

Received March 4, 2019, accepted March 27, 2019, date of publication April 11, 2019, date of current version May 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2909926

# A New Cross Clustering Algorithm for Improving Performance of Supervised Learning

WENSHUO ZHOU, KUANGRONG HAO<sup>ID</sup>, (Member, IEEE), CHUNLI JIANG, LEI CHEN,  
XUE-SONG TANG<sup>ID</sup>, AND XIN CAI

Engineering Research Center of Digitized Textile and Apparel Technology, Ministry of Education, College of Information Science and Technology, Donghua University, Shanghai 201620, China

Corresponding author: Kuangrong Hao (krhao@dhu.edu.cn)

This work was supported in part by the International Collaborative Project of the Shanghai Committee of Science and Technology under Grant 16510711100, in part by the National Natural Science Foundation of China under Grant 61603090 and Grant 61806051, in part by the Fundamental Research Funds for the Central Universities under Grant 2232016D-32, Grant 2232017D-08, and Grant 2232017D-13, and in part by the Shanghai Sailing Program under Grant 17YF1426100.

**ABSTRACT** In this paper, a new clustering algorithm is proposed based on cross clusters without using membership functions. In light of the cross clustering data transformation, the spatial distribution of data is changed while the original data dimension simultaneously is maintained. Combining with the performance index and visual technology, an explanation of the performance improvement of the classification model is presented in accordance with the proposed algorithm. This approach was evaluated on UCR time series datasets, the experiments showed that the algorithm can improve not only the accuracy and the performance of the fully convolutional network and nearest neighbor algorithm, but also the time complexity in time series classification model. It is worth well to apply this method to further research and popularization.

**INDEX TERMS** Cross clustering, time series classification, full convolutional network, nearest neighbor algorithm.

## I. INTRODUCTION

A clustering algorithm is an unsupervised classification algorithm. The classification process is to automatically aggregate data using rules based on data similarity, and then the category of data is defined. Most common clustering algorithms are mainly based on partitioning, hierarchy, density, graph theory, grids, and models [1], [2]. The most popular clustering algorithms include hard clustering methods with clear classification, and soft clustering methods such as fuzzy C-means clustering [3]–[5], whose final result can be transformed into a hard cluster by a membership function.

In recent years, clustering algorithms are sometimes used to process compression transformation on time series data [2], which can reduce the time complexity of subsequent algorithms under the premise that their classification accuracy is affected rarely. For example, as for the time series classification problem, Li and Wan [6] adopted near-neighbor propagation clustering algorithm [7] to select the cluster center points of each cluster, and the cluster center points can represent the original data. They showed that KNN algorithm

based on this method improves the classification performance in the time series classification task. Hwang and Cho [8] used k-means [9] to preprocess the training data set first and exploit KNN to calculate the distance to predict the category. The experimental results show that the classification time is reduced by 85% without loss of precision.

The clustering algorithm is used to extract the representative elements of the data, which is robust to outliers. However, when traditional clustering algorithms extract the data features, the compression of the data is very serious, only the main features of the data are extracted and some necessary features will be lost. Therefore, this paper proposes a cross clustering algorithm that retains more features when extracting the representative elements. Furthermore, this method can extract the representative elements of training set in supervised learning, and then use these representative elements to train the supervised model aiming at improving the model performance.

There are a lot of time series in life, such as the trend of stock prices, the formation of tumors, and changes in the environment [10]. The research of them is of great significance, especially for the prediction. Then the study of time series has received extensive attention in the last decade. The nearest

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Tong.

neighbor algorithms based on Euclidean distance (1NN-ED) [11]–[14] and dynamic time warping (1NN-DTW) [13], [14] are considered to be the simplest and most effective gold baseline model in many time series classification tasks.

Time series classifications have made new progress in recent years for feature-based methods [15]–[18], integrated learning-based methods [19], [20] and deep learning-based methods [14], [21], [22], which have been proven to be feasible. Baydogan *et al.* [15] proposed a time series classification framework (TSBF) based on feature package representation, which has improved classification accuracy in UCR time series datasets [23] compared to other feature representation methods. Lines and Bagnall [20] proposed the proportional elastic ensemble (PROP) model, which combines 11 weak classification models based on elastic distance metrics and tests in 44 UCR time series datasets with improved classification accuracy. Wang *et al.* [14], proposed a time series classification model based on the full convolutional network (FCN) whose prediction accuracy is higher in most UCR time series datasets. Ismail Fawaz *et al.* [21] used the FCN network [14] to learn the migration between different datasets in the UCR time series datasets, and indicated that the migration learning based on FCN networks can improve the classification accuracy of partial time series data.

Although clustering algorithms are traditional machine learning model, there is no consensus on the definition of clustering algorithms [24]. Considering that an instance can only belong to one class in the existing hard clustering methods, but for a particular practical problem, it may be more efficient to allow one instance belong to multiple categories. Therefore, this paper proposes a cross clustering model that implements one instance into multiple classes or clusters. The advantage of this model is that the dimensions of data are remained after the centripetal aggregation transformation while the spatial distribution of data changed, which may reduce the amount of data.

In reality, not all data instances can be classified into specific categories, such as temperature scores. Suppose the number of categories is 2, the demarcation threshold is 25 °C. By using hard clustering algorithm, temperature above 25 °C is classified as heat, otherwise is classified as cold. However, 39 °C is closer to cold category than 100 °C, but 39 °C and 100 °C all belong to heat category according to this rule. It is obviously that the hard clustering cannot define this problem very well. Although soft clustering can give the probability of instance belonging to cluster by membership functions, we still hold the view that it makes no sense because 100 °C is not cold at all for human perception. Therefore, a cross clustering algorithm is proposed, which performs autonomous partitioning by similar distance. The proposed cross clustering algorithm does not need to use membership degree and allows crossover between clusters. In other words, an instance object can belong to multiple categories simultaneously.

This article is arranged as follows. In Section II, the basic model and work related to our method are discussed. The cross clustering algorithm is proposed in Section III.

In Section IV, we propose a time classification model based on cross clustering and a time series classification model based on the cross clustering of limitation of distance threshold. In Section V, we present the test results and discuss the reasons for the improvement of classification accuracy. Finally conclusions are provided in Section VI.

## II. DEFINITIONS AND RELATED WORK

This section introduces the definition of time series classification and basic methods used in this paper, which including KNN, cluster center visualization and full convolution network.

### A. TIME SERIES CLASSIFICATION

*Definition 1:* The real value sequence of a variable that changes along the time axis as a function of time  $\mathbf{x} = [x_1, x_2, \dots, x_T]$  is recorded as a time series, where  $T$  denotes the dimension of a time series.

*Definition 2:* A time series data set can be represented as  $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ , its corresponding label set is  $Y = \{y_1, y_2, \dots, y_n\}$ , where  $y_i$  represents the class label corresponding to time series  $\mathbf{x}_i$ . The mapping from the time series data set to the class label set is defined as  $f : X \rightarrow Y$  and the mapping  $f$  is denoted as classification model on time series.

Time series classification has two forms, one is the unlabeled clustering model, and the other is the labeled classification model. The researches generally refer to the labeled classification model in the recent ten years. Time series data sets are diverse and the UCR archive is currently the most used data set [24]. By the end of 2018, this database contains 128 time series data sets.

### B. K-NEAREST NEIGHBOR (KNN)

KNN is an algorithm commonly used for classification which is a relatively simple classical machine learning algorithm [25]. At present, this algorithm is outstanding in the classification of time series, and becomes a new golden baseline model for algorithm comparison. Algorithm 1 shows the implementation of KNN algorithm.

