A Novel Approach for Missing Data Processing based on Compounded PSO Clustering

HUNG-PIN CHIU, TSEN-JEN WEI, HSIANG-YI LEE
Department of Information Management
Nan Hua University
No.32, Chung Keng Li, Dalin ChiaYi, 622
ROC
hpchiu@mail.nhu.edu.tw, angle2567@yahoo.com.tw, hylee@mail.nhu.edu.tw

Abstract: - Incomplete and noisy data significantly distort data mining results. Therefore, taking care of missing values or noisy data becomes extremely crucial in data mining. Recent researches start to exploit data clustering techniques to estimate missing values. Obviously the quality of clustering analysis significantly influences the performance of missing data estimation. It was proven that clustering problem is NP-hard. Particle swarm optimization (PSO) is the recently suggested heuristic search process for solving data clustering problems. In this paper, a compounded PSO (CPSO) clustering approach is proposed for the missing value estimation. Normalization methods are first utilized to filter outliers and prevent some attributes from dominating the clustering result. Then the *K*-means algorithm and reflex mechanism are combined with the standard PSO clustering so that it can quickly converge to a reasonable good solution. Meanwhile, an iteration-based filling-in value scheme is utilized to guide the searching of CPSO clustering for the optimal estimate values. Effectiveness of the proposed approach is demonstrated on some data sets for four different rates of missing data. The empirical evaluation shows the superiority of CPSO over the well known *K*-means, PSO, and SOM-based approaches, and it is desirable for solving missing value problems.

Key-Words: - Particle swarm optimization, Data clustering, Missing values, Iteration-based filling-in scheme

1 Introduction

Data mining has attracted a great deal of attention in the information industry and in society as a whole in recent years. This is due to the wide availability of huge amounts of data and the imminent need for transferring such data into useful knowledge [1]. Real world data tend to be incomplete, noisy, and inconsistent. Clearly, incomplete and noisy data significantly distort data mining results. Therefore, data preprocessing of missing values or noisy data is very critical in successful data mining [2]. We can directly remove the entire data record if it has a missing value. However, the available information is reduced and important information may be lost. We can also simply replace each missing value with the attribute mean, but no relationships between attributes are exploited. Therefore, it is a common practice to fill in the missing value with the most probable value obtained by exploring and preserving the relationships between attributes.

During the past few years, several methods have been proposed to exploit attribute relationships for processing incomplete data sets [3-7]. In [3], Chen et al. presented a method to estimate null values in the distributed relational database environments. In this method, a rule base is used to indicate the relationships in which some attributes determine other attributes. All of the rules in the rule base, including the weights of the attributes, are given by domain experts. The data record which has a null value is compared to each rule in the rule base to see which one is the closest rule. Then, the null value can be estimated by the degree of closeness. In [4], Chen and Huang further used genetic algorithms to generate weighted fuzzy rules from relational database systems for estimating null values. The main drawbacks of these approaches are that the functional dependencies between attributes are not easily determined in most real life situations, and these methods can only estimate one null value for an attribute at one time.

Clustering is a popular unsupervised pattern classification technique which partitions the input attribute space into a number of regions based on some similarity metric such that similar data records are placed in the same cluster while the dissimilar ones are placed in separate clusters [1]. It is believed that the data records in the same cluster possess common characteristics, and can be utilized to predict and estimate the missing data. Many clustering algorithms exist in the literature, such as the *K*-means algorithm [1, 6], self-organizing feature map (SOM) neural network [1, 7], and so on.

Two types of clustering-based techniques for estimating missing values have been proposed. The first-type approaches cluster data based on the dependent variable (attribute) in advance, so that they only need to process the most proper clusters instead of all the data for estimating null values. Chen and Hsiao [5] used an automatic clustering algorithm to partition the database according to the value of a dependent variable. In each cluster, the concepts of “correlation” and “coefficient of determination” of the regression analysis were applied to derive the important information from independent variables for estimating null values of the dependent variable. Based on the same idea, Cheng and Wang [6] utilized two clustering algorithms to cluster data, and used fuzzy correlation and distance similarity to estimate null values of the dependent variable. The limitations of these approaches are that the functional dependencies between attributes must be determined in advance, and all the missing data distributed over different attributes can not be estimated at the same time.

In the second type, clustering analysis is performed based on all the attributes, and the common characteristic in each cluster is used to estimate missing data simultaneously. Lin and Hsieh [7] proposed a generalized approach to estimate missing values based on SOM networks. First, all missing values are replaced with the corresponding attribute mean. Second, SOM neural networks with different numbers of nodes cluster the data records into many subsets respectively. A trial-and-error scheme uses Euclidean distance function to determine the best clustering topology which has the minimum distance. Then, each missing value is refilled in with the mean value of its corresponding attribute in the cluster. The process is repeated until a termination criterion is satisfied. Their empirical evaluation showed the SOM-based approach works well for missing value estimation.

It is apparent that the quality of clustering analysis significantly influences the performance of missing value estimation. It was proven that clustering analysis is an NP-hard problem [8]. Various heuristic algorithms, such as particle swarm optimization (PSO) [9-11], genetic algorithm [12] and simulated annealing [13] were used to solve clustering problems. PSO is suggested because it is easy to implement and fewer parameters are required [11]. Recently, Merwe and Engelbrecht improved the performance of the PSO clustering by using *K*-means clustering to seed the initial swarm [10]. Kao et al. [11] proposed two reflex schemes to improve the efficiency of PSO algorithm on data clustering problems.

