An Ensemble Method Based on SVC and Euclidean Distance for Classification Binary Imbalanced Data

Lei Zhao[1], Lei Wang[1,2], and Guan Gui[1](✉)

[1] Key Lab of Broadband Wireless Communication and Sensor Network Technology,
Nanjing University of Posts and Telecommunications, Nanjing, China
guiguan@njupt.edu.cn

[2] National Mobile Communications Research Laboratory, Southeast University,
Nanjing, China

Abstract. In recent years, ensemble methods have been widely applied to classify binary imbalanced data. Traditional ensemble rules have manifested performance in dealing with imbalanced data. However, shortage appears that only the results of base classifiers is considered, while these traditional ensemble rules ignore the Euclidean distance between the new data and train data as well as the relations of majority and minority classes in the train data. So we proposed a novel ensemble rule which take Support Vector Classification (SVC) as base classifier. Moreover, the distance between the new data and train data and relations of majority classes and minority classes are taken into account to overcome conventional drawbacks. Simulation results are provided to confirm that the proposed method has better performance than existing ensemble methods.

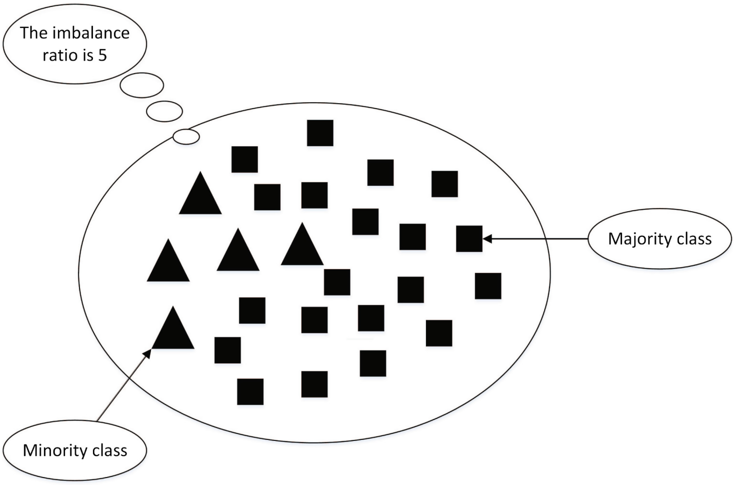
Keywords: Binary imbalanced data · SVC · Euclidean distance
Ensemble rule

1 Introduction

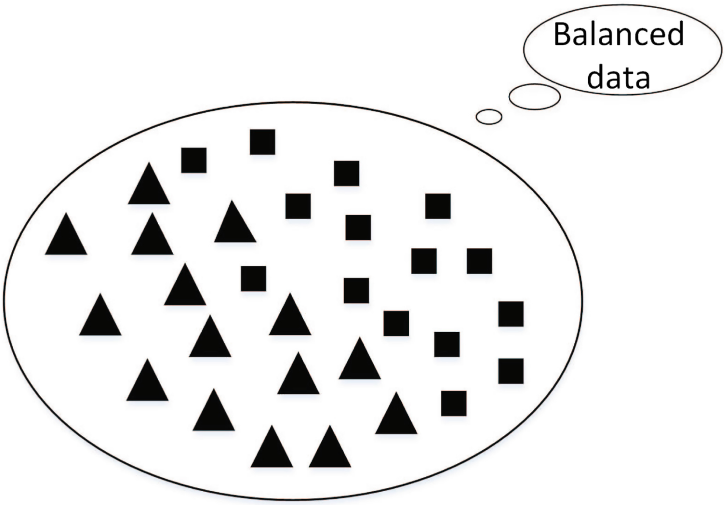
In recent years, due to the importance of imbalanced data in civilian and government applications, such as facial age estimation [1], speech recognition [2] and governmental decision-making support systems [3], imbalanced problems have been developed tremendously. So, researchers have shown greater interest on this crucial area.

