Hindawi
Scientific Programming
Volume 2017, Article ID 3704525, 8 pages
https://doi.org/10.1155/2017/3704525

*Research Article*

# Adaptive Ensemble Method Based on Spatial Characteristics for Classifying Imbalanced Data

**Lei Wang,[1,2] Lei Zhao,[1] Guan Gui,[1] Baoyu Zheng,[1] and Ruochen Huang[1]**

[1]*National Local Joint Engineering Research Center for Communication and Network Technology, Nanjing University of Posts and Telecommunications, Nanjing, China*
[2]*The State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, China*

Correspondence should be addressed to Guan Gui; guiguan@njupt.edu.cn

Received 15 September 2017; Accepted 3 December 2017; Published 26 December 2017

Academic Editor: Anfeng Liu

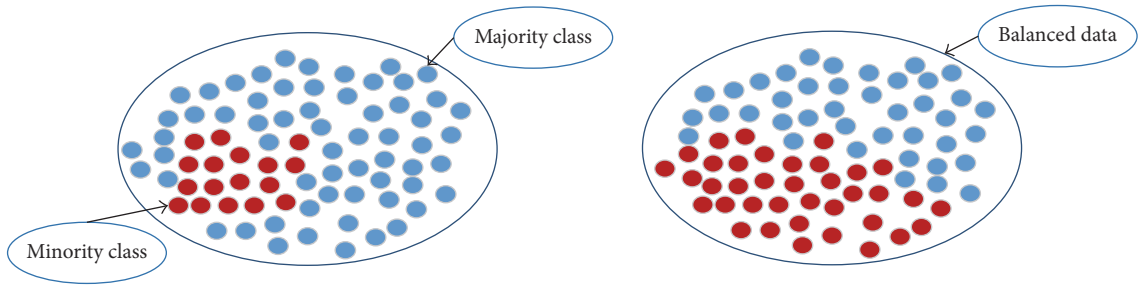
Copyright © 2017 Lei Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The class imbalance problems often reduce the classification performance of the majority of standard classifiers. Many methods have been developed to solve these problems, such as cost-sensitive learning methods, synthetic minority oversampling technique (SMOTE), and random oversampling (ROS). However, the existing methods still have some problems due to the possible performance loss of useful information and overfitting. To solve the problems, we propose an adaptive ensemble method by using the most advanced feature of self-adaption by considering an average Euclidean distance between test data and training data, where the average distance is calculated by $k$-nearest neighbors (KNN) algorithm. Simulation results are provided to confirm that the proposed method has a better performance than existing ensemble methods.

## 1. Introduction

Imbalanced data refers to a data set that has great differences in the number of the classes. Currently, imbalanced data has been applied to real-word domain and plays a key role in civilian and government applications, such as text classification [1], facial age estimation [2], speech recognition [3], and governmental decision-making support systems [4]. The research of imbalanced data is of great significance in the fields of credit fraud, data mining, and illegal account invasion. Hence, more and more researchers pay great attentions to class imbalanced issues due to the fact that the traditional classification of imbalanced data processing is not suitable for classifying minority classes. Imbalanced problem also caught the attention of related areas such as machine learning and data mining [4].

The present study focuses more on binary class imbalance problem, where data set is sorted into majority classes and minority classes. In the data set, the traditional balanced data means that the numbers of each class are equal, and the imbalanced data means that the numbers of the various classes are significantly different. The details of binary class imbalanced and balanced data are shown in Figures 1(a) and 1(b). The traditional classification algorithms, such as naive Bayes [5], random forest [6], $K$-nearest neighbors (KNN) [7], and RIPPER [8], aim at generating models that can optimize the accuracy over classification, but they neglect the minority class. In order to solve the problem mentioned above, many methods have been proposed about binary class imbalanced data in data level and algorithm level, respectively. In the data level, the major idea is to transform imbalanced into balanced data mainly by using sampling method or to create new examples for imbalanced into balanced data, such as SMOTE and ROS, while the algorithm-level solutions primarily include ensemble learning methods [9] and cost-sensitive analysis. Generally speaking, these methods solved the problem of imbalanced data in the accuracy of minority classes. However, there still exist some drawbacks in these traditional imbalanced data classification methods for handling binary class imbalanced data problems [10]. For example, boosting and bagging based ensemble methods may lose some valuable information in the iteration process owing to the use of sampling methods. As a result, this may cause the data overfitting problem. Moreover, it is hard to




(a) An example of binary imbalanced data, where majority class is 58 and minority class is 17

(b) An example of binary balanced data, where majority data is equal to minority data

Figure 1: A figure illustration of imbalanced data (a) and balanced data (b).

get the optimized misclassification cost in the cost-sensitive learning methods, and different misclassification costs lead to different classification results. Hence, the classification results are not stable.

To overcome the above problems, we proposed an adaptive ensemble method that is an improvement of existing ensemble method [10]. *Our main idea is to transform imbalanced binary problem into multiple balanced problems, which neither reduce the number of majority classes nor increase that of the minority classes*. Then we build multiple base classifiers to deal with these balanced problems and lastly we use an adaptive ensemble rule to assemble the base classification results obtained from base classifiers. Common ensemble rules including Max Rule, Majority Vote Rule, Product Rule, Min Rule, and Sum Rule were proposed in [11] and several novel ensemble rules including MaxDistance Rule, MinDistance Rule, ProDistance Rule, MajDistance Rule, and SumDistance Rule were put forward in [10]. In [10], the test results indicate that their methods have a better performance compared with many conventional imbalanced data processing methods over some standard imbalanced data sets. Meanwhile, the results of their experiments proved that SplitBal + MaxDistance has a better performance than other combinations. Throughout this paper, SplitBal + MaxDistance is referred to as SMD.

