Journal of Quantum Computing
DOI:10.32604/jqc.2020.09717
Article

Tech Science Press

Weighted Particle Swarm Clustering Algorithm for Self-Organizing Maps

Guorong Cui, Hao Li, Yachuan Zhang, Rongjing Bu, Yan Kang*, Jinyuan Li and Yang Hu

Department of Software, Yunnan University, Kunming, 650500, China
*Corresponding Author: Yan Kang. Email: kangyan@ynu.edu.cn

Received: 6 April 2020; Accepted: 12 August 2020

Abstract: The traditional K-means clustering algorithm is difficult to determine the cluster number, which is sensitive to the initialization of the clustering center and easy to fall into local optimum. This paper proposes a clustering algorithm based on self-organizing mapping network and weight particle swarm optimization SOM&WPSO (Self-Organization Map and Weight Particle Swarm Optimization). Firstly, the algorithm takes the competitive learning mechanism of a self-organizing mapping network to divide the data samples into coarse clusters and obtain the clustering center. Then, the obtained clustering center is used as the initialization parameter of the weight particle swarm optimization algorithm. The particle position of the WPSO algorithm is determined by the traditional clustering center is improved to the sample weight, and the cluster center is the "food" of the particle group. Each particle moves toward the nearest cluster center. Each iteration optimizes the particle position and velocity and uses K-means and K-medoids recalculates cluster centers and cluster partitions until the end of the algorithm convergence iteration. After a lot of experimental analysis on the commonly used UCI data set, this paper not only solves the shortcomings of K-means clustering algorithm, the problem of dependence of the initial clustering center, and improves the accuracy of clustering, but also avoids falling into the local optimum. The algorithm has good global convergence.

Keywords: Self-organizing map; weight particle swarm; K-means; K-medoids; global convergence

1 Introduction

With the advent of the era of big data and artificial intelligence, information has been growing exponentially. We are faced with a huge amount of text, video, pictures, and audio data. How to dig out the information with real value from the massive data, it has gradually become one of the research topics in the field of computer. Traditional data analysis, which relies on personal experience and teamwork to identify and determine results. It is not only unsatisfactory but also a waste of time and human resources. Therefore, data mining technology was born to help people extract valuable information from massive data. As an effective tool of data mining, clustering algorithm has been widely applied in many fields, including machine learning, pattern recognition, image analysis, information retrieval, computer vision, etc. Efficient data mining ability has attracted more and more attention [1].

As a common technical means in the field of data mining, so far, scholars at home and abroad have proposed many classic clustering algorithms, such as k-means [2] and k-medoids [3]. These algorithms are simple in calculation and fast in convergence. However, there are also two inherent disadvantages: (I) The determination of cluster number K, and the selection of K according to what index will directly affect the clustering accuracy; (II) Selection of initial clustering center. The clustering effect depends on the initialization of the clustering center.

In recent years, domestic and foreign scholars have proposed improvement schemes for the shortcomings of the K-means algorithm. For example, the x-means algorithm proposed by Pelleg [4] successfully solved the K value problem with the help of Bayesian information criteria (BIC) based on K-means. Park [5] chose a new center point and calculate the distance relationship between objects, which improved the efficiency of the algorithm, but failed to improve the clustering accuracy. Merwe et al. [6] proposed the clustering algorithm of particle swarm optimization (PSO)and K-means fusion, which effectively improved the convergence speed and the effectiveness of clustering to a certain extent.

2 Related Research

As an important branch of data mining, clustering algorithm has been studied for many years. A large number of clustering algorithms have emerged so far. However, since the data itself has its form and dimension, no algorithm is universal to the data, and all kinds of algorithms have some defects. For example, some clustering algorithms have significant results on middle and low dimensional data but do not perform well in high-dimensional data. Some clustering algorithms can only deal with the data of special distribution structure and cannot deal with the data of other distribution well.

These defects require scalability of the algorithm, the ability to process different types of data, and the ability to discover clusters of various shapes to solve “noise” and outliers. Traditional clustering algorithms have been unable to solve the above problems. Some scholars conducted clustering by integrating swarm intelligent optimization algorithms and found that a better clustering effect could be achieved.

2.1 Traditional Clustering Algorithm

Traditional clustering algorithms are generally based on partition, hierarchy, density, grid, and model clustering algorithms.

