Combining PSO and k-means to Enhance Data Clustering

Alireza Ahmadyfard, Hamidreza Modares
Department of Electrical Engineering and Robotics,Shahrood University of Technology, Shahrood, Iran.
Emails: ahmadyfard@shahroodut.ac.ir, reza_modares@yahoo.com

Abstract— **In this paper we propose a clustering method based on combination of the particle swarm optimization (PSO) and the k-mean algorithm. PSO algorithm was showed to successfully converge during the initial stages of a global search, but around global optimum, the search process will become very slow. On the contrary, k-means algorithm can achieve faster convergence to optimum solution. At the same time, the convergent accuracy for k-means can be higher than PSO. So in this paper, a hybrid algorithm combining particle swarm optimization (PSO) algorithm with k-means algorithm is proposed we refer to it as PSO-KM algorithm. The algorithm aims to group a given set of data into a user specified number of clusters. We evaluate the performance of the proposed algorithm using five datasets. The algorithm performance is compared to K-means and PSO clustering.**

***Keywords*: data clustering, articles, particle swarm optimization, K-means.**

I. INTRODUCTION

Clustering is an unsupervised classification technique which deals with pattern recognition problems. When used on a set of objects, it helps identify some inherent property present in the objects by classifying them into subsets that have some meaning in the context of a particular problem. More specifically, objects are represented by a set of features which characterize them. The object features are usually represented as a data point in a multi-dimensional space. So clustering can be considered as partitioning of data points based on a homogeneity criterion. When the number of clusters, K, is known as a priori knowledge, clustering is formulated in such a way that objects in the same cluster being more similar in some sense than those in different clusters. This involves minimization of some extrinsic criterion. The K-means algorithm, starting with k arbitrary cluster centres in space, partitions the set of given objects into k subsets based on a distance metric. The centres of clusters are iteratively updated based on optimization of an objective function. This method is one of the most popular clustering techniques which are used widely. Since it is easy to implement and very efficient, with linear time complexity [1]. However, the K-means algorithm suffers from several drawbacks. The objective function of the K-means is not convex and hence it may contain many local minima. Consequently, in the process of minimizing the objective function, there exists a possibility of getting stuck at local exterma and saddle points [2]. The outcome of the K-means algorithm, therefore, heavily depends on the initial choice of the cluster centres. By selecting initial centres near to the optimum solution, convergence to the solution is guaranteed. Recently, many clustering algorithms based on evolutionary computing such as genetic algorithms have been introduced, and only a couple of applications opted for particle swarm optimization [3]. Unlike the Genetic algorithm (GA), PSO does not have complicated evolutionary operators such as crossover and mutation [4]. In the PSO algorithm, the potential solutions called particles, are obtained by ‘‘flowing’’ through the problem space by following the current optimum particles. Generally speaking, the PSO algorithm has a strong ability to find the most optimistic result, but it suffers from converging to a local optimum. By suitably modulating the PSO parameters, convergence can be speeded up and the ability to find the global optimistic result can be enhanced. However, because the PSO algorithm has several parameters to be adjusted by empirical approach, if these parameters are not appropriately set, the search will become very slow near the global optimum. The K-means algorithm, on the contrary, has a strong ability to find local optimistic result, but its ability to find the global optimum is weak.

In this paper we aim to propose a clustering approach based on combining the PSO and the K-means algorithms. The algorithm is referred to as PSO–KM. The motivation for this idea is the fact that PSO at the beginning stage of algorithm is able to search whole space for the optimum solution. When the PSO algorithm reaches to a solution roughly close to the optimum solution, the clustering process switches to K-means algorithm to finish the process faster and more accurately. A proper stage for switching the clustering process is sensed by inspecting the PSO fitness function along the process.

The result of experiments on real and synthetic data reveals that the proposed algorithm outperforms stand alone the PSO and k-means clustering algorithms.

The paper has been organized as follows. In the next section we introduce standard PSO algorithm. In Section 2 we review the k-means algorithm. We explain the proposed algorithm in Section3. In Section 4 we present the result of experiments on synthetic and real data sets. Finally we draw the paper to the conclusion in Section 5.