We have two improvement points for SMD and we define it as SplitBal + MaxDistanceandAvePr (SMDA), which shares the same process with SMD except the ensemble rule. By using base classification algorithms including naive Bayes, random forest, logistic regression, and SVC [12], empirically, our proposed method is evaluated over 38 highly imbalanced data sets. After that, the numerical results show that our method is superior to SMD.

The rest of this paper is organized as follows. Section 2 introduces the works related to our research. Section 3 shows the proposed method. Section 4 reports our experimental procedure, describes details on the setup of experiments, and analyzes the processed data results. Finally, in Section 5, we summarize the study and draw the conclusion.

## 2. Related Work

Over the past decades, the imbalanced data problem has always been a difficult problem in data mining. There are also other data characteristics such as data shift [13] and class overlapping [14], which can influence the performance of conventional classification algorithms for dealing with imbalanced problems. However, we still focus on the imbalance characteristic between classes.

Table 1: Related work list.

| | |
|---|---|
| Data level | [9, 13, 14] |
| Algorithm level | |
| Ensemble learning | [10, 11] |
| Cost-sensitive learning | [15] |

So far, many measures have been proposed to solve the binary class imbalance problem [10–14, 16–21]. These measures can be broadly by data level and algorithm level, as shown in Table 1. The existing measures can adapt the imbalanced class in algorithm level, while preprocess of measures can adjust data from being imbalanced to balanced in data level. Our methods could be regarded as in algorithm level; in this section, we will introduce some methods that belong to algorithm level.

The algorithm level includes cost-sensitive learning, ensemble learning, and recognition-based learning. (1) Cost-sensitive learning approaches obtain the lowest classification error by adjusting the class misclassification cost. MetaCost [15] is a kind of this algorithm, which uses cost-sensitive procedure to make the classification algorithm cost-sensitive. (2) Ensemble learning is used to reduce the variance and bias by integrating the results of many classification algorithms on imbalanced data. Representatively, boosting can adaptively identify the samples, which is classified as error, so it can obtain a good performance on imbalance problem. Bagging improves the classification performance by processing the base classifiers. (3) Autoassociation, RIPPER, and recognition based learning provide the discrimination model created on the examples of the target class alone which have been certified to be effective in dealing with high-dimensional and complicated binary imbalanced data.

However, these ensemble methods may have some unavoidable drawbacks such as changing the raw data spatial distribution or lead to overfitting caused by sampling methods. In addition, these ensemble algorithms may lose the connection between the test data and training data. In other

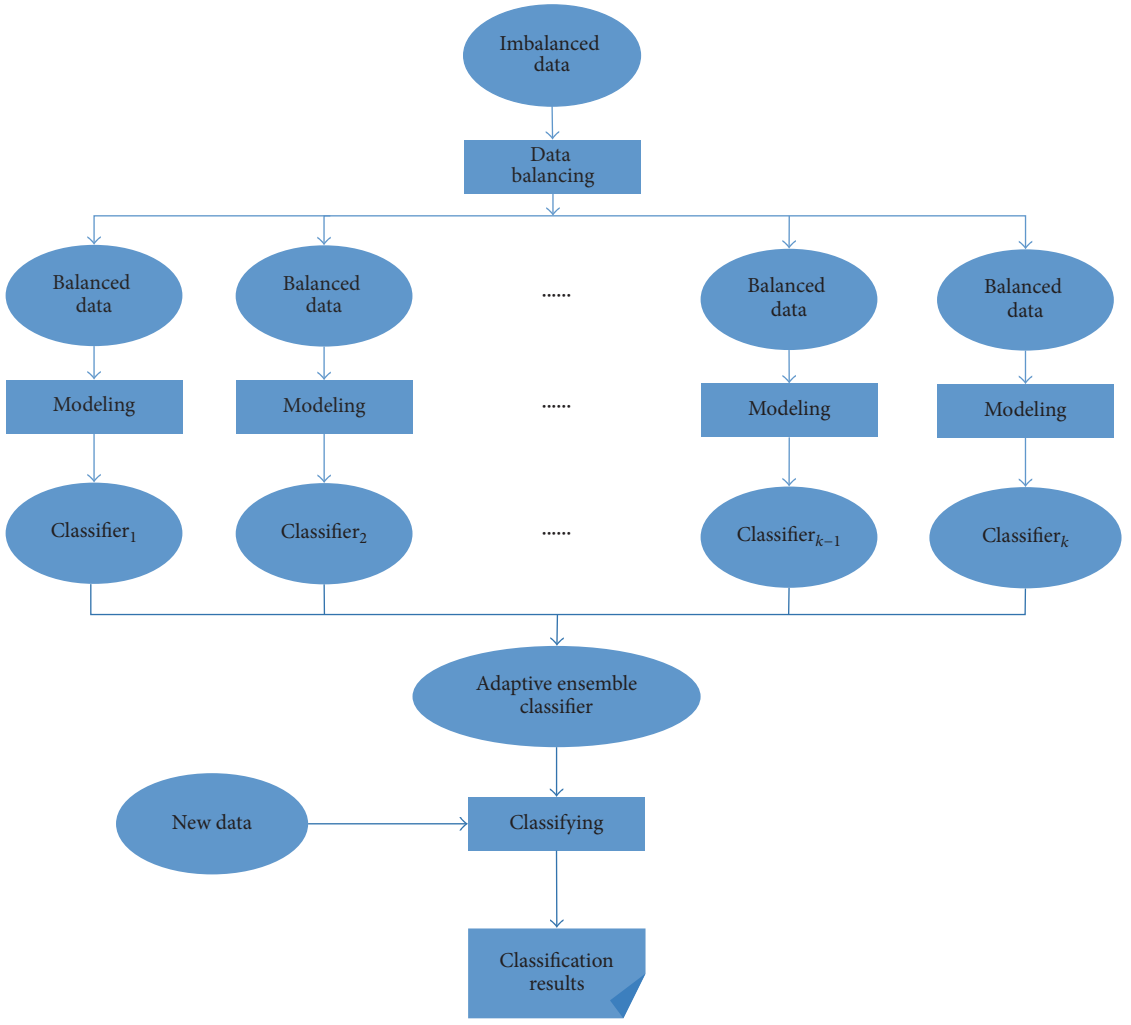



Figure 2: Framework of our proposed adaptive ensemble method for handling binary imbalanced data.

