IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 9, 2019, accepted August 30, 2019, date of publication September 4, 2019, date of current version September 20, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2939481*

# An Economic Operation Analysis Method of Transformer Based on Clustering

**JUNDE CHEN, DEFU ZHANG, AND YASER AHANGARI NANEHKARAN**
School of Informatics, Xiamen University, Xiamen 361005, China

Corresponding author: Defu Zhang (dfzhang@xmu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Project 61672439, and in part by the Fundamental Research Funds for the Central Universities under Grant 20720181004.

**ABSTRACT** The economic operation of power transformers is analyzed in the present paper, which is performed by the clustering analysis method. In order to overcome the disadvantages of the conventional k-means algorithm lacking the stability and accuracy, we propose a novel boost k-means algorithm by optimizing the choice of initial cluster centers, and no additional parameters are required. The proposed approach outperforms the conventional approach in most experiments, for the best one, the accuracy of the proposed approach is 20.37% higher than that of the traditional approach. More importantly, empirical research is conducted in the paper. The index system reflecting the load characteristics of power transformers is established, and using the boost k-means algorithm, the economic operation analysis of power transformers is conducted. The clustering results of different transformers are obtained and the relevant suggestions are given as well. The empirical analysis results prove the validity of the proposed approach, and it can be efficiently applied for the economic operation analysis of transformers.

**INDEX TERMS** Index system, boost k-means, transformer, economic operation.

## I. INTRODUCTION

The power transformer is one of the most critical equipment in the power system, and the energy loss of the transformer occupies a large proportion in the distribution network. In general, the energy loss of the transformer takes possession of about 50% in the entire energy loss of area power system and about 10% of the total installed capacity [1]. Obviously, the economic operation of transformers plays an important role in the energy conservation of power system [2], [3]. Based on the actual load and combined with the parameter information of transformers, the traditional method mainly evaluates the economic operation of the transformer by the size of the load rate. This kind of method may depict the current operating state of the transformers, but it is difficult to reflect the historical situation and the future trend of the transformer load. Therefore, it is necessary to conduct a comprehensive load analysis for the power transformers, seek more scientific characteristic indicators which can reflect the load situation, and establish a data analysis model for the economic operation of the power transformers.

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Abdur Razzaque.

At present, many data mining techniques have presented promising performance to analyze the power transformer operation data. In the literature [4], [5], the fuzzy set methods are applied to monitor the transformer condition based on IEC/IEEE standards. The fuzzy method improves the identification accuracy of transformers and gives remarkable effects, whereas it needs to determine the input/output membership functions, diagnostic rules, and defuzzification. Additionally, the classification algorithms, particularly various artificial neural networks (ANNs), have been widely used in this study [6]–[9], because of their learning capability, parallel distributed process, recognition performance, and nonlinear classification ability. The limitations of ANNs are the required training process, determining a proper design, and network parameters assignment [10]. Clustering analysis is an important branch of data mining, and clustering algorithms play a critical role in data analysis, data mining and engineering signal processing, etc. Among them, k-means is one of the most popular in reality because of its simplicity and effectiveness [11], it is a partition-based and fast convergence speed clustering algorithm that can effectively handle large data [12], [13]. However, due to different settings of the parameters and random selection of initial cluster centers,

the conventional k-means algorithm is not stable, and it may produce different clustering partitions for the same dataset. Therefore, it is necessary to improve the performance of the algorithm before the economic operation analysis of power transformers.

This paper makes two main contributions. First, we improve the conventional algorithm and propose a boost k-means to optimize the choice of initial cluster centers. Through calculating the distance between different points and sets, the nearest points are classified as a set and the unclassified point is moved to the sets using the fast-moving approach, then the average value can be got as the initial center. The procedure is repeated until all the initial cluster centers are found. Moreover, the index system for the economic operation analysis of power transformers is established, and based on the load intensity and time change information, the characteristic indicators including *ALR*, *LRF*, *LRG,* etc. are extracted, which reflect the situation of load intensity, load dispersion, and load change trend separately. Thus, according to the characteristic indicators, the various transformers are clustered using the proposed boost k-means method, and the final results are obtained to determine whether the transformers are economically operated.

## II. PROBLEM DESCRIPTION

Generally, there are two major problems for the classical k-means algorithms, one is the selection of initial cluster centers, which is selected at random [14], the other is the determination of cluster number. The basic idea of this algorithm is to give a database $D$, the $k$ value of clusters is input by the user, at the beginning, the $D$ is divided into $k$ parts randomly, then the division is adjusted by updating the center of the clusters, when the global difference function converges, the process is ended. The sum function of squared errors is used as the objective function of k-means algorithms, as expressed in Eq. (1).

$$J_c = \sum_{i=1}^{k} \sum_{p \in C_i} \|p - M_i\|^2 \tag{1}$$

where $M_i$ is the mean value of the data in class $C_i$, $p$ is the point in class $C_i$.

By minimizing the objective function with multiple iterations, the cluster centers of the algorithm are constantly updated, that is, the algorithm is to find the clustering center vector $V = (v_1, v_2, \ldots, v_k)^T$ of categories to minimize the objective function $J_c$. The search direction of the objective function is always along the direction of decreased error square, and different initial values make the vector $V$ proceed in different paths, in this case, the final clustering result of the object depends largely on the initial partition or the choice of seed points. Therefore, the conventional k-means algorithm is sensitive to the initial clustering center in theory based on the above analysis.

There are also many research efforts focusing on the selection of initial cluster centers. For example, the k-means++ algorithm [15] is proposed to address the challenge, in this algorithm, only the first cluster center is randomly selected while the remainder initial cluster centers are selected as far as possible from the first point. However, random selection is still commonly used in practice [16]. Erisolgu *et al.* [17] proposed an incremental approach for computing initial cluster centers. In this approach, the reduced dataset is partitioned one by one until the number of clusters equals the predefined number of clusters. But the number of clusters must be known in advance, and how to get it is not given, which falls into the egg-chicken loop again. The others include using an optimization algorithm to fine-tune the initial cluster centers, such as in [18], the genetic algorithm (GA) is used for the selection. However, the parameters of optimization algorithms are much, which even exceed that of the k-means algorithm itself. Actually, it is considered that the algorithm should be free of parameters [19].

Additionally, the determination of the cluster number is the other problem for the analysis. The number of clusters in the data to be analyzed must be known in advance because many clustering algorithms require the number of clusters as an input parameter to run the algorithms [19]. However, the number of clusters that exist in real data is usually unknown. Therefore, a number is often guessed in practical cluster analysis, which often results in unsatisfactory results. Although several methods for estimating the number of clusters in data have been developed [20]–[24], they either produce incorrect results or are difficult to use in real applications. Thus, finding the correct number of clusters from real data remains a traditional problem in cluster analysis. It is also an active research topic.