The current research in imbalanced problems focuses more on binary imbalanced data, where data set is sorted into majority classes and minority classes. From Fig. 1(a) and (b), we can see the difference between binary imbalanced data and balanced data. Conventional binary imbalanced data classification methods mainly appear in data level and algorithm level. In data level, the main idea is to transform the imbalanced data into balanced data by increasing data or reducing data, such as synthetic minority over sampling technique (SMOTE) which increases minority classes by using k-nearest neighbors (KNN) algorithm. In contrast, while the main idea in algorithm level is to improve algorithm such

M. Huang et al. (Eds.): WICON 2016, LNICST 214, pp. 312–320, 2018.
https://doi.org/10.1007/978-3-319-72998-5_34



(a) An example of binary imbalanced data, where majority class is 25 and minority one is 5.



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

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

as ensemble learning methods [4], and cost-sensitive analysis. But these methods mentioned above share a collaborative shortage that they ignore the Euclidean distance of the new data and train data, and the number of majority and minority classes which gather around the new data.

To overcome above problems, we proposed a novel ensemble method which take Support Vector Classification (SVC) as base classifier and take the distance between the new data and train data into account. In our method, KNN is taken as a filter to wipe out some majority classes whose Euclidean distance close to minority classes; then SVC is employed as the base classifier to classify the obtained balanced data; finally, we use our proposed ensemble rule to combine the classification results obtained by base classifiers. Our ensemble method is innovative while not only takes the relationships between majority classes and minority classes in train data but also that of new data and train data into account. The numerical analysis indicates that our ensemble method has an obvious improvement compared with four traditional ensemble rules including Max rule, Min rule, Product rule, and Sum rule [5]. At the same time, we also compare proposed method with another ensemble method (Splitbal + MaxDistance) [6], where the area under curve (AUC) is validated to be enhanced in comparison.

The rest of this paper is organized as follows: Sect. 2 introduces the related work; Sect. 3 presents our proposed method; Sect. 4 reports the experimental procedure, describes the detailed experimental setup and analyzes the corresponding results; finally, in Sect. 5 we summarize the study and draw the conclusion.

2 Related Work

Over past decades, the binary imbalanced problem has always been a difficult problem in data mining. So far, many methods have been proposed for handling binary imbalanced problem. Traditional binary imbalanced data classification methods mainly include data level and algorithm level [3]. In algorithm level, methods modify existing classification algorithms for adapting them to binary class imbalanced problem, while in data level methods aim to turn the imbalanced data into balanced data. Our method belongs to algorithm level. Next, we will introduce the present situation in algorithm level.

At present, the algorithm level methods include ensemble learning [5], cost-sensitive learning [7] and recognition-based learning [3]. (1) Ensemble learning is used to reduce the variance and bias by integrating the results of many classification algorithms on imbalanced data. Representatively, boosting uses the iterative method to focus on the samples as classified error, so it can obtain a good performance on imbalance problem. (2) Cost-sensitive learning approaches obtain the lowest classification error by adjusting the class misclassification cost. (3) Recognition based learning, RIPPER [6] and auto association provide the discrimination model created on the examples of the target class alone. They have been certified to be effective in dealing with complicated binary imbalanced data in high dimension.

However, these algorithms have collaborative drawbacks that they all ignore the Euclidean distance of the new data and train data as well as the relations of majority and minority classes in the train data. Our method solve both two problems mentioned above simultaneously by adding the Euclidean distance of the new data and train data and the number of majority and minority classes which gather at the new data into the final classification results.