In this paper, we combine PSO clustering with the *K*-means and reflex schemes to form a compounded PSO (CPSO) clustering method for improving the quality of clustering results. Then a simple and effective CPSO clustering-based approach is proposed to estimate missing data simultaneously. The CPSO clustering process is iterative. At the end of iteration, the global best particle maintains a set of highly fit clusters found so far. An iteration-based filling-in scheme is proposed to iteratively update the missing data, and gradually guide the searching of the CPSO clustering for the fittest estimate values. The Effectiveness of the proposed approach is demonstrated on one artificial and two real life data sets having different characteristics. A series of experiments have been conducted to compare the performance of the proposed novel approach with other approaches for a variety of missing rates. From the experimental results, it can be seen that the novel approach is desirable for solving missing value problems.

2 Background

Clustering is a well known exploratory data analysis method where the objective is to partition the data into a number of clusters. Let the input space *S* be represented by *n* points $\{x_1, x_2,\ldots, x_n\}$, and the *K* clusters be represented by $C_1, C_2,\ldots, C_K$. Then

$$C_i \neq \emptyset \quad \text{for } i = 1,\ldots, K,$$

$$C_i \cap C_j = \emptyset \quad \text{for } i, j = 1,\ldots, K \text{ and } i \neq j, \text{ and}$$

$$\bigcup_{i=1}^{K} C_i = S\ .$$

Several algorithms for clustering data are available in the literature. The *K*-means algorithm, one of the most widely used ones, attempts to solve the clustering problem by optimizing a metric given for minimizing the distance between data point and cluster centers. However, the known limitation of the *K*-means algorithm is that it may get stuck at sub-optimal solutions depending on the choice of the initial cluster centers [1]. In the mean time, the PSO clustering method is quite simple and effective, and it can avoid the minimum local values [9-11]. Recently, few researches proposed different techniques to improve PSO clustering, and showed that it is both robust and suitable for solving data clustering problems.

2.1 Standard PSO Algorithm

PSO is a population-based stochastic search process, modeled after the social behavior of a bird flock [9-11, 14]. The algorithm maintains a population of particles, where each particle represents a potential solution to an optimization problem. The aim of the PSO is to find the particle location that results in the best evaluation of a given fitness evaluation function. The PSO process is iterative; and the entire set of iterations is called a run.

Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. The information is called *pbest*. The overall best value, and its location, obtained so far by any particle in the population, is called *gbest*. All particles search solutions through the search space to find optimal solution by changing continuously the velocity and location of each particle toward its *pbest* and the *gbest* locations. The process for implementing the PSO is as follows [14]:

1) Initialize a population of particles with random positions and velocities on *d* dimensions in the problem space.
2) For each particle, evaluates the desired optimization fitness function in *d* variables.
3) Compare particle's fitness evaluation with particle's *pbest*. If current value is better than *pbest*, then set *pbest* value equal to the current value, and the *pbest* location equal to the current location in *d*-dimensional space.
4) Compare fitness evaluation with the population's overall previous best. If current value is better than *gbest*, then reset *gbest* equal to the current particle's location and value.
5) Change the velocity and position for each particle according to equations (1) and (2), respectively:

$$v_{id}^{new} = v_{id}^{old} + c_1 \times rand \times \left(p_{id} - x_{id}^{old}\right) + c_2 \times rand \times \left(p_{gd} - x_{id}^{old}\right) \tag{1}$$

$$x_{id}^{new} = x_{id}^{old} + v_{id}^{new}, \tag{2}$$

where $c1$ and $c2$ are two positive constants between 0 and 2. v_{id}^{old} and x_{id}^{old} are the previous velocity and position of *i*-th particle in dimension *d*. p_{id} is the *pbest* location of particle *i* and p_{gd} is the *gbest* location in dimension *d*. After v_{id}^{new} is computed by equation (1), x_{id}^{new} can be updated by equation (2) which is the new position of each particle.

6) Repeat step 2)-step 5) until a stop criterion is met, usually a sufficiently good fitness or a predefined number of iterations.

2.2 Clustering Optimization using PSO

A few PSO-based clustering methods had been proposed to search the cluster centers in the data sets automatically [9-11]. A single particle represents the *K* cluster center vectors, where *K* is the number of clusters. That is, each particle x_i is represented as follows:

$$x_i = (z_{i1}, \ldots, z_{ij}, \ldots, z_{iK}) \tag{3}$$

where z_{ij} refers to the *j*th cluster centroid vector of the *i*th particle in cluster C_{ij}. Therefore, a swarm represents a number of candidate clusterings for the current input data vectors. The fitness value of each particle can be computed by following the fitness function:

$$fitness = \sum_{j=1}^{K} \sum_{p \in C_j} \left| p - z_j \right|, \tag{4}$$

where *p* is a data vector in problem space, and each data vector is assigned to the cluster to which the data vector is the most similar. This criterion measures the distances between data vectors within each cluster. Minimizing the intra-cluster distances corresponds to compact clusters that are well separated. Using the standard PSO, data vectors can be clustered as follows:

1) Initialize each particle to contain *K* randomly selected cluster center positions.
2) Evaluate the fitness function for each particle using equation (4).
3) Update the global *gbest* and local *pbest* positions.
4) Update the cluster centers for each particle using equations (1) and (2). After the update, it is possible that the cluster centers in a particle may be out of the search space, where the search space is defined by the largest and the smallest coordinate values in all dimensions among all data points in the datasets. If that happens, then the coordinate values which are out of boundary are reset to the boundary value [11].
5) Repeat step 2)-step 4) until a stop criterion is satisfied or a predefined number of iterations is completed.

