

PSO+K-means Algorithm for Anomaly Detection in Big Data

Rasim M. Alguliyev, Ramiz M. Aliguliyev*, Fargana J. Abdullayeva

Institute of Information Technology, Azerbaijan National Academy of Sciences, Azerbaijan

Abstract The use of clustering methods in anomaly detection is considered as an effective approach. The choice of the cluster primary center and the finding of local optimum in the well-known k-means and other classic clustering algorithms are considered as one of the major problems and do not allow to get accurate results in anomaly detection. In this paper to improve the accuracy of anomaly detection based on the combination of PSO (particle swarm optimization) and k-means algorithms, the new weighted clustering method is proposed. The proposed method is tested on Yahoo! S5 dataset and a comparative analysis of the obtained results with the k-means algorithm is performed. The results of experiments show that compared to the k-means algorithm the proposed method is more robust and allows to get more accurate results.

Keywords Anomaly Detection, Big Data, Particle Swarm Optimization, Clustering, k-means.

AMS 2010 subject classifications 62H30, 91C20

DOI: 10.19139/soic.v7i2.623

1. Introduction

The data clustering is an unsupervised classification method, the purpose of which is to divide the dataset into clusters by using the similarity measure between the objects. Cluster analysis has become an important tool for data analysis, pattern recognition, machine learning, image segmentation, neural computations and of the other areas of science [1]. The purpose of the clustering algorithm is to maximize the inter-cluster distance and minimize the intra-cluster distance. The well-known k-means algorithm has been applied to many practical clustering problems [2, 3, 4, 5, 6, 7, 8]. The purpose of this method is to automatically divide the dataset into k groups. k-means algorithm can generate fast and effective results.

Clustering in k-means algorithm is performed by minimizing the average square distance between each point and the center closest to that point. Although the k-means is a well-known algorithm, this algorithm has many drawbacks. The k-means algorithm depends on initial cluster center selection and requires determination of cluster center previously.

Since the objective function of the k-means algorithm is not convex, this algorithm has many local minimum points. To overcome these type problems of the k-means algorithm evolutionary algorithms are widely used [9, 10]. PSO is one of the evolutionary algorithms and based on the thinking and behavior of the swarm. Literature analysis shows that PSO based clustering allows getting higher results compared to existing clustering methods [11].

Anomaly detection is one of the major fields of research and researchers are trying to find new algorithms for detecting anomalies. Clustering techniques of data mining is an interested area of research for detecting possible intrusions and attacks. This paper presents a new clustering approach for anomaly detection by using the hybridization of k-means and PSO methods. Numerous clustering methods have been proposed to detect anomalies based on hybridization of PSO and k-means algorithms [12, 13, 14, 15, 16, 17]. Differences between existing

*Correspondence to: Ramiz M. Aliguliyev (Email: r.aliguliyev@gmail.com). Institute of Information Technology, Azerbaijan National Academy of Sciences, 9A, B. Vahabzade Street, Az1141, Baku, Azerbaijan.

clustering algorithms proposed on the basis of PSO are in their objective functions. In [9], a data clustering method by applying the PSO algorithm is proposed. From the experiments carried out here, it is clear that due to the lack of weight values of the criteria there are many instances identified incorrectly. In addition, as the weight of criteria to be minimized in data clustering is taken as equal in current algorithms, it is impossible to regulate the higher optimality of the objective function. Therefore, to demonstrate the importance of the criteria in the data clustering process it is necessary to assign the appropriate weight coefficients to these criteria [18]. Here, to indicate the importance of the criteria in the clustering process the weight coefficients are used. Assigning the weights to the criteria allows getting a better optimal solution.

In this paper, to eliminate the above-mentioned clustering problems (clustering accuracy, the presence of many local minimum points, predefined cluster centers) a multi-criterion optimization method based on the weighted PSO algorithm is proposed. In this method minimizing the intra-cluster distance and maximization, the inter-cluster distance has been selected as optimization criteria.

The main contributions of this work are:

- The weighted detection method of anomalies occurred in the cloud environment is proposed. Here the optimization is achieved through the minimization of the two criteria. In this method, the sum of the minimum value of intra-cluster distance and the maximum value of the inter-cluster distance is used as an objective function.
- Based on the proposed method, the PSO algorithm for data clustering is constructed.
- The capabilities of the proposed method are evaluated in the Matlab software package.
- The ability to clustering data of any size of the standard PSO algorithm is demonstrated.
- A new clustering algorithm based on the PSO algorithm is proposed. Here, to form the initial population the k-means algorithm is used.

2. Related works

Hybridization of algorithms for improving the clustering performance is not a novel idea. The contribution of our proposed method is using a PSO based swarm intelligence algorithm, with k-means in order to optimize clustering results based on two contemporary criteria: well-separated clusters by low intra-cluster and high inter-cluster distances. Besides these, another aim of this paper is achievement the local optimization by validity indexes. We apply a new fitness function for PSO algorithm to select the best solution. Then, we propose an anomaly detection method by the combination of PSO and traditional k-means algorithms. We design this hybrid system over cloud computing environment to find the optimal number of clusters with high separation from neighbor clusters and low compactness of local data points, increase detection rate, and decrease the false positive rate at the same time.