The clustering algorithm based on partition divides the sample set into several disjoint clusters according to the distance rule, and iterates until the target function stops the clustering division at the minimum. This method is easy to implement and converges quickly, but its complexity is linearly related to sample size, sample dimension and clustering center. Its representative algorithms include K-means [2], K-medoids [3], CLARANS [7], etc.

Hierarchical clustering algorithms can be divided into agglomerating hierarchical clustering and splitting hierarchical clustering. Early clustering algorithms include AGNES and DIANA clustering proposed by Kaufman et al. [8]. Afterward, BIRCH algorithm proposed by Zhang et al. [9] made use of clustering features and clustering feature trees for hierarchical clustering. Guha et al. CURE [10] algorithm, ROCK [11] algorithm, and Karypis et al. CHAMELEON [12] algorithm are also three famous clustering algorithms.

Compared with partition clustering and hierarchical clustering, the density-based clustering algorithm is not only applicable to convex sample sets but also can find clusters of various shapes and sizes in noisy data. DBSCAN [13] algorithm is a very typical density clustering algorithm, which requires two parameters: distance parameter and density threshold parameter, and divides the “density reachable” samples in space into one class. OPTICS [14], as an extension of DBSCAN, has improved the sensitivity to parameter Settings of DBSCAN.

The grid-based clustering algorithm quantifies the object space into a finite number of units, which form the network structure on which all clustering operations are carried out. STING [15] algorithm is a typical representative of the grid-based clustering algorithm. CLIQUE [16] combines the idea of the grid and density clustering, and it can cluster large-scale high-dimensional data.

The model-based clustering algorithm uses a statistical model and neural network to obtain clustering distribution information of data. The statistical method includes COBWEB algorithm. The network neural method has self-organizing maps (SOM) algorithm.

2.2 Integrated Clustering Research

In recent years, the shortcomings of traditional clustering algorithms are gradually exposed. The selection of the initial clustering center is sensitive, the number of clusters is difficult to determine, and the high requirements on data format will be a huge challenge to the clustering field. The integrated clustering algorithm improves the accuracy of the algorithm, avoids falling into local optimization, and has better convergence. Therefore, the integrated clustering algorithm provides stronger robustness and stability in different fields and data. Because the K-means algorithm is based on the extreme value of fitness function to optimize the objective function, it is prone to fall into local optimization and low efficiency in processing massive data. In recent years, the improvement of this algorithm is a hot research topic in the field of clustering. A clustering algorithm combining genetic algorithm, PSO algorithm, artificial immune algorithm, ant algorithm and its related improvement algorithm with K-means has emerged.

In 2002, Omran et al. [17] proposed an unguided image classification algorithm based on particle swarm optimization, which is the origin of the PSO clustering algorithm. In terms of the disadvantages of traditional clustering algorithms, PSO optimization can achieve certain results. For example, Gehad Ismail Sayed et al. [18] proposed an algorithm based on hybrid particle swarm optimization and K-means to remove residual stains and interphase cells from metaphase chromosome images, so that they were only concentrated on chromosomes, and the segmentation accuracy reached 95%. Liu et al. [19] proposed a new particle cluster clustering algorithm with good global convergence, which not only effectively overcomes the problem that the traditional K-means algorithm is prone to fall into the local minimum and is sensitive to the initial value, but also has a fast convergence rate. Literature [20] proposed a PSO hybrid K-means clustering algorithm and realized MPI based parallelization of the hybrid clustering algorithm to improve the execution efficiency of the algorithm. Literature [21] adopts the classical particle swarm optimization algorithm to improve the initial clustering center of the K-means algorithm and improve the accuracy of clustering results. Literature [22] studied the K-means clustering algorithm based on the improved particle swarm optimization algorithm, and processed particles trapped in local extreme values to make them jump out of the local optimal solution. Although the algorithm inherited the global search ability of the PSO algorithm, it did not fully and effectively utilize the local search ability of the K-means algorithm.

3 Algorithm Model

3.1 SOM-WPSO Model

This paper studied the traditional PSO-Kmeans clustering algorithm and found that the particle position was composed of the clustering center, the particle swarm size was manually set. And the particle moving position every time was a process of optimizing the clustering center, ignoring the optimization of sample weight, resulting in low efficiency of the algorithm and no obvious improvement of the clustering effect.