The search of PSO algorithm starts from multiple positions in parallel. The population-based searching characteristic reduces the effect that initial conditions have, as opposed to the *K*-means algorithm. On the other hand, the *K*-means algorithm tends to converge faster than the PSO clustering, but usually with a less accurate clustering. Merwe and Engelbrecht [10] showed that the performance of the *PSO*-clustering can further be improved by seeding the initial swarm with the result of the *K*-means algorithm. The hybrid algorithm first executes the *K*-means algorithm once. The result of the *K*-means algorithm is then used as one of the initial particles, while the rest of the swarm is initialized randomly. The PSO clustering algorithm as presented above is then executed

Kao et al. [11] pointed out that, although PSO is a good clustering method, it does not perform well when the dataset is large or complex. It is because that when particles reach the boundary of search space, they tend to stay there without turning to other directions for better solutions. As a result, the search time is spent without improving the quality of solutions. Therefore, they proposed two reflex schemes in PSO such that particles will bounce back from the boundary and move to other directions. The first scheme proposed to reflex *α* percent of the range in search space. However, determining the appropriate *α* value is not an easy question. The second scheme determined the reflecting range by comparing the particle location with the global best particle location. It was defined as following:

$$x_{id}^{new}=\begin{cases} x_{id}^{old}+\left|p_{gd}-x_{id}^{old}\right|\times rand,\ if\ moveout\,of\ LB \\ x_{id}^{old}-\left|p_{gd}-x_{id}^{old}\right|\times rand,\ if\ moveout\,of\ UB \end{cases}, \quad (5)$$

where *LB* and *UB* are the lowest and highest values of each vector in data sets. The study showed that the hybrid reflex mechanism and PSO clustering method are better than the genetic algorithm-based approach developed by Sanghamitra and Ujjwal [12] and the simulated annealing-based approach developed by Ujjwal and Malay [13].

3 A Novel Missing Value Estimation Approach

Data clustering technique is the partitioning of a dataset into subsets so that the data in each subset share common pattern. The shared characteristic can be utilized to predict and estimate the missing values. In this paper, we introduce a PSO clustering-based approach to solve the missing value problems. The main idea is that the global best particle in the population represents the best clustering result obtained so far. We iteratively use the current best clustering information to update and guide the missing data to fit the original values. Within each iteration, we fill in missing values using the attribute mean for all data records belonging to the same cluster as the given record with missing values. Moreover, we combine the *K*-means algorithm, reflex mechanism and standard PSO clustering so that it can quickly converge to a reasonable good solution.

A flowchart of the proposed approach is provided in Fig. 1. After data preprocessing, all missing data is first replaced with the corresponding attribute mean. Then, the compounded PSO (CPSO) clustering method clusters the data records into many subsets, and each missing value is refilled in with the mean of its corresponding attribute in the cluster. We exploit an iteration-based filling-in strategy to continuously guide the estimation process. The proposed extension makes the CPSO clustering method more robust and suitable for missing value problem. The different steps of the proposed estimation approach are now described in detail as follows:

(a) Data preprocessing: For distance measurement-based clustering methods, normalization is particularly useful because it helps prevent attributes with initially large ranges from outweighing attributes with initially smaller ranges [1]. In this study, we use min-max normalization to perform a linear transformation on the original data, while preserves the relationships among the data values. However, outliers are able to dominate the min-max normalization, hence we first employ *z*-score normalization to filter outliers. In *z*-score normalization, a value, *v*, of attribute *A* is normalized to *v'* by computing

$$v'=\frac{v-\overline{A}}{\sigma_A}, \quad (6)$$

where $\overline{A}$ and σ_A are the mean and standard deviation, respectively, of attribute *A*. If *v'* is greater than 3 or less than -3, then *v* is regarded as outlier and will be removed. Thereafter, min-max normalization is applied. Suppose that min_A and max_A are the minimum and maximum values of an attribute, *A*. Min-max normalization maps a value, *v*, of *A* to *v'* in the range [$nmin_A$, $nmax_A$] by computing