words, test data can be classified as the class which is closer than the spatial distribution distance. Our method avoids some weaknesses of these traditional imbalanced problems processing methods mentioned previously by transforming the imbalanced problem into several balanced ones; thus it is not like the existing imbalance problems handling methods. Furthermore, our method takes into account the distance between data factors in the ensemble rule, because, in theory, the closer they are, the more similar they are, and our ensemble method is adaptive, which is different from SMD.

## 3. Our Proposed Method

Our proposed method includes three parts: data balancing, modeling, and classifying. Figure 2 describes the frame of the proposed method. For data balancing, in the process of our method, we first divide the majority data set into several parts which are equal to the amount of minority class data. Then we combine the part with the minority class into a new balanced data set. So many balanced data sets are received. For modeling, next, each new balanced data set is used to create a base classifier with a given base classification algorithm. As for classifying, lastly, these base classification results are put into an adaptive ensemble classifier to classify test data. In the modeling component, we directly apply a base algorithm to every balanced data set. Subsequently, we will introduce two procedures of *data balancing* and *classifying* as follows.

*3.1. Data Balancing.* Existing measures to balance the imbalanced data usually lead to the loss of information as well as overfitting. Therefore, it can be realized to transform imbalanced data set into multiple balanced data sets without importing noise data or lessening the raw data. It is well known that the majority of class data sets are usually more than the minority in an imbalanced data set. So we divide the majority class data set into multiple sets, and each set is equal to minority class in number. Considering the similarities of a class, we can split the majority class data set into multiple sets (SplitBal). Then each set is added to the minority class data

Table 2: The strategies and descriptions for MaxDistance Rule.

| Rule | Strategy |
|---|---|
| MaxDistance | $R_1 = \arg\max_{1\le i\le k} \frac{p_{i1}}{D_{i1}+1}$, $R_2 = \arg\max_{1\le i\le k} \frac{p_{i2}}{D_{i2}+1}$ |

Table 3: The strategies for AdaptiveMaxDistance Rule.

| Rule | Strategy |
|---|---|
| AdaptiveMaxDistance | $R_1 = \arg\max_{1\le i\le k} \frac{p_{i1}}{D_{i1}+\alpha} + \text{average}(P_{i1})$ $R_2 = \arg\max_{1\le i\le k} \frac{p_{i2}}{D_{i2}+\alpha} + \text{average}(P_{i2})$ |

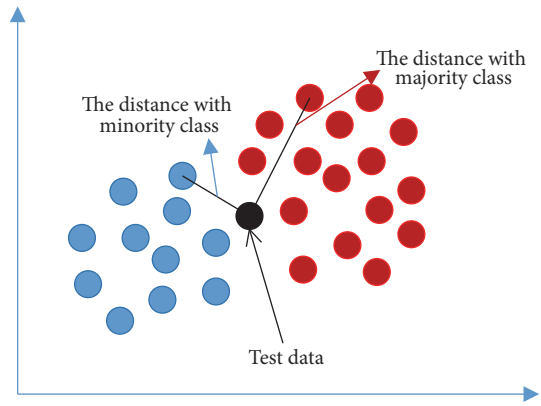



Figure 3: Figure illustration of the average distance.

set to build a new balanced data set. Lastly, we could obtain multiple balanced data sets.

*3.2. Classifying.* After modeling, we can build multiple classifiers with the processed balanced data acquired from data balancing. Then we can get some classification results from these classifiers. Next we combine these classification results together. Like [9], we make some assumptions as follows: assume that there are $k$ binary class data sets and two class labels; the class labels are $C_1$ and $C_2$. Then we could get $K$ base classifiers with a given base algorithm. For the $i$th classifier ($1 \le i \le k$), it will classify the test data as $C_1$ with the probability $P_{i1}$ and as $C_2$ with the probability $P_{i2}$. Moreover, $R_1$ and $R_2$ represent the ultimate ensemble results for classes $C_1$ and $C_2$, respectively. Five ensemble rules and their description and details are shown in [10]. But these ensemble rules just adopt the results of the classification while overlooking the connection between the test data and training data. The test data tends to be grouped into the class whose average Euclidean instances are closer to the test data. A new multilabel classifier that uses neighbor distance was mentioned in [22]. Considering the average Euclidean instances between the test data and the training data, five novel ensemble rules were proposed in [9]. In these ensemble methods, $D_{ij}$ ($1 \le i \le k$, $1 \le j \le 2$) indicate the average Euclidean distance between new data and the data with class label $C_j$ in the $i$th data. From Figure 3, we can learn the process of obtaining $D_{ij}$ with KNN. The details of MaxDistance are shown in Table 2.