### C. CLUSTER CENTER VISUALIZATION

The spatial distributions of the new sample data and the original data before and after using the proposed clustering algorithm are compared to illustrate the validity of our proposed algorithm. Class aggregation of data is evaluated based on the visualization method of clustering centers proposed in 2014 [24]. This method can be used to visually present the distribution of class centers of the data and then guide us to choose the number of classes. The core idea of this method is to draw the decision graph with two indicators of local density and distance, and select the optimal number of clustering through the decision graph.

**Algorithm 1** K-Nearest Neighbor Algorithm

**Input:** trainset  $S(n)$ , testset  $X(m)$ , Number of neighbors  $k$   
**Output:** The category of  $x$

**Process:**

1. **for**  $i = 1, 2, \dots, m$  **do**
2.     Calculate Euclidean distance  $d(i)$  between the test sample point  $\mathbf{x}_i$  and the training sample
3.     Sort  $d(i)$  in ascending order
4.     Select the  $k$  points with the smallest distance from the current test sample point
5.     Determine the frequency of occurrence to the category of the first  $k$  points
6.     Return the category with the highest frequency of occurrence of the current  $k$  point as the predicted classification of the current point  $\mathbf{x}_i$
7. **end for**

*Definition 3:* Local density  $\rho_i$  represents the number of data point  $\mathbf{x}_i$  within a certain truncation threshold.

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \quad (1)$$

$$\chi(\Delta d) = \begin{cases} 1, & \Delta d < 0 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$d_c$  is the truncation distance,  $d_{ij}$  is the distance between data points  $\mathbf{x}_i$  and  $\mathbf{x}_j$ .

*Definition 4:*  $\delta_i$  represents the minimum distance of data point  $\mathbf{x}_i$  from other points with higher local density.

$$\delta_i = \min_{j: \rho_j > \rho_i} (d_{ij}). \quad (3)$$

If data point  $\mathbf{x}_i$  is the largest local density, then

$$\delta_i = \max (d_{ij}). \quad (4)$$

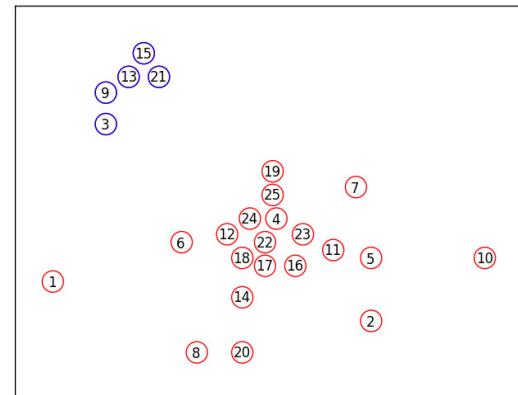
Make the two-dimensional decision graph with the local density  $\rho_i$  as abscissa and distance  $\delta_i$  as vertical coordinate. The number of possible class centers, the degree of class center visibility and outliers can be clearly indicated from the decision graph.

The spatial distribution of the two-dimensional data is shown in Fig. 1 (a), and Fig. 1 (b) shows the decision map of the data in Fig. 1 (a), where the data with larger local density and distance is regarded as the class center point, such as points 22 and 13 while the small local density and large distance as the outliers, such as points 1 and 10.

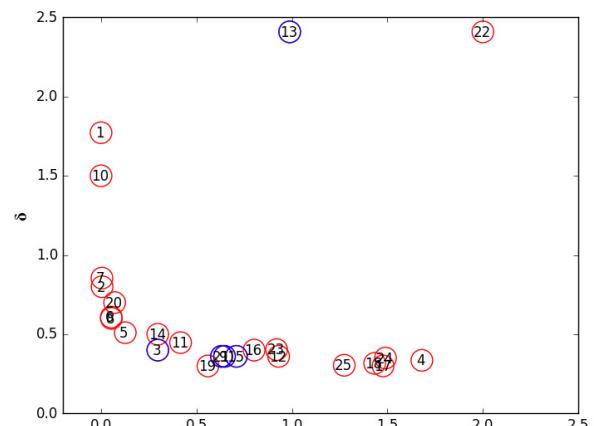
There are two characteristics of a class center in this method. The first case is that the density of class center is larger than that of its neighbors. The second case is that the distance between the class center and other denser data point is relatively large.

**D. FULL CONVOLUTION NETWORK (FCN)**

In recent years, FCN performs well in image semantic segmentation [26], time series classification and other



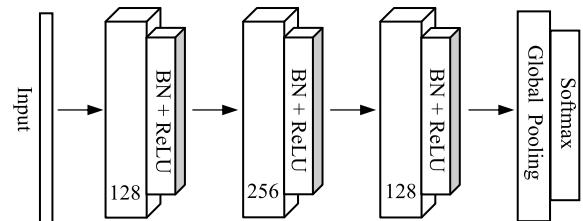
(a)



(b)

**FIGURE 1.** The algorithm in two dimensions. Different colors correspond to different clusters. (a) Point distribution. (b) Decision graph for the data in (a).

fields [14], [21]. FCN [14] for a comparative experiment is exploited due to its best results in most dataset.



**FIGURE 2.** Full Convolution Network (FCN).

In Fig. 2, FCN consists of three convolution layers, a global pooling layer and a Softmax layer. Softmax layer plays a role in classification, and convolution layer can extract the features of data. A basic convolutional layer includes convolution operation, batch normalization layer [27] and ReLU layer, as shown in Eq. (5).

$$\begin{aligned} \mathbf{y} &= \mathbf{W} \otimes \mathbf{x} + \mathbf{b} \\ \mathbf{s} &= \text{BN}(\mathbf{y}) \\ \mathbf{h} &= \text{ReLU}(\mathbf{s}). \end{aligned} \quad (5)$$

where  $\otimes$  represents the convolution operation. The convolution operation is fulfilled by three 1-D kernels with the size {8,5,3}, and the three convolutions have a stride equal to 1. The number of channels per layer is {128, 256, 128}, and there is no pooling layer among them [28]. The batch normalization layer can speed up network training and improve generalization. The convolutional layer is followed by a global average pooling layer rather than the full connection layer which aims to reduce network parameters and further improves the network training speed. The final Softmax layer is responsible for classifying features.

### III. CROSS CLUSTERING ALGORITHM

In this section, the cross clustering algorithm is proposed, which performs autonomous partitioning by similar distance. It is a clustering algorithm that does not need to specify the number of categories in advance, and allows crossover among clusters. In other words, an instance object can belong to multiple categories simultaneously.

**Definition 5:** A cluster represents a collection of elements with a high degree of similarity in the data set. Let  $C_i = \text{SLink}(\mathbf{x}_i)$  indicate a cluster set  $C_i$  that is obtained by passing an instance  $\mathbf{x}_i$  from a similar relationship.

Let  $S = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  denote the whole data set and calculate the similarity  $d_{ij}$  between instances in advance when performing cross clustering.

$$d_{ij} = \text{dist}(\mathbf{x}_i, \mathbf{x}_j). \quad (6)$$

Here, “dist” denotes the Euclidean distance between  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . After calculation, the similarity matrix  $D$ , consisting of  $d_{ij}$ , is generated. The next step is to traverse each instance according to the following rules. Take an instance  $\mathbf{x}_a$  for example:

**Step 1:** look for instance  $\mathbf{x}_b$  closest to its similarity and instance  $\mathbf{x}_c$  closest to instance  $\mathbf{x}_b$ , if  $a = c$ , the search is stopped.

