# Clustering Center Optimization under-Sampling Method for Unbalanced Data

Haitao Li, Mingjie Zhuang*
College of Engineering, Huaqiao University , Quanzhou, 362021, Fujian, China.

* Corresponding author. Tel.: +86 13799761631; email: haitaoli51123@163.com; mjzhuang176@163.com
Manuscript submitted February 19, 2020; accepted April 20, 2020.
doi: 10.17706/jsw.15.3.74-85

**Abstract:** When the number of data in one class is significantly larger or less than the data in other class, under learning algorithm for classification, a problem of learning generalization occurs to the specific class and this is called imbalanced data problem. In this paper, a method of under-sampling based on the optimization cluster center selection (BCUSM) is proposed. First of all, the cluster center selection of K-means clustering algorithm is optimized, the initial cluster center is obtained by calculation, instead of random selection. The optimized method is called OICSK-means. And then use it to cluster the negative samples by setting the same number of clusters as positive samples. According to the cosine similarity, select the most similar samples from each cluster with cluster centers as the negative training samples, and a new training set is established with the positive samples. Finally, training with a new training set. This work selected some data from the UCI database of the University of California, Irvine, and used the support vector machine (SVM) classifier for experimental simulation, and compared the classification effects of this method with other four methods such as synthetic oversampling method (SMOTE). The experimental results demonstrate that the BCUSM has certain effectiveness.

**Key words:** Clustering, unbalanced data, under-sampling, classification.

## 1. Introduction

In the unbalanced data classification, the larger number of samples is usually referred to as negative classes, and the smaller number of samples is referred to as positive classes. For some traditional machine learning classification methods such as support vector machine (SVM) and Naive Bayes, class imbalance causes the classifier to pay too much attention to the negative class samples and ignore the positive class samples, which leads to the classification effect of positive classes is poor, and in practical applications, people will pay more attention to those positive samples, because the misclassification cost of positive samples is often higher than the misclassification cost of negative samples. For example, judging a hacker's intrusion as a normal visit will lead to extremely serious cybersecurity incidents, and if a doctor mistakes a cancer patient for a normal person, it may lead the patient to miss the best treatment opportunity [1], [2]. Therefore, for the unbalanced data classification task, designing and implementing an effective unbalanced data processing algorithm has important practical significance.

In recent years, scholars have proposed many solutions to the classification of unbalanced data. Random oversampling (ROS) and random under-sampling (RUS) are conventional resampling methods for solving class imbalance problems. They increase the imbalance of data by adding positive samples or reducing negative samples [3]. Although these two methods are simple and easy to operate, it is easy to cause

over-fitting of the model or loss of critical samples. In [4] proposed a synthetic minority oversampling technique (SMOTE), which linearly interpolates between a positive class sample and its nearest neighbor to synthesize a new positive sample, thereby achieving a balance between positive and negative classes. The SMOTE algorithm effectively avoids the over-fitting problem caused by random oversampling, but does not consider that the neighboring cases of positive samples located near the boundary are likely to cause overlapping of sample categories. In [5] proposed a boundary oversampling method Borderline-SMOTE based on the SMOTE algorithm. This method only synthesizes new samples for a small number of samples near the boundary, because Han believes that the key to the classification surface as those that are located on the classification boundary. This method avoids sampling overlap problem of the synthesis new samples for all positive samples. An improved SMOTE algorithm ADA-SYN(adaptive synthetic sampling) proposed in [6], which used density distribution information to determine the frequency of samples, can adaptively determine the frequency of positive samples, but this method amplifies the information of the noise samples, thereby affecting the quality of the classification. In [7], an under-sampling method based on traditional K-means clustering is proposed, which takes different cluster numbers clusters the negative classes, and then uses the cluster center as the negative sample. This place improves the imbalance of data and better reduces the possibility of loss of the most important samples. In [8] integrated the idea of ant colony optimization algorithm into under-sampling, and proposed an adaptive under-sampling method ACOSampling algorithm, which does not need to know the true distribution of samples, and determines the sample information through the classification feedback results, so as to avoid the problem of the loss of important information caused by the traditional under-sampling method. The nature of resampling is a preprocessing method for data, so it is not the same as the special algorithm.