## III. PROPOSED APPROACH

The basic purpose of clustering is to divide the data into a set of clusters in which the objects in the same clusters are close to each other, whereas the objects in different clusters are far from each other. Just as mentioned in Section II, the conventional k-means algorithm is sensitive to the initial cluster center and which makes the algorithm results unstable. So, by optimizing the choice of initial cluster center, the paper proposes a boost k-means algorithm to improve the conventional algorithm. This approach first finds the nearest points and separates them from the original dataset to form a new set, the other points in original dataset are moved to the formed set until the number of which reaches the maximum. Then, the processes are repeated until all the points are grouped to the sets, and the average values of all the sets are computed as the initial cluster centers of cluster algorithm. The main processes are listed as follows.

1. Set the total sample set as $U$. Calculate the distance between the sample pairs and sort them by distance, as computed in Eq. (2).

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2} \tag{2}$$

where $d(x, y)$ is the distance, $x$ and $y$ are any two points.

2. The closest points are found, which forms the sample set $A_m(1 \le m \le k)$, then they are removed from the total sample set.

3. Calculate the distance from the sample points in set $U$ to set $A_m$, and then the objective function is defined

$$\text{Min.}L_r = \text{Min.} \sum_{d_i \in C_r} d_i \tag{3}$$

Under the condition of $L_r$ reducing, the close points are merged into the set $A_m$ and removed from the set $U$. Particularly, the distance from a sample point $i$ to a set $C$ is defined as the following Eq. (4).

$$d(i, C) = \min(d(i, j), j \in C) \tag{4}$$

4. Repeat step 3 until the number of samples in $A_m$ reach a certain threshold $\varepsilon$, which can be computed using Eq. (5).

$$\varepsilon = N/K \tag{5}$$

where $N$ is the total number of samples, $k$ is the class number

5. If $m$ is less than $k$, then $m$ is equal to $m + 1$. Compute the closest points from the updated set $U$, which forms a new set and these points are deleted from the set $U$. Return to step 3 and implement.

6. The arithmetic mean values of the final $k$ sets is computed using Eq.(6), which can be used as the initial cluster centers and input to the k-means for the clustering.

$$c_i = \sum A_m/|A_m| \tag{6}$$

Generally, the details of boost k-means are presented in Algorithm 1.

Moreover, the convergence of the algorithm can be further analyzed. As stated before, giving the loss $L_r = \sum_{d_i \in C_r} d_i$, we have the objective function

$$\text{Min.} \sum_{x_i \in U} \|C_r - x_i\| \tag{7}$$

accordingly, Eq. (7) can be transformed as

$$\text{Min.}L = \frac{1}{2} \sum_{r=1}^{K} \sum_{i=1}^{N} (C_r - x_i)^2 \tag{8}$$

Then, pick $x_i$ in random ($x_i \in U$) and check $\Delta L$ when moving $x_i$ from set $S_u$ to $S_v$. When the value of $L$ is declined, this algorithm process is continuous, otherwise, it ends, as expressed in Eq. (9).

$$\gamma(x_n) = \begin{cases} 1, & if \arg\min_r \|x_n - C_r\|^2 \\ 0, & otherwise \end{cases} \tag{9}$$

In reality, the value of $\Delta L$ cannot decrease all along, so the algorithm is convergent. Additionally, the calculation and updating of the initial cluster center take the form:

$$c_i = \sum_n \gamma_{nr} x_n / \sum_n \gamma_{nr}, \tag{10}$$

which is also the optimized solution of the current objective function. Basically, this is a coordinate descent process, for every iteration, the value of the objective function will be decreased until it reaches a minimum. Thus, the algorithm process is convergent, as depicted in Figure 1.

**Algorithm 1:** Boost k-Means

**Input:**
*Data size: N.*
*No. threshold of each set: ε*
**Begin**
Calculate the pairs and the closest pairs is formed the set $A_m$.
$A_m$ is removed from total set $U$. Initialize $t = 0$.
**Repeat** {
$t \leftarrow t+1$
**For each** $x_j \in U$ do
$j^* \leftarrow argmin_i \{|| \leftarrow x_j C_i \leftarrow^2 ||\}$ *//Assign xj to the closest cluster.*
$A_m = A_m \cup \{x_j\}$, $|A_m| \le \varepsilon$ *//end for each.*
**For each** $i = 1$ *to* $k$ do
$c_i \leftarrow \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$ *//Get the initial centers as the input of km*
**Until** $\sum_{i=1}^{k} |C_i| \le N$ *//Until all the points are clustered.*
}
**Function** KM($k$, $c_i$) *//conduct k-means algorithm,*$c_i$ *is the*
k-means $\leftarrow c_i$ *do k-means // calculated centers,* $k = N/\varepsilon$.
**Output:**
The final clustering result.
**End.**

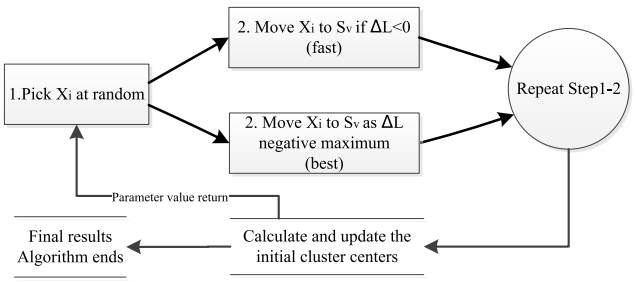



**FIGURE 1.** The fast and best convergence approach.

There will be two approaches for the point moving, the "fast" moving and "best" moving. If the value of the objective function is decreased ($\Delta L < 0$) when moving the point $x_i$ from one set to another, it is defined as the "fast" moving (Only the partial points outside of objective sets need to be compared.). On the other hand, for the "best" moving approach, it requires a negative maximum when moving the point $x_i$ from one set to another ($\Delta L < 0$ & $\Delta L = \min(\Delta L)$). In this case, all the points outside of the objective set need to be compared and the closest points are merged into the objective set.

For the approach of "fast" moving, once the arithmetic mean values of each set is calculated, we can use it as the initial cluster center of classical k-means and input to run the

algorithm further, the obtained results are the final results. For the approach of ''best'' moving, the partitioned categories can be regarded as the approximate clustering result and directly used as well. At present, the approach of ''fast'' move is employed in our practice. A brief example is illustrated as follows. Table 1 lists a randomly generated two-dimensional dataset, and Figure 2 shows its distribution.