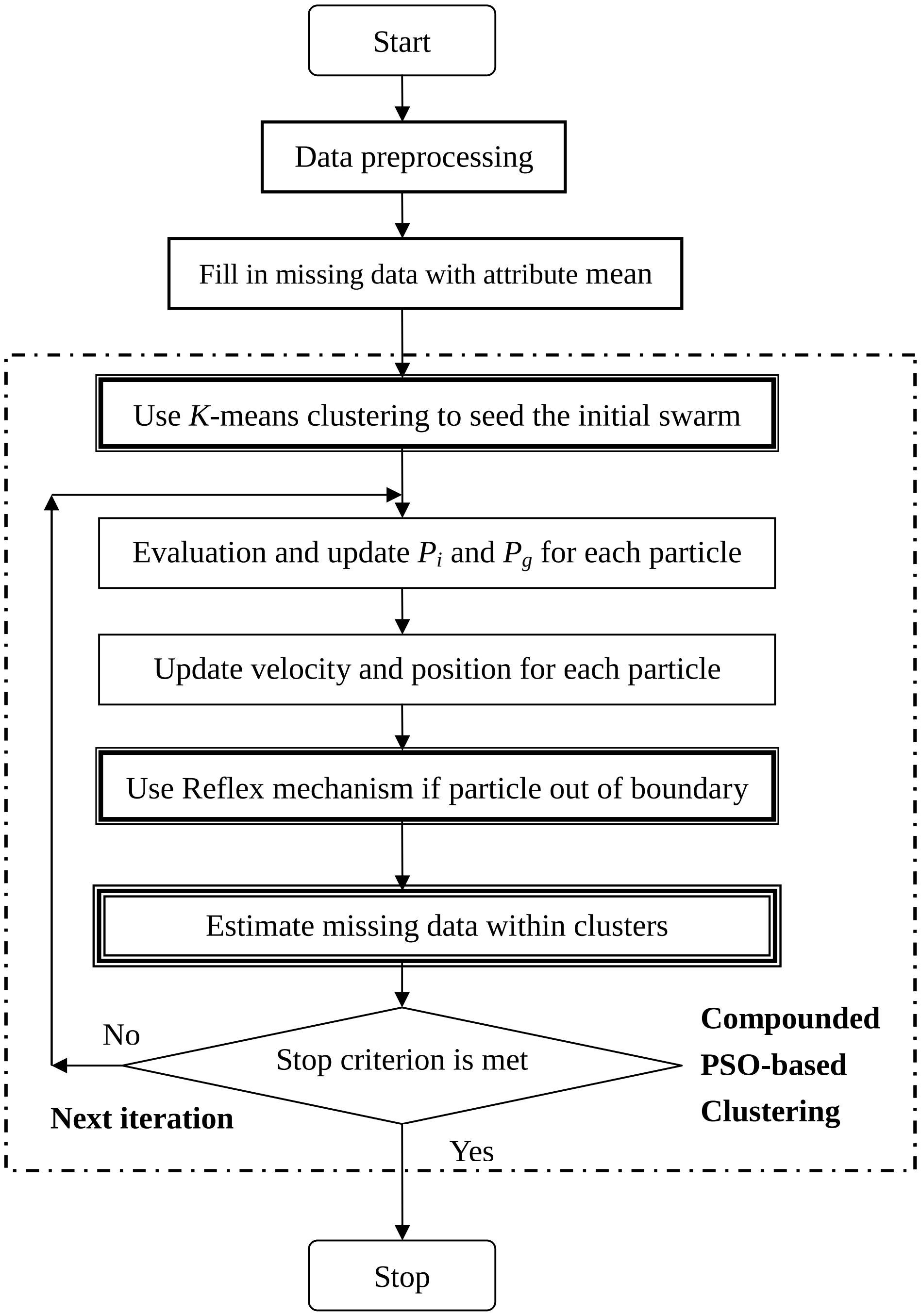

Start
Data preprocessing
Fill in missing data with attribute mean
Use K-means clustering to seed the initial swarm
Evaluation and update P_i and P_g for each particle
Update velocity and position for each particle
Use Reflex mechanism if particle out of boundary
Estimate missing data within clusters
Stop criterion is met
No
Next iteration
Yes
Compounded PSO-based Clustering
Stop

Fig. 1. Flowchart of the proposed approach for missing data processing

$$v' = \frac{v - \min_A}{\max_A - \min_A}\left(n\max_A - n\min_A\right) + n\min_A, \quad (7)$$

where $nmin_A$ and $nmax_A$ are 0 and 1, respectively.

(b) Initial replacement: We calculate the attribute mean to fill in the missing values.

(c) Population initialization: Each particle represents a feasible solution of the problem. In this study, a particle is assumed to encode the centers of *K* clusters as defined in equation (3). For example, let the four cluster centroid vectors for a two-dimensional problem with four clusters be (0.1, 0.5), (0.4, 0.2), (0.7, 0.9), and (0.8, 0.4), then the particle's position looks like [0.1, 0.5, 0.4, 0.2, 0.7, 0.9, 0.8, 0.4]. In order to improve the clustering performance, we use the *K*-means clustering to seed the initial swarm as presented in [10]. The clustering result of *K*-means algorithm is used as one of the initial particles, while the rest of the swarm is initialized randomly. This technique is basically amount to use PSO to refine the clusters formed by *K*-means [10].

(d) Evaluation and update P_i and P_g: The fitness value of each particle is computed by the criterion function defined in equation (4). The criterion tries to make the resulting *K* clusters as compact and as separate as possible. The value of P_i and P_g will be updated if the new value is better than the old ones.

(e) Update velocity and position: All particles search solutions through the search space by following the current best particle. The particle's velocity and position are continuously updated based on equation (1) and (2) for attempting to find optimal solutions.

(f) Reflex mechanism: Kao et al. [11] pointed out when particles reach the boundary of search space, they tend to stay there. In order to improve the efficiency, we adopt the reflex schemes developed in [11] to make particles bounce back from the boundary and move to other directions for better solutions. The reflecting range is calculated according to equation (5).

(g) Estimate missing data within clusters: At the end of iteration in PSO clustering analysis, a set of highly fit clusters found by the global best particle is built. Each missing value is then refilled in with the mean value of its corresponding attribute in the cluster.

(h) Iteration-based filling scheme: In this study, we propose an iteration-based filling scheme to update the missing values. The aim of the missing value problem is to estimate the missing data as close as possible. Replacing the missing values with the attribute mean in a cluster makes the corresponding estimate closer to the optimum. The replacement mechanism can be incorporated into the PSO process to gradually guide the searching to move toward the optimum solution. Therefore, we iteratively cluster and estimate the missing data based on the current global best particle so that the proposed approach has much potential for solving the missing value problem.

In order to verify the performance of the proposed approach for estimating missing values, we utilize mean of absolute error (*MAE*) as evaluation criterion [7], which is defined as follow:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|\hat{e}_i - o_i\right|, \qquad (8)$$

where $\hat{e}_i$ is the estimating value, o_i is the original value, and *m* denotes the size of missing data.

4 Implementation and Evaluation

The effect of the proposed estimate approach was investigated through empirical simulations. The experimental settings and the related simulation results were presented in this section.