PSO with k-means with various parameters is applied in different domains. Comparison of these methods is summarized in Table 1.

3. Optimization method based on swarm intelligence

The particle swarm optimization (PSO) algorithm is proposed by Eberhart and Kennedy in 1995, is a population-based stochastic search process algorithm and is modeled on the basis of the social behavior of birds [20, 21]. The population of particles is the basis of the algorithm. Each of these particles is the possible solution to the optimization problem. In the PSO algorithm, the swarm generates several possible solutions to the optimization problem. Each of these possible solutions is called particles. The purpose of the PSO algorithm is to find the best solution, satisfying to the objective function.

Each particle is the one position in a N dimensional space and moves by regulating its position in the multidimensional search field according to the following:

- The best position of the particle;
- The best position of the particle's neighbors.
- Each i th particle consists of:

Table 1. Hybridization PSO with k-means for clustering

Method	Dataset	Criteria	Parameters value	Cost function	Contribution
[19]	A commercial website log file with 726 clients and 52 pages which clustered separately to 15, 25 and 35 classes	Intra-cluster distance	Iterations number is 50	$F(Z_i) = \sum_{j=1}^m \sum_{\forall x_n} d(x_n, z_{ij})$, x_n is the data point, and z_{ij} refers to the j th cluster centroid of the i th particle, and d is the position of the particles. The function depicts the sum of all the intra-cluster distance, in which lower is better	Incorporating the multidimensional a synchronism and stochastic disturbance method to the velocity in the PSO, which keep populations diversity and ability of search global optimum. This method is called MSPSO-K.
[12]	City coordinates of Hopfield-10 TSP and Iris plants database	$max(d^-(Y_i)) = \max_{j=1,2,\dots,k} \left(\sum_{\forall x_i \in y_j} \frac{d(x_i, y_j)}{ y_j } \right)$ - the maximum value of the mean of distances within same classes; $\min(d_2(Y_1)) = \min(d(y_i, y_j)), \forall i, j, i \neq j$ - is the minimum value of distances between classes	Acceleration constant is $c_1 = c_2 = 1.3$, Inertia weight is $\omega = 0.3$	$f(Y_i) = \frac{max(d_1^-(Y_i))}{\min(d_2(Y_i))}$	A combination of the core idea of k-means with PSO, which leads to the clustering algorithm with low error rate as compared to k-means
[13]	KDDCup 1999 dataset	$J_K = \sum_{j=1}^K \sum_{X_i \in C_j} d(X_i, Z_j)$	Inertia weight w decreases linearly	$f(S_j) = \frac{1}{1+J_K}$ minimizing an objective function distance measure (Euclidean distance is chosen in this paper) between a data point X_i and the cluster center Z_j .	The anomaly intrusion detection system based on the combination of PSO (for initializing K cluster centroids) and K-means (for local search ability to stable the centroids). The results show a false positive rate of 2.8% and the detection rate of 86%
Proposed PSO & k-means	Yahoo! S5	$J_1 = (1 - \alpha) \times \sum_{i=1}^n \sum_{j=1}^K \ x_i - c_j\ $ - intra-cluster distance. $J_2 = \alpha \times \sum_{k,j=1}^K \ c_k - c_j\ $ - inter-cluster distance.	Inertia weight is $\omega = 0.7298$, acceleration constant is $\gamma_1 = \gamma_2 = 1.4962$	$f = (1 - \alpha) \times \sum_{i=1}^n \sum_{j=1}^K \ x_i - c_j\ + \alpha \times \sum_{k,j=1}^K \ c_k - c_j\ \rightarrow \min$	A hybrid PSO-k-means anomaly detection algorithm optimizes clustering results based on two contemporary criteria: well-separated clusters by low intra-cluster and high inter-cluster distances and achievement the local optimization by validity indexes.

- x_i - the current position of the i th particle;
- v_i - the current velocity of the i th particle;
- y_i - the personal best position of the i th particle;
- g - the best position of the swarm.

The position of the particle is regulated according to the following parameters:

$$v_{il}(t + 1) = \omega * v_{il}(t) + \gamma_1 * r_1(t) * (y_{il}(t) - x_{il}(t)) + \gamma_2 * r_2(t) * (g_l(t) - x_{il}(t)) \tag{1}$$

$$x_{il}(t + 1) = x_{il}(t) + v_{il}(t + 1) \tag{2}$$

where ω is the inertia weight ($\omega = 0.7298$), γ_1 (cognitive coefficient – weight of local information) and γ_2 (social coefficient – weight of global information) are acceleration constants, $r_1(t), r_2(t) \sim U(0, 1)$ are uniformly distributed random variables over an interval $(0, 1), l = 1, \dots, N, i = 1, \dots, P, P$ is the swarm size.