However, in the rules $R_1 = \arg\max_{1\le i\le k}(p_{i1}/(D_{i1}+1))$ and $R_2 = \arg\max_{1\le i\le k}(p_{i2}/(D_{i2}+1))$, a fixed value should be added in the denominator with the purpose of preventing the average Euclidean distance from being equal to 0. Then the results of experiment in [10] show that the value can be defined anywhere from 0 to 1, so they add the distance with 1. But we argue that the value added to average in the denominator could be defined with 0, and the value also should be adaptive from 0 to 1 with different classification algorithms. The reasons are as follows. (1) The average distance $D_{ij}$ could not be 0, because from Figure 3 we can know that it is impossible for the new data to be the same as all the train data. (2) When we use different base classification algorithms, the value added to average in the denominator should be mutative. So we define the added value as $\alpha$, which ranges from 0 to 1. (3) From the MaxDistance Rule, we can find that the effect of $P_{il}$ is weak even though it has been considered as the best important decision element in most traditional algorithms. And in [22], we find that EMLA (average of $P_{il}$) always has a better performance than other ensemble rules, so we combine the EMLA with $R_1 = \arg\max_{1\le i\le k}(p_{i1}/(D_{i1}+\alpha))$ and $R_2 = \arg\max_{1\le i\le k}(p_{i2}/(D_{i2}+\alpha))$ as shown in Table 3. Finally, the classification results $R_1$ and $R_2$ are obtained with the ensemble rules in Tables 2 and 3, respectively. If $R_1 \ge R_2$, the test data is considered as $C_1$; otherwise it is considered as $C_2$.

## 4. Numerical Simulation

In this paper, we have adopted 38 public imbalanced data sets which came from Keel data set repository [23]. The details of these 38 data sets are shown in Table 4, including imbalance radio, total attributes (ATT), total number of data sets, and the number of minority (positive) class data sets. For more detailed information about the employed data sets, interested authors are referred to http://sci2s.ugr.es/keel/imbalanced.php.

We use the 5-fold cross-validation strategy in the following experiment. Four different base classification algorithms, naive Bayes, random forest, logistic regression, and SVC, were selected as base classifiers. We use AUC [24] as our algorithm metric which has more advantages than $G$-Mean and $F$-Measure [25]. In our experiment, every AUC of every data set will be tested repeatedly and then take an average.

Our study in this paper made up two experiments. The first experiment is to determine the added value in our rule. Then the second experiment is to compare the proposed method SMDA with SMD method when handling the imbalanced binary problems using different base classification algorithms.

*Experiment 1.* We first use the data set yeast3 (shown in Table 5) to test the AUC value of our method by using different values in our ensemble rules. Then we choose the

Table 4: Statistic summary of the 38 highly imbalanced data sets.

| Data set | ATT | Instance | Minority | IR |
|---|---|---|---|---|
| (1) Yeast3 | 9 | 1484 | 163 | 8.10 |
| (2) Ecoli03 | 8 | 336 | 35 | 8.60 |
| (3) Yeast2vs4 | 9 | 514 | 51 | 9.08 |
| (4) Ecoli067vs35 | 8 | 222 | 22 | 9.09 |
| (5) Ecoli0234vs5 | 8 | 202 | 20 | 9.10 |
| (6) Glass015vs2 | 10 | 172 | 17 | 9.12 |
| (7) Yeast0359vs78 | 9 | 506 | 50 | 9.12 |
| (8) Yeast0256vs3789 | 9 | 1004 | 99 | 9.14 |
| (9) Yeast02579vs368 | 9 | 1004 | 99 | 9.14 |
| (10) Ecoli046vs5 | 7 | 203 | 20 | 9.15 |
| (11) Yeast1289vs7 | 9 | 947 | 30 | 30.57 |
| (12) Ecoli0267vs35 | 8 | 224 | 22 | 9.18 |
| (13) Glass04vs5 | 10 | 92 | 9 | 9.22 |
| (14) Ecoli0346vs5 | 8 | 205 | 20 | 9.25 |
| (15) Ecoli0347vs56 | 8 | 257 | 25 | 9.28 |
| (16) Yeast05679vs4 | 9 | 528 | 51 | 9.35 |
| (17) Vowel0 | 14 | 988 | 90 | 9.98 |
| (18) Ecoli067vs5 | 7 | 220 | 20 | 10.00 |
| (19) Glass016vs2 | 10 | 192 | 17 | 10.29 |
| (20) Led7digit | 8 | 443 | 37 | 10.97 |
| (21) Ecoli01vs5 | 7 | 240 | 20 | 11.00 |
| (22) Glass06vs5 | 10 | 108 | 9 | 11.00 |
| (23) Glass0146vs2 | 10 | 205 | 17 | 11.06 |
| (24) Glass2 | 10 | 214 | 17 | 11.59 |
| (25) Ecoli0147vs56 | 7 | 332 | 25 | 12.28 |
| (26) Ecoli0146vs5 | 7 | 280 | 20 | 13.00 |
| (27) Shuttlec0vs4 | 10 | 1892 | 123 | 13.87 |
| (28) Yeast1vs7 | 8 | 459 | 30 | 14.30 |
| (29) Glass4 | 10 | 214 | 13 | 15.46 |
| (30) Ecoli04 | 8 | 336 | 20 | 15.80 |
| (31) Pageblock13vs4 | 11 | 472 | 28 | 15.86 |
| (32) Glass016vs5 | 10 | 184 | 9 | 19.44 |
| (33) Glass5 | 10 | 214 | 9 | 22.78 |
| (34) Yeast2vs8 | 9 | 482 | 20 | 23.10 |
| (35) Yeast4 | 9 | 1484 | 51 | 28.10 |
| (36) Yeast5 | 9 | 1484 | 44 | 32.73 |
| (37) Ecoli0137vs26 | 8 | 281 | 7 | 39.14 |
| (38) Yeast6 | 9 | 1484 | 35 | 41.40 |

Table 5: The data set of Yeast3.

| Data set | ATT | Instances | Minority | IR |
|---|---|---|---|---|
| Yeast3 | 9 | 1484 | 163 | 8.10 |

fixed value which can make the best AUC. In experiment, the values are 0, 0.2, 0.4, 0.6, 0.8, and 1.0. From Figure 4, we can know that $\alpha$ should be defined as 1, 0, 1, and 0 when the base classification algorithms are naive Bayes, random forest, logistic regression, and SVC, respectively.

