

A boundary restricted adaptive particle swarm optimization for data clustering

S. Rana · S. Jasola · R. Kumar

Received: 25 July 2011 / Accepted: 18 May 2012 / Published online: 8 June 2012
© Springer-Verlag 2012

Abstract Data clustering is the most popular data analysis method in data mining. It is the method that parts the data object to meaningful groups. It has been applied into many areas such as image processing, pattern recognition and machine learning where the data sets are of many shapes and sizes. The most popular K-means and other classical algorithms suffer from drawback of their initial choice of centroid selection and local optima. This paper presents a new improved algorithm named as Boundary Restricted Adaptive Particle Swam Optimization (BR-APSO) algorithm with boundary restriction strategy. The proposed BR-APSO algorithm is tested on nine data sets, and its results are compared with those of PSO, NM-PSO, K-PSO and K-means clustering algorithms. It has been found that the proposed algorithm is robust, generates more accurate results and its convergence speed is also fast compared to other algorithms.

Keywords Adaptive PSO · Data clustering · Cluster centroid · K-means clustering · Nelder mean method

1 Introduction

Data Clustering is an unsupervised classification method whose aim is to part the data set into clusters on the basis of similarities and dissimilarities among the data objects. It helps to find the hidden relationship present in the data objects and divides them in the subsets which have some meaning in context with a particular problem [2].

The clustering algorithm falls in two categories: hierarchical clustering and partitional clustering. The hierarchical clustering approach is a group of nested clusters arranged as a tree known as dendrogram. Each cluster node of the tree consists of child nodes [26]. The idea is to find the data on different levels of granularity. In contrast, partitional clustering methods assume the given number of clusters to be found and then look for the optimal partition based on the objective function. The hierarchical clustering can be considered as a series of partitional clustering and any member of hierarchical clustering can be considered as partitional clustering [8].

Data clustering relies on many assumptions, characteristics, information found in the data that explain the data set and its relationship to each other. The objective of clustering algorithm is to maximize the inter cluster distance and minimize the intra cluster distance as shown in Fig. 1 [3].

K-means clustering is one of the older partitioning methods, which is commonly used to automatically divide the data set into k groups. K-means algorithm generates a fast and efficient solution. The basic K-means algorithm works with the objective to minimize the mean squared distance from each data point to its nearest centre. Although K-means is a very popular algorithm, it suffers from several drawbacks. K-means algorithm is highly dependent upon its initial selection of cluster centres and

S. Rana · S. Jasola
School of ICT, Gautam Buddha University, Greater Noida, India
e-mail: srana.it@gmail.com

S. Jasola
e-mail: sjasola@gbu.ac.in

R. Kumar (✉)
Department of Electrical Engineering, Malaviya National
Institute of Technology, Jaipur, India
e-mail: rkumar.ee@gmail.com

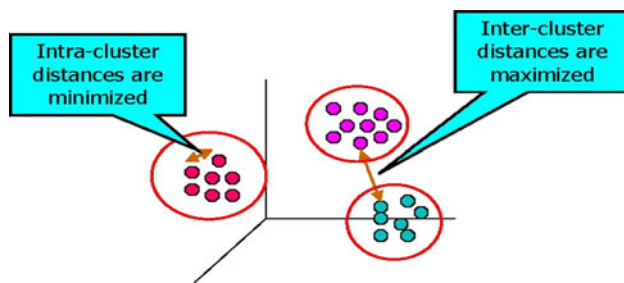


Fig. 1 Clustering analysis process

pre-knowledge of fixed cluster centres. It also suffers from dead unit problem [21]. The K-means algorithm may contain several local minima, as the objective function of K-means is not convex [9, 10]. To solve such problems, evolutionary algorithms such as genetic algorithm (GA) and particle swarm optimization (PSO) have been introduced. Genetic algorithm starts its search with a random set of solutions, instead of just one solution. Once a population of solutions (a random set of binary strings) are created at random, each solution is evaluated in the context of the problem [13]. A termination criterion is then checked. If the termination criterion is not satisfied then three main operators modify the population of solutions and a new (and hopefully better) population is created [17, 18]. Particle swarm optimization (PSO) is a population based algorithm which relies upon the cognitive and social behavior of swarm. PSO imitates the swarming behavior of birds flock, fish and bees, how they are seeking food, defined as cornfield vector [1, 12]. Although, evolutionary algorithms are more efficient and generate the good solutions, they suffer from high computational cost of slow convergence rate. PSO also suffers from slow convergence rate [11]. To deal with slow convergence problem of PSO several hybridizations have been proposed in the literature.

A hybridization of Nelder-Mead simplex method with PSO was proposed by Fan, Liang and Zhara [7]. A hybridization of K-means, Nelder-Mead simplex and PSO was proposed by Kao and Zhara [10]. The idea behind hybridization is to use the merits of all the algorithms and to find the exact and appropriate cluster centres for clustering the arbitrary data. In another approach linearly decreasing weight particle swarm optimization (LDWPSO) was introduced with the concept of linearly decreasing inertia factor to solve the slow convergence problem of PSO. Yang et al. [4] proposed an algorithm named as accelerated linearly decreasing weight particle swarm optimization (ALDWPSO) for improving the performance of PSO and to maintain the global and local search abilities of swarm. This paper presents a hybridization scheme named as Boundary Restricted Adaptive Particle Swarm Optimization (BR-APSO). The objective of proposed BR-

APSO is to find the appropriate cluster centre, solve the slow convergence problem and maintain the equilibrium between global and local search capabilities of swarm. Results shown on different artificial and real life situation data set prove that BR-APSO gives more accurate results as compared to K-means PSO, K-PSO, K-NM-PSO, LDWPSO and ALDWPSO.