4.1 Experimental Settings

Three datasets with a variety of complexity were used to examine the performance of the proposed approaches. Overlapping and non-overlapping data sets were considered. An artificial two-dimension dataset with four unique classes was created. A total of 600 data points were drawn from four independent bivariate normal distributions, where classes were distributed according to:

$$N_2\left(\mu_i = \begin{pmatrix} m_i \\ m_i \end{pmatrix}, \Sigma\begin{bmatrix} 0.5 & 0.05 \\ 0.05 & 0.5 \end{bmatrix}\right),\quad i = 1,\ldots,4\,, \qquad (9)$$

where N_2 means that data points are two-dimension bivariate normal distribution, μ is the mean vector and $\sum$ is the covariance matrix; m_1 = -3, m_2 = 0, m_3 = 3, and m_4 = 6. The dataset is illustrated in Fig.2.

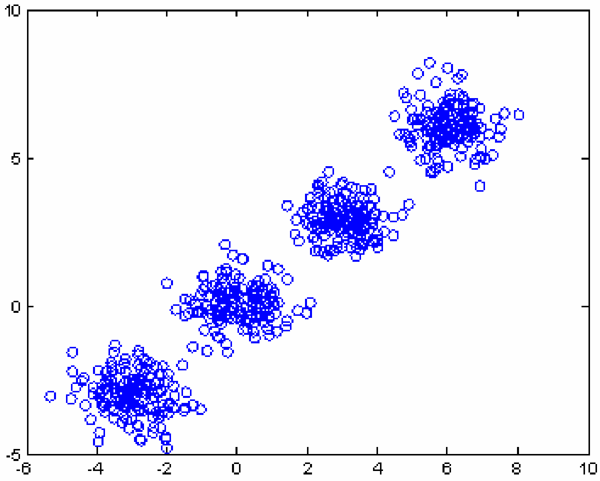


Fig. 2 Artificial dataset

The real-life data sets considered are Iris plants and Vowel. They are available at the ftp site of University of California, Irvine Donald Bren School of Information and Computer Sciences [15]. The Iris plants dataset represents three classes of irises having four feature values. The four feature values represent the sepal length, sepal width, petal length and the petal width in centimeters. It contains three classes Setosa, Versicolor and Virginica, with 50 samples per class. It is known that two classes Versicolor and Virginica have a large amount of overlap while the class Setosa is linearly separable from the other two. The Vowel dataset consists of 871 Indian Telugu vowel sounds. It contains three attributes and six classes. The six classes are known to be linearly inseparable as shown in Fig.3. It is clear from the figure that we can identify the presence of outliers, and the attributes have considerable differences in data ranges. Table 1 lists the characteristics of these datasets.

In addition to the proposed methods, CPSO, we also evaluated the methods of the original PSO clustering and *K*-means algorithm, labeled by PSO and *K*-means, respectively. Four different rates of missing data, including 5%, 10%, 15%, and 20%, are tested in the experiments. For the results reported, average over 20 trials are given. For PSO, both *c*1 and *c*2 are 2. The population size selected was problem-dependent. For simplicity, all empirical experiments were carried out with a population size of 5*N* in this study. *N* is computed as follows [11]:

$$N = K*d \tag{10}$$

where *d* is the dataset dimension and *K* is the anticipated number of clusters.

4.2 Convergence Behaviors

In this study, we incorporate an iteration-based filling-in mechanism into the CPSO clustering process to guide the searching for the fittest estimate values. Since CPSO clustering is a population-based stochastic search process, the performance of the proposed estimation approach varies with that of the CPSO clustering from iteration to iteration. The simulation results of all experiments indicate that the proposed approach quickly converges to a reasonable good solution for both fitness values and *MAE* values. Fig. 4 and Fig. 5 illustrated to a more insight look into the convergence behaviors.