Based on PSO-Kmeans, this paper proposes the weight particle cluster clustering algorithm-WPSO. It integrates the self-organizing and adaptive characteristics of SOM, which not only solves the selection of cluster number and initial cluster center but also avoids falling into local optimization, to achieve relatively high accuracy.

SOM algorithm has the function of dimensionality reduction and can effectively deal with the problem of outlier points, without complex differentiation, integration and other operations. However, the SOM algorithm also has disadvantages such as long training time and possible “dead neurons” in competitive learning.

WPSO algorithms still need to set the initial clustering center and cluster number. Traditional clustering cluster number by artificial selection, clustering cluster number is a very thorny problem, and the choice of initial clustering center tend to be selected at random or choice based on the density, distance, even the initial clustering center is likely to be isolated points, boundary point, the clustering algorithm easy to fall into local optimum, even an empty cluster problems.

Comprehensive SOM and WPSO algorithm, SOM first to coarse clustering of data, the data clustering situation in an unsaturated state. And SOM still iteration, the network will learn data distribution. When SOM reaches the number of iterations, the SOM network will return iterative training weights. The weight is based on the data of competitive learning, and then with the original data is analyzed by weight, it will get the winning neuron. Then clustering center and sample weights initialization WPSO parameters, the number of samples is the number of particles, the particle position as sample weight, particle fitness function for each particle and the center of the cluster the particle belongs to class Euclidean distance. Our goal is to minimum value fitness, the fitness value of the minimum mean particle near the clustering center is very close. WPSO after updating the weight and speed using the K-means clustering division again, using the neighbor's thought will mean clustering mapping to the most recent sample points as the clustering center. It can greatly reduce the effects of noise on the algorithm, also reduce the probability of empty cluster produce. The convergence speed of the algorithm is accelerated, and the algorithm flow chart is shown in Fig. 1.

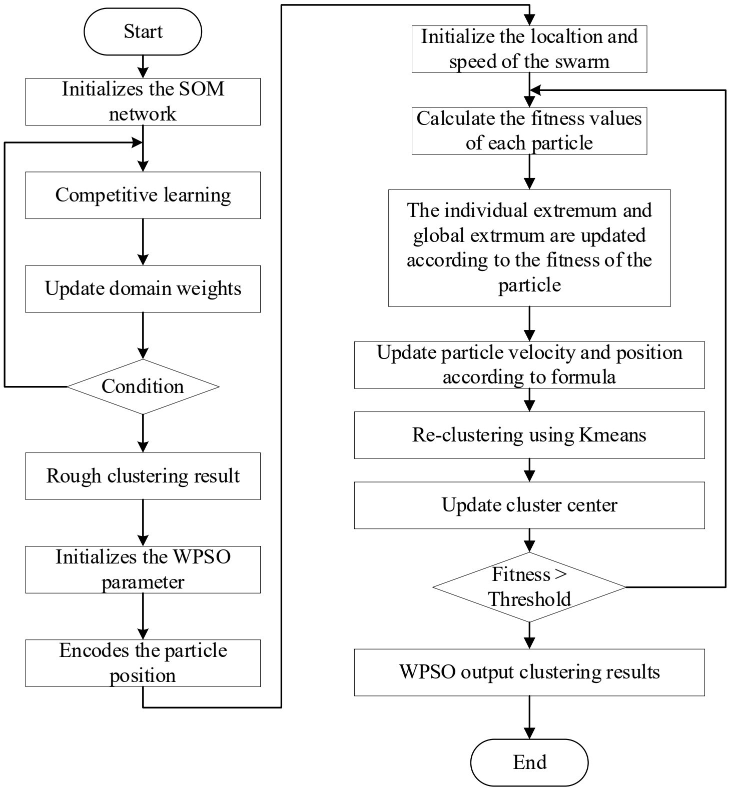
Start
Initializes the SOM network
Competitive learning
Update domain weights
Condition
Rough clustering result
Initializes the WPSO parameter
Encodes the particle position
Initialize the localtion and speed of the swarm
Calculate the fitness values of each particle
The individual extremum and global extrmum are updated according to the fitness of the particle
Update particle velocity and position according to formula
Re-clustering using Kmeans
Update cluster center
Fitness > Threshold
WPSO output clustering results
End

Figure 1: Algorithm flowchart

3.2 Self Organizing Map

SOM (Self-Organizing Map) was obtained by simulating the self-organization mapping of the human cerebral cortex to signals. On the one hand, SOM maps the input pattern of any dimension to a low-dimensional space, which not only reduces the vector dimension but also reduces the computational complexity of iterative training, while maintaining the original topological structure of the sample. On the other hand, the text feature and its neighborhood feature are adjusted by using its self-organizing mapping feature. SOM network structure is shown in Fig. 2.