**Step 2:** set  $\{\mathbf{x}_a, \mathbf{x}_b\}$  is the defined cluster, otherwise it continues to find instance  $\mathbf{x}_d$  which is closest to the similarity with  $\mathbf{x}_c$ , and compared  $\mathbf{x}_d$  with  $\mathbf{x}_a$  and  $\mathbf{x}_b$  next. If  $d = a$  or  $d = b$ , the search is stopped for that set  $\{\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c\}$  is divided into one cluster. Traverse all instances according to such a rule until a closed set is formed which means that the clusters corresponding to instance  $\mathbf{x}_a$  are divided up.

**Step 3:** repeat the above steps to divide the clusters corresponding to the next instance, and loop until all the clusters of the instances are obtained. At last, we remove the repeated set, the set of clusters obtained is the result of once cross clustering.

Twice cross clustering can be proceeded for data after once cross clustering. The number of clusters after once cross clustering is often larger and the individual cluster size is smaller. Supposing that  $m_1$  clusters can be obtained after once cross clustering and they can be represented as  $\{C_1^1, C_2^1, \dots, C_{m_1}^1\}$ . And then define the center point of each cluster to calculate the distance between them, which are called similar distances between the clusters. As for a cluster of  $C_i^1 = \{\mathbf{x}_{(i,1)}^1, \mathbf{x}_{(i,2)}^1, \dots, \mathbf{x}_{(i,q)}^1\}$ , where  $q$  represents the

number of instances of the  $i$ -th cluster after first time cross clustering and the center point of the cluster is  $\mathbf{c}_i^1$ , which is calculated as follows:

$$\mathbf{c}_i^1 = \frac{1}{q} \sum_{p=1}^q \mathbf{x}_{(i,p)}^1 \quad (0 < i \leq m_1) \quad (7)$$

Therefore, the cluster center data set becomes  $Z^1 = \{\mathbf{c}_1^1, \mathbf{c}_2^1, \dots, \mathbf{c}_{m_1}^1\}$  after one-time cross clustering and the similarity  $d_{ij}$  between the cluster centers is calculated as follows.

$$d_{ij}^1 = \text{dist}(\mathbf{c}_i^1, \mathbf{c}_j^1) \quad (8)$$

The similarity matrix  $D^1$ , consisting of  $d_{ij}^1$ , is obtained. The center of each primary cross cluster is traversed afterwards to perform the second cross clustering following the previous operation, then  $m_2$  clusters can be described as  $\{C_i^2, C_2^2, \dots, C_{m_2}^2\}$  and each cluster is defined as  $C_i^2 = \{\mathbf{c}_{(i,1)}^1, \mathbf{c}_{(i,2)}^1, \dots, \mathbf{c}_{(i,q)}^1\}$ . Where  $\mathbf{c}_{(i,t)}^1$  indicates the  $t$ -th element of the  $i$ -th cluster of the second cross clustering, which is corresponding to cluster set of  $C_{(i,t)}^1$ . The original data instance cluster set corresponding to the clusters of the secondary cross cluster is as follows.

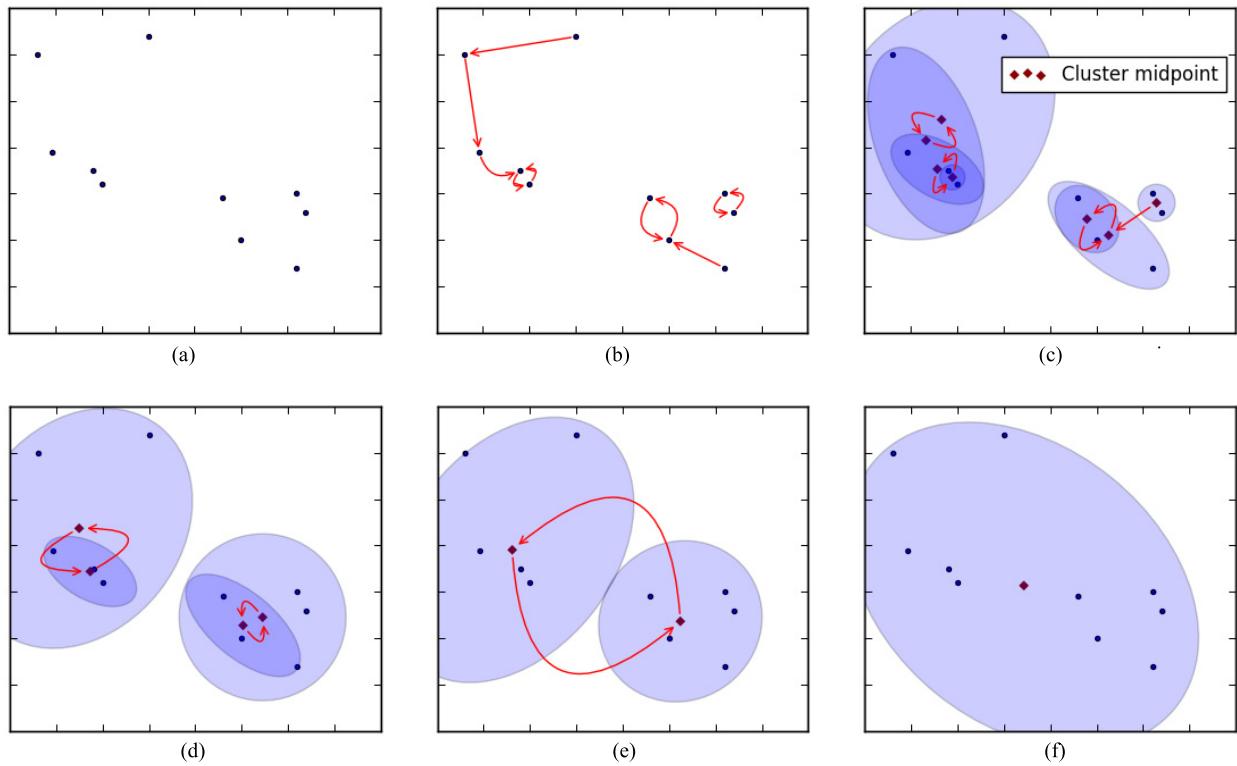
$$C_i^2 = C_{(i,1)}^1 \cup C_{(i,2)}^1 \cup \dots \cup C_{(i,q)}^1 = \{\mathbf{x}_{(i,1)}^2, \mathbf{x}_{(i,2)}^2, \dots, \mathbf{x}_{(i,q-2)}^2\}. \quad (9)$$

For  $w$  times cross clustering, the corresponding instance of each cluster set is

$$C_i^w = \{\mathbf{x}_{(i,1)}^w, \mathbf{x}_{(i,2)}^w, \dots, \mathbf{x}_{(i,q-w)}^w\}. \quad (10)$$

With iterations of cross clustering, the number of instances contained in each cluster is increasing, the number of clusters is decreasing, and the concept represented by each cluster is broader. The cross clustering algorithm is given in Algorithm 2.

Fig. 3 intuitively shows the process of cross clustering where the changes of clusters and center point of each cross clustering are clearly presented. For an instance of the arrow’s starting point, its endpoint is connected to its nearest instance. Fig. 3 (a) shows the distribution of the original data points. In Fig. 3 (b), each cluster is a starting point to find the nearest point along the direction indicated by the arrow, and the obtained points continue to seek its nearest point until a ring is formed. Each ring starting from the starting point is regarded as a cluster and the ellipse shown in Fig. 3 (c), which is formed after once cross clustering. Each ellipse is regarded as a cluster to determine the cluster center point according to equation (7) and it can be found that the distribution of cluster center point is more concentrated than the original data. This is exactly the whole process of one-time cross clustering. After that the center points of each clusters keep to link with their nearest center point until a ring structure is formed which means the search is enough to stop and the result is shown in Fig. 3 (d). Similarly, each ellipse represents a cluster. The cluster center point is calculated in the same way and continued the next cross clustering until all the instances are clustered into one cluster which implies the end of the



**FIGURE 3.** The visualization of cross clustering process. (a) Original Data. (b) Cluster once. (c) Cluster twice. (d) Cluster thrice. (e) Cluster four times. (f) Final result.

entire cross clustering process. It is not difficult to discover that seven clusters formed after the first cross clustering, four clusters formed after the second cross clustering, two clusters formed after the third cross clustering and one cluster formed after the fourth cross clustering, just as shown in Fig. 3 (f). It can easily discover that the number of clusters gradually decreases while the density of centers gradually increases during the process of clustering.