The personal best position of the i -th particle is calculated as follows:

$$y_i(t + 1) = \begin{cases} y_i(t) & f(x_i(t + 1)) \geq f(y_i(t)) \\ x_i(t + 1) & f(x_i(t + 1)) < f(y_i(t)) \end{cases} \tag{3}$$

4. Statement of the clustering problem

In the R^n space, the clustering problem may describe as follows: a given point set including n points $\{x_1, x_2, \dots, x_n\}$, divide these points into K (known constant) sets G_1, G_2, \dots, G_K according to the similarity of them.

Here, during partitioning the following conditions must be satisfied:

$$G_i \neq \emptyset, i = 1, 2, \dots, K; \tag{4}$$

$$G_i \cap G_j = \emptyset, i, j = 1, 2, \dots, K; i \neq j \tag{5}$$

$$\bigcup_{i=1}^k G_i = \{x_1, x_2, \dots, x_n\} \tag{6}$$

Clustering is a process that groups objects into multiple clusters by allowing similar objects to be assigned to the same cluster. K-means is the most popular and widely used clustering algorithm. The k-means algorithm tries to find the c_1, c_2, \dots, c_k cluster centers where the sum of the square distance to the nearest cluster center is minimized:

$$D = \sum_{i=1}^n \left[\min_{k=1,2,\dots,K} d(x_i, c_k) \right]^2 \tag{7}$$

where $d(x_i, c_k)$ is a distance between x_i and c_k . In this paper Euclidean distance is used.

The standard k-means algorithm is summarized as follows:

Step 1. Determination of k clusters and initialization of the $(c_1^{(0)}, c_2^{(0)}, \dots, c_k^{(0)})$ centroids for each cluster. Each cluster center is a m dimensional vector, for example $c_i^{(0)} = \{c_{i1}^{(0)}, c_{i2}^{(0)}, \dots, c_{im}^{(0)}\}$.

Step 2. Calculate the $d_{ki}^{(t-1)}$ distance between i th data set (point in m dimensional space) and k th cluster center. The Euclidean distance is calculated as follows:

$$d_{ki}^{(t-1)} = \|x_i - c_k^{(t-1)}\| = \sqrt{\sum_{j=1}^m (x_{ij} - c_{kj}^{(t-1)})^2} \tag{8}$$

Step 3. Assign each data object to the nearest cluster center.

Step 4. Update of each $c_k^{(t)}$ cluster center by Eq.(9). This equation calculates the average value of all points assigned to that cluster center:

$$c_k^{(t)} = \frac{\sum_{x_i \in k} x_i}{n_k} \tag{9}$$

where n_k is the number of points assigned to k th cluster.

Step 5. Calculation of intra-cluster distance D by Eq. (7).

Step 6. Step 6. If the value of D is satisfied, choosing the final cluster centers. Otherwise, go to $t = t + 1$ iteration and return to step 2.

5. Proposed optimization function for PSO clustering

In PSO clustering algorithm, each $Y_i = (y_1, y_2, \dots, y_k)$ particle represents the centers of the k classes. Here, the particle swarm consists of the plans classified as the several candidate solutions. In the optimization algorithm uses the fitness function of selecting a contingent paying plan from the classified plans as the candidate. For the selection among the classified plans of the plan satisfying to the stated condition, the optimization algorithm is used fitness function. For this purpose, in this paper the following fitness function is proposed:

$$f = (1 - a) \times \sum_{i=1}^n \sum_{j=1}^K \|x_i - c_j\| + a \times \sum_{k,j=1}^K \|c_k - c_j\| \rightarrow \min \tag{10}$$

Here, the goal is to minimize the value of the cost function given by the Eq. (10). That is, it is assumed that clusterization will be carried out more efficiently at the minimum value of this function. $(1 - a)$ and a are the weight ratios of the J_1 and J_2 criteria, respectively, and represent the impact of J_1 and J_2 criteria to the evaluation. As a result of a series experiments, the clustering result in the $a = 0.731$ value of weight ratio was relatively stable and better. For this reason, in this paper, the value of weight ratio is taken as $a = 0.731$.

The minimum value of the function f satisfies the condition that the distance between the points in the same class is small and the distance between the classes is large. The classification plan, which has the lowest f function value, is considered to be the best.

The two criteria that form the evaluation fitness are:

a) *Intra-cluster distance* - the distance between data vectors and their corresponding cluster center within a cluster, where the objective is to minimize the intra-cluster distance. This criterion is calculated by Eq. (11):

$$J_1 = (1 - a) \times \sum_{i=1}^n \sum_{j=1}^K \|x_i - c_j\| \quad (11)$$

where c_j is the j th cluster center, x_i - denotes data points belonging to C_j .

b) *Inter-cluster distance* - the distance between all cluster centers, where the objective is to maximize the distance between clusters. This criterion is calculated by Eq. (12):

$$J_2 = a \times \sum_{k,j=1}^K \|c_k - c_j\| \quad (12)$$

where c_k, c_j are the k th and j th cluster centers, respectively. K is the number of clusters.

6. Proposed optimization model for anomaly detection

Figure 1 shows the proposed optimization model for anomaly detection. As shown in the figure, the model is based on the idea of combining PSO and traditional k-means algorithms. Here the class labels cannot be used directly in the clustering but used in fitness function of PSO to improve the performance of the traditional k-means algorithm.