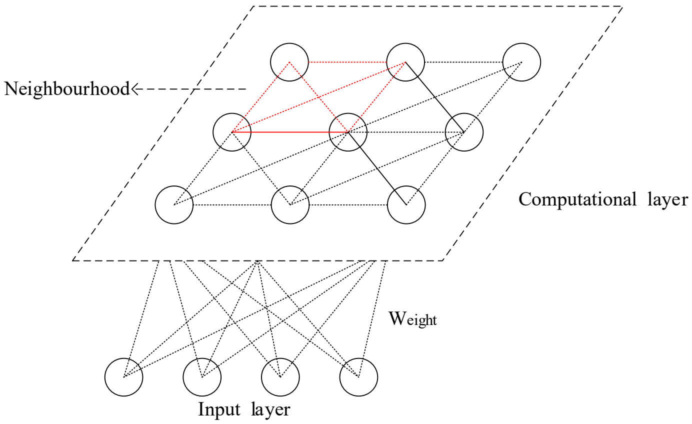

Figure 2: Self-Organizing Map network model

A typical SOM network structure consists of two layers: the input layer and the competition layer. The input layer is mainly responsible for receiving external information. Each neuron of the input layer connects with the neuron of the competition layer for weights and then transmits the external information to the competition layer. The competitive layer is mainly responsible for the analysis of input information, acquiring winning neurons through competitive learning, and inhibiting the excitement of neighboring neurons. The core of the SOM algorithm is competitive learning and neighborhood weight adjustment. The formula is defined as follows:

$$winner = argmax||x_i * w_j|| \tag{1}$$

$$r = C_1\left(1 - \frac{t}{iteration}\right) \tag{2}$$

$$w_{ij}(t+1) = w_{ij}(t) + \sigma(t,N)[x_i - w_{ij}(t)] \tag{3}$$

Formula (1) represents the inner product of text x_i and neuron w_j, and the subscript of the largest inner product is the winning neuron. Formula (2) is the domain radius of the winning neighbor. Formula (3) updates the weights of the winning neuron and the winning neighbor.

3.3 WPSO Algorithm

Traditional PSO clustering algorithm made cluster center as a particle position, calculated the weights of the sample with all the fitness value of particles, and then updated the particle's optimal location and the global optimal position. The number of iterations or fitness threshold algorithm is over, but this way of clustering is strongly dependent on the data pretreatment process. If this is not the same kind of data, the result of clustering is pointless, and the iterative process just moves the clustering center. There is no process of optimizing the weights of the sample, the calculation efficiency is not improved.

Inspired by the PSO clustering algorithm, this paper proposes a new clustering algorithm-WPSO (Weight of Particle Swarm Optimization), using the sample weight instead of particle position. The original clustering center as the particle's traction makes the particles toward the nearest near the particles. And in the iterative process, the velocity of particles is affected by the global optimal particles and individual optimal conditions, and the iteration stops when the optimal fitness value is reached.

The particle of WPSO adopts the encoding format of sample weight, it means the position of each particle is no longer composed of clustering center, but a sample represents a particle. The size of the particle swarm is determined by the number of samples. Besides position, particles also have velocity and fitness values. The particle encoding method is as follows:

$x_1, x_2, x_3, \dots\dots, x_n$	$v_1, v_2, v_3, \dots\dots, v_n$	$fitness(x, c)$

$x_1, x_2, x_3, \ldots\ldots, x_n$ represents the weight of each sample; $v_1, v_2, v_3, \ldots\ldots, v_n$ represents the velocity of the sample, i.e., particle velocity; $fitness(x, c)$ represents the fitness value of the particle from the nearest cluster center. Speed is:

$$v_i(t+1) = w * v_i(t) + c_1 r_1[pbest - x_i(t)] + c_2 r_2[gbest - x_i(t)] \tag{4}$$

