Comparative Study of Pair wise learning in Imbalanced Data Problem Using Different Classification Techniques

Fatima Shakeel
Allana Institute of Management Sciences, Pune, Maharashtra, India.

Abstract: Classification and prediction of rare cases in the patients is a very important task. In such cases the data sets are mostly imbalanced because the number of majority class (negative) outnumbers the minority class (positive) by a large proportion. There is an uneven distribution between the two classes. A lot of research has been conducted on binary class imbalanced data where one class (major) outnumbers the other class (minor) by a huge marginal difference. This research work focuses towards the classification of a data where there are multiple minor classes. Two approaches have been compared using ensemble methods like boosting and bagging and check the overall accuracy of minority classes.

Keywords: Bagging, AdaBoost, One versus One (OVO), One versus All (OVA)

I. INTRODUCTION

The imbalanced data problem can appear in two different types of data sets 1. Binary problems, where one of the two classes comprises considerably more samples than the other. 2. Multi-class problems, where the applications have more than two classes and unbalanced class distribution hinders the classification performance. Most research efforts on imbalanced data sets have traditionally concentrated on two-class problems. However, this is not the only condition where the class imbalance problem prevails. In the case of multi-class data sets, it is much more difficult to define the majority and minority classes. We focus on multi-class classification in imbalanced data sets. Evaluation measures play a crucial role in both assessing the classification performance and guiding the classifier modelling. Traditionally, accuracy is the most commonly used measure for these purposes. The accuracy of a classifier on a given test set is the percentage of test set instances that are correctly classified by the classifier. So, it is important to know that when the performances of all classes are interested, classification performance of each class should be equally represented in the evaluation measure. Though accuracy cannot give us the exact thing that is important here but still we calculate the overall accuracy to check how accurate a learner classifies the data set despite of the fact that data set is an imbalanced one [1][2][3][4]. Mathematically accuracy is defined as:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \qquad (1)$$

Where TP is the number of true positives. TN is the number of true negatives. FP is the number of false positives. FN is the number of false negatives.

Another important measure that actually we are concerned about is the number of minor classes that are classified correctly. So if a minor class instance is classified as major class instance, we have to find and define it as it becomes the false negative instance but actually is not. Therefore, calculation of false negative rate [5][2] becomes important to check how accurately the technique works in mining imbalanced data and where it is lagging.

$$\text{False negative rate} = \frac{FN}{FN+TP} \qquad (2)$$

Equation (1) and (2) gives the mathematical formulation of both accuracy and false negative rate.

II. LITERATURE SURVEY

Research work on patient data set shows the data has multiple majority and minority classes. Whether a patient has type1 diabetes or type2 diabetes or is not diabetic at all is a case of multi-class imbalanced data set. In pair wise learning a data set is segmented into binary class subsets. There are two techniques that have been introduced so far for pair wise learning. These are:- 1) One versus one (OVO): in this pair wise learning each class in the data set is compared with each other class to evaluate the accuracy of segmented data sets. All the cases are combined and the overall result is evaluated. Here the data set gets reduced and therefore it reduces the load on a learner. 2) One versus all (OVA): this is also known as one versus rest. One class is compared with the rest of the classes. All the other classes are combined in one class and each class is then compared with the remaining class taken as one. This might increase the imbalance ratio thereby making a learner's job more tedious.