This paper is organized as follows: In Sect. 2, the K-means algorithm, its details, working and major drawbacks are described. Section 3 details the standard PSO and its hybridization with other algorithms. Section 4 describes the BR-APSO approach, its development and working. Section 5 discusses simulation and experimental results made on some standard test systems and draws inferences on the cluster formation from the results obtained. Finally, Sect. 6 concludes the paper.

2 K-means Algorithm

K-means algorithm has been developed by J. A. Hartigan and M. A. Wong in between 1975 and 1977. It works on the concept that a centre point can represent a cluster. K-means algorithm is an iterative process in which data objects are moved into different clusters, until they reach the appropriate cluster [8, 17].

Given a set of observations (x^1, x^2, \dots, x^n) , where each observation is a d -dimensional real vector. K-means algorithm aims to partition the n observations into c sets ($c < n$) as $Z = (z^1, z^2, \dots, z^c)$ to minimize a measure of dispersion within the clusters. The standard K-means algorithm minimizes the within-cluster sum of squares distance according to the equation (1) given below.

$$f_1 = \arg \min \left(\sum_{j=1}^c \sum_{X^i \in Z^j} \|X^i - \mu^j\|^2 \right) \quad (1)$$

where μ^j is the mean of Z^j .

In K-means algorithm as shown in Fig. 2, k objects are selected as centroids from the given data set and other data objects of data set are assigned to c clusters where maximum similarity is achieved. Once all the data objects are assigned then it calculates the mean squared distance from each data point. The process is repeated until the termination criterion is satisfied. Choosing a proper initial centroid is the key step of the basic K-means procedure.

The main advantage of K-means algorithm is that it generates a fast and efficient solution [24]. The drawback of K-means algorithm is that the effectiveness and quality of result of K-means algorithm is highly dependent upon the objective function used in measuring the distance between objects. Its selection of initial cluster centres and also the number of clusters must be previously known and

Algorithm of K-means

1. Randomly choose k cluster centres.
 2. Assign each object to the closest centroid's cluster.
 3. When all objects have been assigned, recalculate the positions of the centroids.
 4. Go back to Steps 2 unless the centroids are not changing.
-

Fig. 2 K-means algorithm

fixed before clustering process starts [16, 18]. K-means algorithm is also very sensitive in respect to outliers.

3 PSO and its hybridization

PSO is one of the popular algorithms of evolutionary computation techniques which simulates the movement and flocking of birds. PSO performs a global search of solution space and has been widely used to solve multi objective optimization problems [1]. This approach is based upon the cooperation of agents called swarms. In analogy with evolutionary computation methods, a swarm is similar to population and a particle is similar to an individual. PSO follows a stochastic optimization method based on swarm intelligence (SI) [23]. The fundamental idea is that each particle represents a potential solution which it updates according to its own experience and that of neighbours. In PSO, particles are the agents that show the individual solutions. On the other hand, swarm is the collection of particles that shows the solution space. The PSO algorithm searches in parallel using a group of individuals or particles in a swarm, approach to the optimum through its present velocity, previous experience and the experience of its neighbours [5, 19]. PSO searches the problem domain by adjusting the trajectories of moving points in a multidimensional space. The motion of an individual particle for the optimal solution is governed through the interactions of the position and velocity of each individual, its own previous best performance and the best performance of their neighbours. For a swarm the i^{th} particle is represented by a position denoted as $xi = (xi_1, xi_2, \dots, xi_d)$. Except for the position, each particle of a swarm is represented in D dimensional space with a velocity $vi = (vi_1, vi_2, \dots, vi_d)$. The particles explore in the search space with a velocity that is dynamically adjusted according to its own and neighbours' performances. The standard PSO method updates the velocity and position of each particle according to the equations given below.

$$vid(t + 1) = \omega * vid(t) + c_1 * rand() * (pid - xid) + c_2 * rand() * (pgd - xid) \quad (2)$$

$$xid(t + 1) = vid(t + 1) + xid(t) \quad (3)$$

where c_1 and c_2 are two positive acceleration constants,

$rand()$ is a uniformly random number in $[0,1]$, pid and pgd are the best positions found so far by the i^{th} particle and all the particles respectively, t is the iteration count and ω is an inertia weight which is usually, linearly decreasing during the iterations. Equation (2) consists of three parts: the first one shows the current speed of the i^{th} particle i.e. the present state, the second term is known as the cognition term which shows the thought of the particle itself and the last and third term is a social term that shows the ability of information sharing among the swarms. The inertia weight ω plays a role of balancing the local and global search. Elabd and Kamel [6] proposed that the initial value of ω should be high and it should drop linearly and reach its minimum at the end. The higher value at the beginning of iteration helps the particle to move fast and find the global search ability and the small value at the end helps the particles to find the finer local search at the end.

Tsai and Chiu [21] proposed generalized models and techniques for tuning these parameters. PSO also has some drawbacks. PSO finds the best result with interaction of particles, but when the search space is large its convergence speed becomes very slow near global optimum. It also shows poor quality clustering results when it deals with large and complex data sets [19, 25].

Nelder-Mead simplex method, which is a derivative free line search method, is specially used for unconstrained minimization scenarios such as nonlinear least squares, nonlinear simultaneous equation and other types of function minimization [14, 15]. It is used to find the local minima to a function with several variables. The NM simplex method process is based on four basic operations—reflection, expansion, contraction and shrinkage. With the help of these four operations the methods may improve themselves and find the closest local optimum point.