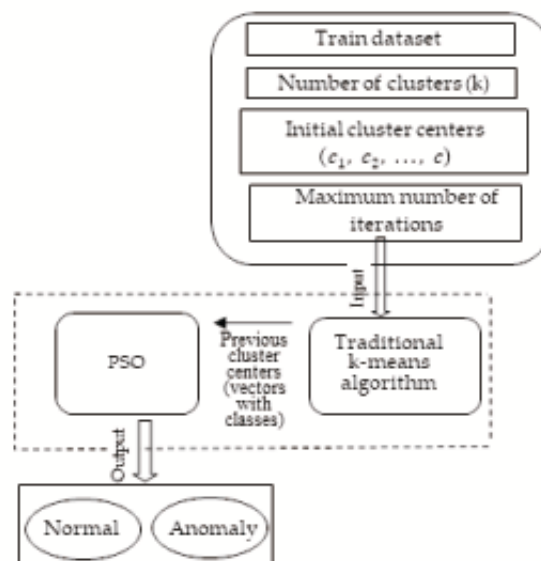


Figure 1. Optimization model for anomaly detection

According to this structure the algorithm of PSO based optimization problem for anomaly detection consists of the following steps:

Step 1. Initialize a population size m , γ_1 and γ_2 learning factors, the number of iterations, a population of particles with small random positions x_p , inertia weight (ω), the number of clusters k , dataset consisting of n points. The position of each particle (x_i) corresponds to the cluster centers of size $k \times m$ and velocity represents the rate of change in particle's position.

Step 2. Run the following steps in the k-means algorithm for every particle in the population:

1. Calculate Euclidean distance measure d_{pi} , between p th cluster center (particle) and i th data point using Eq.(8).
2. Assign each data object x_i to the nearest cluster center.

Step 3. After grouping the data objects based on minimum distance criterion, evaluate the fitness function which is the maximization of classification accuracy as given by Eq. (10).

Step 4. Compare evaluation with the particle's previous best value P_{best} , in terms of fitness value. If the current position (cluster center location) is better than P_{best} , then assign the current position to P_{best} , else retain P_{best} at its old value. This process is carried out for each particle in the population.

Step 5. After updating P_{best} , choose the best fit value (having maximum fitness value) among the particles in P_{best} and assign it as G_{best} . The G_{best} is a single particle of dimension $k \times m$. k being number of clusters identified in partitioning the dataset.

Step 6. The velocity and position of each particle are updated using Eqs. (1) and (2) respectively.

Step 7. Check the convergence criterion, which may be a good fitness value or a maximum number of iterations. If converged, return the G_{best} as the optimal cluster centers, else increment the iteration count $t = t + 1$ and loop to Step 2.

7. Metrics for performance evaluation

Assume that the dataset D is composed of the classes $C^+ = (C_1^+, \dots, C_{K^+}^+)$ (true clustering), and to find clusters $C = (C_1, \dots, C_K)$ in this dataset application of clustering technique is required. There are various indices to compare the two partitions $C = (C_1, \dots, C_K)$ and $C^+ = (C_1^+, \dots, C_{K^+}^+)$ [22].

Purity. The purity of the cluster C_p is defined as follows:

$$purity(C_p) = \frac{1}{|C_p|} \max_{p^+=1, \dots, K^+} |C_p \cap C_{p^+}^+|, \quad p = 1, 2, \dots, K. \tag{13}$$

The value of the purity is always in the interval $[\frac{1}{K^+}, 1]$. A large purity value implies that the cluster is a "pure" subset of the dominant class. The purity of the entire collection of clusters was evaluated as a weighted sum of the individual cluster purities:

$$purity(C) = \sum_{p=1}^K \frac{|C_p|}{n} purity(C_p) = \frac{1}{n} \sum_{p^+=1, \dots, K^+} \max_{p^+=1, \dots, K^+} |C_p \cap C_{p^+}^+|. \tag{14}$$

According to this measure, a higher purity value indicates a better clustering solution.

Entropy. An entropy measure based on information-theoretic considerations can be also used. The entropy of the cluster C_p is defined to be

$$E(C_p) = -\frac{1}{\log(K^+)} \sum_{p^+=1}^{K^+} \frac{|C_p \cap C_{p^+}^+|}{|C_p|} \log \left(\frac{|C_p \cap C_{p^+}^+|}{|C_p|} \right), \quad p = 1, 2, \dots, K. \tag{15}$$

Note that when x is close to 0, then $x \log x$ is close to 0. So it is considered to be $0 \log 0 = 0$.

Since the entropy considers the distribution of semantic classes in a cluster, it is a more comprehensive measure than the purity. Note that to take values between 0 and 1 the entropy is normalized. If almost all the points of cluster C_p belong to the same class C_{p^+} , then $\frac{|C_p \cap C_{p^+}|}{|C_p|}$ is close to 1 for $p = p^+$ and is close to 0 for $p \neq p^+$. So the entropy $E(C_p)$ is close to 0 (since $0 \log 0 = 0$ and $1 \log 1 = 0$). On the other hand, if the points of cluster C_p are randomly divided among all the classes of C^+ , then $\frac{|C_p \cap C_{p^+}|}{|C_p|}$ is close to $\frac{1}{K^+}$ and the entropy of cluster C_p is close to 1.