Alberto Fern´andez proposed an OVO method combined with the SMOTE and has used linguistic fuzzy rule as the base classifier [6]. In the first step data set is divided and OVO result is calculated if the imbalance ratio is less than 1.5 otherwise OVO and SMOTE are used collectively to calculate results. The algorithm used is genetics based namely Fuzzy Hybrid Genetics Based Machine Learning (FH-GBML) ... Xu Ying Liu et al proposed a technique to solve a multi class imbalance problem. This technique is based on pair wise learning in which each binary class data is considered as an individual dichotomy. Weights are assigned to each dichotomy which is then used for decoding. This technique is named as imECOC (imbalanced Error Correcting Output Codes) and has three stages [7]. In the first stage the code matrix is developed. Second is the learning strategy stage and the last is the decoding strategy stage. Li Min Du et al [8] proposed a technique based on the selection of relevant features that are beneficial in recognizing the minority classes. SVM has been used as a base classifier and genetic algorithm is used to select the best suited features. This genetic algorithm is said to improve the fitness function. Hongli Zhang et al [9] proposed a technique based on data gravitation computation (DGC) which is based on the Newton's Law of Gravitation. The results were carried out for both binary class and multi-class imbalanced data sets. An amplified gravitation co-efficient (AGC) is used to hold the imbalance information. It amplifies the gravitational computations. It also optimizes the weight updating procedure to ensure that model would adapt the imbalanced class distribution. It is difficult to find an optimal code matrix. Sampling with ensembles like boosting and bagging have been used to directly mine the imbalanced multi class data sets. One such approach is the AdaBoost.NC which was proposed by Shuo Wang [10]. It uses AdaBoost with negative correlation plus random over sampling. This is a hybrid direct method for multi class imbalanced learning. One more hybrid method is there that uses decision tree as the base classifier. This method is known as Hellinger Distance Decision Tree (HDDT) [11]. This method is specially known as MC-HDDT [12] where MC stands for multi class. Other variants of AdaBoost that were used in mining multi class data sets are AdaBoost.M1 and AdaBoost.M2 [2][10][13]. These overcome the restriction on the limited error weights on the classifiers.

Table 1. Various techniques evolved for multi class imbalanced data problems

S.NO.	Reference NO.	Year	Method	Algorithm used	Pre-processing Technique	Performance Measures
1	[13]	2006	Direct	AdaC2.M1	-	F-measure, Accuracy, G-mean
2	[6]	2010	OVO	Linguistic Fuzzy Rule Based Classification Systems	SMOTE	AUC
3	[14]	2012	Direct	Ramp Loss Multi class SVM	-	AUC, G-mean
4	[10]	2012	Direct	AdaBoost.NC	Random oversampling	Recall, precision, F-measure, G-mean, MAUC
5	[7]	2013	OVO and OVA	ImECOC	-	F-measure, G-mean, MAUC
6	[9]	2014	OVO and OVA	DGC	-	G-mean, AUC
7	[8]	2015	-	Genetic algorithm	Feature Selection	G-mean Extension

From the above table 1 it is evident that sampling is hardly used in the techniques that are used to classify multi-class imbalanced data sets. Recently more research has been done on pairwise learning rather than on a direct method. It is yet a challenge to propose a direct method for multi-class imbalanced data that will prove better than other proposed direct techniques and indirect techniques. Also G-mean, F-measure and AUC are the evaluation measures often used in case of multi class imbalanced data mining.

III. DATA SET COLLECTION

The data set has been collected from KEEL data set repository which is an open source tool. It contains a wide variety of data sets. We have selected a multi-class imbalanced data set 'ECOLI' which is a bacterium found in the lower intestine of warm blooded organisms. The bacteria can lead to severe anaemia and kidney failure and can lead to death. Therefore predicting the presence of these bacteria in a living being is very important. The data set has 7 attributes namely mcg, gvh, lip, chg, aac, alm1 and alm2. It has 8 classes and the number of instances is 336. The class distribution is as following:

Cp: 144, Im:77, ims: 2, ImL: 2, ImU: 35, Om: 20, Oml: 5, PP: 52.
The summary of data set is given in table 2.

The figure 1 below shows the snapshot of the data

Attributes	7	
Classes	8	
Instances	336	
Class distribution	Cp	144
	Im	77
	Ims	2
	Iml	2
	Imu	35
	Om	20
	Oml	5
	pp	52
Data type	real	

Table 2. Information about the data set used

set used.