Cross clustering has its own advantages for that it can be utilized to divide one data into different clusters and exploit the center of clustering to represent the whole cluster thus realize data compression [31]. Subspace reduction algorithm [29] such as PCA can compress data by reducing the dimension of data so that reduce the amount of data that model need process. Moreover, the number and spatial distribution of data can be modified by using the cross clustering algorithm which proceeds cross clustering on original data and regards the centers of clusters as new data. This method can effectively remove some outliers and noise points to optimize the performance of the learner and reduce the time complexity.

#### IV. CLASSIFICATION FOR TIME SERIES DATA BASED ON CROSS CLUSTERING

In this section, the proposed cross clustering algorithm is used to improve the existing model, which proves that the algorithm can improve the accuracy of the existing

classification algorithm and reduce the time complexity of some algorithms.

The proposed cross clustering algorithm has two improved schemes. One is the nearest neighbor algorithm based on cross clustering (CC-ED-1NN). This algorithm is given in Algorithm 3.

For time series classification task, the result is best when  $k = 1$ . Especially when the data samples are unbalanced and the value of  $k$  is larger, the  $k$ -nearest neighbor algorithm is more likely to match the category with a large amount of data rather than the data of a small amount. Therefore, taking  $k$  as 1 can weaken the defect of data imbalance.

The outliers and noise of data always exist, but their proportion will not be too big. Although our proposed cross clustering algorithm improves the accuracy of classification of some models to some extent, it does not perform always well in all models for that the degree of compression of data is still a little high by one-time cross clustering. Therefore, the time series classification model is proposed based on the distance threshold of the cross clustering data transformation to improve the prediction accuracy for get better classifying models. The advantage of this algorithm is that the distance threshold is added to make partial data cross-clustered, and the other part is not cross-clustered. It is far away from other data instances which often regarded as outliers or noise points. The algorithm is given in Algorithm 4.

The purpose of removing duplicate data is to make the data shrink to the center of cluster faster and reduce the

**Algorithm 2** Cross Clustering Algorithm

---

**Input:** dataset  $S = \{\mathbf{x}_i\}_{i=1}^N$

**Output:** Cluster divisions of each cluster:  $C$ , Cluster center dataset:  $Z$

**Process:**

1. Initialize the number of clusters  $C_{num}$  to be equal to the number of dataset instances
2. Number of times of clusters:  $Cc = 0$
3. Cluster center dataset  $Z = \{Z^1, Z^2, \dots\}$
4. Cluster divisions of each cluster:  
 $C = \{C^1, C^2, \dots\}, C^i = \{C_1^i, C_2^i, \dots\}$
5. **while**  $C_{num} > 1$  **do**
6.   Calculate similar distance  $d_{ij} = \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$
7.    $Cc = Cc + 1; C^{Cc} = \emptyset$
8.   **for**  $i = 1, 2, \dots, n$  **do**
9.     Instance  $i$ :  $\mathbf{x}_i$ ; Cluster  $i$ :  $C_i^{Cc} = \emptyset$
10.     $C_i^{Cc} = C_i^{Cc} \cup \{\mathbf{x}_i\}, j = \arg \min(d_{ij})$
11.     $C_i^{Cc} = C_i^{Cc} \cup \{\mathbf{x}_j\}, k = \arg \min(d_{kj})$
12.    **for**  $p = 1, 2, \dots, n - 2$  **do**
13.     **if**  $\mathbf{x}_k \in C_i^{Cc}$  **then**
14.       **break**
15.     **else**
16.        $C_i^{Cc} = C_i^{Cc} \cup \{\mathbf{x}_k\}, k = \arg \min(d_{kp})$
17.     **end if**
18.    **end for**
19.     $C^{Cc} = C^{Cc} \cup \{C_i^{Cc}\}, q = \text{size}(C_i^{Cc})$
20.     $\mathbf{c}^i = \frac{1}{q} \sum_{p=1}^q \mathbf{x}_{(Cc,p)}, Z^{Cc} = Z^{Cc} \cup \{\mathbf{c}^i\}$
21.   **end for**
22.    $C = C \cup \{C^{Cc}\}, Z = Z \cup \{Z^{Cc}\}, S = Z^{Cc}$
23.   Calculate the number of different clusters:  
 $C_{num} = \text{size}(C^{Cc})$
24. **end while**

---

number of data, so as to improve the accuracy of learning model classification and reduce the time complexity of some models. However, too much compression of the data may lose some crucial features of data. The cross clustering algorithm based on limitation of distance threshold may do not compress the original data. For example, only a few data get cross clustered if distance threshold is too big. Besides, there will be no overlapping clusters and does not reduce the number of data which just lead to a slight change of data distribution. It is exactly because of this change that the classification of subsequent learning models is better.

This part of experiments is to detect whether the FCN network can be improved in the accuracy of classification in UCR time data set by comparing the data before and after cross clustering data transformation. The network parameter settings is same to literature [14], which aims to facilitate comparing the results. The result of [14] is currently the best in time series classification algorithms and they set Epochs at 2000, Batch size at 16, Optimizer as Adam [30], the Learning rate as 0.001, the First moment as 0.9,

**Algorithm 3** Nearest Neighbor Algorithm Based on Cross Clustering Data Transformation (CC-ED-1NN)

---

**Input:** train dataset  $S_1 = \{\mathbf{x}_i\}_{i=1}^N$ , test dataset  $S_2 = \{\mathbf{x}_i\}_{i=1}^M$

**Output:** The accuracy of classification of test set:  $acc$

**Process:**

1. Perform cross clustering in each category of the training set
2. Take the cluster center of each class as a new data set
3. Use the new data as a new training set, using 1NN

---

**Algorithm 4** The Time Series Classification Model Based on the Cross Clustering Data Transformation of the Limitation of Distance Threshold

---

**Input:** train dataset  $S_1 = \{\mathbf{x}_i\}_{i=1}^N$ , test dataset  $S_2 = \{\mathbf{x}_i\}_{i=1}^M$

**Output:** The accuracy of classification of test set:  $acc$

**Process:**

1. Calculate similar distance  $d_{ij} = \text{dist}(\mathbf{x}_i, \mathbf{x}_j)$  in each category of the training set
2. Set the threshold:  $T$
3. **if**  $\arg \min(d_{ij}) > T$  **do**
4.   Perform cross clustering on the data and take the cluster center point value  $x'$
5.   Let  $x = x'$ , and remove duplicate data
6. **end if**
7. Calculate the classification model with the updated training set

---

The Second moment as 0.999 and the loss function as cross-entropy.

**V. RESULT AND INTERPRETABILITY**

In this section, the nearest neighbor algorithm is used based on crossover clustering to make a comparative experiment on 80 time series data sets from UCR. The result showed that our algorithm significantly improved the nearest neighbor algorithm of Euclidean distance in the time series data sets with high classification accuracy. Afterwards, the time series classification model based on the crossover clustering of the limitation of distance threshold is used to improve the accuracy of FCN network in time series classification.

Finally, taking the cluster center as new data and get the changes of spatial clustering before and after data processing to illustrate the reason why our algorithm can improve the classification accuracy.