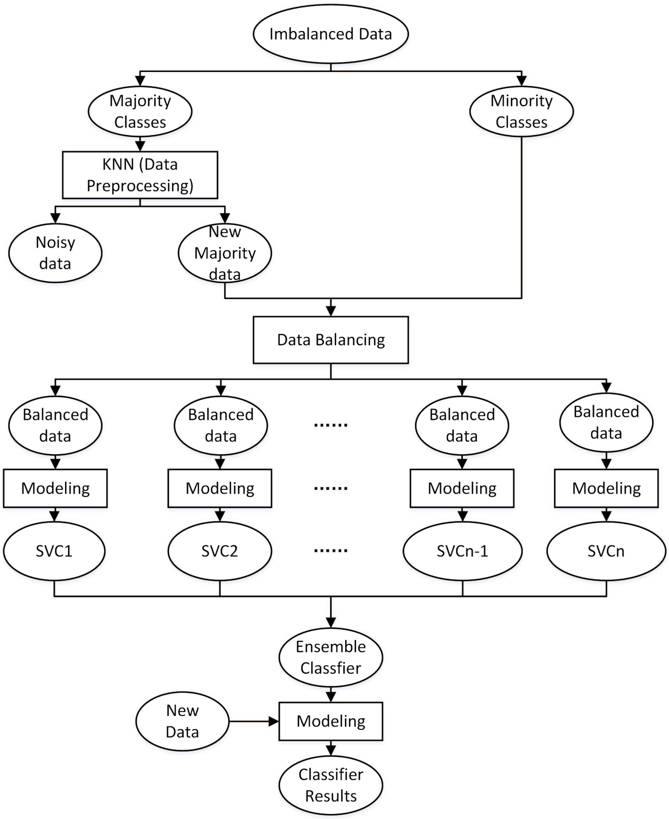


Fig. 2. Framework of our proposed method for handing binary imbalanced data

Table 1. The strategy for traditional rules

Year	World population
Max	$R_1 = argmax_{1\leq i\leq k} P_{i1}, R_2 = argmax_{1\leq i\leq k} P_{i2}$
Min	$R_1 = argmin_{1\leq i\leq k} P_{i1}, R_2 = argmin_{1\leq i\leq k} P_{i2}$
Product	$R_1 = \prod_{i=1}^{k} P^{i1}, R_2 = \prod_{i=1}^{k} P^{i2}$
Sum	$R_1 = \sum_{i=1}^{k} P^{i1}, R_2 = \sum_{i=1}^{k} P^{i2}$

Table 2. The strategy for MaxDistance rule

Rule	Strategy
MaxDistance	$R_1 = argmax_{1\leq i\leq k} \frac{P_{i1}}{D_{i1}+1}, R_2 = argmax_{1\leq i\leq k} \frac{P_{i2}}{D_{i2}+1}$

3 Our Proposed Method

Our proposed method includes four parts: Data Preprocessing, Data Balancing, Modeling, and Classifying. Figure 2 shows the whole process. In our method, we first perform the Data Preprocessing based on KNN; then we divide the majority classes obtained from Data Preprocessing into several sets and each set has the same amount with minority classes; next we use the balanced data set to create the base classifiers with SVC; finally, we use our ensemble rule to combine the results of each base classifiers. In the subsequent, we will introduce the Data Preprocessing and Classifying.

3.1 Data Preprocessing

In this part, we innovatively use KNN as a method of data preprocessing. First of all, we divided train set into majority classes and minority classes. Secondly, we set majority classes as test set, if majority data is classified as minority classes, we judge that such majority data is close to minority data in space and can be stricken out. The reason is that these removed data will influence the classify results if the distances of new data and train data is taken into consideration.

3.2 Classifying

After Modeling, we can obtain several results from each base classifier which based on SVC. Next, we use our ensemble rule to combine these classification results. We make some assumptions as following: assume that there are K SVC classifiers and every data set has two labels, C_1 and C_2. For the ith SVC classifier $(1 \leq i \leq k)$, it classifies the new data as C_1 with the probability P_{i1}, and classifies the new data as C_2 with the probability P_{i2}. In the final classification results, R_1 and R_2 represent the class C_1 and C_2, respectively. In [5], four ensemble rules are described, and the details are shown in Table 1. In our ensemble rule, $D_{ij}(1 \leq i \leq k, 1 \leq j \leq 2)$ represents the average distance between new data and the data with class label C_j in train set, and K_n represents the k closest data around new data using KNN. Moreover, b represents the number of data which belongs to minority classes in the k data. The details of our ensemble rule are shown in Table 3. Table 2 shows another ensemble rule Splital + MaxDistance which comes from [6].