In addition to the resampling method, some scholars have proposed some effective methods from the algorithm level. In [9] proposed an cost-sensitive learning algorithm CS-SVM, which integrates the idea of cost-sensitive learning into the SVM classifier, and optimizes the maximum separation surface of SVM for different classes. The samples are assigned different penalty factors, which improve the SVM's ability to classify unbalanced data. Batuwita proposed a cost-sensitive learning algorithm based on fuzzy weighting FSVM-CIL[10]. This method gives different cost weight of each sample by establishing the form of fuzzy membership function to reflect the importance of different samples. In [11] proposed a fuzzy weighted extreme learning machine algorithm FWELM, which uses the absolute value of the network output to replace the Euclidean distance of each sample to the initial separation surface, and it has a good effect on the classification of unbalanced data. In [12], an one-class support vector machine (OCSVM) algorithm is proposed, which finds the optimal hyperplane by maximizing the minimum Euclidean distance between the origin and the target sample an-d the classification error of the target data is minimized. Tax proposes a support vector data description (SVDD)[13], which is designed to find a minimum hypersphere with the boundary of the minimum hypersphere as the classification boundary between the target sample and the abnormal sample. It has a good effect on the classification of single type unbalanced data. Cao proposed a voting mechanism-based method [14] by constructing multiple independent extreme learning machine(ELM) classifiers, and then voting to select the ELM classifier with better classification effect as the final classifier.

In the under-sampling method described above, although the Borderline-SMOTE method solves the problem that the SMOTE method is easy to cause category duplication, it still cannot avoid the randomness of interpolation and it cannot guarantee that the synthesized sample must have the attributes of the positive class samples. And although the method in [7] can effectively improve the imbalance of data, it needs to determine the number of clusters by experience. If the number of clusters is not well selected, it will seriously affect the final classification effect. And the initial clustering center is randomly set, and it is

possible to select the isolated point or the outlier point as the clustering center, which results in the final selected clustering center not representing the majority class well.

In view of the shortcomings of the above methods, this paper integrates the clustering idea into the under-sampling method based on the traditional K-means clustering algorithm. By optimizing the selection of the initial clustering center, an under-sampling method based on optimized clustering center selection (BCUSM) is proposed. For negative class samples, set the number of clusters to the number of positive samples, and obtain the initial cluster center set by specific calculation, and then perform multiple clustering operations, using the sample with the highest similarity with the cluster center as the training data of the negative class samples, and then they are combined with the positive samples in the original training set to form a new training set, which is used to train the classifier, thereby improving the classification accuracy of the classifiers when the data is unbalanced.

This method is different from traditional SMOTE method is that, first of all, avoid the SMOTE method random interpolation on the boundary of the categories overlap problem, secondly, the clustering center selection of K - means algorithm has been optimized, and avoids the random choice lead to outliers or outlier clustering center, through the calculation of similarity, more accurately determine the clustering center. Then, the optimized k-means algorithm is applied to under-sampling, which not only ensures the purpose of compressing the negative data space, makes the original data set tend to balance, but also ensures that the negative data after compressing the data space is representative to a certain extent

The rest of this paper is organized as follows. Section 2 presents the related methods and principle involved in the BCUSM algorithm. Section 3 describes the basic principles and steps of the BCUSM algorithm in detail. Section 4 provides the experiments and result analysis. Section 5 presents the conclusion and future work.

## 2. Related Methods and Principles

### 2.1. Under-Sampling Method

The resampling method is an important means to solve the problem of unbalanced data classification, including under-sampling and oversampling. It is a method that is different from specialized classification algorithms to solve the problem of unbalanced data classification. It increases the number of sample data of a certain type, so that the heterogeneous samples tend to balance, so it is more like a unique data preprocessing method.

Under-sampling is a commonly used resampling method that is often used to process negative samples. It improves the classification performance of positive samples by reducing the number of negative samples. The under-sampling method is different from the oversampling method. It does not cause the over-fitting problem of the positive class because of the increase of the positive class sample. It can ensure the integrity of the positive class sample information to the greatest extent. However, the under-sampling method also has its shortcomings. It is very likely that some important sample information will be lost in the process of deleting negative samples, which will affect the prediction accuracy of negative classes.

The clustering method is also a commonly used data preprocessing method, which divides the samples with the highest similarity into the same class, thus distinguishing the samples of different categories. Therefore, combining the clustering method with the under-sampling method can retain the important characterization information of the negative class samples to the greatest extent possible. That is, before using the under-sampling method to process the negative class samples, the clustering method is used to divide the negative class samples into independent clusters, and then the under-sampling method is used for each cluster to select the most representative one as a negative class sample. This not only reduces the number of negative samples, but also retains the most representative samples of the negative classes.