**A. RESULTS OF NEAREST NEIGHBOR ALGORITHM BASED ON CROSS CLUSTERING**

The analysis results of the UCR time series in this experiment, shown in Table 1, are proved that the improvement of our algorithm has generality. The contents given in Table 1 are the type of the time series data set, including the number of types (num-set), the average compression ratio of training set data (c-ratio), the result of the nearest neighbor algorithm of the Euclidean distance (ED), and the nearest neighbor algorithm of the Euclidean distance of our algorithm (CC-ED).

**TABLE 1.** Data transformation improves knn classification performance.

| Type      | num-set | c-ratio | ED            | CC-ED         |
|-----------|---------|---------|---------------|---------------|
| Device    | 6       | 0.7683  | <b>0.5473</b> | 0.5504        |
| ECG       | 6       | 0.7307  | 0.1571        | <b>0.1450</b> |
| Image     | 28      | 0.7698  | 0.2939        | <b>0.2876</b> |
| Motion    | 13      | 0.7442  | <b>0.2984</b> | 0.3024        |
| Power     | 1       | 0.7278  | 0.0222        | <b>0.0167</b> |
| Sensor    | 15      | 0.7505  | 0.2422        | <b>0.2385</b> |
| Simulated | 4       | 0.7277  | 0.2055        | <b>0.1969</b> |
| Spectro   | 7       | 0.6958  | <b>0.1966</b> | 0.1993        |
| Average   | -       | 0.7393  | 0.2454        | <b>0.2421</b> |

Note: ‘-’ Indicates inexistence.

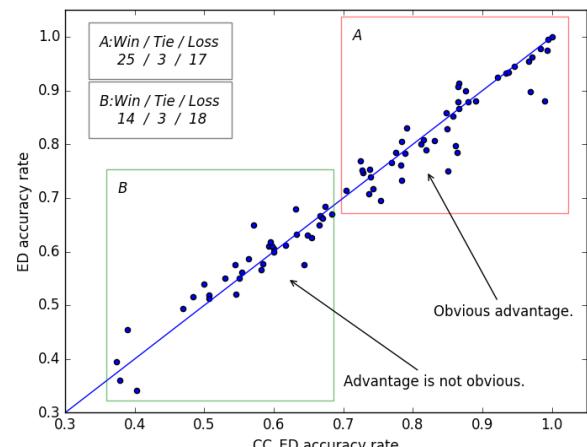
It can be seen from Table 1 that the average compression ratio is 73.93% and the average error is 0.2421 after processed by the nearest neighbor algorithm based on cross clustering, which is slightly better than the result before processing. This shows that the proposed model can improve the accuracy of classification of data set and compress the training data set.

It can be seen from Table 1 that the average compression ratio is 73.93% and the average error is 0.2421 after processed by the nearest neighbor algorithm based on cross clustering, which is slightly better than the result before processing. This shows that the proposed model can improve the accuracy of classification of data set and compress the training data set.

Fig. 4 further illustrates the efficiency of our algorithm which the abscissa representing the accuracy of CC-ED, and the ordinate implying the accuracy of ED without cross clustering. The data sets with same accuracy before and after the cross clustering data transformation are presented with the points where the line passes through. The points above the line represent the data set whose ED accuracy is greater than the CC-ED accuracy while the points below the line are the opposite. The B region in Fig. 4 shows that our algorithm has no advantage when the data set classification accuracy is low. The A region shows that when the data set classification accuracy is better, our algorithm has obvious advantages. This is because in the classification algorithm with Euclidean distance as a measure, the data set with higher classification accuracy has better intraclass similarity and greater distance between classes, so the data set is smoother after cross clustering data transformation and reduce the noise point and outliers with the classification accuracy is improved. However, for data sets with poor classification accuracy, the data distribution overlap between classes is severe. Even after using cross clustering data transformation, the different types of data are not well separated.

### B. RESULTS OF TIME SERIES CLASSIFICATION MODEL BASED ON DISTANCE THRESHOLD LIMITING CROSS CLUSTERING

Compared with the nearest neighbor algorithm of Euclidean distance, the model performs better. In fact, the proposed algorithm can improve the accuracy of classification of models by setting appropriate thresholds.

**FIGURE 4.** Comparison of 1NN classification accuracy before and after data transformation in UCR data set.

The accuracy of classification and the loss of validation are compared before and after the cross clustering data transformation on time series Distal Phalanx Outline Correct (DPOC) of FCN network [14]. Setting a threshold  $T = 0.01$ , the training set data are transformed from 600 samples to 576 samples by cross clustering. The results of several tests are shown in Table 2. Fig. 5 shows the loss and accuracy of one training process. Table 2 contains the final results for 12 times trials and Fig. 5 shows the change process of data in one of the 12 trials.

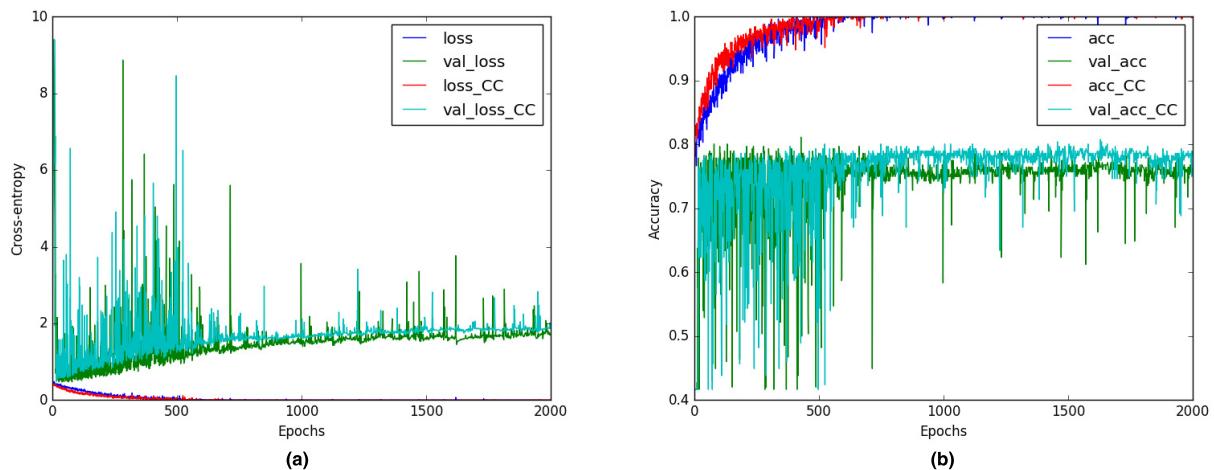
The selection of the threshold  $T$  directly affects the strength of the cross-cluster extracting representative points. If the  $T$  value is too large, the strength of the data processing of the cross clustering will be weakened, and only a small number of outliers will be clustered. As the threshold  $T$  is set to decrease, the data for the cross clustering transformation will increase and the representative points will be reduced. If the threshold  $T$  is too small, the data features will be seriously lost, which will weaken the performance of the supervised model. Therefore, the threshold  $T$  is an important parameter for improving the performance of the supervised model. For data with fewer outliers and small data volume, it is recommended to set a larger threshold to only filter a small number of outliers. It is recommended to set a smaller threshold for data with a large number of outliers and a large amount of data to compress the data and reduce the outliers meanwhile.