	A	B	C	D	E	F	G	H	I	J
1	@relation	Ecoli-Bal								
2	@attribute	mcg	real	[0.0	0.89]	McGeoch's method for signal sequence recognition.				
3	@attribute	gvh	real	[0.16	1.0]	von Heijne's method for signal sequence recognition				
4	@attribute	lip	real	[0.48	1.0]	von Heijne's Signal Peptidase II consensus sequence score.				
5	@attribute	chg	real	[0.5	1.0]	Presence of charge on N-terminus of predicted lipoproteins.				
6	@attribute	aac	real	[0.0	0.88]	score of discriminant analysis of the amino acid content of outer membrane and periplasmic p				
7	@attribute	alm1	real	[0.03	1.0]	score of the ALOM membrane-spanning region prediction program.				
8	@attribute	alm2	real	[0.0	0.99]	score of ALOM program after excluding putative cleavable signal regions from the sequence.				
9	@attribute	class	cp (Cyptoplasm)	im (inner membrane)	pp (periplasm)	imU (inner membrane, non cleavable signal sequence)	om (outer membrane)	omL (outer membrane lipoprotein)	imL (inner membrane lipoprotein)	imS (inner membrane, cleavable signal sequence)
10	@inputs	mcg	gvh	lip	chg	aac	alm1	alm2		
11	@outputs	class								
12	@data									
13	0.49	0.29	0.48	0.5	0.56	0.24	0.35	cp		
14	0.07	0.4	0.48	0.5	0.54	0.35	0.44	cp		
15	0.56	0.4	0.48	0.5	0.49	0.37	0.46	cp		
16	0.59	0.49	0.48	0.5	0.52	0.45	0.36	cp		
17	0.23	0.32	0.48	0.5	0.55	0.25	0.35	cp		
18	0.67	0.39	0.48	0.5	0.36	0.38	0.46	cp		
19	0.29	0.28	0.48	0.5	0.44	0.23	0.34	cp		
20	0.21	0.34	0.48	0.5	0.51	0.28	0.39	cp		
21	0.2	0.44	0.48	0.5	0.46	0.51	0.57	cp		
22	0.42	0.4	0.48	0.5	0.56	0.18	0.3	cp		
23	0.42	0.24	0.48	0.5	0.57	0.27	0.37	cp		
24	0.25	0.48	0.48	0.5	0.44	0.17	0.29	cp		
25	0.39	0.32	0.48	0.5	0.46	0.24	0.35	cp		
26	0.51	0.5	0.48	0.5	0.46	0.32	0.35	cp		
27	0.22	0.43	0.48	0.5	0.48	0.16	0.28	cp		
28	0.25	0.4	0.48	0.5	0.46	0.44	0.52	cp		
29	0.34	0.45	0.48	0.5	0.38	0.24	0.35	cp		

ecoli

Figure 1. Snapshot of the data set

Since in this study we are comparing pair wise learning techniques (OVO and OVA), it is necessary to compute the imbalance ratio between each class and between each class with the rest of the classes.

Table 3. One versus one data segmentation.

	Classes	Imbalance ratio		class	Imbalance ratio
Cp versus	Im	1.85	Ims versus	Iml	1
	Ims	71.5		Imu	17.5
	Iml	71.5		Om	10
	Imu	4.08		Oml	2.5
	Om	7.15		pp	2.65
	Oml	28.5			
	Pp	2.75			

Im versus	Ims	38.50	Imu versus	Om	1.75
	Iml	38.50		Oml	7
	Imu	2.2		Pp	1.48
	Om	3.85			
	Oml	15.4			
	Pp	1.48			
Om versus	Oml	4	Oml versus	pp	10.4
	Pp	2.6			

Table 4. One versus all (rest) data segmentation

	Imbalance ratio
Cp versus rest	1.34
Im versus rest	3.37
Ims versus rest	167.5
Iml versus rest	167.5
Imu versus rest	8.71
Om versus rest	15.85
Oml versus rest	60.4
Pp versus rest	5.48

Research witnesses the effectiveness of ensemble methods when compared with other non-ensemble techniques. Since ensembles use the combination of series of algorithmic and pre-processing approaches. In this study we have used two ensemble methods namely Boosting (AdaBoost) and Bagging with some base classifiers or machine learners.

IV. EXPERIMENTAL SET UP AND RESULTS

Different machine learning classifier for our experiments namely ID3, naïve Bayes, decision tree, K-NN, neural network and SVM with its variants on the selected dataset to classify the instances were used. All classifiers were run on machine environment of 64-bit with 4GB RAM. We apply set of experiments, in the first section. The instances were classified using ensemble method AdaBoost and bagging separately. The AdaBoost and Bagging techniques were used in two different approaches: 1) OVO and 2) OVA without sampling.