From Table 3 we can see that the rule $R_2 = \frac{average(P_{i2})}{D_{i2}} \times (1 + \frac{b}{K_n})$ has a weight of $(1 + \frac{b}{K_n})$. The reason is that the number of majority classes is less than that of majority classes, and such fact will mislead the classification results

Table 3. The strategy for proposed ensemble rule

Rule	Strategy
Proposed ensemble rule	$R_1 = \frac{average(P_{i1})}{D_{i1}}, R_2 = \frac{average(P_{i2})}{D_{i2}} \times (1 + \frac{b}{K_n})$

Table 4. Statistic summary of the 38 highly imbalanced data sets in experimental study

ID	Data set	ATT	Ins	Mi	IR	ID	Data set	Att	Ins	Mi	IR
1	yeast3	9	1484	163	8.10	20	led7digit	8	443	37	10.97
2	ecoli3	8	336	35	8.60	21	ecoli01vs5	7	240	20	11.00
3	yeast2vs4	9	514	51	9.08	22	glass06vs5	10	108	9	11.00
4	ecoli067vs35	8	222	22	9.09	23	glass0146vs2	10	205	17	11.06
5	ecoli0234vs5	8	202	20	9.10	24	glass2	10	214	17	11.59
6	glass015vs2	10	172	17	9.12	25	ecoli0147vs56	7	332	25	12.28
7	yeast0359vs78	9	506	50	9.12	26	ecoli0146vs5	7	280	20	13.00
8	yeast0256vs3789	9	1004	99	9.14	27	shuttlec0vsc4	10	1829	123	13.87
9	yeast02579vs368	9	1004	99	9.14	28	yeast1vs7	8	459	30	14.30
10	ecoli046vs5	7	203	20	9.15	29	glass4	10	214	13	15.46
11	yeast1289vs7	9	947	30	30.57	30	ecoli4	8	336	20	15.80
12	ecoli0267vs35	8	224	22	9.18	31	pageblocks13vs4	11	472	28	15.86
13	glass04vs5	10	92	9	9.22	32	glass016vs5	10	184	9	19.44
14	ecoli0346vs5	8	205	20	9.25	33	glass5	10	214	9	22.78
15	ecoli0347vs56	8	257	25	9.28	34	yeast2vs8	9	482	20	23.10
16	yeast05679vs4	9	528	51	9.35	35	yeast4	9	1484	51	28.10
17	vowel0	14	988	90	9.98	36	yeast5	9	1484	44	32.73
18	ecoli067vs5	7	220	20	10.00	37	ecoli0137vs26	8	281	7	39.14
19	glass016vs2	10	192	17	10.29	38	yeast6	9	1484	35	41.40