978-1-4244-2751-2/08/$25.00 ©2008 IEEE

II. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) is an optimization algorithm which simulates the movement and flocking of birds [5]. The algorithm works by initializing a flock of birds randomly over the searching space, where each bird is referred to as a ''particle''. Consider that a set of ''particles'' fly with a certain velocity algorithm and move to find the global best position in an iterative process. At each iteration of algorithm, the velocity vector for each particle is modified based on three parameters: the particle momentum, the best position reached by the particle and that of all particles up to the current stage. Then based on the determined velocity for each particle, the particle is moved to its next position. Suppose that the search space is n-dimensional, the position and the velocity of i th particle in the swarm at t th iteration of algorithm is denoted by vector $X_i^{(t)} = (x_{i1}^{(t)}, x_{i2}^{(t)}, \ldots x_{in}^{(t)})$ and vector $V_i^{(t)} = (v_{i1}^{(t)}, v_{i2}^{(t)}, \ldots v_{in}^{(t)})$ respectively. Using the optimization objective function, the fitness of each particle as a solution is evaluated at each stage of algorithm. A record of the best position of particle based on the fitness criterion is kept in process. In this regard the best previously visited position of the particle i at current stage of algorithm is denoted by vector $P_i = (p_{i1}, p_{i2}, \ldots p_{in})$ referred to as the personal best. For all particles in the swarm the position of the best fitness until the current stage is also recorded. We refer to it as the global best position denoted by $G = (g_1, g_2, \ldots g_n)$.

At each iteration, the velocity of particle and its new position is defined according to the following equations:

$$V_i^{(t)} = \omega * V_i^{(t-1)} + c_1 * r_1 (P_i - X_i^{(t-1)}) + c_2 * r_2 (G - X_i^{(t-1)}) \qquad (1)$$

$$X_i^{(t)} = X_i^{(t-1)} + V_i^{(t)} \qquad (2)$$

Where, ω is called the inertia weight that controls the impact of previous velocity of particle on its current one. In the references [6,7], several selection strategies of inertial weight ω have been given. Generally, at the beginning stages of PSO algorithm, the inertial weight ω should decrease rapidly, once the swarm converge around the optimum solution, the inertial weight must decrease slowly. r_1 and r_2 are two independently uniformly distributed random variables in range [0,1] . c_1 and c_2 are positive constant parameters called acceleration coefficients which control the maximum step size between successive iterations.

According to Equation (1) the velocity of the particle at each iteration is calculated using three terms: the velocity of the particle at previous iteration, the distance of particle from its the best previous position and the distance from the best position of the entire population. Having the velocity of particle, the particle flies to a new position according to Equation (2). This process is repeated until a termination condition is reached. Two common conditions used for terminating the PSO algorithm are exceeding the number of iterations from a predefined level and negligible change for particles in successive iterations.

III. K-ALGORITHM

At the core of any clustering algorithm places the similarity function based on which closeness of two patterns is measured. The K-means algorithm [2] groups the set of data points in space into a predefined number of clusters. In this regard, the Euclidean distance is commonly used as a similarity measure. The strategy in this algorithm is to group data points in such a way that the Euclidean distance between data points belonging to each group being minimized. The data points in each group (cluster) are represented by the group centre of mass, referred to as the cluster centroid. Hence the k-means algorithm attempts to find the best points in space as the cluster centroids.

The standard K-means algorithm is summarized as follows:

1. Randomly initialize the k cluster centroid in space

$$Z = \{z_1, z_2, \ldots, z_k\}$$

2. Repeat until a termination condition is satisfied

(a) Assign to each data point in space the cluster centroid which has the closest distance to the point. The distance of data point y_p to the centroid in d-dimensional space is given as:

$$D(y_p . z_j) = \sqrt{\sum_{i=1}^{d} (y_{pi} - z_{ji})^2} \qquad (3)$$

(b) Recalculate the cluster centroids based on the definition of centroid. So the centroid for cluster j is determined as follows:

$$z_j = \frac{1}{n_j} \sum_{\forall y_p \in c_j} y_p \qquad (4)$$

Where C_j is the subset of data points belonging to the cluster j and n_j is the number of data points in this cluster.

The clustering process terminates when one of the following conditions is satisfied:

1. The number of iterations exceeds a predefined maximum.
2. When change in the cluster centroids is negligible.
3. When there is no cluster membership change.

be given in a line after affiliations.

IV. HYBRID PSO-KMEANS FOR CLUSTERING

The PSO–BP is an optimization algorithm combining the PSO with the k-means, proposed to solve the problem of data clustering. Similar to the GA, the PSO algorithm is a global search algorithm, which has a strong ability to find global optimistic result. However, the convergence speed of PSO algorithm near to the solution is very slow. The k-means algorithm, on the contrary, converge fast to a local optimum result, but its ability to find the global solution is weak. By combining the PSO and the k-means algorithms, a novel clustering approach is formulated in this paper. We refer to it as PSO–KM hybrid algorithm. The motivation for combining these clustering methods is to have advantage of both PSO and k-means algorithm. We start the data clustering by PSO algorithm it allows to search all space for a global solution. When the region of global optimum is found by PSO we continue the clustering using k-means. This strategy accelerates the convergence speed as well as accuracy. In this way the k-means algorithm finalizes the clustering task.