The hybrid NM-PSO algorithm as shown in Fig. 3, is hybridization of Nelder-Mead (NM) simplex search method and particle swarm optimization in which the population size is set to $3N + 1$. The initial randomly generated $3N + 1$ particles are sorted by fitness and after that top $N + 1$ particles are given to simplex method to improve the $(N + 1)^{\text{th}}$ particle [7]. The remaining $2N$ particles are adjusted by PSO. The process of adjusting these $2N$ particles includes selection of gbest particle and selection of neighbourhood best particle and velocity updates. The gbest particle is then calculated as guided by sorted fitness values and neighbourhood best particle is determined by dividing the $2N$ particles in N neighbourhood.

Algorithm of NM-PSO

1. Initialization
Generate a population of size $3N + 1$.
 2. Evaluation & Ranking
Evaluate the fitness of each particle. Rank them on the basis of fitness.
 3. Simplex Method
Apply NM operator to the top $N + 1$ particles and replace the $(N + 1)^{\text{th}}$ particle with the update.
 4. PSO Method
Apply PSO operator for updating the remaining $2N$ particles.
Selection: From the population select the global best particle and the neighbourhood best particles.
Velocity Update: Apply velocity update to the $2N$ particles with worst fitness according to equations (2) and (3).
 5. If the termination conditions are not met, then go back to 2
-

Fig. 3 Algorithm of NM-PSO

Kao et al. [10] has proposed a hybridization of K-means, Nelder Mead and PSO. The idea behind this is to use the merits of this entire algorithm. In K-NM-PSO $3N$ particles are randomly generated. The clustering process is started by K-means and then NM-PSO is carried out to its completion.

Shi and Eberhart [20] have proposed a linearly decreasing weight particle swarm optimization (LDWPSO) algorithm. In this algorithm concept of linearly decreasing inertia weight was introduced. Results show that LDWPSO is more efficient and accurate as compared to the original PSO and has a great efficiency of balancing the global and local search abilities of the swarm. In LDWPSO w is linearly decreased from 0.9 to 0.4 in the search process. The equation of linearly decreasing inertia weight is as follows:

$$w = (w_{\max} - w_{\min}) \times \frac{\text{Iteration}_{\max} - \text{Iteration}_i}{\text{Iteration}_{\max}} + w_{\min} \quad (4)$$

In this equation w is the inertia weight, w_{\min} is 0.4 and w_{\max} is 0.9.

Yang et al. [4] proposed a hybrid algorithm named as accelerated linearly decreasing weight particle swarm optimization (ALDWPSO) as shown in Fig. 4, is hybridization of linearly decreasing weight and an acceleration strategy to improve the performance of PSO. It involves the four processes—the encoding and initialization of the particle, acceleration strategy, velocity update, position update and fitness evaluation.

Algorithm of ALDWPSO

1. Initialize the position and velocity randomly.
 2. Start clustering of 1/3 of particles on initial population.
 3. For 1/3 particles recalculate the cluster centres and replace the original 1/3 centres.
 4. Start clustering of the fitness value.
 5. Calculate the Pbest and Gbest.
 6. Weight is linearly decreasing between 0.9 to 0.4 for each particle
 7. Velocity and position update.
 8. If the terminating condition is not satisfied then go to step 4
-

Fig. 4 Algorithm of ALDWPSO

4 Proposed algorithm

In standard PSO when evaluation goes on, at each iteration particles of swarm update the velocity based on their momentum, personal best solution and group best solution. In evaluation process, some particles of swarm have a tendency to go outside the boundary in the search of global solution. In Fig. 5 these particles are shown by arrows. These particles will attract another particle and pass the personal best and global best information which will not be accurate as this particle is outside the boundary. Due to this inaccurate information, the quality of clustering result will affect more and the diversity of clustering process will be lost. It also affects the convergence rate of solution and the algorithm is stuck in local minimum.

Another problem with PSO is its slow convergence near global solution [10]. There are several algorithms and their hybridization is available in literature, but these algorithms can improve the global convergence ability of PSO to a certain degree [24]. In proposed boundary restricted adaptive particle swarm optimization algorithm (BR-APSO) as shown in Fig. 6, a boundary restriction strategy has been taken for those particles which go outside the boundary of search space. When evaluation process starts by standard PSO, there are several particles which go outside the boundary and affect the clustering result. In Fig. 7, it is shown that after applying the BR-APSO, there is no element which is outside.

In this approach the algorithm forcibly brings back those particles which go outside in the evaluation process. This process iteratively goes on when the termination criteria are not satisfied. The inertia weight is maximum in beginning of the process and as the process carries on low inertia is suggested for the reasons that it must not jump

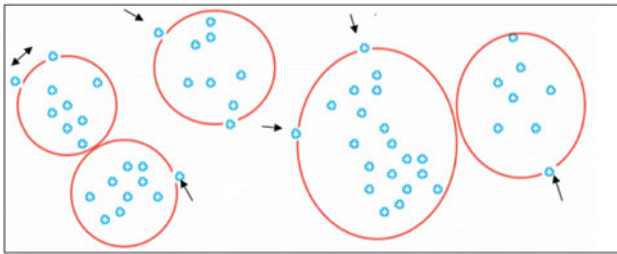


Fig. 5 Standard PSO clustering

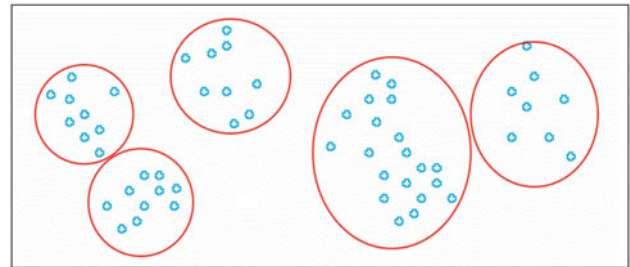


Fig. 7 BR-APSO clustering

Algorithm of the proposed scheme (BR-APSO)