*Experiment 2.* Performance results are evaluated in comparisons of SDMA and SMD. For every imbalanced data set, the detailed AUC values for both methods with the four different base classification algorithms are shown in Table 6. The end of the row represents the average of AUC values of the two methods with each classification algorithm. From Figure 5, we can observe that there are 33 AUC values of SMDA which are greater than or equal to SMD by using logistic regression. In addition, 25 AUC values of SMDA are greater than SDMA using SVC, while they are greater than or equal to SMD using random forest. It is noticed that 30 AUC values of SMDA are great than or equal to SMD using naive Bayes. In Figure 6, we

Table 6: AUC value for SMDA and SMD using different classification algorithms.

| DATA | Logistic regression | | SVC | | Random forest | | Naive Bayes | |
|---|---|---|---|---|---|---|---|---|
| | SMDA | SMD | SMDA | SMD | SMDA | SMD | SMDA | SMD |
| Yeast3 | 0.9617 | 0.9600 | 0.9521 | 0.8529 | 0.9757 | 0.9773 | 0.9093 | 0.8355 |
| Ecoli03 | 0.9259 | 0.9309 | 0.9211 | 0.8997 | 0.9415 | 0.9349 | 0.8955 | 0.9037 |
| Yeast2vs4 | 0.9206 | 0.9192 | 0.9211 | 0.8655 | 0.9755 | 0.9720 | 0.8798 | 0.8487 |
| Ecoli067vs35 | 0.9497 | 0.8697 | 0.8508 | 0.8677 | 0.9551 | 0.9594 | 0.8433 | 0.8326 |
| Ecoli0234vs5 | 0.9398 | 0.8809 | 0.9396 | 0.9425 | 0.9873 | 0.9845 | 0.7885 | 0.8595 |
| Glass015vs2 | 0.5556 | 0.5644 | 0.7600 | 0.7033 | 0.8033 | 0.7678 | 0.7022 | 0.6877 |
| Yeast0359vs78 | 0.8044 | 0.7877 | 0.7012 | 0.5987 | 0.8334 | 0.8457 | 0.7312 | 0.6032 |
| Yeast0256vs3789 | 0.8443 | 0.8429 | 0.8190 | 0.7928 | 0.8473 | 0.8393 | 0.7156 | 0.6893 |
| Yeast02579vs368 | 0.9380 | 0.9379 | 0.9142 | 0.8958 | 0.9579 | 0.9589 | 0.9214 | 0.8251 |
| Ecoli046vs5 | 0.9593 | 0.9343 | 0.9283 | 0.9523 | 0.9903 | 0.9861 | 0.8924 | 0.8792 |
| Ecoli0267vs35 | 0.9198 | 0.8932 | 0.8315 | 0.8797 | 0.9373 | 0.9382 | 0.8369 | 0.8286 |
| Glass04vs5 | 0.9538 | 0.9470 | 0.9667 | 0.9667 | 1.0000 | 1.0000 | 0.9875 | 0.9875 |
| Ecoli0346vs5 | 0.9444 | 0.8375 | 0.9278 | 0.9486 | 0.9833 | 0.9819 | 0.8931 | 0.8986 |
| Ecoli0347vs56 | 0.9388 | 0.8982 | 0.9178 | 0.9371 | 0.9720 | 0.9712 | 0.8896 | 0.8957 |
| Yeast05679vs4 | 0.8479 | 0.8437 | 0.8333 | 0.7439 | 0.9142 | 0.9134 | 0.7835 | 0.7151 |
| Vowel0 | 0.9833 | 0.9826 | 0.9992 | 0.9997 | 0.9991 | 0.9988 | 0.9715 | 0.9340 |
| Ecoli067vs5 | 0.8910 | 0.8359 | 0.9308 | 0.9231 | 0.9654 | 0.9731 | 0.8064 | 0.8385 |
| Glass016vs2 | 0.6529 | 0.6265 | 0.7211 | 0.6760 | 0.7824 | 0.7770 | 0.6897 | 0.6843 |
| Led7digit | 0.9530 | 0.9525 | 0.9619 | 0.9615 | 0.9549 | 0.9538 | 0.9462 | 0.9427 |
| Ecoli01vs5 | 0.9767 | 0.9580 | 0.9488 | 0.9453 | 0.9872 | 0.9918 | 0.8581 | 0.8186 |
| Glass06vs5 | 0.9094 | 0.8404 | 0.9839 | 0.9739 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| Glass0146vs2 | 0.7048 | 0.6663 | 0.7336 | 0.7324 | 0.8276 | 0.8480 | 0.7287 | 0.7169 |
| Glass2 | 0.6923 | 0.6752 | 0.8447 | 0.8175 | 0.8213 | 0.8011 | 0.7252 | 0.7293 |
| Ecoli0147vs56 | 0.9393 | 0.9016 | 0.9236 | 0.9255 | 0.9736 | 0.9742 | 0.8076 | 0.8045 |
| Ecoli0146vs5 | 0.9412 | 0.9284 | 0.9480 | 0.9500 | 0.9863 | 0.9882 | 0.8076 | 0.8045 |
| Shuttlec0vs4 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 0.9757 | 0.9757 |
| Yeast1vs7 | 0.8273 | 0.8268 | 0.7849 | 0.6529 | 0.8306 | 0.8368 | 0.7835 | 0.6054 |
| Glass4 | 0.9316 | 0.8585 | 0.9595 | 0.9662 | 0.9663 | 0.9655 | 0.8688 | 0.8814 |
| Ecoli04 | 0.9831 | 0.9815 | 0.9911 | 0.9100 | 0.9928 | 0.9928 | 0.9203 | 0.9428 |
| Pageblock13vs4 | 1.0000 | 0.9985 | 0.7889 | 0.9119 | 0.9984 | 0.9968 | 0.9623 | 0.8959 |
| Glass016vs5 | 0.8912 | 0.8765 | 0.9526 | 0.9559 | 0.9912 | 0.9882 | 1.0000 | 1.0000 |
| Glass5 | 0.8725 | 0.8600 | 0.9350 | 0.9230 | 0.9925 | 0.9875 | 0.9950 | 0.9950 |
| Yeast2vs8 | 0.8574 | 0.8546 | 0.7663 | 0.7256 | 0.8806 | 0.8795 | 0.8038 | 0.7758 |
| Yeast4 | 0.8740 | 0.8743 | 0.8692 | 0.8154 | 0.9307 | 0.9277 | 0.8368 | 0.7837 |
| Yeast1289vs7 | 0.7782 | 0.7776 | 0.6606 | 0.5877 | 0.7960 | 0.7974 | 0.7741 | 0.5185 |
| Yeast5 | 0.9803 | 0.9859 | 0.9827 | 0.9373 | 0.9912 | 0.9912 | 0.9861 | 0.9524 |
| Ecoli0137vs26 | 0.9164 | 0.9385 | 0.9443 | 0.9702 | 0.9611 | 0.9684 | 0.9572 | 0.9084 |
| Yeast6 | 0.9254 | 0.9235 | 0.9273 | 0.8529 | 0.9482 | 0.9478 | 0.9516 | 0.9394 |
| Average | 0.8919 | 0.8729 | 0.8879 | 0.8674 | 0.9382 | 0.9370 | 0.8654 | 0.8386 |