**TABLE 1. Randomly generated two-dimensional dataset.**

| No. | 2-D data | No. | 2-D data |
|---|---|---|---|
| 1 | 89, 49 | 6 | 65, 44 |
| 2 | 61, 69 | 7 | 7, 10 |
| 3 | 27, 8 | 8 | 11, 52 |
| 4 | 58, 76 | 9 | 94, 40 |
| 5 | 93, 87 | 10 | 80, 96 |

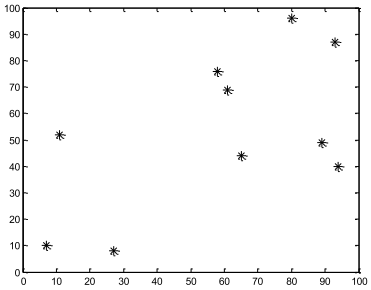


**FIGURE 2. The data distribution of table 1.**

Suppose that the data is divided into $k = 3$ classes, the possible number of isolated points $m$ is taken as 1, and the threshold of sample number is set to 4. Firstly, we can find that point 2 (61,69) and point 4 (58,76) are the closest points to each other. So, the point 2 (61,69) and point 4 (58,76) are selected to form the first sample set $A_1$, which is deleted from the total set $U$. Since the maximum number of samples in each sample set is set to 4, the search for the closest point from $U$ to $A_1$ will be continued, it is easy to know that the point 6(65,44) is the closest, so that the point 6(65,44) is added to set $A_1$ and removed from the set $U$. After removing 2, 4, and 6 points in the set $U$, the nearest points are the point 1(89, 49) and point 9(94, 40), which formed the set $A_2$, likewise, points 5(93, 87), 10(80, 96) are also added to the set $A_2$. Then, points 3(27,8), 7(7,10) formed the $A_3$, in this way, the divided 3 classes are separately obtained and the classification of this dataset is performed. What is more, the arithmetic mean of these sample sets can be computed respectively, and the initial cluster centers for all categories are obtained as (61, 63), (89, 68), (17, 9), thus, the new initial cluster center is generated, which is more consistent with the actual distribution of samples and achieves better clustering results.

## IV. EXPERIMENT ANALYSIS

UCI Machine Learning Repository [25] is an international general database for the algorithm test of machine learning, and the real-world datasets are downloaded from the UCI database. To verify the above points and evaluate the performance of the proposed approach, a lot of experiments were conducted on the downloaded UCI datasets. The algorithms were mainly implemented using *C* language in codeblocks tools, whereas the figures and partial algorithms were conducted in R, Matlab, and SPSS, etc.

### A. VERIFICATION TEST

The iris dataset is a feature set of plant flowers, each sample consists of 4 features: sepal length (SL), sepal width (SW), petal length (PL), and petal width (PW). In this dataset, there are 150 samples classified as 3 clusters and each cluster has 50 samples. Thus, we can define the number of samples as 150, the attribute dimension is $m = 4$, and the number of clusters is $k = 3$. Accordingly, the different initial cluster centers are employed to cluster the samples, in the experiments, the comparison of the final results presents that 91 of the 150 samples are with the different clustering category, so the difference rate is reached to 60.67%, which shows that the different initial cluster centers have a big impact on the final results. Table 2 displays the partial results.

**TABLE 2. Result comparison of different initial cluster centers.**

| SL | SW | PL | PW | CLASS | CAL1 | CAL2 |
|---|---|---|---|---|---|---|
| 5.1 | 3.8 | 1.6 | 0.2 | Iris-setosa | Cluster-2 | Cluster-3 |
| 4.6 | 3.2 | 1.4 | 0.2 | Iris-setosa | Cluster-2 | Cluster-2 |
| 5.3 | 3.7 | 1.5 | 0.2 | Iris-setosa | Cluster-2 | Cluster-3 |
| 5 | 3.3 | 1.4 | 0.2 | Iris-setosa | Cluster-2 | Cluster-3 |
| 7.0 | 3.2 | 4.7 | 1.4 | Iris- versicolor | Cluster-1 | Cluster-1 |
| 6.4 | 3.2 | 4.5 | 1.5 | Iris- versicolor | Cluster-3 | Cluster-1 |
| 6.9 | 3.1 | 4.9 | 1.5 | Iris- versicolor | Cluster-1 | Cluster-1 |
| 5.5 | 2.3 | 4 | 1.3 | Iris- versicolor | Cluster-3 | Cluster-1 |
| 6 | 2.2 | 5 | 1.5 | Iris-virginica | Cluster-3 | Cluster-1 |
| 6.9 | 3.2 | 5.7 | 2.3 | Iris-virginica | Cluster-1 | Cluster-1 |
| 5.6 | 2.8 | 4.9 | 2 | Iris-virginica | Cluster-3 | Cluster-1 |
| 7.7 | 2.8 | 6.7 | 2 | Iris-virginica | Cluster-1 | Cluster-1 |

The initial cluster centers used in these two groups are presented as follows.

1#: (5.1, 3.5, 1.4, 0.2), (4.7, 3.2, 1.3, 0.2), (4.9, 3.0, 1.4, 0.2)

2#: (7.7, 3.0, 6.1, 2.3), (4.4, 3.2, 1.3, 0.2), (5.7, 4.4, 1.5, 0.4)

Then, the good result of the two groups is selected to compare with the actual categories, there are 16 cluster errors in the 150 sample data, and the error rate is 10.67%. In the same way, the wine dataset is tested as well. This dataset is a chemical composition analysis for the wine brewed from three different cultivated plants in the same region of Italy. It contains 178 sample records, 13 chemical species and 3 different types of raw material cultivation, so the attribute dimension is set $m = 13$ and the number of clusters is set $k = 3$. For the results of two initial cluster centers, there are 89 different samples, the difference rate is 50%. Likewise, the good result of the two groups is selected to compare

**TABLE 3.** The tested UCI dataset and the experimental results.

| Dataset | No. of Samples | No. of Features | No. of Clusters | K-means | Boost KM | Difference | Rank |
|---|---|---|---|---|---|---|---|
| Iris | 150 | 4 | 3 | 89.33% | 90.67% | 1.34% | 2 |
| Wine | 178 | 13 | 3 | 57.30% | 70.22% | 12.92% | 5 |
| Balance | 625 | 4 | 3 | 51.62% | 54.86% | 3.24% | 3 |
| Diabetes | 768 | 8 | 2 | 66.28% | 65.76% | -0.52% | 1 |
| Musk | 6598 | 166 | 2 | 53.99% | 74.36% | 20.37% | 7 |
| Pegdigits | 10992 | 16 | 10 | 58.39% | 72.53% | 14.14% | 6 |
| Skin Seg. | 245057 | 3 | 2 | 53.02% | 59.78% | 6.76% | 4 |

with the actual categories, there are 76 cluster errors in the 178 samples, and the error rate is 42.69%.

From the above verification experiments, we can see that the initial cluster centers have a great impact on the final outputs of the algorithms, and the results obtained by various initial cluster centers are quite different. The initial cluster centers are selected at random, which makes the results of the conventional k-means algorithm unstable and affects the accuracy of the algorithm.

