Automatic Clustering Using FSDE-Forced Strategy Differential Evolution

To cite this article: A Yasid 2018 J. Phys.: Conf. Ser. 953 012127

View the article online for updates and enhancements.
Automatic Clustering Using FSDE-Forced Strategy Differential Evolution

AYasid
Department of Information System, University of Trunojoyo Madura, Jl. Raya Telang,, Bangkalan, 69162 Indonesia
Email : ayasid@trunojoyo.ac.id

Abstract. Clustering analysis is important in datamining for unsupervised data, cause no adequate prior knowledge. One of the important tasks is defining the number of clusters without user involvement that is known as automatic clustering. This study intends on acquiring cluster number automatically utilizing forced strategy differential evolution (AC-FSDE). Two mutation parameters, namely: constant parameter and variable parameter are employed to boost differential evolution performance. Four well-known benchmark datasets were used to evaluate the algorithm. Moreover, the result is compared with other state of the art automatic clustering methods. The experiment results evidence that AC-FSDE is better or competitive with other existing automatic clustering algorithm.

1. Introduction
With the rapid development of technology such as the internet, every time huge amounts of data are produced and stored in the database. This data should be processed in order to get useful meaning. One of the important methods for obtaining useful information is data mining. The applications of data mining can get the relationships, data structures and hidden information within data to gain knowledge that can be used in decision making or other needed. Basically, data mining consist of two problems namely supervised learning and unsupervised learning [1]. In the supervised learning, like classification and prediction, the training data and target variables have been defined whereas unsupervised learning approaches there is no data training and target variables. Data clustering is unsupervised techniques that aim to group data or objects, whereas data in one group closely similar to each other and data in another group closely dissimilar to the data in other groups. Data clustering applications have been widely used in the real world such as web mining, market analysis, document clustering, image segmentation etc.

There are hierarchical and partitional methods in clustering algorithms. In the hierarchical algorithm, there are two modes: (1) agglomerative (start with one data point as cluster and concatenate two nearest clusters iteratively until stopping criterion meets); (2) divisive (start with all data points as one cluster, then split the cluster into smaller one until stopping criterion meets). A dendogram is created during the processed that represent the cluster. On the other hand, Partitional algorithm works to find all clusters iteratively begin with a given non-overlapping cluster. $k$-means is a simple and well known partitional clustering algorithm. Begin with a specified number of clusters to partitioned $N$ data set into $k$ different of clusters. However, the main drawback of these algorithms is to set the final number of cluster subjectively.

There have been many clustering algorithm researched deal with the problem of final number of cluster[2, 3]. This problem is known as automatic clustering that attempt to find the final number of
cluster automatically. Nowadays, evolutionary algorithm has been used widely to handle automatic clustering problem such as genetic algorithm [4], artificial bee colony [5], particle swarm optimization [6], differential evolution [7] etc.

Differential Evolution (DE) is an evolutionary algorithm (EA) for a global optimum solution [8]. DE algorithm is quite simple and faster than the other EA algorithm. Its performances are defined by the control parameter and the strategy of test vector in each generation. There are many proposed method to make DE more effective and efficient. One of the newest strategy is forced strategy differential evolution (FSDE) that modify mutation off original DE [9].

The objective of this paper is to study the automatic clustering algorithm. It uses forced strategy differential algorithm to make result better. The paper is organized as follows. A basic concept of clustering is explained is the first section. The second section explains some literature study followed by the proposed method in the third sections. The obtain result is present in the fourth section. Finally, the final remarks are made in fifth sections.

2. Literature Study

Clustering can also classified as hard clustering and soft clustering. In hard clustering, the data becomes a member on one particular cluster. The rules of hard clustering are as follows: (1) one cluster must have at least one data attached; (2) no data point belongs to multi different cluster and; (3) all of the point inserted into a cluster. However, in the soft clustering or fuzzy clustering, each data points can be members of multi cluster based on the degree of membership.

The k-means algorithm is one of the most popular hard clustering algorithms. Begin with initialization final number of clusters, \( k \), and centroids is taken randomly. Iteratively, the distances from each data point to the centroids are calculated. k-means algorithm always uses Euclidean distance that shows the geometric distance in the search space. The Euclidean distance is computed as:

\[
\text{dist}(x, y) = \left[ \sum (x_i - y_i)^2 \right]^{1/2}
\]

(1)

Where \( x_i, y_i \) are data points to be calculated. Data points will be assigns to the corresponding cluster based on the nearest distance. After that, new centroids are produced using the means of all data point in one cluster. This process will be done until stopping criterion is met

Quality of Cluster

To check the result of clustering algorithm, cluster validity assessment is used. The aim is to get the best partitioning fits with the underlying data. Two common criterions on evaluating and validating clustering result are: (1) compactness (each data point in one cluster should be closest as possible to another); (2) separation means that the cluster result should be widely separate to other cluster. There are many clustering validity index to measure the goodness cluster in quantitative manner that rely on the compactness and separation. One of the popular ones is VI index [10].