## 2.2. Traditional K-means Algorithm

The idea of the traditional K-means clustering algorithm is to specify the number *k* of clusters, randomly select *k* initial cluster centers. And then calculate the Euclidean distance of each sample to the cluster center and divide it into the nearest cluster according to the distance. Finally calculate the average distance between the samples in each cluster and the cluster center, and use it as a new cluster center. After several iterations, it will not change until the cluster center. It uses sum of squared errors (SSE) as the objective function of the performance metric. Assume that the data set *D* and the class set *C* is as follows. $D=\{\boldsymbol{x_1}, \boldsymbol{x_2}, ..., \boldsymbol{x_n}\}$, $C=\{C_1, C_2, ..., C_k\}$, $\boldsymbol{x_i}=(a_{i1}, a_{i2}, ..., a_{im})$, $\boldsymbol{c_j}=(b_{j1}, b_{j2}, ..., b_{jm})$ .*n* is the total number of samples, *k* is the number of clusters, $\boldsymbol{x_i}$ presents a sample of the data set, $C_k$ represents the kth cluster, $\boldsymbol{c_j}$ represents the center of the *jth* cluster, and *m* is the number of attributes. Equation (1) is the standard definition of SSE.

$$SSE(C)=\sum_{j=1}^{k}\sum_{\mathbf{x_i}\in C_k}\left\|\mathbf{x_i}-\mathbf{c_j}\right\|^2 \tag{1}$$

In equation (1), $\boldsymbol{x_i}$ and $\boldsymbol{c_j}$ represent the ith element and clustering center of $C_k$, respectively. The updating formula of the clustering center is as follows.

$$\boldsymbol{c_k}=\frac{\sum_{x_i ? C_k} x_i}{|C_k|} \tag{2}$$

Equation (2) represents the calculation of the average value of the data in cluster $C_k$ as a new cluster center point. In (1) and (2) $\boldsymbol{c_j}$ is actually equivalent to $C_k$. The ultimate goal of the K-means algorithm is to find the cluster set that minimizes SSE.

## 2.3. Support Vector Machine

The Support Vector Machine (SVM) is a commonly used machine learning classification method. Its classification performance is much better than other machine learning classification methods when data is balanced. However, when the data are unbalanced, its performance will be significantly affected. Therefore, it is a good choice to use the SVM classifier to verify the performance of the BCUSM algorithm. It finds a maximum interval of separation planes in the sample by a linear equation of the form (3), which is commonly referred to as the maximum interval division hyperplane, through which different classes of samples are separated. Where w is the normal vector of the hyperplane, which determines the direction of the hyperplane, and b is the displacement offset, which determines the distance between the origin and the hyperplane, so the hyperplane can also be represented by (***w***, *b*). Assume that the hyperplane can correctly classify the class of the sample. Taking the two classifications as an example, the positive and negative samples are represented by +1 and -1 respectively. The boundary faces of the positive and negative samples can be represented by (4).

$$w^T x+b=0 \tag{3}$$

$$\begin{cases} w^T x_i+b\ ^3 1 \\ w^T x_i+b\ ?\ \text{-}1 \end{cases} \tag{4}$$

The support vector is the sample that satisfies (3) and is closest to the hyperplane. The sum of the

distances of the positive and negative class support vectors to the hyperplane is called the margin of the category, and the distance from any sample to the hyperplane can be expressed by (5).

$$r=\frac{\left|w^T x+b\right|}{\|w\|} \tag{5}$$

As showed in Fig. 1, the ultimate goal of the support vector machine is to find the hyperplane that maximizes the interval. The solid line represents the hyperplane that needs to find, and the dashed line represents the boundary surface of different categories, where the red solid sample represents the support vector of the positive class, and the red hollow sample represents the support phase vector of the negative class.