### B. COMPARISON EXPERIMENT

To investigate the performance of the proposed algorithms by experiments, seven datasets including Balance, Wine, Musk, Pegdigits, Skin segmentation (seg.), etc. are downloaded from UCI database and used in the experiments. They are frequently used as the benchmark datasets in probing the performance of different algorithms.

The Balance dataset is generated to model psychological experimental results. Each example is classified as having 3 categories: the balance scale tip to the right, tip to the left, or be balanced. There are 4 attributes and 625 samples (49 balanced, 288 left, 288 right) in this dataset.

The Diabetes dataset is obtained from two sources: an automatic electronic recording device and paper records. There are 768 samples divided into 2 clusters in this dataset, and each sample is determined by 8 features.

The Musk dataset consists of 102 molecules (39 musks and 63 non-musks). Musk is a molecule that binds to a target protein. The 102 molecules consist of 6,598 data each of which is represented by a 166-dimensional feature vector derived from their surface properties.

There are 10,992 samples collected in the Pendigits dataset, which represents handwriting digits written by 44 writers. The 10,992 handwriting digits are categorized into 10 clusters with respect to digits between 0 and 9 and each instance is described by 16 features.

The Skin seg. dataset is collected by randomly sampling B, G, R values from face images of various age, gender, and race groups. The dataset is composed of 245,057 samples classified into 2 clusters (50,859 skin samples and 194,198 non-skin samples), and each sample contains 3 features. Table 3 displays the information about the seven datasets.

According to the statistic variables of correct detections (also known as true positives), misdetections (also known as false negatives) and false positives, the accuracy is the most commonly used indicator to evaluate the performance of a data clustering system. It is defined as follows.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where TP(true positive) is the number of instances that actually belong to cluster $C$ and are correctly classified by the clustering algorithm. FP (false positive) is the number of instances that do not belong to cluster $C$ but mistakenly classified as this cluster. TN (true negative) is the number of instances which are not in cluster $C$ in reality, and they are correctly classified. FN (false negative) is the number of data which are in cluster $C$ in reality, but they are incorrectly classified as others.

Thus, for the proposed method, the comparative experiments are performed on the above testing datasets. The 10,992 handwriting digits of Pendigits dataset, which are accurately clustered by the k-means with 6418 instances, the accuracy rate is 58.39%. In contrast, the correct clustering number of boost k-means on Pendigits dataset is 7972, which is more than that of the conventional k-means. The accuracy of boost k-means is 72.53%.

Similarly, considering the 6,598 samples of Musk dataset, 3,562 samples are classified correctly by the conventional k-means algorithm, the correct rate is 53.99%, and the 4,906 samples are classified correctly by the boost k-means (KM) algorithm, the correct rate is 74.36%. Figure 3 presents the results of confusion matrices on different datasets.

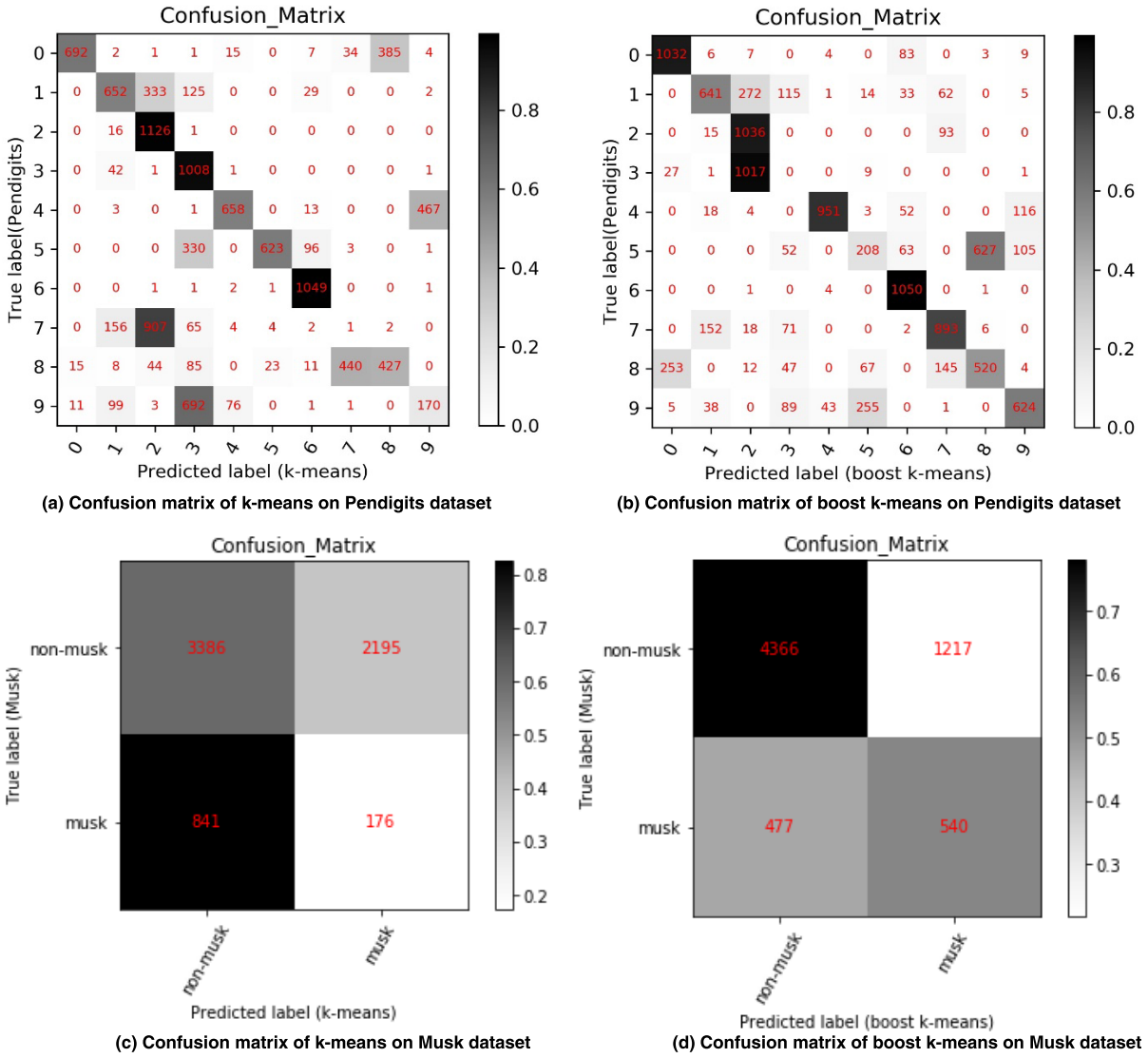
The correct classifying number of the Wine dataset is 102 for the conventional k-means algorithms, the accuracy rate is 57.30%. By contrast, for the boost k-means, the correct number of the dataset is 125 and accuracy is 70.22%., the calculated initial cluster center are:

(13.55, 2.05, 2.43, 17.89, 106.44, 2.64, 2.60, 0.32, 0.32, 5.38, 1.02, 2.98, 1033.53)

(12.67, 2.53, 2.37, 21.08, 95.37, 1.94, 1.50, 0.41, 0.41, 5.02, 0.87, 2.29, 558.78)

(12.88, 2.49, 2.41, 20.08, 103.15, 2.14, 1.66, 0.37, 0.37, 5.28, 0.90, 2.44, 707.02)

In the same way, the experiments are conducted on the other UCI datasets and all the results are displayed in Table 3.