In contrast to the purity measure, an entropy value of 0 means that the cluster is comprised entirely of one class, while an entropy value near 1 implies that the cluster contains a uniform mixture of all classes.

The global clustering entropy of the entire collection is defined to be the sum of the individual cluster entropies weighted according to the cluster size. That is,

$$E(C) = \sum_{p=1}^K \frac{|C_p|}{n} E(C_p). \quad (16)$$

The global entropy also takes values between 0 and 1. A perfect clustering solution will be one that produces clusters that contain documents from only a single class, in which case entropy will be zero. In general, the smaller the entropy, the better the quality of the cluster.

Dunn Index. Dunn Index one of the most cited indices is proposed by [23]. The Dunn index identifies clusters which are well separated and compact. The goal is therefore, to maximize the inter-cluster distance while minimizing the intra-cluster distance. The Dunn index for k clusters is defined by Eq. (17) [24]:

$$DU_K = \min_{i=1, \dots, K} \left\{ \min_{j=1+1, \dots, K} \left(\frac{\text{diss}(c_i, c_j)}{\max_{m=1, \dots, k} \text{diam}(c_m)} \right) \right\} \quad (17)$$

where $\text{diss}(c_i, c_j) = \min_{x \in c_i, y \in c_j} \|x - y\|$ is the dissimilarity between clusters c_i and c_j and $\text{diam}(c) = \max_{x, y \in c} \|x - y\|$ is the intra-cluster function (or diameter) of the cluster. If the Dunn index is large, it means that compact and well-separated clusters exist.

Silhouette index. The Silhouette index (SI) is another well known way of estimating the number of groups in a dataset. The SI computes for each point a width depending on its membership in any cluster. This silhouette width is then an average over all observations. This leads to Eq. (18):

$$SI_k = \frac{1}{n} \sum_{i=1}^n \frac{(b_i - a_i)}{\max(a_i, b_i)} \quad (18)$$

where n is the total number of points, a_i is the average distance between point i and all other points in its own cluster and b_i is the minimum of the average dissimilarities between i and points in other clusters. The partition with the highest SI is taken to be optimal.

8. Experiments

In this section, for evaluating the effectiveness, comparative analysis of the clustering results of the proposed method and the k-means algorithm on real Yahoo! S5 dataset is carried out. For the PSO based clustering algorithm, the acceleration constants are taken as $\gamma_1 = \gamma_2 = 1.4962$.

In the test process, the size of the population was taken 50 and the value of the fitness function calculated by the Eq. (10).

The *real_2.csv* file of the *A1Benchmark Yahoo! S5* dataset contains 1440 rows. The feature vector of this dataset contains two parameters such as timestamp and value, there is also the dataset marked as normal or anomaly. In other words, the dataset is a classified dataset. The real state of the A1Benchmark dataset (without clustering) is as shown in Figure 2.

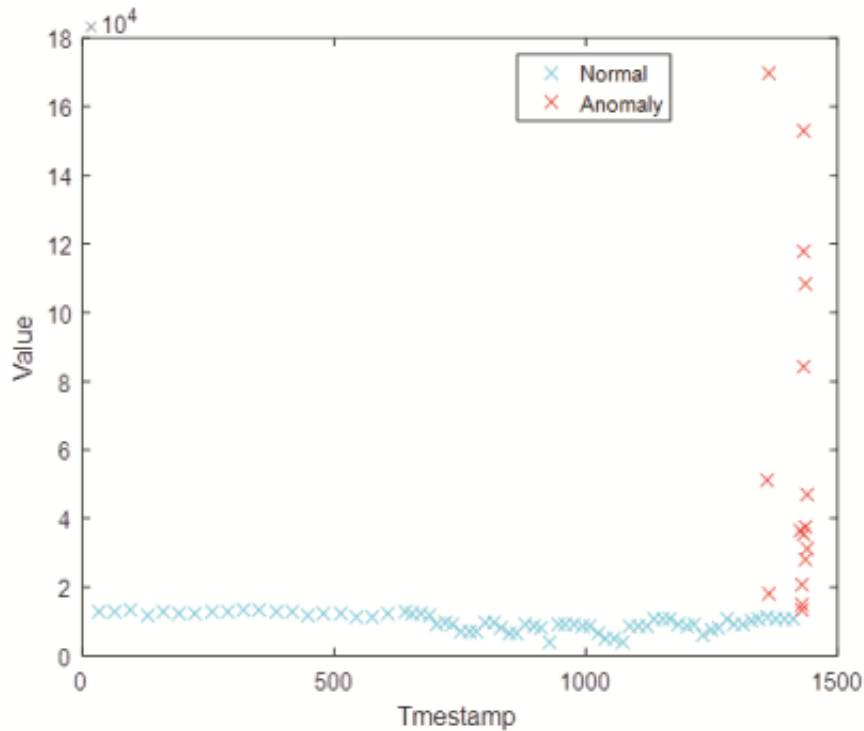


Figure 2. The real state of the A1Benchmark dataset

For the experiments from the Yahoo! S5 dataset was taken overall 84 points, 68 of them which were normal (blue dots) and 16 are anomalous points.