Position:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{5}$$

The choice of the fitness function directly affects the convergence speed of the clustering algorithm and whether it can find the optimal solution. It has an overall understanding of clustering and the judgment of the correct rate of clustering results. Introducing the WPSO algorithm into the K-means algorithm, the criterion function for evaluating the clustering quality can be used as the fitness function of the particle swarm. The intra-class tightness MSE is used to indicate the quality of the cluster, the smaller the MSE, the better the clustering effect.

$$MSE = \sum_{j=1}^{k} \sum_{D_i \epsilon C_{j_i}}^{n} d(D_i, C_j) \tag{6}$$

The fitness value of the particle represents the similarity between the data objects in each class. The smaller the fitness value, the closer the degree of binding of the data objects within the class, and the better the clustering effect. The fitness function can be expressed as:

$$Fitness = MSE \tag{7}$$

Although the moving direction of the particles is pulled by the cluster centroids, the completion of each iteration cannot fall to a specific sample point, which increases the difficulty of cluster centroids selection. To reduce the influence of noise points, this paper uses the idea of K-medoids to select the sample points closest to the cluster mean value as the cluster centroids after each K-means clustering is completed, which not only accelerates the convergence speed but also prevents the occurrence of empty clusters.

$$Center(i, t) = argmin\{dis(avg, x)\} \tag{8}$$

This formula represents the distance between the *i*-th cluster average and all samples during the *t*-th iteration, and the nearest sample is the cluster centroid within the cluster.

3.4 K-means Algorithm

In 1967, MacQueen proposed a classical clustering algorithm based on the partition-K-means algorithm, which is simple in calculation and fast in convergence. The algorithm randomly selected K sample points as the initial cluster centroids and then divided other samples into clusters nearest to K samples according to the nearest neighbor principle. After each iteration, the cluster centroids, namely the mean of all the samples in the cluster, was recalculated. The algorithm stops when the nearest cluster of all samples in the data set is not changed.

K-means algorithm steps are as follows:

Input: The data set D containing n data objects and the number of clusters k.

Output: A set of k clusters that satisfy the convergence of the clustering criterion function.

k samples were randomly selected from the data set D as the initial cluster centroids C_j, j = 1, 2, 3, ..., k.

Calculate the distance (x_i,C_j), I = 1, 2, 3, ..., n of each sample of the data set from the k cluster centroids.

Divide the sample into the nearest class according to the nearest neighbor principle, that is, satisfy distance(x_i,C_j)=min{distance(x_i,C_j), j = 1, 2, 3,..., k}, then $x_i \epsilon y_i$.

Recalculate the cluster centroids:

$$C_j = \frac{1}{n} \sum_{x_i \epsilon y_i} x_i \tag{9}$$

Calculate class cohesion:

$$E = \sum_{i=1}^{n} \sum_{j=1}^{k} ||x_i - C_j|| \tag{10}$$

4 Experimental Results and Analysis

4.1 Experimental Data

To verify the effectiveness and feasibility of the algorithm, Iris data set, Wine data set and Glass data set of UCI were used for experiments in this paper, the basic information was shown in Tab. 1 below.

Table 1: UCI data set

Data Set	Features	Class	Samples	Distribution
Iris	4	3	150	50/50/50
Wine	13	3	178	59/71/48
Glass	9	6	214	70/76/17/13/9/29

4.2 Evaluation Criteria

Different clustering algorithms have different application scenarios, so we need a variety of evaluation criteria to analyze the merits of the algorithms. To verify that the accuracy of the proposed algorithm is higher than other algorithms, Purity, fitness value, Davies-Bouldin index, Dunn’s index, and Silhouette coefficient were adopted as the evaluation criteria for clustering results.

4.3 Experimental Parameters

The clustering algorithm is not universally applicable to all data, so it is necessary to select the appropriate algorithm according to the data and set different parameters for different data. The specific parameters are shown in Tab. 2.