1. Randomly generate $3N$ particles where each particle represents a feasible solution, i.e. cluster solution
2. Initialization of particle position and velocity.
3. Calculate the fitness function

$$\text{objective function } (f_2) = \sum \|x^i - c^j\|, i = 1, \dots, n, j = 1, \dots, z$$
 /* Where n and z are the numbers of datasets and clusters, respectively and x^i is data point and c^j is cluster centre. */
4. The value of objective function is stored as particle personal best and global or swarm best.
5. Calculate the value of the inertia weight exponentially
 $w = \max w * (\exp(-iter));$
6. Update the velocity and position of particle i .
 /* Boundry restriction strategy */
 If $position \leq upbnd$ and $position \geq lwbnd$
 Then $position = position$
 else $position = position - velocity;$
7. Place n points into the space represented by the objects that are clustered with cluster centres obtained from PSO algorithm. These points represent initial group centroids.
8. Assign each object to the group that has the closest centroid. When all objects have been assigned, recalculate the positions of the c centroids.
9. Repeat steps until the centroids no longer move.

Fig. 6 Algorithm of BR-APSO

over the global solution. Previous work adopted the tuning process by linearly decreasing the inertia weight but it is seen that better solutions or swarm crowding near the global solution is observed with small change in the inertia weight and hence the inertia weight adaption function is non-linear. This motivates us to choose exponential

function to update the inertia weight in BR-APSO. Result shows that the performance is significantly improved over other methods of PSO because it effectively balances the global and local searching ability of swarm.

5 Datasets and result discussion

Nine data sets have been taken to validate our algorithm. Data sets are, Vowel, Iris, Crude oil, Contraceptive Method Choice, Glass and Wine. The aim for selecting these data sets is to cover the data set of all dimensions (high, low, medium). The data set with 3 dimensions is grouped into three clusters, so the number of parameters is the product of cluster number and its attributes (features) i.e. $N = k \times d = 3 \times 3 = 9$, for finding the optimal cluster centre. The datasets are available at <ftp://ftp.ics.uci.edu/pub/machine-learning-databases/>.

1. Artificial data set 1 (Art1): This is a 2-dimensional problem with 4 unique classes. The problem is interesting in that only one of the inputs is really relevant to the formation of the classes. Figure 8 shows the artificial data set1. A total of 600 patterns were drawn from four independent bivariate normal distributions, where classes were distributed according to equation (5) for $i = 1, \dots \sim$ where μ is the mean vector and \sum is the covariance matrix; $m1 = -3, m2 = 0, m3 = 3$ and $m4 = 6$.

$$N_2 \left(\mu = \begin{pmatrix} m_i \\ 0 \end{pmatrix}, \sum = \begin{bmatrix} 0.50 & 0.05 \\ 0.05 & 0.50 \end{bmatrix} \right) \quad (5)$$

2. Artificial data set 2 (Art2): This is a three dimensional problem with 5 unique classes. The total number of data sets is $n = 250$, where distribution of features of class is according to Class1 ~ Uniform (85, 100), Class2 ~ Uniform (70, 85), Class3 ~ Uniform (55, 70), Class4 ~ Uniform (40, 55), Class5 ~ Uniform (25, 40). The data sets are shown in Fig. 9.
3. Vowel data set: This data contain 871 Indian Telugu vowels sounds. This is the three dimensional problem where all three attributes are corresponding to the I, II and III vowel frequency and six overlapping classes

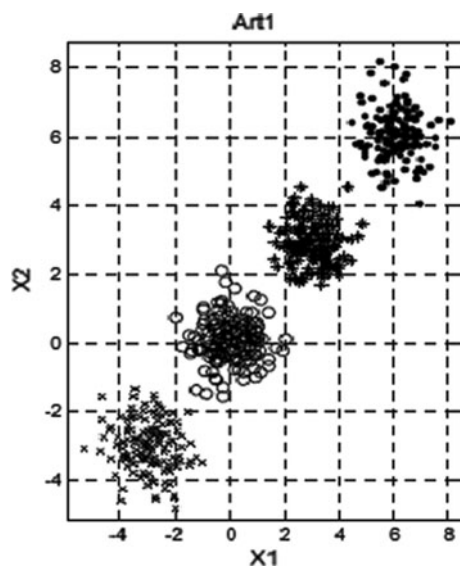


Fig. 8 Artificial dataset 1

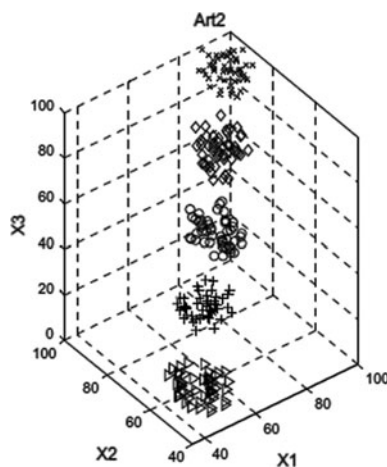


Fig. 9 Artificial dataset 2

{d (72 objects), a (89 objects), i (172 objects), u (151 objects), e (207 objects), o (180 objects)}.