Figure 3 illustrates point anomalies, which are anomalies in a single value in the time series of A1Benchmark dataset.

Based on the results of the k-means algorithm on this dataset 79 points identified as normal 5 points as anomalous (Figure 4). Here, the 11 points have been identified incorrectly.

In the proposed PSO algorithm, 72 points from the total data were identified as normal 12 points as the anomaly (Figure 5). In this algorithm, 4 points have been identified incorrectly. In the figure, the black circles show the cluster centers.

The effectiveness of the clustering was evaluated over four metrics: Dunn’s index, Silhouette index, Purity index, Entropy Index (Table 2).

Table 2. Clustering results of the PSO algorithm on $a = 0.731$

	Total	Normal	Anomaly	Wrong normal	Wrong anomaly	Dunn’s index	Silhouette index	Purity index	Entropy index
K-means	84	79	5	11	-	0.05	0.39	0.87	0.58
PSO	84	72	12	4	-	0.38	0.87	0.95	0.31
Real	84	68	16	-	-	-	-	-	-

Here, according to the clustering performance evaluation metrics, the Dunn Index for the k-means algorithm is 0.0510, and for the PSO algorithm 0,3847. Note that, although the Dunn Index is large, the algorithm is considered to be the most effective. According to the Silhouette index, the k-means algorithm has got 0.3899,

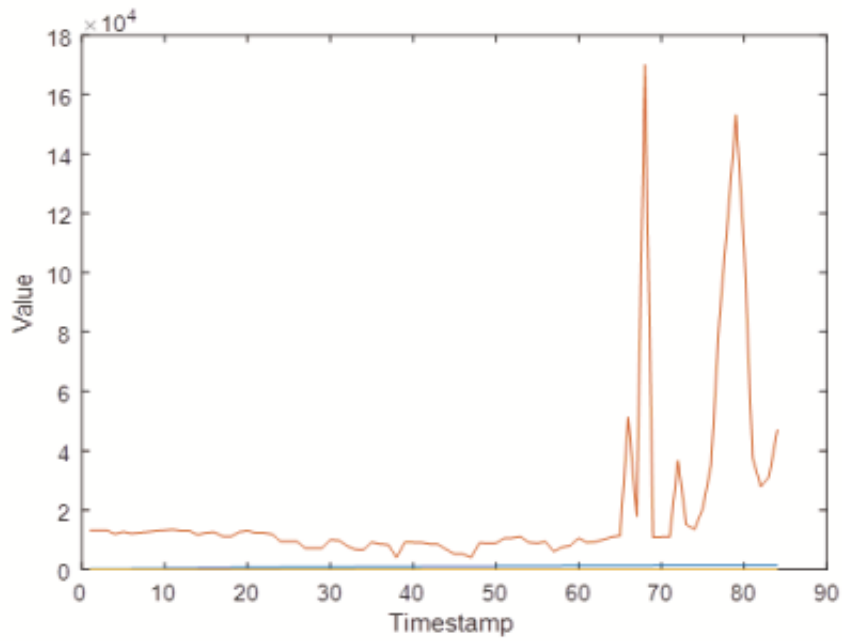


Figure 3. Point anomalies in a single value in the time series of A1Benchmark dataset