Table 2: Parameter settings

SOM Parameters			PSO Parameters				
Dataset	Output	Iteration	W	C1	C2	R1	R2
Iris	1*3	100	0.8	2	2	0.8	0.2
Wine	1*3	100	0.6	2	2	0.7	0.3
Glass	2*3	100	0.8	2	2	0.6	0.3

4.4 Result Analysis

To verify the cluster purity of the algorithm, the algorithm model SOM-WPSO was compared with SOM, PSO/K- means and K-means algorithms in terms of purity respectively, and the data set of each group is repeated 20 times. Take the highest accuracy comparison, as shown in Fig. 3.

It can be seen from Fig. 3 that in the comparison of the algorithm model, SOM worked best on low-dimensional data. On the medium-dimensional data set, the accuracy of the three algorithms is not much different. On the high-dimensional data set, K-means has the lowest accuracy. Combining the advantages of the three algorithm models, this paper improved the PSO algorithm into WPSO and used the PAM idea to make the cluster centroids fall on the specific sample, to avoid the cluster centroids, and found that the accuracy rate was greatly improved.

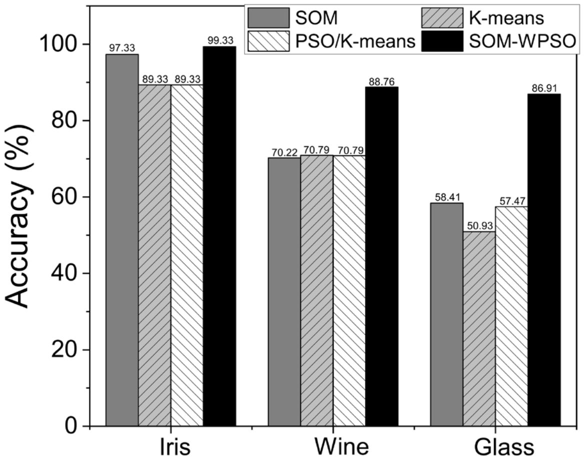

Figure 3: Algorithm purity comparison

Table 3: Model fitness values

Data set	Fitness Value	K-means	pso-km	SOM-WPSO
Iris	highest	123.8498	97.6575	101.9808
	lowest	97.3259	97.2221	97.2725
	average	103.7300	97.3901	100.9184
Wine	highest	18776.94	16960.21	16932.2505
	lowest	16960.20	16940.28	16704.7616
	average	17067.07	16944.78	16770.1669
Glass	highest	215.7317	227.9053	238.7814
	lowest	213.4705	222.3460	233.2041
	average	214.9835	224.3496	235.9792

It can be seen from Tab. 3 that K-means has the lowest accuracy for low-dimensional Iris data set, medium-dimensional Wine data set, and high-dimensional Glass data set. This is because the K-means algorithm is very sensitive to the selection of the initial cluster centroids. The choice of K-means cluster centroids is random and will directly affect the clustering results. By introducing the PSO optimization algorithm into the K-means algorithm, it is found that the pso-km algorithm can eliminate the influence of the cluster centroids on the clustering result to some extent, and determine the cluster centroids by searching the global optimal position of the particle swarm. Based on the idea of pso-km, the algorithm in this paper changed the sample weight to improve the clustering accuracy and found that it was superior to other algorithms on the three data sets.

Tab. 4 shows that on the Iris data set and Wine data set fitness value is relatively small, on the Glass data set is relatively large, the fitness value fluctuations in different dimensions, since both the K-means algorithm and the pso-km algorithm are designed to optimize the cluster centroids, and the algorithm in this paper is to optimize the sample weights. At the same time, in this paper, the cluster centroids are selected by K-means re-clustering, and the idea of K-medoids is used to project the cluster mean to the nearest sample so that the distance of some samples from the cluster centroid becomes far. It also shows that the algorithm of this paper has a better clustering effect on medium- and low-dimensional data.

The experiment will also be compared from the Davies-Bouldin index, Dunn's index and Silhouette coefficient. The smaller the DB, the smaller the distance within the class, and the greater the distance between classes, that is, the smaller the DB, the better the clustering effect. A larger DI means that the distance within the class is smaller, the distance between classes is larger, and the clustering effect is better. The Silhouette coefficient is between [−1,1], and the closer to 1 means that the cohesion and resolution are relatively better.