It can be seen from Table 2 that the average validation accuracy of the DPOC data set is increased by 1.84% when the threshold is set to 0.01. At the same time, Fig. 5 evidently shows that the accuracy of the validation is significantly improved. Fig. 5 and Table 2 together reflect a strange phenomenon that the accuracy of validation increases while the loss of validation set also increases instead of decreasing. After the cross clustering data transformation, the increase in accuracy is due to the fact that our cross clustering changes the spatial distribution of the training set data. Fig. 5 (a) displays that the cross-entropy of the validation data becomes

**TABLE 2.** Training results before and after cross clustering data transformation of dpc dataset by fcn network.

| The numbers of trials | Before transformation |      |               |         | After transformation |      |          |               |
|-----------------------|-----------------------|------|---------------|---------|----------------------|------|----------|---------------|
|                       | loss                  | acc  | val_loss      | val_acc | loss                 | acc  | val_loss | val_acc       |
| 1                     | 3.5350E-05            | 1.00 | 1.7286        | 0.7572  | 1.8155E-05           | 1.00 | 1.8194   | 0.7790        |
| 2                     | 3.2431E-05            | 1.00 | 1.6653        | 0.7428  | 2.3782E-05           | 1.00 | 1.9804   | 0.7681        |
| 3                     | 3.1018E-05            | 1.00 | 1.7337        | 0.7536  | 1.9047E-05           | 1.00 | 2.0220   | 0.7609        |
| 4                     | 1.8372E-05            | 1.00 | 1.7101        | 0.7464  | 1.9166E-05           | 1.00 | 1.8610   | 0.7790        |
| 5                     | 2.7350E-05            | 1.00 | 1.8844        | 0.7355  | 1.4226E-05           | 1.00 | 1.9898   | 0.7645        |
| 6                     | 3.3369E-05            | 1.00 | 1.6282        | 0.7464  | 9.2570E-06           | 1.00 | 1.9325   | 0.7790        |
| 7                     | 3.0748E-05            | 1.00 | 1.6986        | 0.7681  | 2.5276E-05           | 1.00 | 1.8445   | 0.7790        |
| 8                     | 2.9404E-05            | 1.00 | 1.7414        | 0.7645  | 1.9464E-05           | 1.00 | 1.8920   | 0.7790        |
| 9                     | 2.2745E-05            | 1.00 | 1.7395        | 0.7645  | 1.0855E-05           | 1.00 | 1.8574   | 0.7681        |
| 10                    | 3.1157E-05            | 1.00 | 1.7224        | 0.7609  | 1.5120E-05           | 1.00 | 1.9321   | 0.7754        |
| 11                    | 4.3647E-05            | 1.00 | 1.7121        | 0.7572  | 1.5279E-05           | 1.00 | 2.0053   | 0.7754        |
| 12                    | 4.7787E-05            | 1.00 | 1.5883        | 0.7536  | 2.0681E-05           | 1.00 | 1.8193   | 0.7645        |
| Average               | 3.1948E-05            | 1.00 | <b>1.7127</b> | 0.7542  | <b>1.7526E-05</b>    | 1.00 | 1.9130   | <b>0.7726</b> |

Note: 'loss', 'acc', 'val\_loss' and 'val\_acc' respectively represent training set loss, training set accuracy, validation set loss and validation set accuracy.



**FIGURE 5.** FCN training results before and after cross clustering data transformation on data set DPOC. 'loss', 'val\_loss', 'acc' and 'val\_acc' respectively represent the training set loss, validation set loss, training set accuracy and validation set accuracy before cross clustering data transformation. 'loss\_CC', 'val\_loss\_CC', 'acc\_CC' and 'val\_acc\_CC' respectively represent the training set loss, validation set loss, training set accuracy and validation set accuracy after cross clustering data transformation. (a) The Cross-entropy with the number of iterations. (b) The accuracy with the number of iterations.

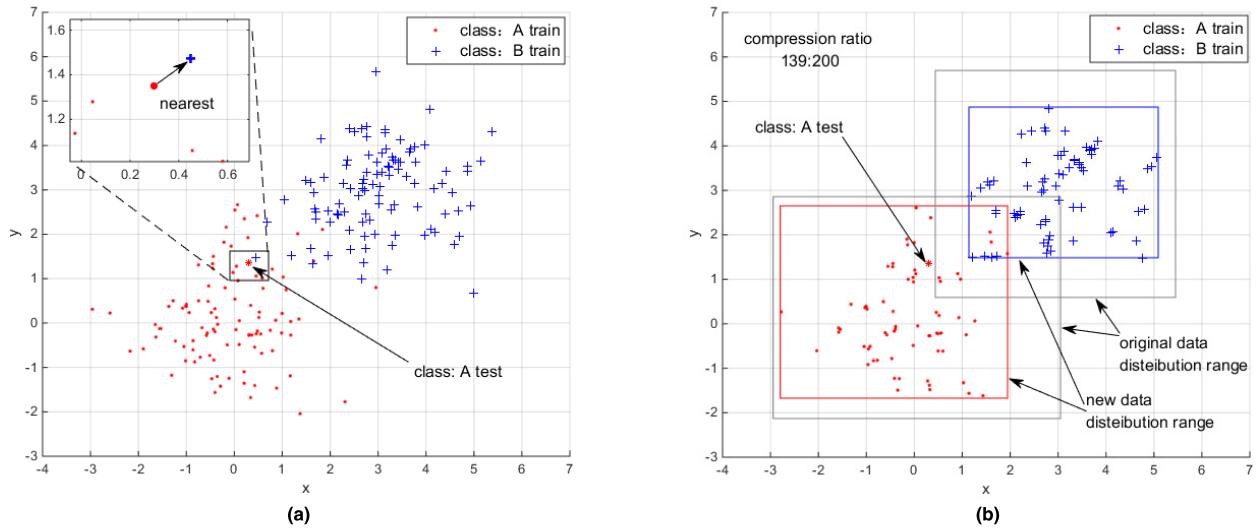
larger after cross clustering data transformation, which is the performance of over-fitting and the validation accuracy is generally reduced.

In fact, this kind of contravention of common sense is explicable for that the distribution of the original train data is changed after the cross clustering data transformation. Then the data itself becomes smoother and more concentric. Furthermore, this process is accompanied by data reduction which results in the better fitness of train set and the worse fitness of test set relatively in FCN network. This phenomenon of “over-fitting” is due to the cross clustering data transformation of train data rather than validation data. It can also be seen from Fig. 5 that the convergence of data is faster after transformation in the process of training.

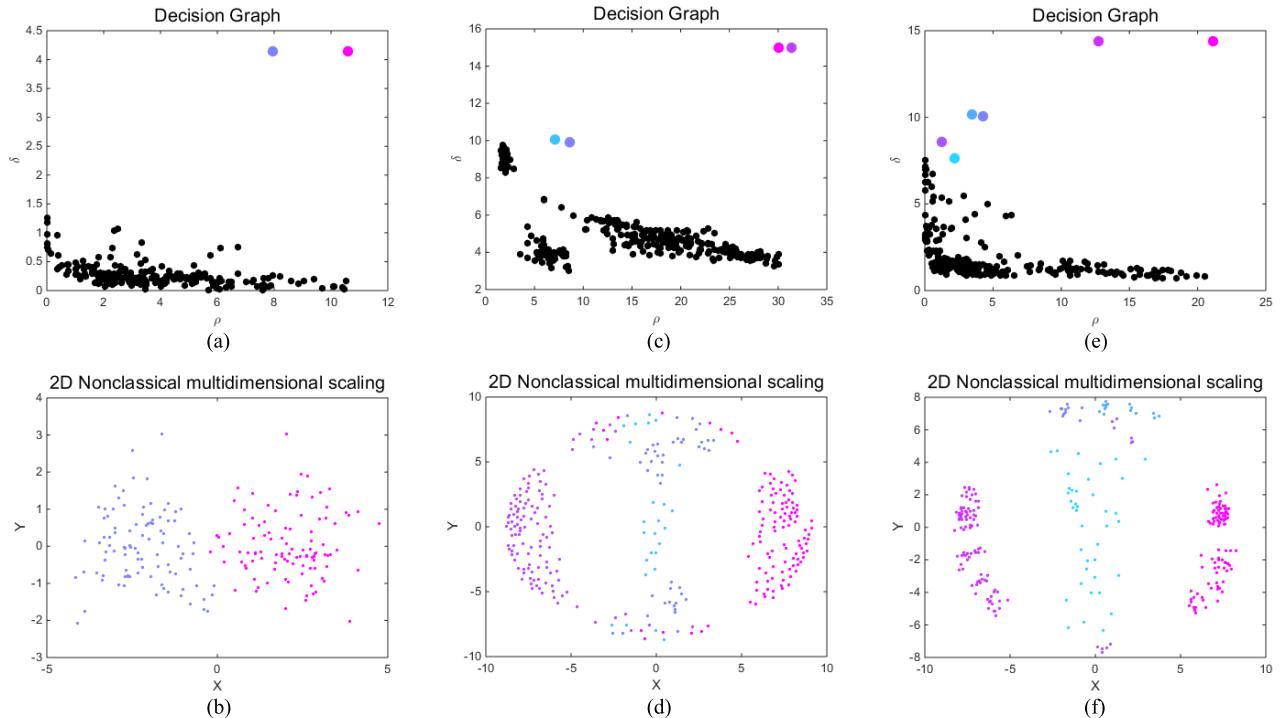
### C. WHY CAN CROSS CLUSTERING ALGORITHM IMPROVE THE ACCURACY OF THE MODEL?