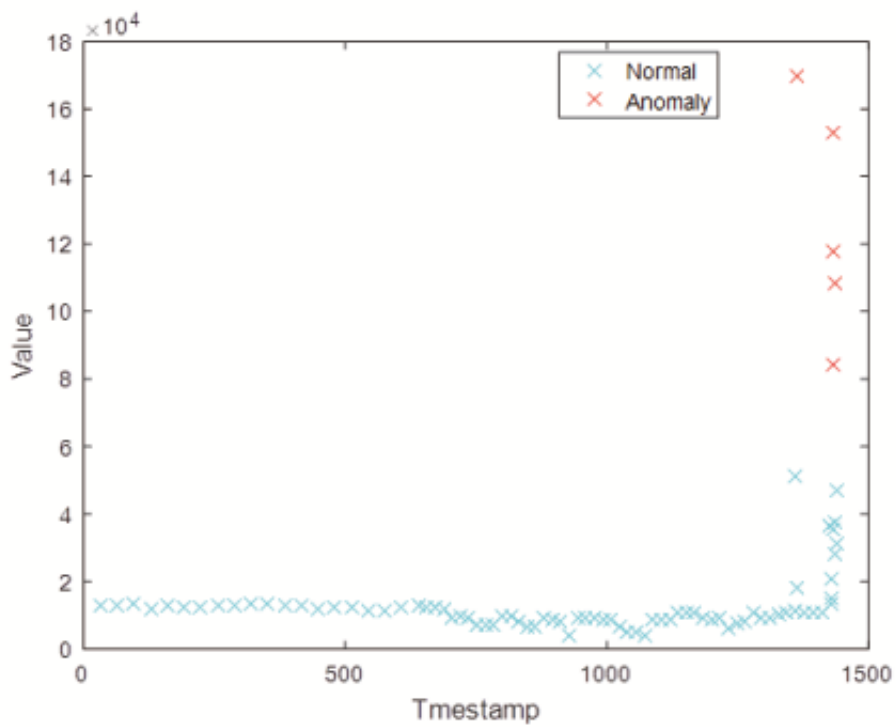


Figure 4. The result of the k-means based clustering

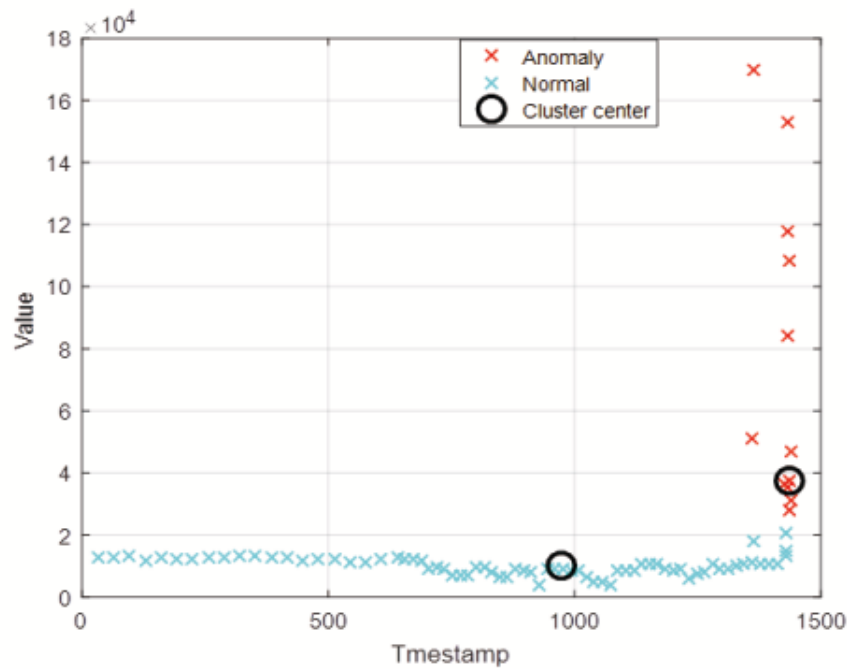


Figure 5. The result of PSO based clustering

the PSO algorithm 0.8722. Good results have been also obtained from the proposed method over other metrics. Thus, according to the Purity index, the k-means algorithm has got 0.8690 and the PSO algorithm 0.9524. But when calculating entropy, the entropy of the k-means algorithm was 0.5821 and the entropy of the PSO algorithm was 0.3096. Low entropy in clustering shows that the method is more effective.

The iteration number of the proposed PSO algorithm is taken 200, and the obtained results improved gradually and the optimal solution (BestCost) has been found (Figure 6).

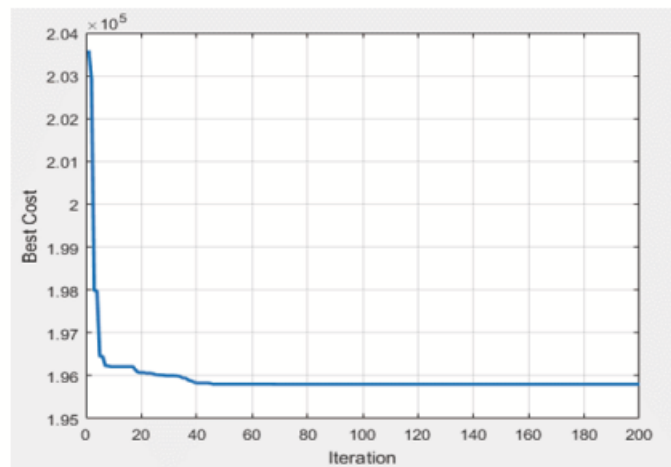


Figure 6. Optimal solution dynamics based on PSO

9. Conclusion

Investigations showed that PSO algorithm, PSO modifications, and hybridisation it with various algorithms provide excellent results in terms of effectiveness and accuracy in solving the optimization problem. Application of this algorithm to data clustering allows for the accurate creation of clusters and accurate forecasting and clustering of data.

In this paper, PSO based multi-criterion optimization method for data clustering was proposed. Comparative analysis of the proposed method with the k-means algorithm on Yahoo! S5 dataset was conducted. The results of experiments showed that the clustering method based on PSO algorithm is better than the k-means algorithm.

Acknowledgement

This work was supported by the Science Development Foundation under the President of the Republic of Azerbaijan-Grant No. EIF-KETPL-2-2015-1(25)-56/05/1.