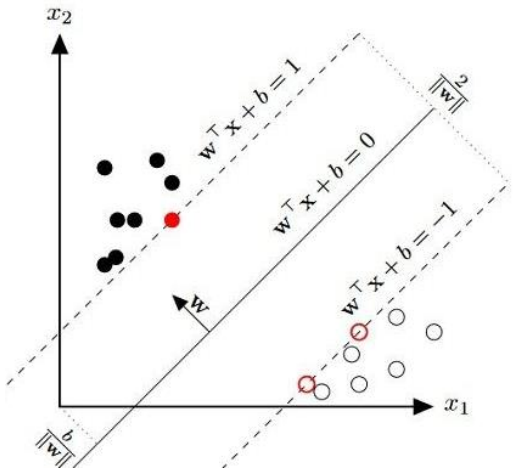



Fig. 1. The support vector machine principle[15].

## 3. An Optimized Classification Method

The idea of BCUSM algorithm is to use the K-means clustering algorithm selected by the optimized initial clustering center to cluster the negative classes, and then select the corresponding samples as new training samples to achieve the purpose of under-sampling so that the original data sets are balanced.

Firstly, the selection of the initial clustering center of the K-means clustering algorithm is optimized, and the negative K-means algorithm is used to cluster the negative classes. And then, the intra-cluster similarity calculation is performed, the sample most similar to the cluster center is selected as the negative sample in each cluster, thereby reducing the negative data and achieving the effect of data balance.

### 3.1. Optimize Cluster Center Selection

Since the traditional K-means algorithm randomly selects the initial clustering center, it is easy to cause the isolated point or the outlier point to be selected as the clustering center, which will result in poor final clustering quality. Therefore, based on the traditional K-means algorithm, this paper optimizes the selection of the initial cluster center according to (6)-(10). And the optimized algorithm is called OICSK-means as showing in Algorithm 1.

The optimization process of the initial cluster center is as follows. First of all, traverse the negative sample set, calculate the sum of Euclidean distances from each sample to other samples, and pick the sample with the smallest sum of distances as the first initial cluster center. The purpose of this step is to make the first cluster center try to avoid isolated points or outliers, and ensure the quality of the cluster center selected later. And then calculate the distance from the first initial cluster center in the remaining

samples, and use the sample with the largest distance from the first cluster center as the second initial cluster center. Finally, the sample with the largest sum of distances from the first and second initial cluster centers is taken as the third initial cluster center. The purpose of these two steps is to make the distance between the cluster centers as far as possible, so that the boundaries of the clusters are clearer. By analogy, the initial cluster center set is finally obtained. Using this concept following rules can be extracted.

$$dist(x_i, x_j) = \sqrt{(x_i - x_j)^T (x_i - x_j)} \tag{6}$$

$$\text{sum\_dist}(x_i) = \sum_{j \in (D - x_i)} dist(x_i, x_j) \tag{7}$$

$$\text{firstC} = \underset{x_i \in D}{\text{argmin}}(sum\_dist(x_i)) \tag{8}$$

$$\text{secondC} = \underset{x_j \in (D - firstC)}{\text{argmax}} (dist(x_j, firstC)) \tag{9}$$

$$c_j = \text{argmax}(dist(x_i, c_{j-1}) + dist(x_i, c_{j-2})) \tag{10}$$

Equation (6) is the Euclidean distance between samples. (7) represents the sum of Euclidean distances from one sample to other samples, and (8) and (9) represent the first initial cluster center and the second initial cluster center, respectively. Specifically, (10) is a recursive formula for the initial cluster center and $3 \le j \le k$, k represents the number of clusters.

**Algorithm 1 OICSK-means**

**Input:** Number of clusters k and data set *D*

**Output:** cluster set *C*

**1:** Calculate the initial cluster center $C_{\substack{j \\ 1 \le j \le k}}$ according to (6)-(10)

**2**: **while** $c_j^{'} == c_j$ **do**

**3:** **for** *i*=1 to n **do**

**4:** **for** *j*=1 to *k* **do**

**5:** **d**= $dist(x_i, c_j)$ divide $x_i$ to the corresponding cluster center $c_j$ according to the value of **d**

**6:** **end for**

**7:** **end for**

**8:** $c_j^{'} = \dfrac{\sum_{x_i \in C_j} x_i}{|C_j|}$ update the cluster center according to (2)

**9: end while**

**10: return** $C = (C_1, C_2, ..., C_k)$

## 3.2. Intracluster Similarity Calculation

In order to ensure as much as possible that important information of negative samples is not lost, and the imbalance of data sets can be minimized, the under-sampling method based on optimized clustering center uses the cosine similarity measure to measure the samples and cluster centers in each cluster. For the similarity, the sample with the highest similarity is selected as the final training sample. Set the number of

clusters k equal to the number of positive samples, obtain k clusters according to Algorithm 1, and calculate the cosine similarity of samples to cluster centers in each cluster according to (11), and select the sample with the highest similarity to its cluster center in each cluster and form the final training set with the positive samples.