Tab. 5 and Tab. 6 shows that the proposed algorithm is superior to other clustering algorithms on the three evaluation index, while performed worse on the Glass data set, which is also consistent with the comparison of the previous fitness values. The algorithm in this paper does improve the clustering accuracy of data, the clustering effect is good on low- and medium-dimensional data set, but the evaluation index for high-dimensional data clustering is not very good, and clustering results on high-dimensional data set is not very stable, so the algorithm in this paper is not very suitable for high-dimensional data.

Table 4: Comparison of iris data sets

	K-means	pso-km	SOM-WPSO
DBI	0.8116	0.6937	0.6680
DVI	04347	2.6776	2.8772
SC	0.4876	0.4927	0.4999

Table 5: Comparison of wine data sets

	K-means	pso-km	SOM-WPSO
DBI	0.5479	0.5313	0.5283
DVI	1.9032	1.9370	2.0675
SC	0.5883	0.5921	0.5611

Table 6: Comparison of glass data sets

	K-means	pso-km	SOM-WPSO
DBI	1.0989	0.9246	1.1960
DVI	0.5741	0.7507	0.7270
SC	0.5180	0.6846	0.2564

To verify the convergence degree of the algorithm, this paper compared it with the pso-km algorithm on the Iris data set, as shown in Fig. 4 and Fig. 5.

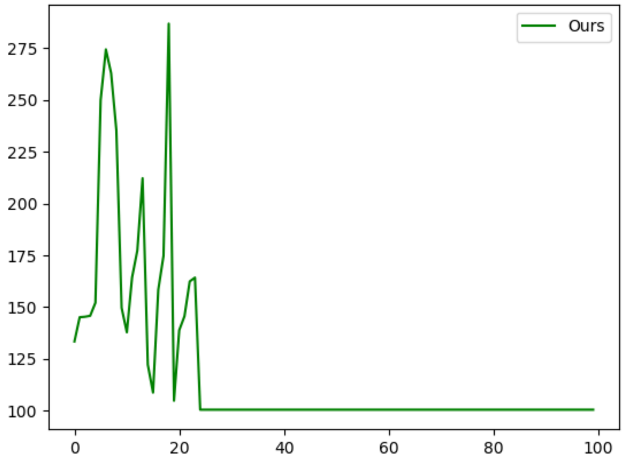

Figure 4: Convergence curve of our algorithm

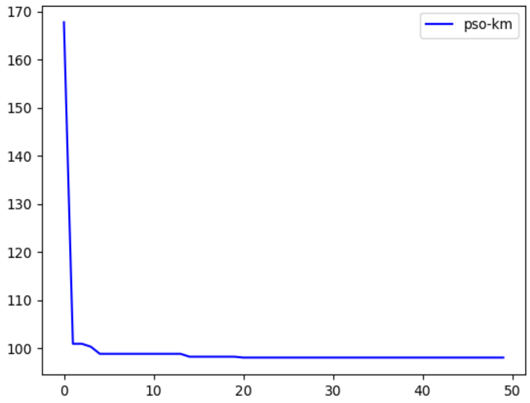

Figure 5: Convergence curve of pso-km

It can be seen from Fig. 4 that the convergence curve of the algorithm fluctuates greatly, and the particles randomly move in the global space until the position with the smallest fitness value is found, and the comparison is within the set threshold range, and if it exceeds the comparison number, it converges. Compared with the pso-km algorithm in Fig. 5, this algorithm converges quickly and does not converge slowly through stepwise iteration, this is due to the characteristics of the algorithm. The goal of this algorithm is to find the best accuracy rate through the fitness value and finally determine the position of the cluster centroids.

5 Summary

Aiming at the sensitivity of the K-means clustering algorithm to the initial cluster centroids, a combined clustering algorithm combining SOM and optimized particle swarm weights is proposed in this paper. The algorithm overcomes the shortcomings of traditional clustering algorithms. Although it is flawed for high-dimensional data, compared with K-means and pso-km algorithms, the accuracy of the clustering algorithm is improved and verifies that the algorithm in this paper is feasible and has a good clustering effect on the medium-dimensional data. Of course, this paper also has some shortcomings. Because the article aims to improve the accuracy of the algorithm and ignores the fitness value of the algorithm and the stability of the convergence curve. In the next work, how to choose a good cluster centroid to reduce the fitness value and let the convergence curve not fluctuate greatly will become the focus of work.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] K. J. Anil, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[2] J. Macqueen, "Some methods for classification and analysis of multi-variate observations," in *Proc. BSMSP*, Berkeley, AC, USA, pp. 621–632,1967.

