle are updated to the new values. The values of the global best cost and position of the entire swarm is updated at every iteration to store the best global value at every iteration.

IV. Experiments

The efficiency of the objective function proposed in this paper is demonstrated vis-a-vis other evolutionary and non evolutionary algorithms on artificial clustering datasets. Standard internal cluster validation metrics such as Separation index S_i, Calinski and Harabasz index CH_i, Entropy of the distribution of cluster memberships E, Pearson Gamma index P and Dunn index D along with distance based statistics such as Within cluster sum of squared distances M_{wcd}, Average distance between clusters M_{avg-CD} and Average silhouette width S_w are used to evaluate [16] [17].

A. Dataset

The artificial benchmarks used for the experiment are given below:

TABLE I
Description of the datasets

Sr. No	Name	Size
1	Banana	4811
2	Cluto	8000
3	Cure	4200
4	Long	1261
5	2d-4c	1000
6	Demo	300

B. Results

1) Distance based Statistics: The within-cluster sum of squares is a measure of the variability of the observations within each cluster. Clusters that have higher values exhibit greater variability of the observations within the cluster. The results of PSO-2 are comparable to other clustering algorithms as seen in Table II.

TABLE II
WITHIN CLUSTER SUM OF SQUARES

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	185.20	3.42	383.25	637.02	20434.32	593.39
PAM	192.88	3.42	402.30	666.56	20434.32	593.39
ART2	190.09	37.78	1712.27	594.87	1127417	996.77
DE	185.21	3.94	385.31	637.53	20434.32	593.39
PSO	185.21	3.43	383.36	637.53	20434.32	593.39
GA	185.21	3.42	383.41	637.52	20434.32	593.39
PSO-2	193.23	3.98	403.29	1051.73	94470.4	**593.38**

Average distance between clusters should be large as this would mean that the clusters are well separated. Table III shows the that PSO-2 produces well separated clusters comparable to other algorithms.

TABLE III
AVERAGE DISTANCE BETWEEN CLUSTERS

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.49	300.36	1.22	1.84	62.26	5.38
PAM	0.50	300.35	1.22	1.85	62.26	5.37
ART2	0.5	NaN	1.18	1.72	**74.44**	4.92
DE	0.49	301.83	1.22	1.85	62.26	5.37
PSO	0.49	300.36	1.22	1.85	62.26	5.37
GA	0.49	300.36	1.22	1.85	62.26	5.37
PSO-2	0.50	**303.46**	**1.22**	**2.26**	66.9	5.37

Silhouette value is a measure of how similar an object is to its own cluster compared to other clusters. The silhouette ranges from 1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters. Clusters produced by PSO-2 have moderate scores of silhoutte width compared to other algorithms as seen in Table IV.

TABLE IV
AVERAGE SILHOUETTE WIDTH

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.46	0.43	0.48	0.43	0.87	0.61
PAM	0.47	0.43	0.45	0.42	0.87	0.61
ART2	0.47	NULL	0.08	0.33	0.56	0.33
DE	0.46	0.41	0.48	0.43	0.87	0.61
PSO	0.46	0.43	0.48	0.43	0.87	0.61
GA	0.46	0.43	0.48	0.43	0.87	0.61
PSO-2	**0.47**	0.41	0.44	0.36	0.86	0.61

2) Internal Cluster Validation Metrics: Separation index is computed on the distances for every point to the closest point not in the same cluster. The separation index is then the mean of the smallest proportion of these. Larger value of separation index indicates better clustering and PSO-2 has exceeded baseline SI set by other algorithms on two datasets as seen in Table V.

TABLE V
SEPARATION INDEX

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.03	9.27	0.04	0.1	15.68	0.85
PAM	0.03	8.79	0.04	0.10	15.68	0.85
ART2	0.02	Inf	0.04	0.06	50.92	0.27
DE	0.03	7.87	0.04	0.12	15.68	0.85
PSO	0.03	8.66	0.04	0.12	15.68	0.85
GA	0.03	8.59	0.04	0.12	15.68	0.85
PSO-2	0.02	8.45	0.03	**0.21**	43.43	**0.85**

Although a higher value of CH index as seen in Table VI is considered a better result, CH criterion is preferred where clusters are spherical. CH index value obtained by PSO-2 is comparable to other algorithms.

TABLE VI
CALINSKI AND HARABASZ INDEX

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	5689.28	11473.7	5683.4	1092.8	36113.1	679.9
PAM	5271.64	11447.1	5374.5	1000.1	36113.1	679.9
ART2	5419.3	NaN	517.4	619.8	730.5	229.0
DE	5688.92	9819.6	5648.6	1091.1	36113.1	679.9
PSO	5688.83	11450.1	5681.6	1091.1	36113.1	679.9
GA	5688.83	11453.2	5680.6	1091.1	36113.1	679.9
PSO-2	5253.3	9691.4	5359.3	268.3	11233.4	**679.9**

A higher value of entrophy as seen in Table VII indicates higher disorder in the clustering and a lower value is preferred. PSO-2 has produced clustering with lower value for entrophy than other algorithms on three datasets.

TABLE VII
ENTROPHY

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.69	2.08	1.78	0.69	1.23	1.1
PAM	0.65	2.08	1.77	0.68	1.23	1.1
ART2	0.66	0	1.31	0.99	0.38	1.34
DE	0.69	2.03	1.77	0.69	1.23	1.1
PSO	0.69	2.08	1.78	0.69	1.23	1.1
GA	0.69	2.08	1.78	0.69	1.23	1.1
PSO-2	**0.65**	**1.98**	1.76	**0.24**	0.98	1.1

Higher values of Pearson Gamma Coefficient obtained by PSO-2 indicates better clustering compared to other approaches on three datasets whereas performance on the remaining is in acceptable range as seen in Table VIII.