Since the aim of this research is to find the possibility of mining multi class imbalanced data without sampling the data set initially, we compare the results (accuracy and false negative rate) of the two proposed methods one-versus-one (OVO) and one-versus-all (OVA) using boosting and bagging ensemble methods. The study was evaluated in rapid miner tool.

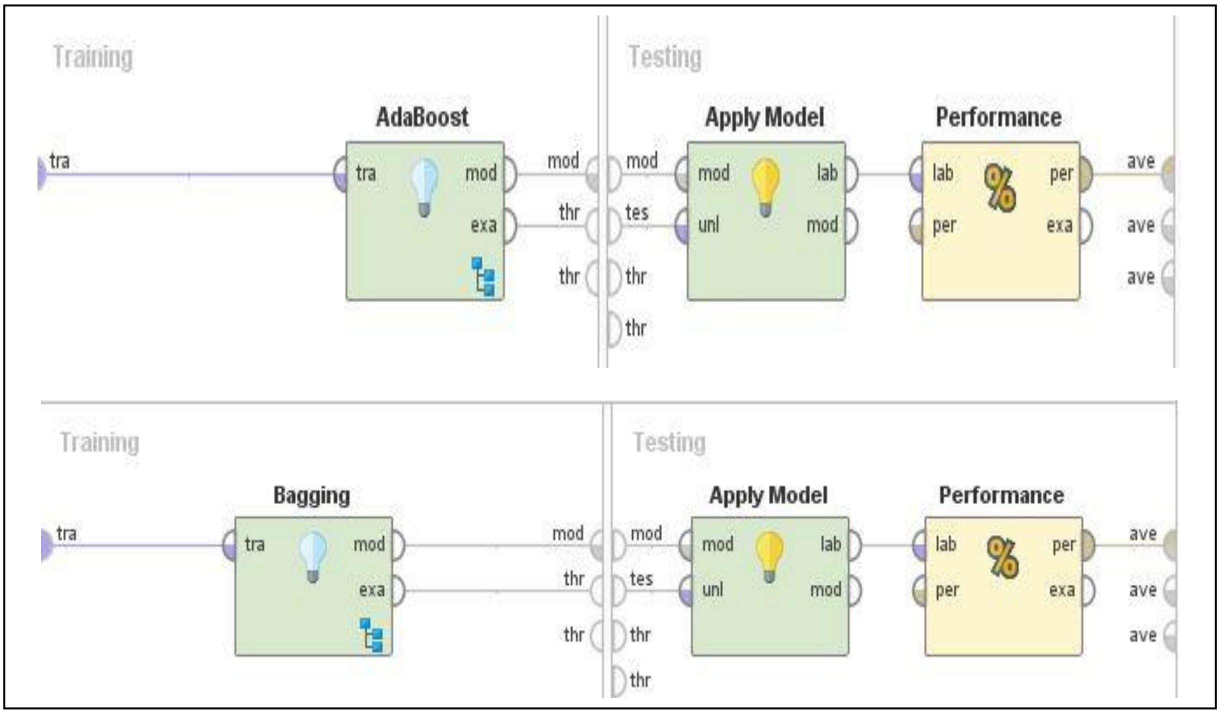

Figure 2. Experimental setup of AdaBoost and Bagging

Experiment 1: OVO

1. In the first experiment data was manually segmented into sub data sets for pair-wise learning i.e; OVO.
2. A total of 28 sub sets were generated and each were fed to read excel operator.
3. A split validation operator with split ratio of 0.7 is used to evaluate the Adaboost containing ID3 and performance operator.
4. Same procedure is repeated using decision tree, K-NN, belief networks, artificial neural networks, svm and svm(pso). All the above process is repeated using bagging. The model construction through rapid miner is shown in fig and the results are formulated in the appendix A.

Experiment 2: OVA

From the original data set, 8 sub data sets were generated. Each sub data set contains the all the instances of one class gaainst the instances of rest classes. 8 data sets were thus fed to the read excel operator seperately to evaluate its over all accuracy and false negative rate using ensembles boosting and bagging with ID3. Tha same process was repeated with decision tree, K-NN, naïve bayes, naïve bayes(kernel), artificial neural network(ANN) , SVM and SVM(PSO). Results are shown in appendic B and appendix C.