4. Fisher Iris data set: This is perhaps the best known database to be found in the pattern recognition literature. The data set contains 3 classes of iris flower of 150 instances each, where each class refers to a type of iris plant (Iris setosa, Iris virginica and Iris versicolor). One class is linearly separable from the other two; the latter are NOT linearly separable from each other. Each class has 50 samples with four features.
5. Crude oil data set: This data set has 56 data with five attributes: vanadium, iron, beryllium, saturated hydrocarbons and aromatic hydrocarbons. The crude oil data set is taken from three zones of sandstone.
6. Contraceptive Method Choice (CMC) data set: This data set is the subset of 1987 National Indonesia

Table 1 Characteristics of data set

Name of data set	No. of classes	No. of attributes (features)	No. of instances in each class
Art1	4	2	600 (150, 150, 150, 150)
Art2	5	3	250 (50, 50, 50, 50, 50)
Vowel	6	3	871 (72, 89, 172, 151, 207, 180)
Iris	3	4	150 (50, 50, 50)
Crude oil	3	5	56 (7, 11, 38)
CMC	3	9	1,473 (629, 334, 510)
Cancer	2	9	683 (444, 239)
Glass	6	9	214 (70, 17, 76, 13, 9, 29)
Wine	3	13	178 (59, 71, 48)

contraceptive Prevalence survey. This data set is the sample of women who are married and were not pregnant or did not know if they were at the time of interview. The problem is to predict the choice of current contraceptive method of women on the basis of their demographic and economic characteristics.

7. Wisconsin breast cancer data set: It contains 683 objects with two categories and nine attributes: clump thickness, cell size uniformity, cell shape uniformity, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. Categories are malignant and benign.
8. Glass data set: The data set was collected from 6 different glasses: building windows float processed (70 objects), building windows non-float processed (76 objects), vehicle windows float processed (17 objects), containers (13 objects), tableware (9 objects), and headlamps (29 objects), where each data has nine features: refractive index, sodium, magnesium, aluminum, silicon, potassium, calcium, barium, and iron.
9. Wine dataset: 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 (inputs) found in each of the three types of wines (classes). These data are collection of 178 instances (data vectors). Hence, this is a classification problem with “well behaved class structures”. There are 13 inputs, 3 classes and 178 data vectors.

Kao et al. [10] summarized the details of these nine data sets for better understanding the characteristics and further analysis. The inputs, classes and data vector details of the different data sets are shown in Table 1.

In this section, details of the overall results of the proposed algorithm are discussed. The performance and comparison of the result of different algorithms such as

K-means, PSO, NM-PSO, K-PSO, K-NM-PSO, LDWPSO, ALDWPSO are evaluated. The clustering quality is also compared. The criteria for evaluating of quality are:

Sum of intra cluster distance: The sum of distances between data object and centre of cluster. If the sum is minimum then the quality of clustering is high.

Error rate: The total number of misplaced data objects of the cluster divided by the total number of data as according to equation (6)

$$ER = \left(\sum_{i=1}^n (if(A_i = B_i) \text{ then } 0 \text{ else } 1) / n \right) \times 100 \quad (6)$$

Here n is the total number of data objects. A_i and B_i is the dataset with i^{th} data object before and after clustering.

The program has been written in MATLAB. The results of proposed algorithm are average of 20 simulation runs.

Kao et al. [10] and Yang et al. [4] have taken $10 \times N$ iterations for each run for all dataset in N dimensional problem. On the other hand proposed algorithm finds the global optimal solution in just 100 iterations as compared

to $10 \times N$ iterations. The experimental results also show that its convergence is very fast and provide more efficiency and effectiveness as compared to K-means, K-PSO, K-NM-PSO, LDWPSO, ALDWPSO.

The Intra cluster distances of all seven algorithms for nine datasets are summarized in Table 2. The obtained values are the result of inter cluster distance and its standard deviation and average of intra cluster distance over 20 simulation runs. Result shows that for all experimental data set BR-APSO gives better result by little difference between the averages and less standard deviations in comparison of other seven algorithms. Proposed algorithm is more effective and faster to find the best global optima as compared to PSO, NM-PSO, K-PSO, K-NM-PSO LDWPSO and ALDWPSO. Table 3 shows an example of error rate calculation. In this table data points (2, 6) and (1, 7) are misplaced and the error rate is 2/5.

Table 4 describes the error rate of all the algorithms over 20 simulations with mean and standard deviation and best solution among all. For all data sets, proposed

Table 2 Comparison of intra cluster distance for K-means, PSO, NM-PSO, K-PSO, K-NM-PSO, LDWPSO, ALDWPSO and BR-APSO

Data set	Criteria	K-means	PSO	NM-PSO	K-PSO	K-NM-PSO	LDWPSO	ALDWPSO	BR-APSO
Art1	Average	721.57	627.74	515.88	515.88	515.88	515.88	515.88	515.88
	(Std)	295.84	180.24	7.14E−08	5.60E−05	7.14E−08	7.14E−08	7.14E−08	0
	Best	516.04	515.93	515.88	515.88	515.88	515.88	515.88	515.88
Art2	Average	2,762	2,517.2	1,910.4	2,067.3	1,746.9	1,746.9	1,745.9	1,744.69
	(Std)	720.66	415.02	296.22	343.64	3.6	3.6	3.6	3.42
	Best	1,746.9	1,743.2	1,743.2	1,743.2	1,743.2	1,743.2	1,743.2	1,743.2
Vowel	Average	159,243	168,477	151,984	149,376	149,141.4	152,393	148,985.5	148,985.4
	(Std)	916	3,715.3	4,386.43	155.56	120.38	4,935.47	30.67	29.72
	Best	149,422	163,882	149,240	149,206	149,005	149,041	148,967.2	148,967.2
Iris	Average	106.05	103.51	100.72	96.76	96.67	96.67	96.66	96.66
	(Std)	14.11	9.69	5.82	0.07	0.008	0.03	0.0009	0.0007
	Best	97.33	96.66	96.66	96.66	96.66	96.66	96.66	96.65
Crude oil	Average	287.36	285.51	277.59	277.77	277.29	277.24	277.24	267.91
	(Std)	25.41	10.31	0.37	0.33	0.095	0.043	0.039	0.33
	Best	279.2	279.07	277.19	277.45	277.15	277.21	277.21	266.94
CMC	Average	5,693.6	5,734.2	5,563.4	5,532.9	5,532.7	5,532.18	5,532.18	5,532.18
	(Std)	473.14	289	30.27	0.09	0.23	9.50E−05	2.00E−06	2.00E−07
	Best	5,542.2	5,538.5	5,537.3	5,532.88	5,532.4	5,532.18	5,532.18	5,532.18
Cancer	Average	2,988.3	3,334.6	2,977.7	2,965.8	2,964.7	2,964.39	2,964.39	2,963.7
	(Std)	0.46	357.66	13.73	1.63	0.15	0.0001	6.80E−06	1.00E−06
	Best	2,987	2,976.3	2,965.59	2,964.5	2,964.5	2,964.39	2,964.39	2,963.70
Glass	Average	260.4	291.33	248.96	207.35	200.5	200.5	200.5	199.87
	(Std)	36.82	6.82	6.82	5.12	2.26	2.26	2.26	1.09
	Best	215.68	243.45	243.45	203.37	199.68	199.68	199.68	199.64
Wine	Average	18,061	16,311	16,303	16,294	16,293	16,292.7	16,292.38	16,292.41
	(Std)	793.21	22.98	4.28	1.7	0.46	0.64	0.24	0.21
	Best	16,555.7	16,294	16,292	16,292	16,292	16,292.2	16,292.19	16,292.19