(a) Confusion matrix of k-means on Pendigits dataset

(b) Confusion matrix of boost k-means on Pendigits dataset

(c) Confusion matrix of k-means on Musk dataset

(d) Confusion matrix of boost k-means on Musk dataset

**FIGURE 3.** The results of confusion matrices on different datasets.

As seen from the above Table 3, the accuracy results on the seven datasets for the traditional k-means algorithm and boost k-means algorithm are given separately. Thus, the method of statistic test can be used to determine whether there is a significant difference between the two algorithms. Wilcoxon signed-ranks test [26], [27], which is one of the most widely used statistical tests in behavioral studies, is a nonparametric statistical test. It makes weaker assumptions about the distribution of data than other groups of statistical tests (for example, tests based on the normal distribution include t-test, ANOVA and linear regression, etc.) [28]. Because of these features, the Wilcoxon test has been widely used in many fields, especially in algorithm comparison analysis. The brief description of the test process is illustrated as follows.

Let $d_i$ be the difference of clustering performance between the two algorithms on the $i$-th dataset, and arrange the absolute values of their difference from small to large (Take the average value if the rank is the same.), then the rank of each dataset is calculated. Let $R+$ represent that the sum of rank for the first algorithm is better than that of the second one, and R- is on the contrary, as expressed in Eq. (12).

$$R^+ = \sum_{d_i>0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i)$$
$$R^- = \sum_{d_i<0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i) \tag{12}$$

The above seven datasets are calculated below.

$$R^+ = 2+5+3+7+6+4 = 27$$
$$R^- = 1 \tag{13}$$

Let $T = \min(R^+, R^-)$, it is easy to know the $T$ value is 1. Then, the critical value table of the Wilcoxon test is checked, and under the condition of $a = 0.05$, the difference between

**TABLE 4.** The calculation time of initial cluster centers (s).

| Iris | Wine | Balance | Diabetes | Musk | Pegdigits | Skin Seg |
|---|---|---|---|---|---|---|
| 0.155 | 0.237 | 0.475 | 0.646 | 6.611 | 9.676 | 13.665 |

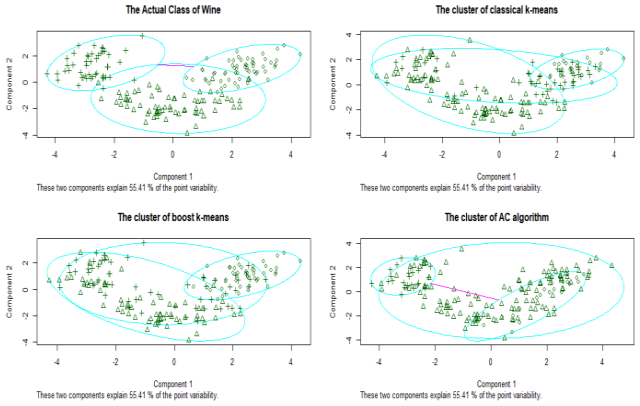



**FIGURE 4.** The clustering chart of algorithms.

the algorithms is significant when $T$ is less or equal to 2 ($T \leq 2$). Here, the $T$ value satisfies the condition and it is less than the critical value. Therefore, the result of the statistical test rejects the null hypothesis, indicating the significant differences between the two algorithms. It means that the boost k-means outperforms the conventional algorithm in a statistical sense. Besides, there are six datasets for the boost k-means is superior to the conventional k-means, in terms of quantity, it is more than that of the conventional algorithm. On the other hand, the calculation time of initial cluster centers for the different datasets is displayed in Table 4.

Furthermore, to compare with the other algorithms, the Analog Complexing (AC) cluster algorithm [29] is selected in our experiments. This algorithm assumes each sample as a pattern, by computing the similarity between patterns, the more similar patterns are grouped into one class, and the less similar patterns are classified into different classes. The earliest application of AC algorithm in clustering analysis was by Lemke *et al.* [30] and Ivakhnenko [31], and many good effects were obtained for this algorithm after continuous development and improvement. Therefore, with the conventional k-means and proposed boost k-means together, the AC algorithm is implemented in the experiments using the KnowledgeMiner software [29]. Figure 4 depicts the clustering effect diagram of each algorithm on the wine dataset. For the conventional k-means and boost k-means, the clustering number was set k $=$ 3 and the maximum iteration number was 10. The AC algorithm was computed with 95% similarity, and a total of 93 categories were clustered. Considering the actual categories, there were total 3 categories for this dataset, the categories 1 and 81 with the largest number were corresponded to the actual first and third class respectively, the other categories were processed as the second class. The detailed results of this phase are described in subsequent sections.

It can be seen from the above figure that the performance of boost k-means is most consistent with the actual category status, and the number of clustered category that matches the real category is 125, the matching rate is 70.22%. The AC algorithm determines the clustering number automatically, but the clustering results are too detailed, the matched number of this algorithm is 103, the matching rate is 57.87%. For the conventional k-means algorithm, the matched number is 102 and the matching rate is 57.30%. In addition to the comparison with the actual category, the compactness and separation validity function [32] can be used to evaluate the clustering results, as defined in Eq. (14).

$$s(U,k) = \frac{\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}(u_{ij})^2 \left|x_i - c_j\right|^2}{\min\limits_{p,q=1,2,\ldots,k} \left|c_p - c_q\right|^2} \tag{14}$$

where the $c_p$, $c_q$, and $c_j$ refer to the cluster centers, $x_i$ is any sample in the dataset, $k$ is the number of class and $U$ is the sample set. Then, the Xie-Beni ($XB$) index is used for the evaluation of cluster effects, as expressed in Eq. (15).

$$XB = \max_{k}\{\max_{\Omega} S(U,k)\}, \quad k = 2, 3, \ldots, n-1 \tag{15}$$

The $S(U,k)$ is the ratio of the average distance between data objects and their corresponding cluster centers to the minimum distance of cluster centers, in principle, when the $S(U,k)$ is smaller, the clustering quality will be higher. Table 5 displays the calculated XB values of the above algorithms.