can see that the average AUC values of our method are greater than SMD overall. Therefore, we can obtain that that SMDA has a better performance than SMD in dealing with the data sets mentioned above.

## 5. Conclusion

An adaptive ensemble method based on spatial characteristics for dealing with the binary class imbalanced problems has been given in this paper. Different from the existing methods mentioned in this paper, our method firstly uses an adaptive ensemble rule for dealing with imbalanced binary problem. Furthermore, our method neither alters the raw data distribution nor suffers from unexpected mistakes or data loss.

Our method applies random splitting to the majority class instances to transform the imbalanced binary class data into multiple balanced binary class data. After that, we use a base classification algorithm to build multiple base

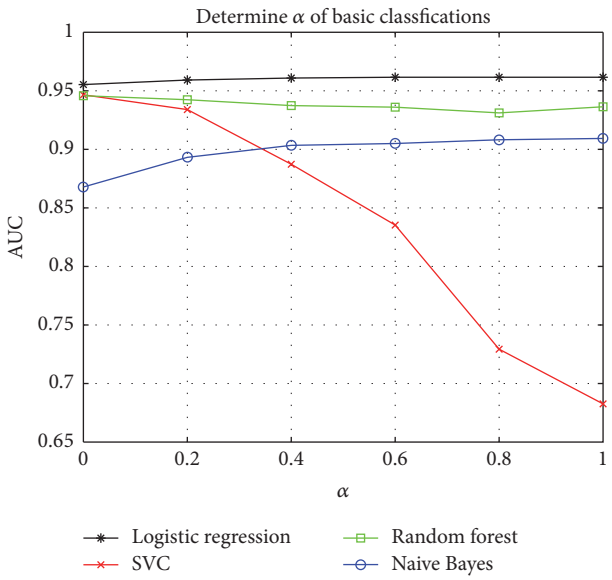



Figure 4: The AUC of different algorithms.

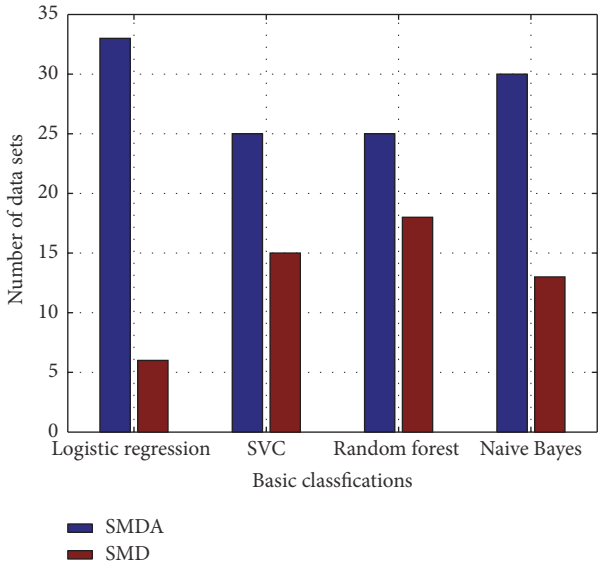



Figure 5: The performance comparison of SMDA (our proposed) and SMD (existing method [10]).

classifiers. Finally, we use the proposed adaptive ensemble rule to assemble the classification results received from base classifiers. The experimental results show that (i) the added variable value to the distance in our methods is adaptive, which changes with the classification algorithm and ranges from 0 to 1, and (ii) our ensemble rule SMDA has a better performance than SMD, so we could obtain that the proposed method currently performs better than the existing methods mentioned in this paper.