$$cs(x_i, c_j) = \frac{x_i ? c_j}{\| x_i \| ? \| c_j \|} = \frac{\sum_{h=1}^{m} a_{ih} ? b_{jh}}{\sqrt{\sum_{h=1}^{m} a_{ih}^2} ? \sqrt{\sum_{h=1}^{m} b_{jh}^2}} \tag{11}$$

Different from the traditional K-means clustering algorithm, the initial cluster center set is obtained by accurate calculation, rather than randomly selected, because random selection may select samples located near the boundary, which will affect the final clustering effect. The BCUSM algorithm as showing in Algorithm 2 and it's model as showing in Fig. 2

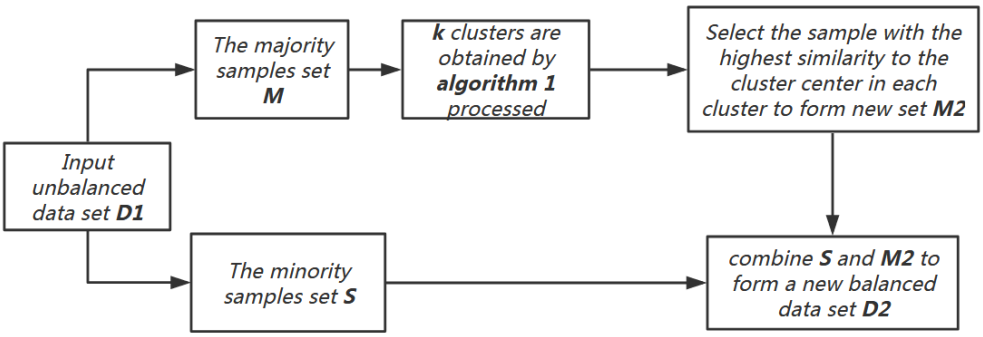



Fig. 2. The model of BCUSM algorithm.

**Algorithm 2 BCUSM**

**Input:** Data set *D*, positive sample set $D^+$, negative class sample set $D^-$, cluster number $k=|D^+|$, cluster center set *C*=**NULL**, new negative class sample set $D'^-$ =**NULL**

**Output:** Balanced data set $D'$

**1:** C= **OICSK-means** ( $D^-$ ,k). Use algorithm 1 for negative class samples to get clustered set *C*

**2: for** *j*=1 **to** k **do**

**3:** **for** *i*=1 **to** $|C_j|$ **do**

**4:** $D'^- = D'^- \cup cs(x_i, c_j)$

**5:** **end for**

**6: end for**

**7:** $D' = D^+ \cup D'^-$

**8: return** $D'$

## 4. Experimental

### 4.1. Data Set

In order to verify the validity of the BCUSM algorithm, we use the data in the UCI standard database to obtain three data sets with different balance rates, and use SVM classifier classification. The data set is shown in Table 1. In order to make the experimental data more operational, we use (12) to normalize the data in the data set. The hyper-parameters used in the experiment, except that the number of clusters *k* is set to the number of positive samples. The others are set as default.

Table 1. Data Set Description

| Data set | # of samples | # of attributes | IR |
|---|---|---|---|
| Ecoli | 336 | 7 | 3.36 |
| Vehicle | 846 | 18 | 3.25 |
| Yeast | 1484 | 8 | 2.43 |

Table 2. Confusion Matrix

| result \ category | Positive | Negative |
|---|---|---|
| True | True Positive(TP) | True Negative(TN) |
| False | False Positive(FP) | False Negative(FN) |

$$x_i^{'} = \frac{x_i - x_{i\_\min}}{x_{i\_\max} - x_{i\_\min}} \tag{12}$$

$x_{i\_\min}$, $x_{i\_\max}$ represent the minimum attribute value and the maximum attribute value, respectively. In the binary classification problem, the imbalance rate is defined as (13).

$$IR = \frac{|D^-|}{|D^+|} \tag{13}$$

In order to more realistically demonstrate the performance of the training model and make the test results of the model more convincing, this work used the self-help method in the experimental test. According to the size *n* of the data set *D*, there are *n* times of back-sampling on the data set *D*, so that a data set *D*′ is obtained as large as the original data set, and use *D*′ as the training set, and *D*-*D*′ As a test set. Although some samples in *D* will appear multiple times in *D*′, by probability calculation, approximately 36.8% of the samples in data set *D* will not be sampled, so multiple training sets can be obtained by this method and the accuracy of the classification model is improved. This work repeated 4 times for each data set on the basis of *n* times of put back sampling, and compared the classification effects of the other four methods.