We detect the proper stage for switching from PSO to k-means, using PSO fitness function. When the value of fitness function for a number of successive iterations changes negligibly the clustering algorithm switches to k-means.

Similar to the PSO, in PSO–KM searching process, we start with initializing a group of random particles in solution space. First, all the particles are updated according to the Equations (1) and (2), until a new generation set of particles are generated. The flying particles are used to search the global best position in the solution space. Finally the k-means algorithm is used to search around the global optimum. In this way, the proposed hybrid algorithm would find the optimum solution more quickly.

The procedure for this PSO–KM algorithm can be summarized as follows:

Step 1: Initialize the position and velocity of particles randomly. Each particle is a potential solution for clustering problem in hand. In the context of clustering, a single particle represents the centroid of clusters. Hence i th particle is initialized as follows:

$$X^{(0)}{}_{i} = (z^{(0)}{}_{i1}, z^{(0)}{}_{i2}, ... z^{(0)}{}_{ik}) \qquad (5)$$

Where $z^{(0)}{}_{ij}$ refers to the j th cluster centroid in solution suggested by the i th particle. Therefore a swarm suggests a number of candidates for clustering centroids.

Step 2: Evaluate the fitness for each particle based on clustering criteria. The fitness of particle i in swarm is defined as below:

$$F(i) = \frac{\sum_{j=1}^{k} \sum_{\forall y_p \in C_{ij}} (y_p - z_{ij})^2}{N_p} \qquad (6)$$

where N_p is the number of data points as inputs to clustering process. By minimizing the fitness function, the dispersion of clusters would be minimized.

Step 3: If the number of iterations exceeds a predefined level go to Step 7, otherwise go to Step 4.

Step 4: The position of best particle among the particles in swarm is stored. Then the position of all the particles are updated according to Equations (1) and (2).

If a particle flies beyond the boundary $[X_{\min}, X_{\max}]$,(the range of possible solutions) then the position of particle is set to the $X_{\min}$ or $X_{\max}$; similarly if a new velocity is beyond the boundary $[V_{\min}, V_{\max}]$, the new velocity will be set to $V_{\min}$ or $V_{\max}$.

Step 5: Reduce the inertia weight, ω, according to the strategy described in Section 2.

Step 6: If the global best of particles, G , remains unchanged for a number of iterations (ten in our implementation) go to Step 7; otherwise go to Step 3.

Step 7: Use the k-means algorithm to finish clustering task. The clustering terminates when one of conditions stated in Section 3 reaches.

V. EXPERIMENS

In order to evaluate the performance of the proposed clustering algorithm, we conducted two experiments using synthetic and real data. In these experiments we compare the proposed PSO-KM method with stand alone PSO clustering and k-means clustering. We constructed three synthetic data sets (SET I, SET II and SET III). The distribution of data points in each cluster of these sets is normal $N(\mu, \Sigma)$ with mean vector μ and covariance matrix Σ. The parameters for the clusters in the three data sets are shown in Table1.

Fig. 1 illustrates the data points in SET III including five clusters in three dimensional space. The second experiment was conducted using two real datasets Iris and Cancer. The datasets are available online [9]. The information of synthetic and real datasets are tabulated in Tabel 2. The second and the third columns of Table 2 show the number of data points in each dataset and in each individual cluster respectively. The complexity of clustering problem from separately point of view shown in the last column. The result of clustering on these datasets using the proposed hybrid PSO-KM, PSO and k-means are presented in Table 3.

We evaluated the performance of each method using both error rate and mean square error criterion. Except for SET I in which the clusters are well separated, the result of clustering for PSO-KM is better than PSO and k-means.