**TABLE 5.** The XB evaluation value.

| Algorithms | Iterations/ Similarity | Category Samples | | | XB |
|---|---|---|---|---|---|
| | | C1 | C2 | C3 | |
| Classical k-means | 8 | 27 | 102 | 49 | 0.70666 |
| Boost k-means | 4 | 47 | 69 | 62 | 0.70452 |
| AC algorithm | 95% | 27 | 126 | 25 | 2.16944 |

From the $XB$ values calculated in Table 5, it can be seen that the differences between the algorithms are remarkable. The $XB$ value of AC algorithm is the largest while that of boost k-means is the smallest, which indicates that the boost k-means outperforms the other algorithms in the experiment. Therefore, based on the cluster analysis diagram, the statistical matching results with actual categories, and the $XB$ evaluation indicator, our method shows a significant performance comparing with the conventional algorithm and another method. Experimental results verify the effectiveness and superiority of the proposed method. Thus, it can be finally applied in the empirical analysis.

## V. EMPIRICAL ANALYSIS AND DISCUSSION

### A. BUILDING INDEX SYSTEM

In order to evaluate the operation status of power transformers effectively, it is necessary to sort out the characteristic indicators reflecting the economic operation of transformers.

Initially, the raw data are collected. Including the maximum load of transformers, installation capacity, voltage grade, and others, the relevant variables are selected for the analysis, and the data preprocessing works such as checking for abnormal data, imputation of missing data, revision of holiday data, etc. are performed. Then, using the index calculation formula to calculate the characteristic indicators, the index system is established. On the basis of this, the model is built and the results are analyzed in depth. Overall, according to the information contained in the transformer records, the following characteristic attributes are collected.

1. The load intensity information. This can be obtained from the daily maximum load and the capacity parameters of the transformers.

2. The time change information. It reflects the time-series change of analysis variables, and which is obtained from the data collecting time.

Thereby, considering the load intensity and time change information, the current load level of the transformer can be divided into three levels: high, medium and low. Relatively, the load trend is classified as fast, slow and stable separately. In this way, a Nine-square grid map is formed, as depicted in Figure 5. Thus, there are 9 categories in total, and the number of clusters can be set to 9 accordingly.

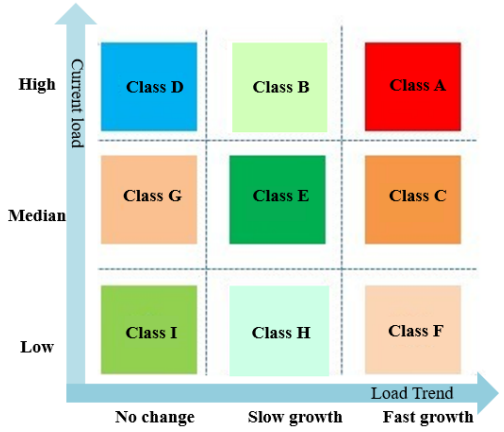



**FIGURE 5.** The nine-square grid map.

In this figure, the class A shows that the current load and load trend are both high, which reflects the load of this category is large and the load trend change is on the rise. Therefore, this category needs to be focused and expanded capacity when necessary. Similarly, the situation of class I is exactly the opposite, both the current load and load trend are low, so it is recommended to reduce capacity if necessary, class E is the economic operation category currently.

Moreover, the transformer load rate is the ratio of the actual maximum load to the transformer load volume, and from which three variables reflecting the economic operation rule of the transformers are derived. These three variables are the average load rate (*ALR*), load rate fluctuation (*LRF*) and load rate gradient (*LRG*) respectively, which are calculated as follows.

#### 1) THE CALCULATION OF AVERAGE LOAD RATE

The average load rate is defined using Eq. (16).

$$ALR = \frac{1}{m}\sum_{j=1}^{m}[MLR_{i1}]_j, \tag{16}$$

where $j = 1, 2, 3, \ldots, 365$, and $m$ represents the number of days, $MLR_{i1}$ is the load ratio of the $i$-th transformer. This method is simple and the result can be directly calculated according to the transformer load record.

#### 2) THE CALCULATION OF LOAD RATE FLUCTUATION

Referring to the formula

$$LRS = \sqrt{\frac{1}{m}\sum_{j=1}^{m}\left([MLR_{i1}]_j - ALR\right)^2}, \tag{17}$$

the standard deviation of load rate can be calculated, where the *LRS* denotes the standard deviation, $j = 1, 2, 3\ldots, 365$, $m$ is the number of days, $MLR_{i1}$ is the load ratio of the $i$-th transformer, *ALR* is average load rate. Thus, based on the average load rate and standard deviation of load rate, the load rate fluctuation can be calculated in Eq. (18).

$$LRF = LRS/ALR \times 100\% \tag{18}$$

The load rate fluctuation is interpreted as the degree of load dispersion per unit average load, which reflects the relative size of load rate dispersion.

#### 3) THE CALCULATION OF LOAD RATE GRADIENT

The load rate gradient is calculated as

$$LRG = (ALR_2 - ALR_1)/ALR_1 \tag{19}$$

where *LRG* represents the load rate gradient, $ALR_1$ is the average load rate in the first half period of the transformers, $ALR_2$ is the average load rate in the second half period of the transformers. The load rate gradient reflects the changing trend of transformer load rate, if the value of load rate gradient is 1, it implies that the trend of transformer load rate is getting larger, and this situation is probably due to the increase of user's power load. If the load rate gradient value is 0, it means that the trend of the transformer load rate is getting smaller, and this situation may be due to the decreasing power load of users.

### B. MODEL CALCULATION

After the index system reflecting the load characteristics of the transformers is established, it can be further analyzed. For example, if the statistical period is one year, the daily average load rate can be obtained as $\overline{MLR^j_{i1}} = \sum_{j=1}^{365} MLR^j_{i1}/365$, and when the transformer load rate is $\overline{MLR^j_{i1}} \in [30\%, 70\%]$, the operation state of the transformer is economical. However, when the transformer load rate is $\overline{MLR^j_{i1}} < 30\%$, it means that the transformer is not running economically. Additionally, when the transformer load rate is $\overline{MLR^j_{i1}} > 70\%$, it implies that the load of the transformer is high. In this case, the transformer is easy to be damaged, and the factors should be

further investigated to improve the operation economy of the transformer.

Based on the establishment method of index system mentioned in the previous section, the data of special transformers are extracted and the relevant characteristic indicators including average load rate, load rate fluctuation and load rate gradient are calculated separately. The partial data are displayed in Table 6.