## 4.2. Performance Metrics

For the classification of unbalanced data, the traditional evaluation mechanism based on the overall classification accuracy rate is not suitable for the evaluation of the classification method of unbalanced data. In this paper, the confusion matrix is used as the basis for evaluating the performance of the proposed method. As showing in Table 2. where *TP* indicates the correct number of positive samples, *TN* indicates the correct number of negative samples, *FP* represents the number of samples misclassified into positive classes, and *FN* indicates the number of samples misclassified into negative classes.

According to the confusion matrix, it can get two performance metrics, namely the overall performance *g-mean* value and the positive class accuracy rate *pr* as showed in (14) and (15). The reason for adopting these two indicators is that for the problem of non-equilibrium data classification, people will pay more attention to the classification accuracy of the positive class in practical applications, although the purpose of

the improvement is to improve the classification accuracy of the positive class data, the accuracy of the overall classification accuracy cannot be lowered.

$$g-mean=\sqrt{\frac{TP}{TP+FN}\times\frac{TN}{TN+FP}} \tag{14}$$

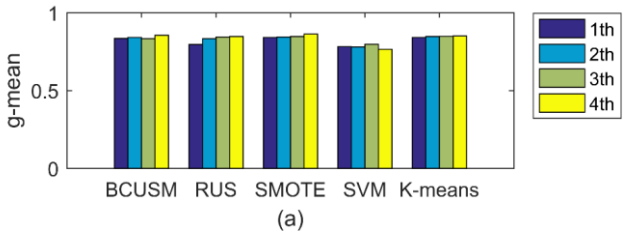
$$pr=\frac{TP}{TP+FP} \tag{15}$$

### 4.3. Results and Analysis

The experiment compares the under-sampling method proposed by this paper with RUS, SMOTE and under-sampling method based on ordinary K-means algorithm and the classification result of SVM classification without processing the original data. The *g-mean* and *pr* values of the five methods on different data sets are shown in Fig. 3~Fig. 6. From the comparison of *g-mean* values, it can be seen that the BCUSM under-sampling method is very close to the other four methods in terms of overall classification performance, and the classification effect on the three data sets is also relatively stable. It can be seen from (a), (c), (e) and (g) that the *g-mean* values of the five methods are very close, indicating that the BCUSM under-sampling method is very close to the other four methods in terms of overall classification performance, and the classification effect on three data sets is also relatively stable.

From the comparison of *pr* values in (b), (d), (f) and (h), it can find that compared with RUS, SMOTE and the under-sampling method of traditional K-means, BCUSM has improved the classification performance of positive class, which indicates that BCUSM can not only improve positive class classification accuracy but also can maintain the stability of the overall classification performance. This shows that by optimizing the initial cluster center, BCUSM can find representative samples of the negative class, which not only improves the imbalance of the data set, but also reduces the possibility of losing important information about the negative sample. The above analysis shows that the BCUSM under-sampling method can improve the classification accuracy of positive data and has a certain utility.

It can also be found that the RUS random under-sampling method has certain sensitivity to the data set by comparing the *pr* values. And its classification performance is not stable for different data sets. However, the classification performance of BCUSM under-sampling method is similar to that of different data set in the experiment, which indicates that BCUSM under-sampling method is more universal than RUS random under-sampling method, and it also reflects that the RUS random under-sampling method easily loses important sample information when the training data has fewer feature attributes, resulting in poor classification. In addition, the SVM's classification effect on the balanced data set is significantly better than the direct SVM classification of the original data set. This shows that SVM is very sensitive to unbalanced data. When no processing is performed on the original training set, the classification accuracy of the SVM for the positive class is greatly reduced, but it also shows that the SVM has better classification performance when the data set is balanced.