algorithm has smaller mean and standard deviation among all seven algorithms such as K-means, NM-PSO, K-NM-PSO, LDWPSO, ALDWPSO.

The convergence graph of the BR-APSO, NM-PSO, K-PSO, K-NM-PSO, K-means, PSO are shown in Figs. 10 and 11. Figure 10a shows the convergence behavior of the algorithms for data set Art1. In all the algorithms the convergence speed of K-means algorithm is fast but it is stuck in local optimum. The other algorithms PSO & NM-

PSO take about 50 iterations for global convergence. K-PSO and K-NM-PSO take about 10 iterations to reach the global optimum, whereas proposed BR-APSO takes about 7 iterations to converge to global optimum. On the other hand, in Fig. 10b the algorithms BR-APSO, NM-PSO, K-PSO & K-NM-PSO correctly cluster the data of Art1 dataset into four clusters. In Figs. 10c, d the PSO clusters the dataset with 25 % error rate & K-means clusters the dataset with 25.67 % error rate.

Figure 11a shows the convergence graph of all algorithms for Art2 data set. K-means algorithm is fast but stuck in local minima. PSO and NM-PSO takes 80 iterations for finding global optimum, whereas, BR-APSO takes about 10 iterations for convergence to global optimum. Figure 11b, c, d shows the clustering results of all the algorithms. BR-APSO, K-PSO, NM-PSO, K-means, K-NM-PSO clusters the correct dataset in five clusters whereas K-means shows 40 % error rate and PSO shows the 20 % error rate to clusters the dataset in five clusters.

Table 3 Error rate calculations

I	Data point	Ai	Bi	Not misplaced (0)/ Misplaced (1)
1	(2, 6)	2	1	1
2	(6, 3)	2	2	0
3	(1, 7)	2	1	1
4	(5, 4)	1	1	0
5	(8, 7)	1	1	0

Number of misplaced point: 2

Table 4 Comparison of error rate for K-means, PSO, NM-PSO, K-PSO, K-NM-PSO, LDWPSO, ALDWPSO and BR-APSO

Data set	Criteria	K-means	PSO	NM-PSO	K-PSO	K-NM-PSO	LDWPSO	ALDWPSO	BR-APSO
Art1	Average	13	7.57	0	0	0	0	0	0
	(Std)	17.78	12.18	0	0	0	0	0	0.00
	Best	0	0	0	0	0	0	0	0.00
Art2	Average	34	22	4.04	10	0	0	0	0.00
	(Std)	13.45	11.35	8.52	10.32	0	0	0	0.00
	Best	20	0	0	0	0	0	0	0.00
Vowel	Average	44.26	44.65	41.96	42.24	41.94	42.25	41.83	41.83
	(Std)	2.15	2.55	0.98	0.95	0.95	1.47	0.32	0.29
	Best	42.02	41.45	40.07	40.64	40.64	40.18	40.87	40.34
Iris	Average	17.8	12.53	11.13	10.2	10.07	10.13	10	10.00
	(Std)	10.72	5.38	3.02	0.32	0.21	0.27	0	0.0
	Best	10.67	10	8	10	10	10	10	9.33
Crude oil	Average	24.46	24.64	24.29	24.29	23.93	26.52	26.34	26.34
	(Std)	1.21	1.73	0.75	0.92	0.72	0.66	0.79	0.007
	Best	23.21	23.21	23.21	23.21	23.21	25	25	26.32
CMC	Average	54.49	54.41	54.47	54.38	54.38	54.38	54.38	54.38
	(Std)	0.04	0.13	0.06	0	0.05	0	0	0.0002
	Best	54.45	54.24	54.38	54.38	54.31	54.38	54.38	54.31
Cancer	Average	4.08	5.11	4.28	3.66	3.66	3.51	3.51	3.49
	(Std)	0.46	1.32	1.1	0	0	0	0	0.00
	Best	3.95	3.66	3.66	3.66	3.66	3.51	3.51	3.49
Glass	Average	37.71	45.59	40.89	32.17	30.45	30.45	30.45	29.01
	(Std)	13.75	15.62	15.58	15.52	14.13	14.13	14.13	14.13
	Best	11	11.21	9.35	12.62	12.62	12.62	12.62	12.61
Wine	Average	31.12	28.71	28.48	28.48	28.37	28.51	28.31	28.48
	(Std)	0.71	0.41	0.27	0.4	0.27	0.4	0.28	0.24
	Best	29.78	28.09	28.09	28.09	28.09	28.09	28.08	28.08