REFERENCES

1. E. Macedo, *Two-step semidefinite programming approach to clustering and dimensionality reduction*, Statistics, Optimization and Information Computing, vol.3, no. 3, pp. 294–311, 2015.
2. N. Karmitsa, A.M. Bagirov, and S. Taheri, *Clustering in large data sets with the limited memory bundle method*, Pattern Recognition, vol. 83, pp. 245-259, 2018.
3. N. Karmitsa, A.M. Bagirov, and S. Taheri, *New diagonal bundle method for clustering problems in large data sets*, European Journal of Operational Research, vol. 263, no. 2, pp. 367-379, 2017.
4. C. Atilgan and E. Nasibov, *A space efficient minimum spanning tree approach to the fuzzy joint points clustering algorithm*, IEEE Transactions on Fuzzy Systems, 2018. DOI: 10.1109/TFUZZ.2018.2879465.
5. E.N. Nasibov and C. Atilgan, *A note on fuzzy joint points clustering methods for large datasets*, IEEE Transactions on Fuzzy Systems, vol. 24, no. 6, pp. 1648-1653, 2016.
6. R.M. Alguliyev, R.M. Aliguliyev, A.M. Bagirov, and R.R. Karimov, *Batch clustering algorithm for big data sets*, Proceedings of the 2016 IEEE 10th International Conference on Application of Information and Communication Technologies, pp.79-82, 2016.
7. R.M. Alguliyev, R.M. Aliguliyev, Y.N. Imamverdiyev, and L.V. Sukhostat, *Weighted clustering for anomaly detection in big data*, Statistics, Optimization and Information Computing, vol. 6, no. 2, pp. 178-188, 2018.
8. R.M. Alguliyev, R.M. Aliguliyev, and L.V. Sukhostat, *Anomaly detection in big data based on clustering*, Statistics, Optimization and Information Computing, vol. 5, no. 4, pp 325-340, 2017.
9. S. Rana, S. Jasola, and R. Kumar, *A boundary restricted adaptive particle swarm optimization for data clustering*, International Journal of Machine Learning and Cybernetics, vol. 4, no. 4, pp. 391-400, 2013.
10. A.A. Esmin, R.A. Coelho, and S. Matwin, *A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data*, Artificial Intelligence Review, vol. 44, no. 1, pp 23-45, 2015.
11. R.J. Kuo, M.J. Wang, and T.W. Huang *An application of particle swarm optimization algorithm to clustering analysis*, Soft Computing, vol. 15, no. 3, pp. 533-542, 2011.
12. P. Zhenkui, H. Xia, and H. Jinfeng, *The clustering algorithm based on particle swarm optimization algorithm*, Proceedings of the International Conference on Intelligent Computation Technology and Automation, pp. 148-151, 2008.
13. L. Xiao, Z. Shao, G. Liu, *k-means algorithm based on particle swarm optimization algorithm for anomaly intrusion detection*, Proceedings of the Sixth World Congress on Intelligent Control and Automation, pp. 5854-5854, 2006.
14. C. Koliass, G. Kambourakis, and M. Maragoudakis, *Swarm intelligence in intrusion detection: A survey*, Computers and Security, vol. 30, no. 8, pp. 625-642, 2011.
15. Z. Li, Y. Li, and L. Xu, *Anomaly intrusion detection method based on k-means clustering algorithm with particle swarm optimization*, Proceedings of the International Conference on Information Technology, Computer Engineering and Management Sciences, pp. 157-161, 2011.
16. S.H. Li, Y.C. Kao, Z.C. Zhang, Y.P. Chuang, and D.C. Yen, *A network behavior-based botnet detection mechanism using PSO and k-means*, ACM Transactions on Management Information Systems, vol. 6, no. 1, pp. 3-30, 2015.
17. A. Karami and M. Guerrero-Zapata, *A fuzzy anomaly detection system based on hybrid PSO-Kmeans algorithm in content-centric networks* Neurocomputing, vol 149, pp. 1253-1269, 2015.
18. R.M. Alguliyev, Y.N. Imamverdiyev, and F.C. Abdullayeva, *Multicriteria optimization method for load balancing in cloud computing*, Problems of Information Technology, no. 2, pp. 3-15, 2017.
19. J. Chen, *Hybrid clustering algorithm based on PSO with the multidimensional asynchronism and stochastic disturbance method*, Journal of Theoretical and Applied Information Technology, vol. 46, no.1, pp. 434-440, 2012.
20. J. Kennedy and R. Eberhart, *Particle Swarm Optimization*, Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942-1948, 1995.
21. R. Eberhart, Y. Shi, and J. Kennedy, *Swarm Intelligence*, (1st edition), 512 p., 2002.

22. R.M. Aliguliyev, *Performance evaluation of density-based clustering methods*, Information Sciences, vol. 179, no. 20, pp. 3583-3602, 2009.
23. J.C. Dunn, *Well separated clusters and optimal fuzzy partitions*, Journal of Cybernetics, vol. 4, no. 1, pp. 95-104, 1974.
24. S. Saitta, B. Raphael, and F.C. Smith, *A bounded index for cluster validity*, Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition, pp. 174-185, 2007.