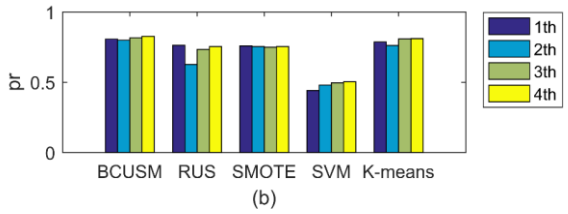



Fig. 3. (a) is the *g-mean* value of the different methods on the Ecoli dataset and (b) represents the *pr* value.

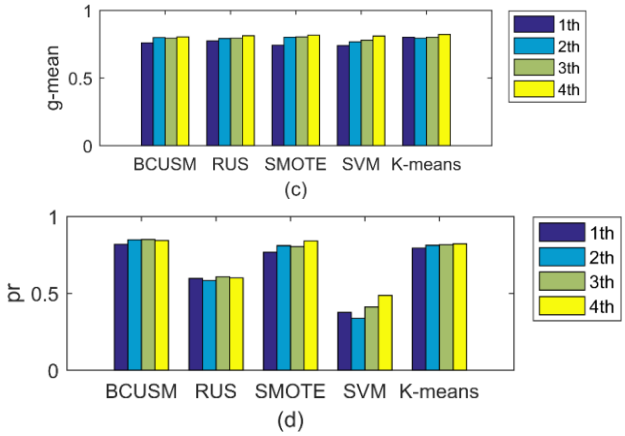



Fig. 4. (c) represents the *g-mean* value of the different methods on the data set Vechile, and (d) represents the *pr* value.

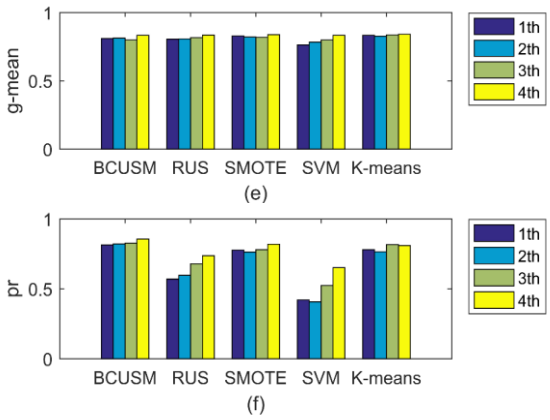
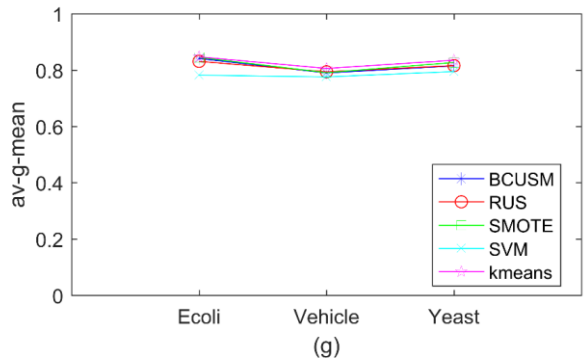



Fig. 5. (e) and (f) respectively represent the *g-mean* and *pr* values of different methods on the data set Yeast.



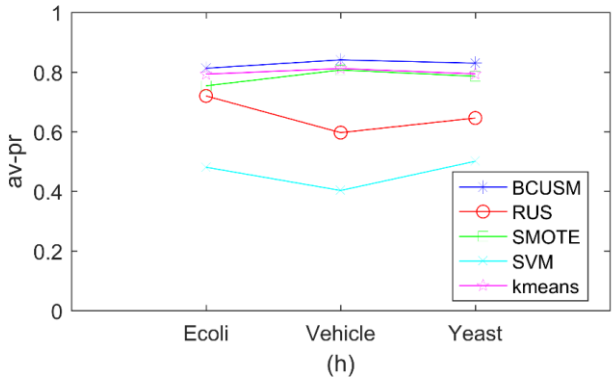



Fig. 6. Average g-mean and average pr values for different methods on the data set.

## 5. Conclusion

In this paper, the idea of clustering algorithm is integrated into the under-sampling method, and an under-sampling method BCUSM based on the optimization of clustering center selection is proposed from the data level. The experimental results show that the BCUSM under-sampling method has a certain effect in the face of unbalanced data classification, which can improve the classification accuracy of positive samples and maintain good overall classification performance. Due to the selection of the initial cluster center and the similarity comparison, a lot of distance calculations are added, which increases the time complexity of the algorithm operation. So the next work will focus on how to effectively reduce the time complexity of the algorithm running in the case of a good classification performance. In addition, applying this algorithm to different types of data sets would be a good attempt, and the future work will try to use the BCUSM algorithm to handle larger unbalanced data.