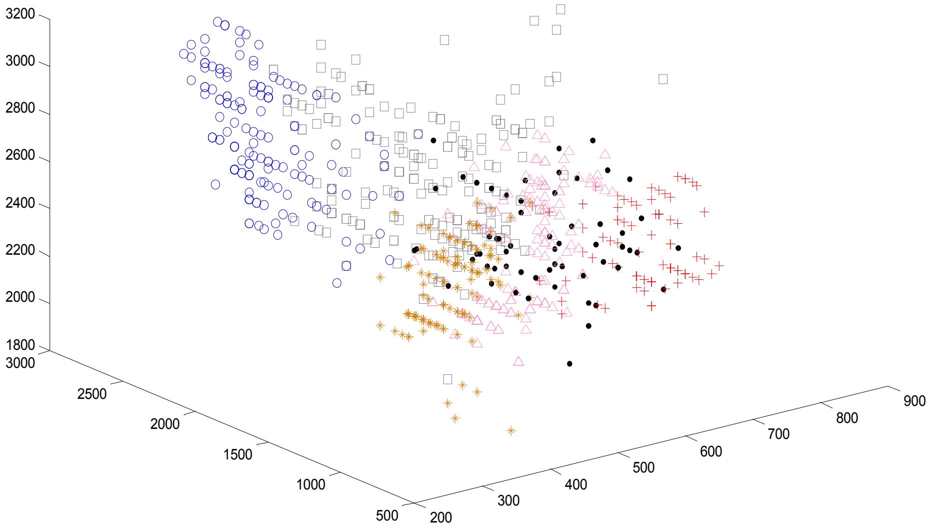
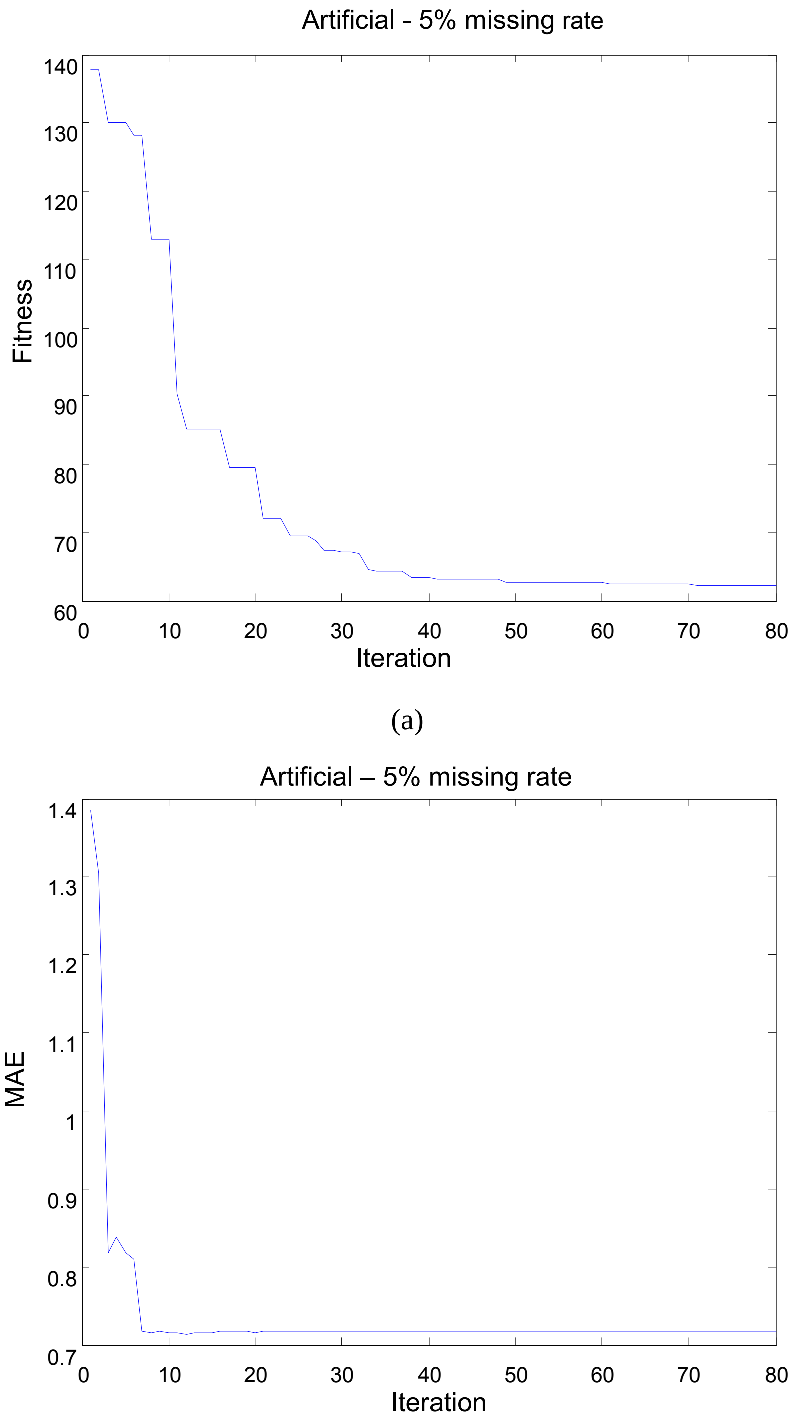


Fig. 3 Vowel dataset

Table 1: Characteristics of the data sets

dataset	# classes	# attributes	# data records
Artificial	4	2	600(150,150,150,150)
Iris plants	3	4	150(50,50,50)
Vowel	6	3	871(72,89,172,151,207,180)

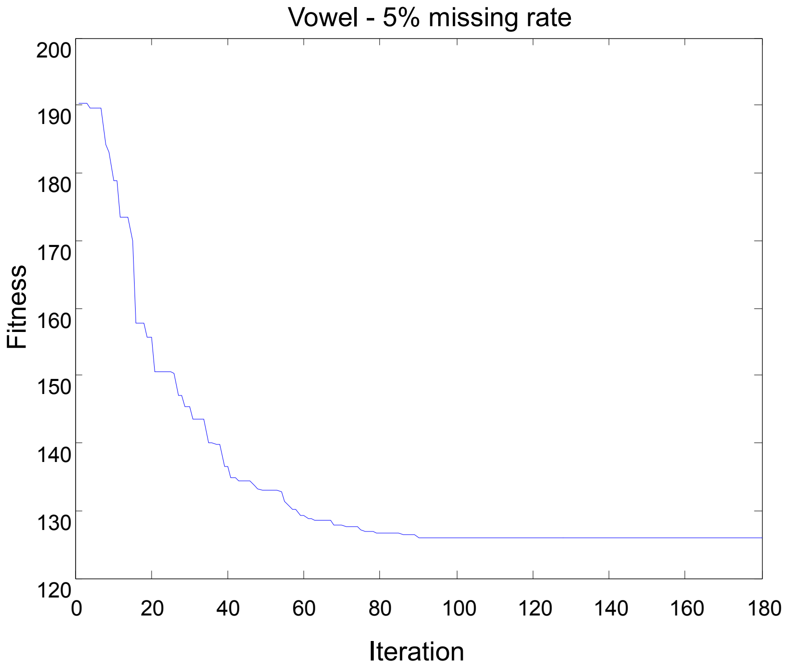


(a)