**TABLE 6. Calculation of the characteristic index and clustering result.**

| Transformer_id | *ALR* | *LRF* | *LRG* | Cluster |
|---|---|---|---|---|
| 66_0319_03_1 | 40.19 | 33.60 | -5.85 | 5 |
| 66_0319_03_3 | 46.71 | 34.57 | -4.44 | 5 |
| 1169_0319_03_2 | 1753.24 | 26.69 | 25.41 | 1 |
| 10_0319_03_2 | 70.66 | 22.18 | 9.80 | 5 |
| 1171_0319_03_1 | 1126.22 | 27.41 | 14.38 | 6 |
| 1185_0319_03_1 | 1903.44 | 57.80 | 21.58 | 1 |
| 1195_0319_03_1 | 232.58 | 34.31 | 48.33 | 2 |
| 21_0319_03_3 | 56.53 | 26.14 | 3.53 | 5 |

After performing the data preprocessing, the feature samples are clustered by the boost k-means, and the parameter of this algorithm is set as: the distance function is applied as the Euclidean distance, the maximum number of iterations is 500, the number of clusters is set to 9, the number of seeds is 10. Thus, the clustering results are obtained as the last column of Table 6.

Furthermore, the category 5 is randomly selected for the economic operation analysis, and the frequency density of the three characteristic variables is depicted in Figure 6. It can be observed from this figure that the average load rate of category 5 is mainly concentrated between 30% and 70%, which reflects that the transformer operation is economical and ideal. The load rate fluctuation is concentrated at 20% to 40%, which means the transformer operation is smooth and the power supply is normal. The load rate gradient is basically between -10% and 30%, and most of which is concentrated at 0 to 30%, it implies that the load trend is relatively stable, and some of the power supply load is increased, but the increased margin is not big.

For category 5, the distribution of the average load rate is depicted in Figure 7.

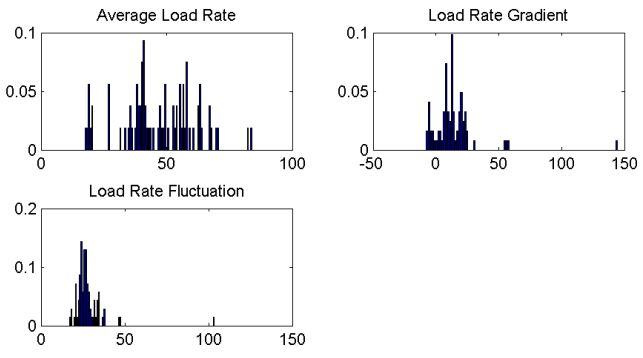



**FIGURE 6. The chart of frequency density.**

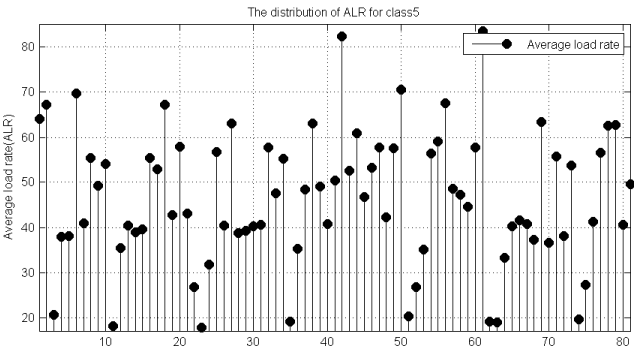



**FIGURE 7. The distribution chart of ALR for category 5.**

As stated before, the *ALR* (average load rate) range of the economic operation for power transformers is in [30%,70%], it is uneconomical when *ALR* is less than 30%, while overloaded when *ALR* is higher than 70%. According to this rule, most of the transformer samples in category 5 are within this range, it indicates that most of them are economical to operate and in good working condition. However, the transformers with less than 30% *ALR* still account for a certain proportion, which is not economical, and maybe the electricity consumption of users is decreasing. It is recommended to observe closely for a period of time, and if necessary, the capacity reduction can be suggested. In addition, as for the *ALR* of individuals is more than 70%, the overall load is higher, it is recommended to pay attention to these sites, and the capacity expansion is considered when necessary.

Similarly, the same analysis is performed on the other categories. After establishing index system and applying the boost k-means clustering on the load analysis of transformers, the overall operation characteristics and optimization suggestions are given, which provides an effective way for the economic operation analysis of power transformers.

## VI. CONCLUSION

Accurate characteristic parameters are the key factors to analyze the operation situation of transformers, which often affect the final analysis results. Considering the load intensity and time change information, the variables including the average load rate (*ALR*), load rate fluctuation (*LRF*) and load rate gradient (*LRG*) are extracted as the characteristic indicators of transformers. A characteristic analysis method for the economic operation of transformers is developed, and the specific calculation processes are given, so the feature engineering is established. Meanwhile, to further monitor the operation status of transformers efficiently, the data mining techniques are introduced into the analysis, in particular, the clustering analysis method is discussed in theory and the relevant literature is reviewed. Then, on the basis of the above, the boost k-means is proposed in the paper and the comparative experiments are performed as well. After that, the empirical research is conducted in our work. The performance of experimental analysis was remarkably improved by the calculated

initial cluster center using our algorithm. The clustering results are stable and the error is small, which overcomes the shortcomings of traditional algorithms. Additionally, through considering the load intensity and time change information, the number of clusters is determined according to the feature analysis results, which shows certain theoretical and practical significance. In the future, we will explore the possibility of developing a general method to automatically determine parameters and make it usable in real applications.

## ACKNOWLEDGMENT

The authors would like to thank the project chance provided by Guangxi Electric Power Research Institute and thank Mr. Zhang Liang-jun, the chairman of Guangzhou TipDM Intelligent Technology Company Ltd., for valuable discussion and contribution to the successful delivery of the project. They would also like to thank all the editors and anonymous reviewers for their constructive advice.

## REFERENCES

[1] Y. Weipeng and Z. Yao, "Economic operation of transformers in the area power network based on real-time analysis and control," in *Proc. China Int. Conf. Electr. Distrib.*, Guangzhou, China, Dec. 2008, pp. 1–5.

[2] M. Dong, H. Zheng, Y. Zhang, K. Shi, S. Yao, X. Kou, G. Ding, and L. Guo, "A novel maintenance decision making model of power transformers based on reliability and economy assessment," *IEEE Access*, vol. 7, pp. 28778–28790, 2019.

[3] W. Zhang, T. Li, and X. Yuan, "Study on a united evaluation algorithm and its applications for economic operation of transformer," *Energy Procedia*, vol. 16, pp. 2073–2080, Jan. 2012.

[4] A. D. Ashkezari, H. Ma, T. K. Saha, and C. Ekanayake, "Application of fuzzy support vector machine for determining the health index of the insulation system of in-service power transformers," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 20, no. 3, pp. 965–973, Jun. 2013.

[5] Z. Xuewei and L. Hanshan, "Research on transformer fault diagnosis method and calculation model by using fuzzy data fusion in multi-sensor detection system," *Optik*, vol. 176, pp. 716–723, Jan. 2019.

[6] Y.-C. Huang, "A new data mining approach to dissolved gas analysis of oil-insulated power apparatus," *IEEE Trans. Power Del.*, vol. 18, no. 4, pp. 1257–1261, Oct. 2003.