## Conflicts of Interest

The authors declare that there are no conflicts of interest.

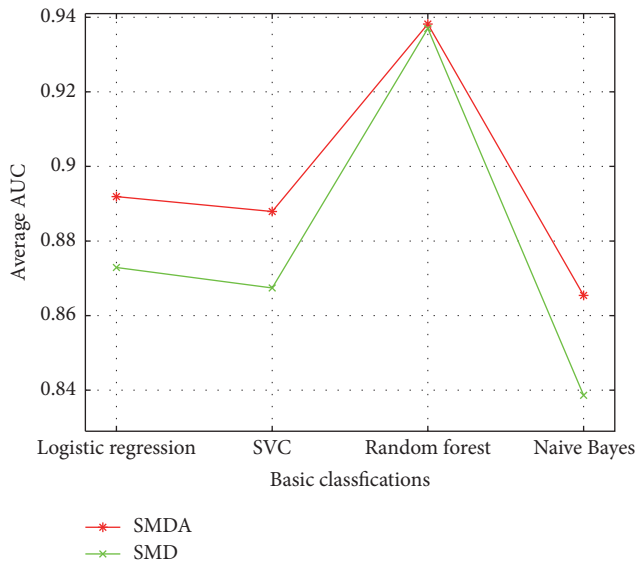



Figure 6: The average value of AUC.

## Authors' Contributions

Lei Wang and Lei Zhao conceived and designed the experiments; Lei Zhao and Guan Gui performed the experiments; Baoyu Zheng and Ruochen Huang analyzed the data and also gave comprehensive comments and suggestions; Lei Wang wrote the paper.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grants nos. 61671253 and 61401069), the Priority Academic Development Program of Jiangsu Higher Education Institutions, China, the Major Projects of the Natural Science Foundation of the Jiangsu Higher Education Institutions (16KJA510004), the Open Research Fund of the State Key Laboratory of Integrated Services Networks, Xidian University (ISN17-04), and the Open Research Fund of National Local Joint Engineering Research Center for Communication and Network Technology, Nanjing University of Posts and Telecommunications (TXKY17005).

## References

[1] M. D. D. Castillo and J. I. Serrano, "A multi strategy approach for digital text categorization," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 15–32, 2004.

[2] W.-L. Chao, J.-Z. Liu, and J.-J. Ding, "Facial age estimation based on label-sensitive learning and age-oriented regression," *Pattern Recognition*, vol. 46, no. 3, pp. 628–641, 2013.

[3] A. An, N. Cercone, and X. Huang, "A case study for learning from imbalanced data sets," in *Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence*, vol. 2056 of *Lecture Notes in Computer Science*, pp. 1–15, Springer, Berlin, Germany, 2001.

[4] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[5] Z. Zheng, "Naive bayesian classifier committees," *Lecture Notes in Computer Science*, vol. 1398, pp. 196–207, 1998.

[6] A. Liaw and M. Wiener, "Classification and regression by random forest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[7] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, article 1883, 2009.

[8] W. W. Cohen, "Fast effective rule induction," in *12th International Conference on Machine Learning*, pp. 115–123, Miami, Fl, USA, 2013.

[9] Z. Sun, Q. Song, and X. Zhu, "Using coding-based ensemble learning to improve software defect prediction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 1806–1817, 2012.

[10] Z. Sun, Q. Song, X. Zhu, H. Sun, B. Xu, and Y. Zhou, "A novel ensemble method for classifying imbalanced data," *Pattern Recognition*, vol. 48, no. 5, pp. 1623–1637, 2015.

[11] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[12] C. Ullrich, "Support vector classification," in *Forecasting and Hedging in the Foreign Exchange Markets*, pp. 65–82, Springer, Berlin, Germany, 2009.

[13] V. López, A. Fernández, and F. Herrera, "On the importance of the validation technique for classification with imbalanced datasets: addressing covariate shift when data is skewed," *Information Sciences*, vol. 257, no. 2, pp. 1–13, 2014.

[14] R. Alejo, R. M. Valdovinos, V. García, and J. H. Pacheco-Sanchez, "A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios," *Pattern Recognition Letters*, vol. 34, no. 4, pp. 380–388, 2013.

[15] P. Domingos, "Metacost:a general method for making classifiers cost sensitive," in *Proceedings of the International Conference on Knowledge Discovery & Data Mining*, pp. 155–164, San Diego, Cal, USA, 1999.

[16] A. Liu, Z. Chen, and N. N. Xiong, "An adaptive virtual relaying set scheme for loss-and-delay sensitive WSNs," *Information Sciences*, vol. 424, pp. 118–136, 2018.

[17] M. Zhou, M. Zhao, A. Liu, M. Ma, T. Wang, and C. Huang, "Fast and efficient data forwarding scheme for tracking mobile targets in sensor networks," *Symmetry*, vol. 9, no. 11, article 269, 2017.

[18] D. Wu, J. Wang, R. Q. Hu, Y. Cai, and L. Zhou, "Energy-efficient resource sharing for mobile device-to-device multimedia communications," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 5, pp. 2093–2103, 2014.

[19] D. Wu, L. Zhou, Y. Cai, R. Hu, and Y. Qian, "The role of mobility for D2D communications in LTE-advanced networks: energy vs. bandwidth efficiency," *IEEE Wireless Communications Magazine*, vol. 21, no. 2, pp. 66–71, 2014.

[20] L. Zhou, R. Q. Hu, Y. Qian, and H.-H. Chen, "Energy-spectrum efficiency tradeoff for video streaming over mobile ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 5, pp. 981–991, 2013.