The VI index computes the ratio between inter and intra cluster distance. Intra is the average of minimum distance between each point to the centroid in a cluster, whereas inter is the distance between centroid in one cluster to other cluster. By using this index not only number of cluster will be gained, but also the best final partition of data. It is formulated as:

\[
VI = \left( c \times N(0,1) + 1 \right) \times \frac{\text{intra}}{\text{inter}},
\]

(2)

Let VI is the fitness function to be minimized. To avoid too small clustering result, punishment strategy, \((c \times N(0,1) + 1)\), is applied. The parameter \( c \) is set to 30 and \( N(0,1) \) is the Gaussian function. For
data set that have small data point, \( N(0,1) \) is adopted such as iris and wine data set. Intra cluster can be formulated as:

\[
\text{intra} = \frac{1}{N} \sum_{k=1}^{K} \sum_{x \in C_k} \| x - m_k \| \tag{3}
\]

Calculate the Euclidian distance between all data point, \( x \), to the centroid, \( m_k \). Then sum up all the shortest distance of each point to the centroid of cluster, after that divided it by the number of data tuples, \( N \).

Inter cluster is computed as:

\[
\text{inter} = \min \{ d(\bar{m}_k, \bar{m}_{k'}) \}
\]

\[\forall k = 1, 2, ..., K - 1 \]

\[kk = k + 1, ..., K \]

Where \( m_k, m_k \) are the two cluster centroids that have to calculate its distance and \( K \) is the number of clusters.

**Differential Evolution Algorithm**

Like other evolution based algorithm, DE candidate solution, \( X_{i,G} \), is generated from number of population, \( N_p \). Where \( i = 1, 2, ..., N_p \) and \( G \) represent the generation of population. DE proses mainly fall into three phases that are mutation, selection and production. The mutation process computes different weighted vector population using scale factor, \( F \). Begin by selecting three different individuals population randomly \( X_{j1,G}, X_{j2,G} \) and \( X_{j3,G} \) to produce the donor vector \( V_{i,G} \) so that is formulated as:

\[
V_{i,G} = X_{j1,G} + F \times (X_{j2,G} - X_{j3,G}) \tag{5}
\]

Where:

\[i = 1, 2, ..., N_p \]

\[j1, j2, j3 \in \{1, ..., N_p \}, \text{ randomly selected and satisfy } j1 \neq j2 \neq j3 \]

\( F \) is the constant control parameter \([0,1]\).

Crossover operation produce the trial vector, \( U_{i,G} \), to increase the perturbed parameter vectors. It computed as shown in equation (5).

\[
U_{j,i,G+1} = \begin{cases} 
V_{j,i,G+1} & \text{if } \text{rand}_{i,j}[0,1] \leq C, \text{ or if } j = I_{\text{rand}} \\
X_{j,i,G+1} & \text{if } \text{rand}_{i,j}[0,1] > C, \text{ or if } j \neq I_{\text{rand}} 
\end{cases} \tag{7}
\]

Production in DE will select the trial vector then compared it to the target vector in current population. The lowest function value will be chosen to be the new solution for the next generation as shown as follow:

\[
X_{i,G+1} = \begin{cases} 
U_{i,G+1} & \text{if } f(U_{i,G+1}) \leq f(X_{i,G}) \text{ where } i = 1, 2, ..., N \\
X_{i,G} & \text{otherwise}
\end{cases} \tag{8}
\]
FSDE-Forced Strategy Differential Evolution

The FSDE strategy implemented in mutation operator to enhanced the original DE performance [9]. It modify the mutation formula by adding the best solution, $X_{G,best}$, to compute the donor vector. It also attached new mutation factor, $N$, that takes in varying value lies between $[0,1]$ and $F$ parameter used constant value of 0.6. It’s computed as follow:

$$V_{i,G} = X_{i,G} + N.(X_{G,best} - X_{j2,G}) - F.(X_{G,best} - X_{j3,G})$$  \quad (9)

3. Proposed Method

Automatic Clustering based FSDE

The rest of proposed AC-FSDE clustering is as follows:

Step 1) Initialize $k$ (randomly generate) number of cluster centroid of chromosome and $k$ (randomly generate) activation threshold.

Step 2) For each chromosome, $V_{ik}$, find out the active chromosome using rule as shown in equation (7)

Step 3) For $G=1$ to $G_{max}$ do

a) Assign data points to the shortest centroid distance (compute by Euclidian distance between each data point to all actives centroids)

b) Change the population using FSDE. Use the fitness function to compare the better candidate solution.

c) Apply k-means algorithm to do local search. Use the actives cluster to be the input of k-means clustering.