Table 5. The AUC of our ensemble method and the SMD method with 38 data sets

ID	Data set	SMD	SVC-KNN	ID	Data set	SMD	SVC-KNN
1	yeast3	0.8648	**0.9346**	20	led7digit	0.9599	**0.9600**
2	ecoli3	0.9206	**0.9329**	21	ecoli01vs5	**0.9505**	0.9453
3	yeast2vs4	0.8441	**0.9227**	22	glass06vs5	0.9548	**0.9789**
4	ecoli067vs35	**0.8923**	0.8597	23	glass0146vs2	0.7630	**0.8162**
5	ecoli0234vs5	0.9383	**0.9419**	24	glass2	0.7851	**0.8104**
6	glass015vs2	0.7272	**0.7978**	25	ecoli0147vs56	0.9242	**0.9245**
7	yeast0359vs78	0.6999	**0.7512**	26	ecoli0146vs5	**0.9520**	0.9475
8	yeast0256vs3789	0.7875	**0.8107**	27	shuttlec0vsc4	1.0000	1.0000
9	yeast02579vs368	0.9169	**0.9210**	28	yeast1vs7	0.6158	**0.8006**
10	ecoli046vs5	**0.9607**	0.9523	29	glass4	**0.9659**	0.9413
11	yeast1289vs7	0.5761	**0.7375**	30	ecoli4	0.8692	**0.9867**
12	ecoli0267vs35	**0.8891**	0.8523	31	pageblocks13vs4	0.9135	**0.9627**
13	glass04vs5	**0.9667**	0.9633	32	glass016vs5	**0.9529**	0.9324
14	ecoli0346vs5	**0.9500**	0.9444	33	glass5	**0.9325**	0.9125
15	ecoli0347vs56	**0.9390**	0.9323	34	yeast2vs8	0.7547	**0.7765**
16	yeast05679vs4	0.7870	**0.8478**	35	yeast4	0.8341	**0.8704**
17	vowel0	0.9996	**0.9999**	36	yeast5	0.9516	**0.9865**
18	ecoli067vs5	**0.9410**	0.9250	37	ecoli0137vs26	**0.9768**	0.9620
19	glass016vs2	0.7401	**0.8113**	38	yeast6	0.8236	**0.9246**
	Average	**SMD: 0.8742**		**SVC-KNN: 0.9020**			

tending to majority classes, so we add a weight into to balance the imbalance ratio. Finally, the final classification result R_1 and R_2 are obtained with the ensemble rules in Tables 1 and 2, if $R_1 \geq R_2$, the new data is seen as C_1, otherwise C_2. In our experiments, we set K_n as 5, for the specific reasons, please see the fourth section.

4 Simulation Study

4.1 Data Sets

In our paper, we have adopted several binary imbalanced data sets which come from Keel data set repository [8]. The attributes of these data sets are shown in Table 4. Including imbalance radio (IR), total attributes (ATT), total instances, and the number of minority (positive) class instances. For more details of the adopted data sets, please referred to the following cite http://sci2s.ugr.es/keel/imbalanced.php.

In experiments, we used the 5 fold cross validation strategy, furthermore, we set SVC as our basic algorithm. Otherwise, we select AUC [9] as our performance evaluation, because of its superiority to G-mean and F-measure [10] in performance evaluation.

4.2 Experimental Design

Our works in this paper consists of three investigations. The first investigation is to test which value of is better in. The second investigation is to compare our method with Splital + MaxDistance when the basic algorithm is SVC. The third investigation is to make a compare of our ensemble rule with traditional ensemble rules including Max rule, Min rule, Product rule, and Sum rule.