[7] A. J. X. Guo and F. Zhu, "Spectral-spatial feature extraction and classification by ANN supervised with center loss in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1755–1767, Mar. 2019.

[8] R. J. Haddad, B. Guha, Y. Kalaani, and A. El-Shahat, "Smart distributed generation systems using artificial neural network-based event classification," *IEEE Power Energy Technol. Syst. J.*, vol. 5, no. 2, pp. 18–26, Jun. 2018.

[9] W.-M. Lin, "Transformer-fault diagnosis by integrating field data and standard codes with training enhancible adaptive probabilistic network," *IET Proc.-Gener. Transmiss. Distrib.*, vol. 152, no. 3, pp. 335–341, May 2005.

[10] R. M. A. Velásquez, J. Vanessa, M. Lara, and A. Melgar, "Converting data into knowledge for preventing failures in power transformers," *Eng. Failure Anal.*, vol. 101, pp. 215–229, Jul. 2019.

[11] Y.-K. Lam, P. W. M. Tsang, and C.-S. Leung, "PSO-based *K*-Means clustering with enhanced cluster matching for gene expression data," *Neural Comput. Appl.*, vol. 22, nos. 7–8, pp. 1349–1355, Jun. 2013.

[12] S. Wang, M. Li, N. Hu, E. Zhu, J. Hu, X. Liu, and J. Yin, "*K*-Means clustering with incomplete data," *IEEE Access*, vol. 7, pp. 69162–69171, 2019.

[13] J. Feng, Y. Zhang, G. Yue, X. Liu, H. Su, and P.-F. Zhang, "Atherosclerotic plaque pathological analysis by unsupervised *K*-Means clustering," *IEEE Access*, vol. 6, pp. 21530–21535, 2018.

[14] H. Jw, *Data Mining: Concepts and Techniques*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann, 2006, pp. 383–386.

[15] D. Arthur and S. Vassilvitskii, "*k*-Means++: The advantages of careful seeding," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, New Orleans, LA, USA, Jan. 2007, pp. 1027–1035.

[16] M. A. Masud, J. Z. Huang, C. Wei, J. Wang, I. Khan, and M. Zhong, "I-nice: A new approach for identifying the number of clusters and initial cluster centres," *Inf. Sci.*, vol. 466, pp. 129–151, Oct. 2018.

[17] M. Erisoglu, N. Calis, and S. Sakallioglu, "A new algorithm for initial cluster centers in *k*-Means algorithm," *Pattern Recognit. Lett.*, vol. 32, no. 14, pp. 1701–1705, Oct. 2011.

[18] W. Kwedlo and P. Iwanowicz, "Using genetic algorithm for selection of initial cluster centers for the *K*-means method," in *Proc. Int. Conf. Artif. Intell. Soft Comput. (ICAISC)*, 2010, pp. 165–172.

[19] P. Fränti and P. SamiSieranoja, "How much can *k*-means be improved by using better initialization and repeats?" *Pattern Recognit.*, vol. 93, pp. 95–112, Sep. 2019.

[20] R. C. de Amorim and C. Hennig, "Recovering the number of clusters in data sets with noise features using feature rescaling factors," *Inf. Sci.*, vol. 324, pp. 126–145, Dec. 2015.

[21] C. Hennig and T. F. Liao, "How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification," *J. Roy. Stat. Soc. C, Appl. Stat.*, vol. 62, no. 3, pp. 309–369, May 2013.

[22] C.-W. Tsai, W.-L. Chen, M.-C. Chiang, "A modified multiobjective EA-based clustering algorithm with automatic determination of the number of clusters," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2012, pp. 14–17.

[23] L. Wang, C. Leckie, K. Ramamohanarao, and J. Bezdek, "Automatically determining the number of clusters in unlabeled data sets," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 3, pp. 335–350, Mar. 2009.

[24] Y. Ye, J. Z. Huang, X. Chen, S. Zhou, G. Williams, and X. Xu, "Neighborhood density method for selecting initial cluster centers in *K*-means clustering," in *Proc. 10th Pacific-Asia Conf. Knowl. Discovery Data Mining (PAKDD)*. Berlin, Germany: Springer-Verlag, 2006, pp. 189–198.

[25] C. J. Merz. (1996). *UCI Repository of Machine Learning Database*. [Online]. Available: https://www.ics.uci.edu/mlearn/MLRepository.html

[26] L. Deng, J. Pei, J. Ma, and D. L. Lee, "A rank sum test method for informative gene discovery," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2004, pp. 410–419.

[27] S. Li, X. Wu, and X. Hu, "Gene selection using genetic algorithm and support vectors machines," *Soft Comput.*, vol. 12, no. 7, pp. 693–698, May 2008.

[28] E. EiitiKasuya, "Wilcoxon signed-ranks test: Symmetry should be confirmed before the test," *Animal Behav.*, vol. 3, no. 79, pp. 765–767, 2010.

[29] F. Lemke and J. A. Müeller, "Self-organising data mining," *Syst. Anal. Model. Simul.*, vol. 43, no. 2, pp. 231–240, 2010.

[30] F. Lemke, J.-A. Müller, and F. List-Platz, "Self-organizing data mining based on GMDH principle," *Mathematik*, 2019. [Online]. Available: https://pdfs.semanticscholar.org/a86d/dc1aaa3a68c37f0ad32be1f41cc1d66481a3.pdf

[31] A. G. Ivakhnenko, "Sorting methods in self-organization of models and clusterizations (review of new basic ideas)-iterative (multirow) polynomial GMDH algorithms," *Soviet J. Automat. Inf. Sci.*, vol. 22, no. 4, pp. 1–12, 2002.

[32] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *J. Intell. Inf. Syst.*, vol. 17, nos. 2–3, pp. 107–145, Dec. 2001.

**JUNDE CHEN** received the bachelor's degree from Xiangtan University, in 2004, and the master's degree from Sichuan University, in 2010. He is currently pursuing the Ph.D. degree with the School of Informatics, Xiamen University. His research interests include the aspects of data mining and image processing.

**DEFU ZHANG** is currently with the School of Informatics, Xiamen University. He has published articles in the following journals: the *INFORMS Journal on Computing*, *Computers & Operations Research*, the *European Journal of Operational Research*, and *Expert System with Applications*. His research interests include all aspects of computational intelligence, image analysis, and data mining.

**YASER AHANGARI NANEHKARAN** received the B.E. degree in power electrical engineering from the IAU of Ardabil Branch, Ardabil, Iran, and the M.Sc. degree in IT from Cankaya University, Ankara, Turkey. He is currently pursuing the Ph.D. degree with the Department of Computer Science, Xiamen University, Xiamen, China. His research interests mainly include data mining, big data, and deep learning techniques.