[3] P. Arora, Deepali and C. Varshney, "Analysis of k-means and k-medoids algorithm for big data," *Procedia Computer Science*, vol. 57, no. 2, pp. 1–8, 2016.

[4] T. Ishioka, "Extended K-means with an efficient estimation of the number of clusters," in *Proc. IDEAL*, Hong Kong, NT, China, pp. 1–8, 2000.

[5] H. S. Park and C. H. Jun, “A simple and fast algorithm for k-medoids clustering,” *Expert Systems with Applications*, vol. 36, no. 2, pp. 3336–3341, 2009.

[6] D. W. V. D. Merwe and A. P. Engelbrecht, “Data clustering using particle swarm optimization,” in *Proc. CEC*, Portland, OR, USA, pp. 1–8, 2004.

[7] T. Ng. Raymond and J. W. Han, “CLARANS: a method for clustering objects for spatial data mining,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 5, pp. 3–16, 2002.

[8] L. Kaufman and P. J. Rousseeuw, “Finding groups in data: An introduction to cluster analysis,” *Journal of the American Statistical Association,* vol. 1, no. 1, pp. 2–10, 1990.

[9] T. Zhang, R. Ramakrishnan and M. Livny, “BIRCH: An efficient data clustering method for very large databases,” in *Proc. ICMD*, New York, NY, USA, pp. 103–114, 1996.

[10] S. Guha, R. Rastogi and K. Shim, “Cure: An efficient clustering algorithm for large databases,” *Information Systems*, vol. 26, no. 1, pp. 35–58, 2001.

[11] S. Guha, R. Rastogi and K. Shim, “ROCK: A robust clustering algorithm for categorical attributes,” *Information Systems*, vol. 25, no. 5, pp. 345–366, 2000.

[12] G. Karypis, “Chameleon: Hierarchical clustering using dynamic modeling,” *IEEE Computer*, vol. 32, no. 1, pp. 1–8, 1999.

[13] M. Ester, H. P. Kriegel and X. Xu, “A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proc. ICKDDM*, New York, NY, USA, pp. 1–10, 1996.

[14] M. Ankerst, M. M. Breunig, H. P. Kriegel and S. Jörg, “OPTICS: Ordering points to identify the clustering structure,” in *Proc. ICMD*, Philadelphia, USA, pp. 1–10, 1999.

[15] S. Rajagopal, T. Mythili, and R. D. Priyanka, “Statistical information grid approach to segment large dataset,” *International Journal for Research and Development in Technology*, vol. 5, no. 1, pp. 46–50, 2016.

[16] R. Agrawal, J. E. Gehrke, D. Gunopulos and P. Raghavan, “Automatic subspace clustering of high dimensional data for data mining applications,” *Data Mining and Knowledge Discovery*, vol. 27, no. 2, pp. 94–105, 1998.

[17] G. O. Mahamed, S. Ayed and P. E. Andries, “Image classification using particle swarm optimization,” in *Proc. APCSEL*, Gui Lin, China, pp. 370–374, 2004.

[18] G. I. Sayed, A. E. Hassanien and M. I. Shaalan, “Particle swarm optimization and k-means algorithm for chromosomes extraction from metaphase images,” in *Proc. ICAISI*, Cairo, Egypt, pp. 1–8, 2018.

[19] J. M. Li, L. C. Huang and L. W. Huang, “A new clustering algorithm—Particle clustering algorithm,” *Computer Engineering and Applications*, vol. 41, no. 20, pp. 183–185, 2005.

[20] Y. Q. Li and J. X. Li, “Parallel pso combined with k-means clustering algorithm based on mpi,” *Journal of Computer Applications*, vol. 31, no. 2, pp. 428–431, 2011.

[21] F. U. Tao, W. J. Sun, D. O. Computer and N. A. University, “Pso-based k-means algorithm and its application in network intrusion detection,” *Computer Science*, vol. 1, no. 1, pp. 1–8, 2013.

[22] X. Q. Cao and J. H. Zhang, “Clustering algorithm based on improved particle swarm optimization algorithm,” *Journal of Computer Research and Development*, vol. 49, no. 1, pp. 1–8, 2012.