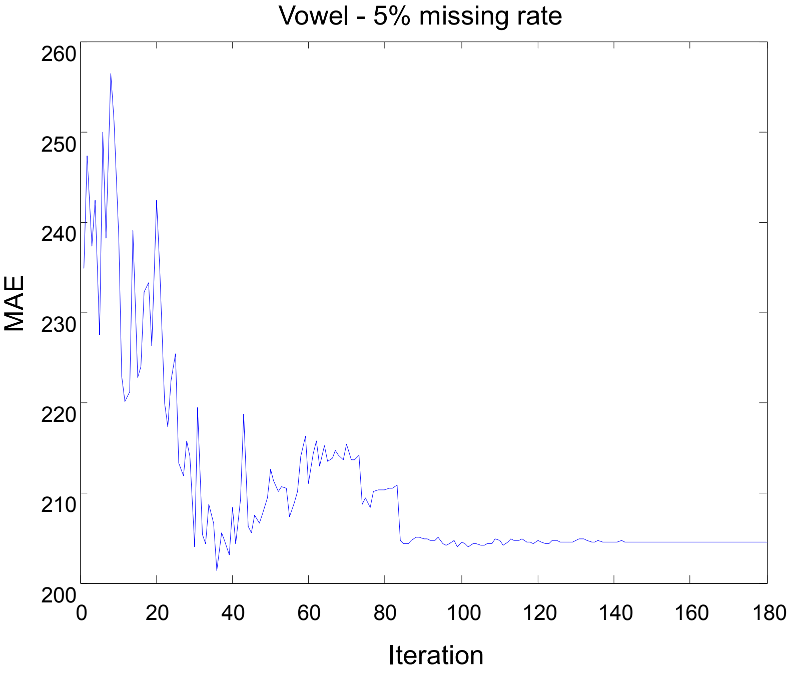
(b)

Fig. 4 Convergence behaviors in artificial dataset for the 5% missing rate: (a) performance graph of *fitness* values and (b) performance graph of *MAE* values.

Fig. 4(a) and (b) show plots of the best values of the fitness function and the corresponding *MAE* values across 80 iterations in artificial dataset. As we can see, the two curves descend rapidly at the beginning, and then as the population converges on the nearly optimal solution, they descend more slowly, and finally flatten at the end.



(a)



(b)

Fig. 5 Convergence behaviors in Vowel dataset for the 5% missing rate: (a) performance graph of *fitness* values and (b) performance graph of *MAE* values.

Fig. 5(a) shows fitness function values across 180 iterations in Vowel dataset. It presents the similar convergence behavior with the non-overlapping artificial dataset, but converges slower to better fitness values. However, Fig. 5(b) shows the erratic behavior of the *MAE* value curve.

The reason is mainly due to the iteration-based filling scheme and the significantly overlapping data. The process of alternate clustering and replacement iteratively changes the missing values, and leads to a dynamically unstable curve in the front part of iterations. Nevertheless, the *MAE* curve still presents a descending trend, and finally converges to a good enough solution with the help of the proposed guided mechanism.

4.3 Implementation Results

In order to examine the effectiveness of the proposed CPSO estimate approach, we performed on four different missing rates to observe the performances for three datasets. The average *MAE* values obtained from the CPSO clustering-based estimation approach are shown in Table 2. The results are averaged over 20 trails. The estimate results obtained from the *K*-means algorithm and the PSO clustering are also given for comparison. As can be seen clearly, CPSO is the best in every aspect, especially in Vowel dataset. In artificial dataset, CPSO and PSO offer the similar results. They have better performances than *K*-means. In Iris plants dataset, competitive results were observed with all methods. However, the performance of CPSO is slightly superior to those of PSO and *K*-means. In Vowel dataset, CPSO performed significantly well compared with *K*-means and PSO. In general, CPSO and PSO outperform *K*-means, and CPSO is better than PSO. It means that PSO-based estimate approach provides the superiority over *K*-means approach, meanwhile, combining the *K*-means algorithm and the reflex mechanism into PSO clustering effectively improve the estimate performance. The experimental results show the CPSO clustering is a suitable process for estimating missing values.

4.4 Comparison with Neural Networks Approach

The proposed CPOS approach can estimate all missing data simultaneously. As discussed in the introduction section, the SOM neural network approach proposed by Lin and Hsieh [7] can also estimate missing values simultaneously. Their empirical study showed the estimate performance of SOM is better than that of the *K*-means algorithm. However, the approach needs lots of time to training different SOM neural networks, and employs a trial-and-error scheme to determine the best SOM clustering topology. In this paper, for fair comparison, we compared our method with the SOM approach using a dataset given in their work.

Table 2 Comparison of the performances based on the three clustering methods
(Results in bold are the best in the Table)

Dataset	Missing rate	*MAE*		
		K-means	PSO	CPSO
Artificial	5%	2.456	0.959	**0.923**
	10%	1.770	1.132	**1.069**
	15%	1.714	1.205	**1.124**
	20%	1.763	1.249	**1.208**
Iris plants	5%	0.293	0.296	**0.289**
	10%	0.253	0.252	**0.240**
	15%	0.332	0.322	**0.297**
	20%	0.317	0.316	**0.289**
Vowel	5%	280.903	208.845	**185.303**
	10%	240.402	215.635	**177.501**
	15%	230.465	196.206	**179.740**
	20%	231.175	187.671	**185.053**

The tested dataset used in [7] contains three experiment parameters from an industrial manufacturing. In their study, the estimate was performed for 5% missing rate, that is, three randomly selected attribute values was regarded as the missing values. The original dataset and the selected missing values are shown in Table 3. Noticeably, the attribute *RR* has initially larger data range than attribute *NU* and *Tan/CU*. The published results found in [7] are given to compare with our approach. Table 4 and 5 list the comparisons of the estimated results and the corresponding *MAE* values. It should be noted that the estimated result of CPSO is significantly better in the attribute *RR*, and relatively competitive results are observed in the other two attributes. In general, the *MAE* of the CPSO is clearly less than that of the SOM. That is, it is observed from the comparison results that CPSO performed significantly well in estimating missing values compared with SOM.