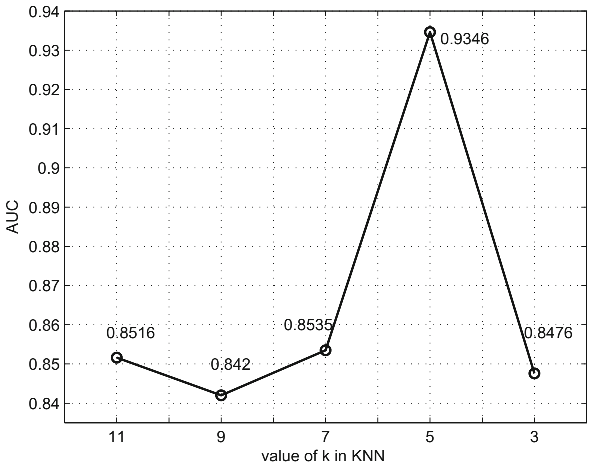


Fig. 3. The results while K_n take different values

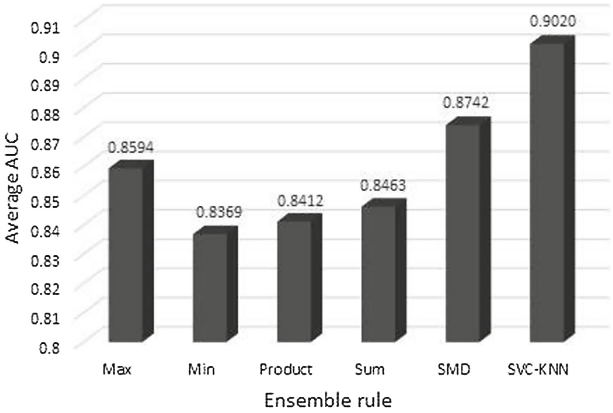


Fig. 4. The average AUC of different ensemble rule with 38 data sets

Investigation 1: We use the data set yeast3 to test the AUC value of our ensemble methods when takes different values (under the situation that its value must be odd). The experiment results are shown in Fig. 3. From the Fig. 3, we can figure out that our method could get best performance when K_n is set to 5.

Investigation 2: In this part, we make a compare of our method with the method Splital + MaxDistance (SMD) and the results are shown in Table 5. From Table 5, we can see that our method has a better performance and there is 2.78% increase than another ensemble method. Moreover, the deep color represent the better performance at each data set.

Investigation 3: For the four traditional ensemble rules, we obtained its average AUC value while SVC is set as basic algorithm with 38 imbalanced data sets. From Fig. 4 we can see that our method (SVC-KNN) acquired the best performance and SMD ranked second.

5 Conclusion

An ensemble method based on SVC and Euclidean distance for classification of binary imbalanced data has been shown in this paper. Different from the existing methods, our proposed ensemble method is innovative while not only takes the connections between majority classes and minority classes in train data but also that of new data and train data into account.

In this paper, we can learn that our proposed method has a visible improvement compared with existing methods while dealing with binary imbalanced data set. But there some problems need to be solved, such as whether our proposed ensemble rule suit to other basic algorithms (Naive Bayes, Random Forest, and so on). So further research on this area is expected in the future.

Acknowledgment. This work was sponsored by National Natural Science Foundation of China (61271240, 61671253); 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 National Mobile Communications Research Laboratory, Southeast University (2016D01); The Open Research Fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology (NUPT), Ministry of Education (NYKL201509).

References

1. Castillo, M.D.D., Serrano, J.I.: A multi strategy approach for digital text categorization. ACM SIGKDD Explor. Newsl. **6**(1), 15–32 (2004)
2. An, A., Cercone, N., Huang, X.: A case study for learning from imbalanced data sets. In: Stroulia, E., Matwin, S. (eds.) AI 2001. LNCS (LNAI), vol. 2056, pp. 1–15. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45153-6_1
3. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
4. Sun, Z., Song, Q., Zhu, X.: Using coding-based ensemble learning to improve software defect prediction. IEEE Trans. Syst. Man Cybern. Part C **42**(6), 1806–1817 (2012)
5. Kittler, J., Hatef, M., Duin, R.P., Matas, J.: On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **20**(3), 226–239 (1998)
6. Sun, Z., Song, Q., Zhu, X., et al.: A novel ensemble method for classifying imbalanced data. Pattern Recogn. **48**(5), 1623–1637 (2015)
7. Zheng, J.: Cost-sensitive boosting neural networks for software defect prediction. Expert Syst. Appl. **37**(6), 4537–4543 (2010)
8. Alcal, J., Fernndez, A., Luengo, J., Derrac, J., Garca, S., Snchez, L.: KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J. Multiple-Valued Logic Soft Comput. **17**(2), 255–287 (2011)
9. Huang, J., Ling, C.X.: Using AUC and accuracy in evaluating learning algorithms. IEEE Trans. Knowl. Data Eng. **17**(3), 299–310 (2005)
10. He, H., Ma, Y.: Imbalanced Learning: Foundations, Algorithms, and Applications. Wiley, Hoboken (2013)