Step 4) Show the final result (minimum fitness function)

The representation of chromosome is based on [7]. Every individual chromosome is a real number vector contain activation threshold and data dimensions, $v_{ik} = (v_{ik}^T \times d_i)$. It is produced randomly generated lies number $[0,1]$. The activation threshold acts as control variable to inform that the cluster active or inactive. After the algorithm is completed, the activation threshold is calculated and the result will be the final number of cluster. Activation threshold has a following rule:

$$\text{if } v_{ik} > 0.5, \text{ then } \text{ cluster center } v_{ik} \text{ is active}$$
$$\text{else } v_{ik} \text{ is inactive}$$  \quad (10)

4. Computational Result

Four well known benchmark data set namely iris, wine, vowel and glass was validated to this proposed method. To algorithm was run 30 times to get its accuracy. VI index is applied to compute the fitness function. The initial cluster number is determined by using $\sqrt{Np}$ formula and the result is compared with automatic clustering using differential evolution (ACDE).

| Datasets | Data Points | Dimensions | Cluster Number | Composition for each cluster |
|----------|-------------|------------|----------------|------------------------------|
| Iris     | 150         | 4          | 3              | 50,50,50                     |
| Wine     | 178         | 13         | 3              | 59,71,48                     |
| Glass    | 214         | 9          | 6              | 70,76,17,13,9,29             |
| Vowel    | 871         | 3          | 6              | 72,89,172,151,207,180        |
Table 2. Tuning Parameter

| Parameter       | Values                      |
|-----------------|-----------------------------|
| Size of population | 10 x dimension             |
| F (ACDE)        | [0.5, 1.0]                  |
| F (AC-FSDE)     | [0.5, 1.0]                  |
| N(AC-FSDE)      | [0.0, 1.0]                  |
| $k_{\text{min}}, k_{\text{max}}$ | $2; \sqrt{Np}$          |

Table 3. Number of cluster

| Dataset | Algorithm | Average number of cluster |
|---------|-----------|---------------------------|
| Iris    | ACDE      | 2.8333 ± 0.4611           |
|         | AC-FSDE   | 3 ± 0.3713                |
| Wine    | ACDE      | 2.8677 ± 0.4342           |
|         | AC-FSDE   | 3.0667 ± 0.3651           |
| Glass   | ACDE      | 5.5333 ± 0.6288           |
|         | AC-FSDE   | 6.0667 ± 0.4498           |
| Vowel   | ACDE      | 6.2667 ± 1.0806           |
|         | AC-FSDE   | 6.2 ± 0.4842              |

5. Conclusion

This study has demonstrated an automatic clustering algorithm using the variant of DE algorithm namely FSDE to obtain the final number of cluster. The result shows that automatic clustering using forced strategy of differential evolution (AC-FSDE) more competitive than automatic clustering algorithm based differential evolution (ACDE). For the next further research can be investigated on more benchmark data as well as the application of real life data set to improve the algorithm.

References

[1] Jain, A.K., Data clustering: 50 years beyond K-means. Pattern Recognition Letters, 2010. 31(8): p. 651-666.
[2] He, H. and Y. Tan, A two-stage genetic algorithm for automatic clustering. Neurocomputing, 2012. 81(Supplement C): p. 49-59.
[3] Peng, H., et al., An automatic clustering algorithm inspired by membrane computing. Pattern Recognition Letters, 2015. 68(Part 1): p. 34-40.
[4] Liu, Y., X. Wu, and Y. Shen, Automatic clustering using genetic algorithms. Applied Mathematics and Computation, 2011. 218(4): p. 1267-1279.
[5] Kuo, R.J., et al., Automatic kernel clustering with bee colony optimization algorithm. Information Sciences, 2014. 283(Supplement C): p. 107-122.
[6] Thong, P.H. and L.H. Son, A novel automatic picture fuzzy clustering method based on particle swarm optimization and picture composite cardinality. Knowledge-Based Systems, 2016. 109(Supplement C): p. 48-60.
[7] Das, S., A. Abraham, and A. Konar, Automatic Clustering Using an Improved Differential Evolution Algorithm. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 2008. 38(1): p. 218-237.
[8] Storn, R. and K. Price, Differential Evolution &ndash; A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. J. of Global Optimization, 1997. 11(4): p. 341-359.
[9] Ramadas, M., A. Abraham, and S. Kumar, *FSDE-Forced Strategy Differential Evolution used for data clustering*. Journal of King Saud University - Computer and Information Sciences, 2016.

[10] Kuo, R.J., et al., *Integration of particle swarm optimization and genetic algorithm for dynamic clustering*. Information Sciences, 2012. 195(Supplement C): p. 124-140.