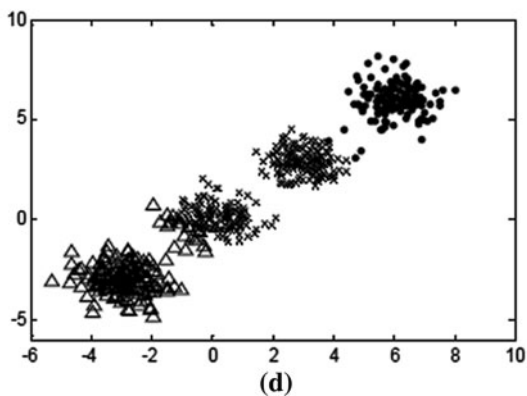
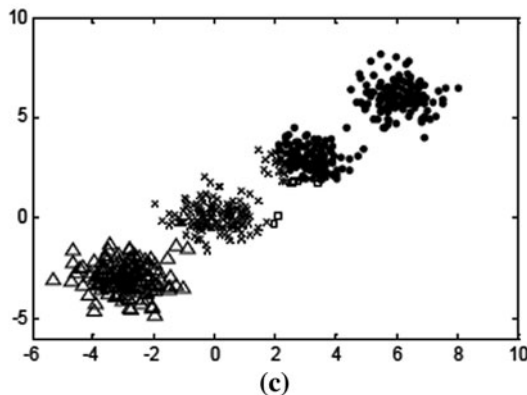
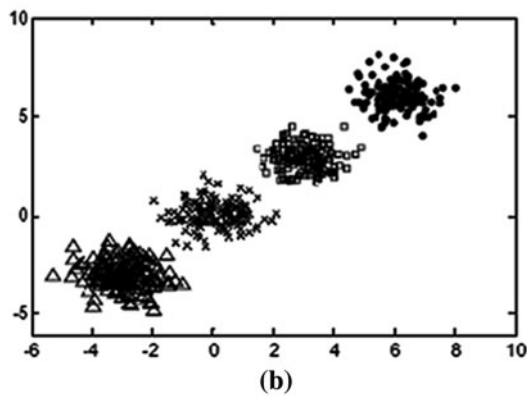
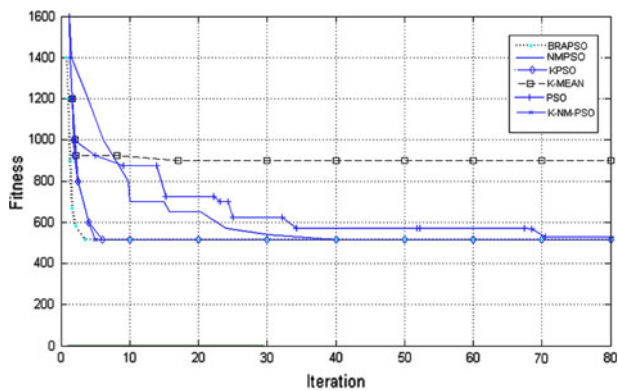


Fig. 10 **a** Conversion graph of BR-APSO, K-means, PSO, K-NM-PSO for Art 1; **b** BR-APSO, NM-PSO, K-PSO and K-NM-PSO with 0 % error rate; **c** PSO with 25 % error rate; **d** K-means with 25.67 % error rate

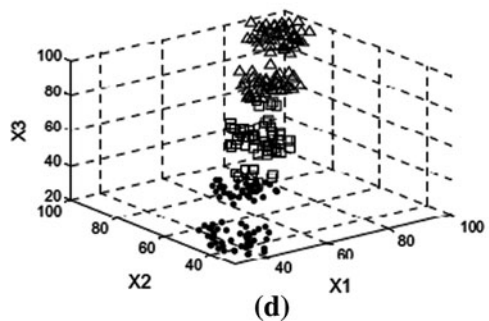
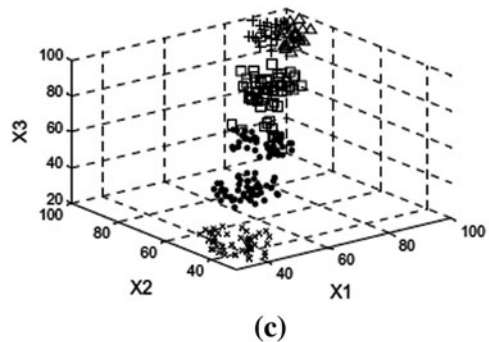
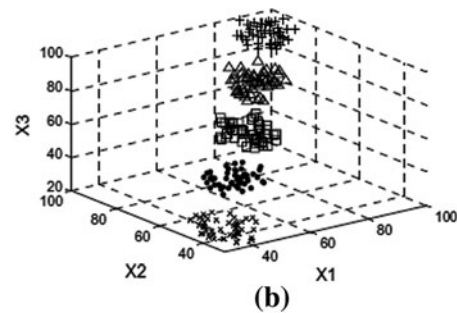
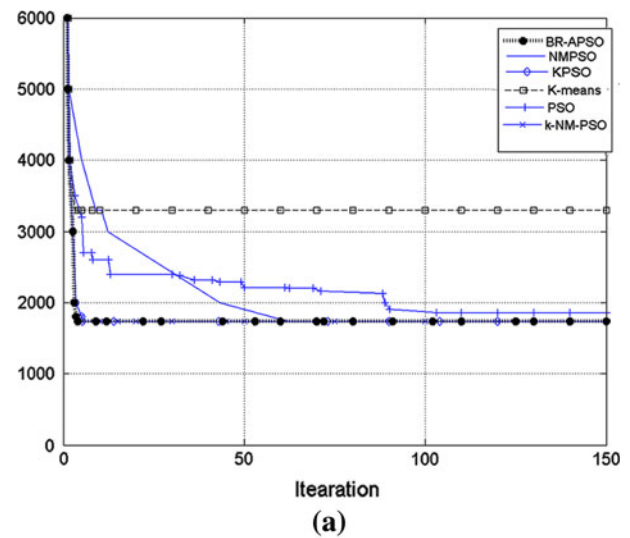