In this part, using a simple experiment to explain why the algorithm can improve the accuracy of the classification model after the cross clustering data transformation. Then the method in [24] is used to visualize the changes of data centers before and after cross clustering, and the distribution of non-classical multidimensional scaling transformations (non-classical MDS) is used to observe the changes of distribution of data before and after cross clustering in two-dimensional plane.

First, constructing a two-dimensional standard positive distribution of two 200 data samples in which each dimension is an independent one-dimensional standard positive



**FIGURE 6.** Comparison of data distribution before and after cross clustering data transformation. (a) The distribution of raw data. (b) The distribution of data after one-time cross clustering.



**FIGURE 7.** Class-centered decision graphs and the distribution of non-classical multidimensional scaling transformations (non-classical MDS). (a) The class-centered decision graphs of two-dimensional data in Fig. 6 (a). (b) The non-classical MDS of two-dimensional data in Fig. 6 (a). (c) The class-centered decision graphs of synthetic control dataset in UCR. (d) The non-classical MDS of synthetic control dataset in UCR. (e) The class-center decision graph of synthetic control dataset in UCR after one-time cross clustering data transformation. (f) The non-classical MDS of synthetic control dataset in UCR after one-time cross clustering data transformation. (NotečžFor data set synthetic control, the accuracy of the nearest neighbor algorithm is 88%, and the accuracy of the nearest neighbor algorithm after one-time cross cluster data transformation is 99%).

distribution, as shown in Eq. (11).

$$\varphi(x) = \frac{1}{2\pi} e^{-\frac{x^2}{2}} (-\infty < x < +\infty) \quad (11)$$

We translate one of the mean values to point (3, 3) and the other is unchanged. Each two-dimensional positive distribution represents a type of data sample, as shown in Fig. 6 (a).

As shown in Fig. 6 (a), one point "class: A test" in the test set of the class of "A" will be classified in the class of "B"

by mistake due to the crossover of the distribution of data in these two classes when the nearest neighbor algorithm is applied.

Cross clustering is utilized in each class of the original data and the center point of each cluster is taken as a new sample point, which is shown in Fig. 6 (b). At this point, the data of test set "class: A test" will be correctly matched with the class of "A" in the new sample space because the

distribution of space of train data has changed. This kind of data compression transformation is essentially a process to reduce the outliers of the data which improve the accuracy of the model by using the transformed data. It can also be seen from Fig. 6 (b) that the distribution of data is significantly more aggregated after cross clustering data transformation.

In addition to determine whether the performance of the learner is improved after data compression transformation, it can also make an intuitive judgment through visualization technology to evaluate the effect of cross clustering data transformation. This visualization method is generally applicable to two-dimensional or three-dimensional. Fig. 7 depicts the decision graphs and the distribution of non-classical MDS in synthetic control dataset. The decision graph utilizes a method for detecting the center point proposed by ... [24] which can be used to observe the transformation of distribution of the cluster center before and after the cross clustering data compression. The non-classical multidimensional scaling is to transform the high-dimensional spatial data into the low-dimensional space and maintain the distance between the data. By using this method, the original high-dimensional data is reduced to the two-dimensional space aiming to visualize the distance between the data to explain the impact of cross clustering data transformation.

It is clear and prominent for the class center in Fig. 7 (a) and the data is clustered in the vicinity of the two classes in the space, indicating that the decision graph can accurately identify the class center. Fig. 7 (b) is a non-classical multidimensional scaling of raw data where only the rotation, flipping and translation occurred after the transformation of the similarity matrix compared with Fig. 6 (a) and the distribution of data is the same as the original data due to its 2-dimensional form. Fig. 7 (c) is the test set data of the synthetic control dataset in the UCR database that the sequence length is 60, which is divided into 6 categories and it has a total of 300 data instances. There are four obvious centers of data and two subclasses which cannot be clearly separated from the data, which may cause the nearest neighbor algorithm can not accurately classify the data.

Fig. 7 (d) is the non-classical MDS distribution of synthetic control where the 60-dimensional data is compressed into a 2-dimensional Euclidean space for visual display. Fig. 7 (e) is a class-centered decision graph for cross clustering of synthetic control and six class centers can basically recognize compared with the original data, which makes the nearest neighbor algorithm better classify. Fig. 7 (f) displays a non-classical MDS distribution map after cross clustering for synthetic control where the aggregation of data is significantly stronger than the original data.

## VI. CONCLUSIONS

This paper mainly proposes a cross clustering algorithm. The difference with traditional clustering algorithm is that it considers that an instance can belong to multiple classes or clusters that allows crossover between clusters. And it is also different from soft clustering for that it does not need

to calculate the membership degree but directly cluster by similarity distance between instances.

The proposed clustering algorithm effectively avoids the problem that the traditional clustering algorithm loses too many necessary features when extracting data representative points. This method extracts the representative elements of the training set in supervised learning, and uses these representative elements to train supervision aiming at improving the model performance. The experiments results showed that this transformation has a crucial effect on time series classification without changing the data dimension for that the accuracy of classification is improved. The time complexity of testing is reduced. In addition, this algorithm weakens the effect of over-fitting of FCN, improves the accuracy of network training and speeds up the network training at the same time.

Finally, a variety of ways has been used to visually explain how the performance of the classification model is improved. The most important is that we refer to an article [24] to use their cluster center visualization method to illustrate the data processed by our method. The result shows that the class center is more clear and obvious. Then we apply the non-classical multi-dimensional scaling algorithm to transform the data into two dimensions where we observe that the data aggregation is stronger.

The proposed cross clustering algorithm not only improves the clustering performance, but also points out an effective direction for time series classification. However, the setting of the distance threshold parameter in the cross clustering application still does not have an accurate selection method, which is a focus of future research. Moreover, this algorithm may be effective for other classification tasks or regression tasks, such as multi-agent systems [32] and streaming data [33], which is a problem worthy of further study.