Table 3 Industrial manufacturing dataset
(The selected attribute values are in bold)

	RR	NU	Tan/Cu
1	294	14.3	4
2	289	15.7	4.3
3	314	23.2	5.6
4	375	12.1	3.7
5	437	**8.7**	4.9
6	498	6.5	6.1
7	481	8.99	4.2
8	588	11.8	4.3
9	660	12.4	5.3
10	242	16.2	4.6
11	268	26.9	**4.1**
12	340	10.5	5.3
13	377	16.9	3.9
14	434	5.06	4.7
15	494	7.08	5.4
16	**483**	8.76	5.2
17	580	15.1	4.6
18	651	5	5.8

Table 4 Comparison of the estimated results of CPSO with SOM
(Results in bold are the closest)

	RR	NU	Tan/Cu
Real data	483	8.7	4.1
SOM	450.67	**7.6**	4.49
CPSO	**475.67**	7.15	**4.35**

Table 5 Comparison of the estimated *MAE* value of CPSO with SOM
(Results in bold are the best in the Table)

	RR	NU	Tan/Cu	*MAE*
SOM	32.33	**1.1**	0.39	11.27
CPSO	**7.33**	1.55	**0.25**	**3.04**

5 Conclusions and Future Works

In this paper, we have studied the applicability of the simple CPSO clustering to estimate missing data. The proposed CPSO clustering approach combines the best features of *K*-means, reflex mechanism and PSO clustering. An iteration-based filling-in scheme is utilized to guide the searching of CPSO to move toward the reasonable good solutions. The effectiveness of the proposed approach is demonstrated on both artificial and real life datasets with a variety of complexity. Both overlapping and non-overlapping datasets are considered for this purpose. The empirical evaluation shows the superiority of the proposed approach over well known *K*-means and PSO. We also compared our methods with the SOM-based estimation approach, developed by other study and the results are also encouraging. The experimental results show the novel approach is valuable and desirable to estimate missing data.

In the future, investigation of proper representation of categorical data for PSO-based clustering is needed and more complicated cases will be tested to evaluate the performance of the proposed data estimating approach. Also we will like to explore and compare the other data filling-in scheme, such as the run-based strategy. Moreover, study on dynamically determining the optimal number of clusters for CPSO will be conducted. The adaptive PSO clustering algorithm is expected to further improve the performance of estimating missing values.

References:

[1] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2007.

[2] W. Wei and Y. Tang, A Generic Neural Network Approach for Filling Missing Data in Data Mining, *Proceeding of the 2003 IEEE International Conference on Systems, Man and Cybernetics*, 2003, pp.862-867.

[3] S.M. Chen and H.H. Chen, Estimating Null Values in the Distributed Relational Databases Environments, *Cybernetics and Systems: An International Journal*, 31(8), 2000, pp. 851-871.

[4] S.M. Chen and C.M. Huang, Generating Weighted Fuzzy Rules from Relational Database Systems for Estimating Null Values Using Genetic Algorithms, *IEEE Transaction on Fuzzy Systems*, Vol.11, No.4, 2003, pp. 495-506.

[5] S.M. Chen and H.R. Hsiao, A New Method to Estimate Null Values in Relational Database Systems Based on Automatic Clustering Techniques, *Information Sciences*, 2005, pp. 47-69.

[6] C.H. Cheng and J.W. Wang, A New Approach for Estimating Null Value in Relational Database, *Soft Computing*, 2006, pp.104-114.

[7] C. N. Lin and K. L. Hsieh, The Generalized Approach based on Artificial Neural Networks to the Problem of Missing Data, *Journal of Commercial Modernization*, 3(2), 2005, pp. 108-114. (*in Chinese*)

[8] A.B. Adib, NP-hardness of the Cluster Minimization Problem Revisited, *Journal of Physics A: Mathematical and General*, no.40, 2005, pp. 8487-8492.

[9] C.Y. Chen and Fun Ye, Particle Swarm Optimization Algorithm and Its Application to Clustering Analysis, *Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control*, Taipei, Taiwan, March 21-23, 2004, pp.789-794.

[10] DW van der Merwe and AP Engelbrecht, Data Clustering using Particle Swarm Optimization, *Proceedings of the 2003 IEEE International Conference on Evolutionary Computation*, 2003, pp. 215-220.

[11] I.W. Kao, C.Y. Tsai and Y.C. Wang, An Effective Particle Swarm Optimization Method for Data Clustering, *Proceeding of the 2007 IEEE Industrial Engineering and Engineering Management*, 2007, pp. 548-552.

[12] B. Sanghamitra and M. Ujjwal, An Evolutionary Technique based on *K*-means Algorithm for Optimal Clustering in R^N, *Information Sciences*, vol.146, 2002, pp. 221-237.

[13] M. Ujjwal and K. P. Malay, Clustering Using Simulated Annealing with Probabilistic Redistribution, *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, no. 2, 2001, pp. 269-285.

[14] R.C. Eberhart and Y. Shi, Particle Swarm Optimization: Developments, Applications and Resources, *Proceedings of the 2001 IEEE Congress on Evolutionary Computation*, 2001, pp.81-86.

[15] ftp://ftp.ics.uci.edu/pub/machine-learning-datasets/