Fig. 11 **a** Conversion graph of BR-APSO, K-means, PSO, K-NM-PSO for Art 2; **b** BR-APSO, NM-PSO, K-PSO and K-NM-PSO with 0 % error rate; **c** PSO with 20 % error rate; **d** K-means with 40 % error rate

6 Conclusion

This paper has proposed a cluster analysis algorithm based on PSO named as BR-APSO. The performance of this algorithm is validated on nine data sets based on intra cluster distance criterion in N dimensional Euclidian space. The K-means and other algorithms suffer from initial center selection and slow convergence rate near global optima. The experiment results by using nine data sets show that proposed algorithm has better performance and convergence rate compared to K-means K-PSO, K-NM-PSO, LDWPSO, ALDWPSO and least error rate compared to these algorithms.

Acknowledgments The authors would like to thank to Erwie Zahara, Department of Industrial Engineering and Management, St. John's University, Tamsui, Taiwan for providing data set and other help regarding this work.

References

- Ahmadi A, Karray F, Kamel M (2007) Multiple cooperating swarms for data clustering. In: Proceedings of IEEE swarm intelligence symposium, pp 206–212
- Ahmadyfard A, Modares H (2008) Combining PSO and k-means to enhance data clustering. In: International symposium on telecommunications, pp 688–691
- Bandyopadhyay S, Maulik U (2002) An evolutionary technique based on K-means algorithm for optimal clustering in RN. *Inf Sci* 146:221–237
- Yang CH, Hsiao CJ, Chuang LY (2010) Accelerated linearly decreasing weight particle swarm optimization for data clustering. *Proc Int Multi Conf Eng Comput Sci (Hongkong)* 1:1–6
- Eberhart RC, Shi Y (2001) Tracking and optimizing dynamic systems with particle swarms. In: Proceedings of the congress on evolutionary computation, Seoul, Korea, pp 94–97
- El-abd M, Kamel M (2005) Information exchange in multiple cooperating swarms. In: IEEE swarm intelligence symposium, pp 138–142
- Fan S-KS, Liang Y-C, Zahara E (2004) Hybrid simplex search and particle swarm optimization for the global optimization of multimodal functions. *Eng Optim* 36:401–418
- Jain AR, Murthy MN, Flynn PJ (1999) Data clustering: a review. *ACM Comput Surv* 31(3):265–323
- Jinxin D, Qi Minyong (2009) A new algorithm for clustering based on particle swarm optimization and K-means. *IEEE Int Conf Artif Intell Comput Intell* 4:264–268
- Kao Yi-Tung, Zahara E, Kao I-Wei (2007) A Hybridized Approach to Data Clustering. *Expert Syst Appl* 34:1754–1762
- Kennedy J, Eberhart RC (1995) Particle swarm optimization. *Proc IEEE Int Joint Conf Neural Netw* 4:1942–1948
- Kennedy J (1997) Minds and Cultures: Particle Swarm Implications. In: Socially Intelligent Agents Papers AAAI Fall Symposium Technical Report FS-97-02, AAAI Press, Menlo Park, pp 67–72
- Murthy CA, Chowdhury N (1996) In search of optimal clusters using genetic algorithms. *Pattern Recogn Lett* 17:825–832
- Nelder JA, Mead R (1965) A simplex method for function minimization. *Comput J* 7:308–313
- Olsson DM, Nelson LS (1975) The Nelder–Mead simplex procedure for function minimization'. *Technometrics* 17:45–51
- Paterlini S, Krink T (2006) Differential evolution and particle swarm optimization in partitional clustering'. *Comput Stat Data Anal* 50:1220–1247
- Rana S, Jasola S, Kumar R (2010) A hybrid sequential approach for data clustering using K-Means and particle swarm optimization algorithm. *Int J Eng Sc Technol* 2(6):167–176
- Rana S, Jasola S, Kumar R (2011) A review on particle swarm optimization algorithm and their application to data clustering. *Artificial Intelligence Review Archive* 35(3):211–222
- Selim SZ, Ismail MA (1984) K-means type algorithms: a generalized convergence theorem and characterization of local optimality. *IEEE Transact Pattern Anal Mach Intell* 6:81–87
- Shi Y, Eberhart, RC (1998) Parameter selection in particle swarm optimization. *Evolutionary programming*, vol 1441 of lecture notes in computers science, Springer, pp 591–600
- Tsai CY, Chiu CC (2008) Developing a feature weight self-adjustment Mechanism for k-means clustering Algorithm. *Comput Stat Data Anal* 52:4658–4672
- Zalik KR (2008) An efficient kmeans clustering algorithm. *Pattern Recogn Lett* 29:1385–1391
- Wang XZ, He YL, Dong LC, Zhao HY (2011) Particle swarm optimization for determining fuzzy measures from data. *Inf Sci* 181(19):4230–4252
- Liang J, Song W (2011) Clustering based on Steiner points. *Int J Mach Learn Cyber*. doi:10.1007/s13042-011-0047-7
- Graaff AJ, Engelbrecht AP (2011) Clustering data in stationary environments with a local network neighborhood artificial immune system. *Int J Mach Learn Cyber*. doi:10.1007/s13042-011-0041-0
- Guo G, Chen S, Chen L (2011) Soft subspace clustering with an improved feature weight self-adjustment mechanism. *Int J Mach Learn Cyber*. doi:10.1007/s13042-011-0039-7