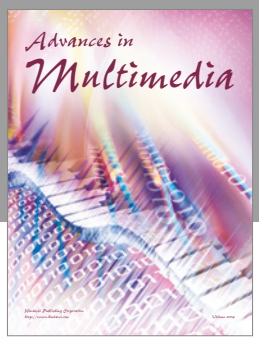
[21] L. Zhou, "Mobile device-to-device video distribution: theory and application," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 3, article 38, pp. 1253–1271, 2016.

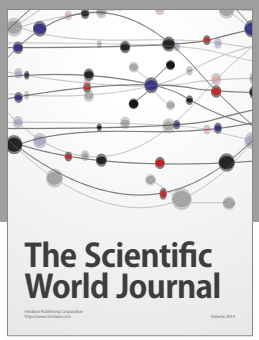
[22] M. A. Tahir, J. Kittler, and A. Bouridane, "Multilabel classification using heterogeneous ensemble of multi-label classifiers," *Pattern Recognition Letters*, vol. 33, no. 5, pp. 513–523, 2012.

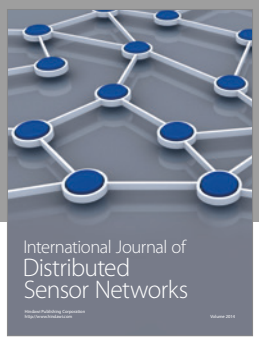
[23] J. Alcal, A. Fernndez, J. Luengo, J. Derrac, S. Garcła, and L. Snchez, "KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework," *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, no. 2, pp. 255–287, 2011.

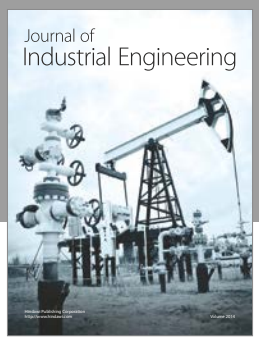
[24] J. Huang and C. X. Ling, "Using AUC and accuracy in evaluating learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 299–310, 2005.

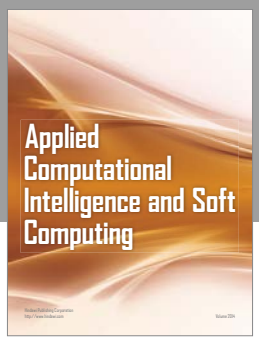
[25] H. He and Y. Ma, *Imbalanced Learning: Foundations, Algorithms, and Applications*, John Wiley and Sons, Hoboken, NJ, USA, 2013.

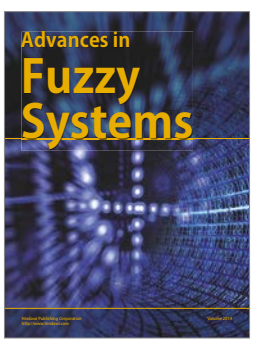


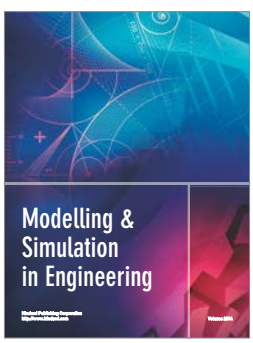


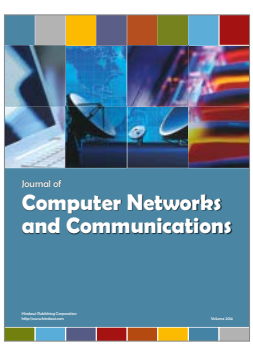
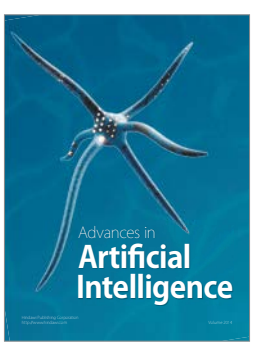
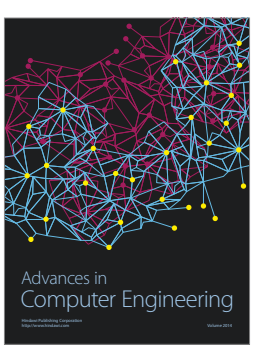
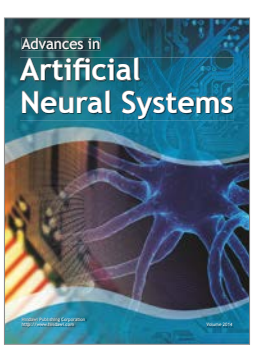
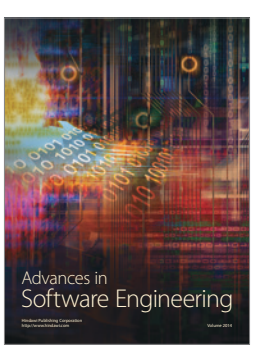
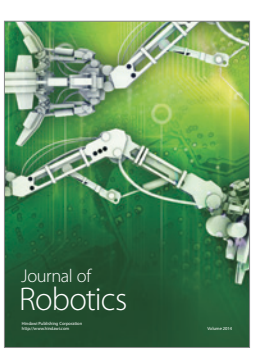
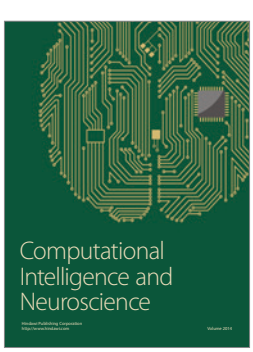












Hindawi
Submit your manuscripts at
https://www.hindawi.com