## REFERENCES

- [1] S. Kausar, X. Huahu, I. Hussain, W. Zhu, and M. Zahid, "Integration of data mining clustering approach in the personalized E-learning system," *IEEE Access*, vol. 6, pp. 72724–72734, 2018.
- [2] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering—A decade review," *Inf. Syst.*, vol. 53, pp. 16–38, Oct./Nov. 2015.
- [3] J. C. Bezdek, W. Full, and R. Ehrlich, "FCM: The fuzzy c-means clustering algorithm," *Comput. Geosci.*, vol. 10, nos. 2–3, pp. 191–203, 1984.
- [4] T. Yu, J. Yang, and W. Lu, "Dynamic background subtraction using histograms based on fuzzy c-means clustering and fuzzy nearness degree," *IEEE Access*, vol. 7, pp. 14671–14679, 2019.
- [5] X. Li, J. Song, F. Zhang, X. Ouyang, and S. U. Khan, "MapReduce-based fast fuzzy c-means algorithm for large-scale underwater image segmentation," *Future Gener. Comput. Syst.*, vol. 65, pp. 90–101, Dec. 2016.
- [6] H. Li and X. Wan, "Classification for time series data based on center sequences of clusters," *J. Univ. Electron. Sci. Technol. China*, vol. 46, pp. 625–630, 2017.
- [7] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.
- [8] S. Hwang and S. Cho, "Clustering-based reference set reduction for k-nearest neighbor," in *Proc. ISNN*, Nanjing, China, 2007, pp. 880–888.
- [9] A. K. Jain, "Data clustering: 50 years beyond K-means," in *Proc. ICPR*, vol. 2008, Tampa, FL, USA, 2008, pp. 651–666.
- [10] C. Liu, J. Tan, H. Shi, and X. Wang, "Lithium-ion cell screening with convolutional neural networks based on two-step time-series clustering and hybrid resampling for imbalanced data," *IEEE Access*, vol. 6, pp. 59001–59014, 2018.

- [11] F. Petitjean, G. Forestier, G. I. Webb, A. E. Nicholson, Y. Chen, and E. Keogh, "Dynamic time warping averaging of time series allows faster and more accurate classification," in *Proc. IEEE ICDM*, Shenzhen, China, Dec. 2014, pp. 470–479.
- [12] Y.-H. Lee, C.-P. Wei, T.-H. Cheng, and C.-T. Yang, "Nearest-neighbor-based approach to time-series classification," *Decis. Support Syst.*, vol. 53, pp. 207–217, Apr. 2012.
- [13] E. Keogh and C. A. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowl. Inf. Syst.*, vol. 7, no. 3, pp. 358–386, 2005.
- [14] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *Proc. IJCNN*, Anchorage, AK, USA, 2017, pp. 1578–1585.
- [15] M. G. Baydogan, G. Rungger, and E. Tuv, "A bag-of-features framework to classify time series," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2796–2802, Nov. 2013.
- [16] P. Schäfer, "The BOSS is concerned with time series classification in the presence of noise," *Data Mining Knowl. Discovery*, vol. 29, no. 6, pp. 1505–1530, 2015.
- [17] P. Schäfer, "Scalable time series classification," in *Proc. ECMLPKDD*, Riva del Garda, Italy, 2016, pp. 1273–1298.
- [18] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing SAX: A novel symbolic representation of time series," *Data Mining Knowl. Discovery*, vol. 15, no. 2, pp. 107–144, 2007.
- [19] A. Bagnall, J. Lines, J. Hills, and A. Bostrom, "Time-series classification with COTE: The collective of transformation-based ensembles," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 9, pp. 2522–2535, Sep. 2015.
- [20] J. Lines and A. Bagnall, "Time series classification with ensembles of elastic distance measures," *Data Mining Knowl. Discovery*, vol. 29, no. 3, pp. 565–592, 2015.
- [21] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Transfer learning for time series classification," in *Proc. IEEE Big Data*, Seattle, WA, USA, Dec. 2018, pp. 1367–1376.
- [22] Y. Geng and X. Luo. (2018). "Cost-sensitive convolution based neural networks for imbalanced time-series classification." [Online]. Available: <https://arxiv.org/abs/1801.04396>
- [23] Y. Chen et al., "The UCR time series classification archive," Tech. Rep., Jul. 2015.
- [24] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.
- [25] X. Wu et al., "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2008.
- [26] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.
- [27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, vol. 37, Jul. 2015, pp. 448–456.
- [28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Las Vegas, NV, USA, 2016, pp. 770–778.
- [29] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, Dec. 2000.
- [30] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: <https://arxiv.org/abs/1412.6980>
- [31] R. Tang and S. Fong, "Clustering big IoT data by metaheuristic optimized mini-batch and parallel partition-based DGC in Hadoop," *Future Gener. Comput. Syst.*, vol. 86, pp. 1395–1412, Sep. 2018.
- [32] B. Wu, K. Hao, X. Cai, and T. Wang, "An integrated algorithm for multi-agent fault-tolerant scheduling based on MOEA," *Future Gener. Comput. Syst.*, vol. 94, pp. 51–61, May 2019.
- [33] W. Wang, K. Velswamy, K. Hao, L. Chen, and W. Pedrycz, "A hierarchical memory network-based approach to uncertain streaming data," *Knowl.-Based Syst.*, vol. 165, pp. 1–12, Feb. 2019. Accepted: Nov. 2018.



**WENSHUO ZHOU** received the B.S. degree in automation from the Shandong University of Science and Technology, Tai'an, China, in 2017. He is currently pursuing the M.S. degree in control science and engineering with the College of Information Sciences and Technology, Donghua University, Shanghai, China. His research interests include time series prediction method and deep learning.



**KUANGRONG HAO** (M'17) received the B.S. and M.S. degrees in mechanical engineering from the Hebei University of Technology, Tianjin, China, in 1984 and 1989, respectively, the M.S. degree in applied mathematics and computer sciences from the Ecole Normale Supérieure de Cachan, France, in 1991, and the Ph.D. degree from the Ecole Nationale des Ponts et Chaussées, Paris, France, in 1995. She is currently a Professor with the College of Information Sciences and Technology, Donghua University, Shanghai, China. She has published more than 180 technical papers, and five research monographs. Her scientific interests include intelligent perception, intelligent systems and network intelligence, robot control, and intelligent optimization of textile industrial process.



**CHUNLI JIANG** received the M.S. degree in control science and engineering from the Guangxi University of Science and Technology, Liuzhou, China, in 2017. She is currently pursuing the D.S. degree in control science and engineering with the College of Information Sciences and Technology, Donghua University, Shanghai, China. Her research interests include intelligent control, the Internet of Things technology, and multiobjective optimization methods.



**LEI CHEN** received the B.S. degree in electrical engineering and automation and the Ph.D. degree in control theory and control engineering from Jiangnan University, Wuxi, Jiangsu, China, in 2006 and 2014, respectively. From 2011 to 2013, she was a Visiting Student with the University of Alberta, Edmonton, AB, Canada. She is currently with the College of Information Sciences and Technology, Donghua University, Shanghai, China. Her research interests include process control, system identification, soft sensor, and state estimation.



**XUE-SONG TANG** received the B.S. degree in telecommunication engineering from Wuhan University, Wuhan, China, in 2006, the M.S. degree in telecommunication engineering from Melbourne University, Melbourne, VIC, Australia, in 2007, and the Ph.D. degree in computer science from Fudan University, Shanghai, China, in 2015. He is currently a Lecturer with the College of Information Sciences and Technology, Donghua University, Shanghai. His scientific interests include deep learning, machine vision, image processing, inter-discipline research on brain science, and artificial intelligence.



**XIN CAI** received the Ph.D. degree in control theory and control engineering from Shanghai Jiao Tong University, Shanghai, China, in 2015. He is currently a Lecturer with the College of Information Sciences and Technology, Donghua University, Shanghai. His research interests include model predictive control, performance analysis, and optimization of complex systems.