Table 1. Information for sysnthetic and real datasets

Data Set	Criteria	k-means	PSO	PSO-KM
SET I	Error rate MSE	0% 1.11	0% 1.11	0% 1.11
SET II	Error rate MSE	8.2% 1.22	7.6% 1.18	7.2% 1.15
SET III	Error rate MSE	16% 41.59	14.3% 38.72	12.1% 38.43
Iris	Error rate MSE	13.2% 0.54	12% 0.52	10.5% 0.52
Cancer	Error rate MSE	5% 29.12	4.7% 28.24	3.7% 26.30

Table 2. The parameters for three sysnthetic data set

Data Set	Number of clusters	space dimensions	Cluster parameters
I	3	2	$\mu_1 = [-10,-10], \Sigma_1 = 1$ $\mu_2 = [-6,-6], \Sigma_2 = 1$ $\mu_3 = [-3,-3], \Sigma_3 = 1$
II	3	2	$\mu_1 = [-10,-10], \Sigma_1 = 1$ $\mu_2 = [-8.5,-8.5], \Sigma_2 = 1$ $\mu_3 = [-3,-3], \Sigma_3 = 1$
III	5	3	$\mu_1 = [-20,-20,-20], \Sigma_1 = 4$ $\mu_2 = [-10,-10,-10], \Sigma_2 = 4$ $\mu_3 = [-5,-5,-5], \Sigma_3 = 4$ $\mu_4 = [0,0,0], \Sigma_4 = 4$ $\mu_5 = [19,19,19], \Sigma_5 = 4$

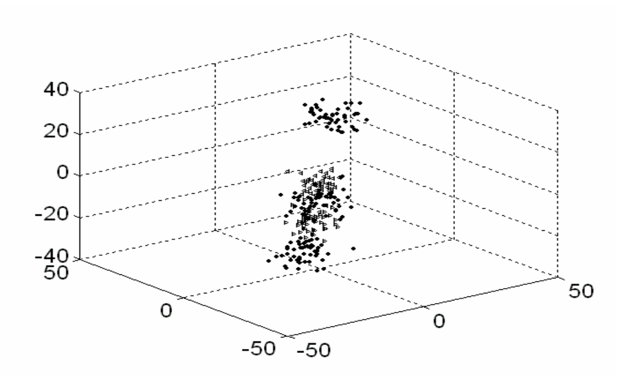

Figure1. Clusters in synthetic dataset SET III

Table 3. The performance of three clustering method

Data set	# Data in set	# Data in Clusters	# Clusters	# Space dimension	Cluster Overlapping
SET I	210	Each 70	3	2	No
SET II	210	Each 70	3	2	Yes, Medium
SET III	250	Each 50	5	3	Yes, Medium
Iris	150	Each 50	3	4	Yes, Little
Cancer	683	444 & 239	2	9	Yes, Little

VI. CONCLUSION

We addressed the clustering problem in this paper. We proposed a method based on combination of the particle swarm optimization (PSO) and the k-mean algorithm. We showed that the combined method has the advantage of both PSO and k-means methods while does not inherent their drawbacks. As the PSO algorithm successfully searches all space during the initial stages of a global search we used PSO algorithm at earlier stage of PSO-KM. As long as the particles in swarm being close to the global optimum, the algorithm switches to k-means as it can converge faster than PSO algorithm. We detected the proper stage for switching from PSO to k-means using the fitness function. The result of experiment on five datasets including real and synthetic data showed the hybrid algorithm outperforms K-means and PSO clustering.

REFRENCES

[1] Chen, C.-Y., and Ye, F, " Particle swarm optimization algorithm and its application to clustering analysis," in *Proc.* the IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan , pp. 789–794,2004.

[2] Selim, S. Z., and Ismail, M. A., "K-means type algorithms: a generalized convergence theorem and characterization of local optimality". *IEEE Transaction of Pattern Analysis Machine Intelligent*, 6, pp 81–87.

[3] Paterlini, S., and Krink, T., "Differential evolution and particle swarm optimization in partitional clustering" *in Proc. 2006 Computational Statistics and Data Analysis*, 50, pp 1220–1247.

[4] D.W. Boeringer, and D.H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," IEEE Transaction of Antennas Propagation 52 (3) (2004) pp 771–779.

[5] J Kennedy, and RC Eberhart, "Particle Swarm Optimization," in Proc. the IEEE International Joint Conference on Neural Networks, Vol. 4, pp 1942–1948, 1995.

[6] Y. Shi, and R.C. Eberhart, "A modified particle swarm optimizer," in *Proc.* IEEE World Conf. on Computation Intelligence (1998) pp 69–73.

[7] R.C. Eberhart, and Y. Shi, "Comparing Inertia Weights and Constriction Factors in Particle swarm Optimization," in Proc. congress on Evolutionary Computing, vol. 1 (2000) pp 84–88.

[8] Kaufman, L., and Rousseeuw, P. J., "Finding groups in data: Anintroduction to cluster analysis" New York: John Wiley & Sons (1990).

[9] ftp://ftp.ics.uci.edu/pub/machine-learning-databases/