TABLE VIII
PEARSON GAMMA COEFFICIENT

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.66	0.51	0.62	0.53	0.89	0.78
PAM	0.67	0.51	0.61	0.51	0.891	0.78
ART2	0.67	NA	0.25	0.44	0.54	0.56
DE	0.66	0.51	0.62	0.53	0.89	0.78
PSO	0.66	0.51	0.62	0.53	0.89	0.78
GA	0.66	0.51	0.62	0.53	0.89	0.78
PSO-2	**0.67**	**0.53**	0.62	0.39	**0.93**	0.78

Higher value of Dunn index as seen in Table IX indicates better clustering results and PSO-2 has obtained clusters with dunn index comparable to those produced by other approaches.

TABLE IX
DUNN INDEX

Name	Banana	Cluto	Cure	Long	2d-4c	Demo
KMeans	0.01	0.004	0.002	0.003	0.45	0.06
PAM	0.003	0.003	0.005	0.01	0.45	0.06
ART2	0.002	Inf	0.002	0.004	0.605	0.01
DE	0.002	0.003	0.002	0.012	0.45	0.06
PSO	0.002	0.01	0.003	0.01	0.45	0.06
GA	0.002	0.001	0.003	0.01	0.45	0.06
PSO-2	**0.004**	0.003	**0.004**	0.005	**0.94**	0.05

V. CONCLUSION

Meta heuristics are considered as efficient alternatives to classical techniques in terms of traversing a search space which is non convex and not solvable in polynomial time. The inherent nature of such algorithms gives them capabilities such as parallelized implementation, fast convergence, ability to consider multiple solutions and the capability to avoid local optima. The objective function for such algorithms has to be chosen to achieve both fast computation as well as achieve a trade-off between multiple objectives. PSO-2 implemented in this paper seeks to uncover clustering structure while respecting constraints of efficiency. The cost function provides modified definitions of intra and inter cluster distances that are calculated in linear time. This achieves an improvement over the existing methods that need $O(n^2)$ to compute both the within and between cluster distances. Clustering structure uncovered by PSO-2 shows compact and well seperated clusters. It however fails to discover non convex clusters in the data. Yet the performance of PSO-2 is comparable to the standard PSO which is based on minimizing within cluster distance only. The experiments have demonstrated that clustering output provided by PSO-2 is comparable to other evolutionary as well as non evolutionary algorithms.

REFERENCES

[1] Hruschka, E.R., Campello, R.J. and Freitas, A.A., 2009. A survey of evolutionary algorithms for clustering. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 39(2), pp.133-155.

[2] Chakrabarti, Deepayan, Ravi Kumar, and Andrew Tomkins. "Evolutionary clustering." In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 554-560. ACM, 2006.

[3] Ma, P.C., Chan, K.C., Yao, X. and Chiu, D.K., 2006. An evolutionary clustering algorithm for gene expression microarray data analysis. IEEE Transactions on Evolutionary Computation, 10(3), pp.296-314.

[4] Zhang, Xingyi, et al. "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization." IEEE Transactions on Evolutionary Computation (2016).

[5] Luo, J., Jiao, L. and Lozano, J.A., 2016. A sparse spectral clustering framework via multiobjective evolutionary algorithm. IEEE Transactions on Evolutionary Computation, 20(3), pp.418-433.

[6] Lin, K.P., 2014. A novel evolutionary kernel intuitionistic fuzzy c-means clustering algorithm. IEEE Transactions on Fuzzy Systems, 22(5), pp.1074-1087.

[7] Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S. and Coello, C.A.C., 2014. A survey of multiobjective evolutionary algorithms for data mining: Part I. IEEE Transactions on Evolutionary Computation, 18(1), pp.4-19.

[8] Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S. and Coello, C.A.C., 2014. Survey of multiobjective evolutionary algorithms for data mining: Part II. IEEE Transactions on Evolutionary Computation, 18(1), pp.20-35.

[9] Li, H., Zhang, Q. and Deng, J., 2017. Biased multiobjective optimization and decomposition algorithm. IEEE transactions on cybernetics, 47(1), pp.52-66.

[10] Siddiqi, U.F. and Sait, S.M., 2017. A New Heuristic for the Data Clustering Problem. IEEE Access.

[11] Eberhart, R. and Kennedy, J., 1995, October. A new optimizer using particle swarm theory. In Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on (pp. 39-43). IEEE.

[12] Bratton, D. and Kennedy, J., 2007, April. Defining a standard for particle swarm optimization. In Swarm Intelligence Symposium, 2007. SIS 2007. IEEE (pp. 120-127). IEEE.

[13] Clerc, M., 2010. Particle swarm optimization (Vol. 93). John Wiley & Sons.

[14] Storn, R. and Price, K., 1997. Differential evolutiona simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization, 11(4), pp.341-359.

[15] Whitley, D., 1994. A genetic algorithm tutorial. Statistics and computing, 4(2), pp.65-85.

[16] Hennig, C. and Liao, T. (2013) How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification, Journal of the Royal Statistical Society, Series C Applied Statistics, 62, 309-369.

[17] Hennig, C. (2013) How many bee species? A case study in determining the number of clusters. In: Spiliopoulou, L. Schmidt-Thieme, R. Janning (eds.): "Data Analysis, Machine Learning and Knowledge Discovery", Springer, Berlin, 41-49.