### Conflict of Interest

The authors declare no conflict of interest.

### Author Contributions

This work was done by all authors, Haitao Li carried out the research, obtained relevant data sets, performed simulation experiments, and gave a first draft of the paper. Mingjie Zhuang was responsible for the verification of the entire paper and gave relevant suggestions for modification.

### Acknowledgment

This work supported by the Technology Research and Development Projects of Quanzhou city(2018Z011) Fujian province, Fujian Provincial Academic Engineering Research, and Huaqiao University 2018 Postgraduate Innovation Ability Cultivating Projects(No.1811322002).

### References

[1] Dal, P. A., Boracchi, G., Caelen, O, *et al.* (2017). Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Transactions on Neural Networks and Learning Systems*, *29(8)*, 3784-3797.

[2] Bahnsen, A. C., Aouada, D., & Stojanovic, A. (2016). Feature engineering strategies for credit card fraud detection. *Expert Systems with Applications an International Journal*, *51*, 134-142.

[3] Mirza, B., Kok, S., Lin, Z., *et al*. (2016). Efficient representation learning for high-dimensional imbalance data. *Proceedings of the 2016 IEEE International Conference on Digital Signal Processing*, 511-515.

[4] Chawla, N. V., Bowyer, K. W., Hall, L. O., *et al*. (2002). SMOTE: Synthetic minority over-sampling

technique. *Journal of Artificial Intelligence Research*, *16(1)*, 321-357.

[5] Rivera, W. A., & Xanthopoulos, P. (2016). A priori synthetic over-sampling methods for increasing classification sensitivity in imbalanced data sets. *Expert Systems with Applications*, *66*, 124-135.

[6] He, H., Bai, Y., & Garcia, E. A., *et al*. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*.

[7] Li, C. X., Xie, l. S., & Lu, C. B. (2019). An undersampling method based on clustering for unbalanced data sets. *Practice and Understanding of Mathematics*, *49(1)*, 203-209.

[8] Yu, H., Ni, J., & Zhao, J. (2013). ACOSampling: An ant colony optimization-based undersampling method for classifying imbalanced DNA microarray data. *Neurocomputing*, *101(2)*, 309–318.

[9] Iranmehr, A., Masnadi, S. H., & Vasconcelos, N. (2019). Cost-sensitive support vector machines. *Neurocomputing*, *343*, 50-64.

[10] Batuwita, R., & Palade, V., (2010). FSVM-CIL: Fuzzy support vector machines for class imbalance learning. *IEEE Transactions on Fuzzy Systems*, *18(3)*, 558-571.

[11] Yu, H. L., Qi, Y. S., Yang, X. B., *et al*. (2017). Algorithm research of class unbalance fuzzy weighted limit learning machine. *Computer Science and Exploration*, *11(4)*, 619-632.

[12] Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, *13(7)*, 1443-1471.

[13] Tax, D. M. J., & Duin, R. P. W. (1999). Support vector domain description. *Pattern Recognit Lett*, *20(11–13)*, 1191-1199.

[14] Cao, J., Lin, Z., Huang, G. B. *et al*. (2012). Voting based extreme learning machine. *Information Sciences*, *185(1)*, 66-77.

[15] Keerthi, S. S., Shevade, S. K., & Bhattacharyya, C. *et al*. (2001). Improvements to platt\"s SMO algorithm for SVM classifier design. *Neural Computation*, *13(3)*, 637-649.

**Haiato Li** was born in 1995, he received the B.S. degree from Huaihua University, Huaihua, China, in 2018. He is currently pursuing the M.S. degree with the Huaqiao University. He is current research interests include machine learning and large-scale multiple input multiple output antenna selection technology.

**Mingjie Zhuang** was born in 1964, he received the B.S. degree from Department of Electronic Engineering, Fudan University, Shanghai, China, in 1982, and the Ph.D. degree in information and communications engineering, Xiamen University, Xiamen, China, 2001. His research interests include are wireless communication technology, space-time processing, stochastic processes, the internet of things technology, antenna technology, satellite navigation technology.