To study the output of ensemble methods i.e: AdaBoost and Bagging, various analysis were conducted. The performance measures used are:

I. Overall accuracy

It is the the degree to which the result of a measurement, calculation, or specification conforms to the correct value or a standard. In diagnosing a disease it is very important that the technique gives a good accuracy of predicting both the positivr cases as well as the negative cases. Although classification error of negative case wont effect the critical disease diagnosis but it would be better that if the technique can predict both positive and negative cases accurately as much as possible. Therefore, in this casse study it is very important to evaluate the overall accuracy of a machine learner. The more the accuracy, the better is the learner. Comparing the results or outputs, it is evident from the results table that svm(pso) used with AdaBoost gives the good overall accuracy. For example if we take the imbalance ratio between cp and oml, the ratio is 28.5%. sVM (PSO) with AdaBoost gave a good overall accuracy of 99.18%. Similarly in case of cp versus ims where ratio is 71.5 svm(pso) gives the best accuracy of 97.65%.

II. False negative rate

False negative(FN) is the sample that has been falsely predicted as negative and is actually a positive instance. This performance measure is very important in this case study after all a disease is being predicted in patients. The less the FNR the better is the prediction of minor class or cases in the classification. In this case study since we are predicting a disease, it is very important that the positive cases are predicted as positive and not negative. However, if negative cases are misclassified as positive, it wont damage in real sense but the opposite is not true. Here we are comparing various machine learners using ensemble methods to detect which among the learners is the best classifier. From our result table it is evident that SVM(PSO) produces lowest FNR (zero in maximum cases) in a large number of cases compared to others.

A. Result Analysis

After analysing the results, the performance of any of the machine learners used depends on the data set size and the imbalance between each class of the data set. From figure 3, figure 4, figure 5 and figure 6, we can infer that SVM(PSO) shows good accuracy and less FNR when the imbalance ratio is high more than 4. If the imbalance ratio is less than 2, even ANN, bayesian classification works good.

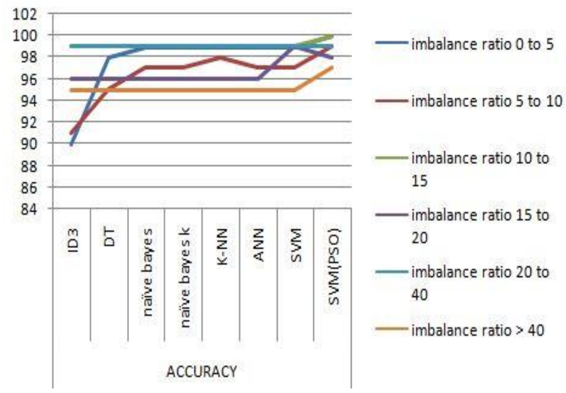

Figure 3. Accuracy of learners versus imbalance ratio using AdaBoost

It is noted that when IMS and IML were compared against all others (OVA) where the imbalance ratio was found to be 167.5 , none of the compared learners chould actually find the minority classes(IMS and IML). It may be because of the fact that SVM works on the instances on or close to the hyperplane. Since there were only 2 instances of IMS and IML, it is possible that these instances were far from seperating hyperplane and therefore got missclassified. Comparing the results of bagging and boosting, AdaBoost showed a good overall accuracy in OVO learning with a less rate of false negatives. This is shown in figure 4 and figure 5.

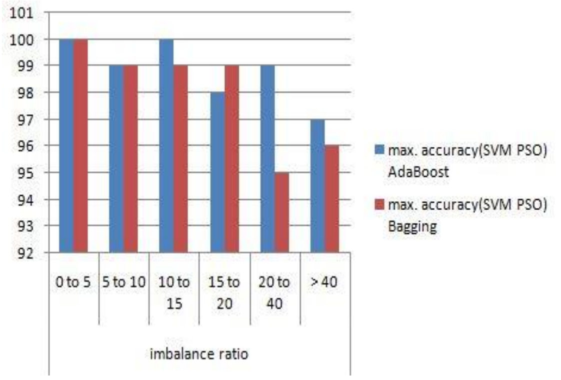

Figure 4. AdaBoost and Bagging accuracy of SVM (PSO) in OVO method

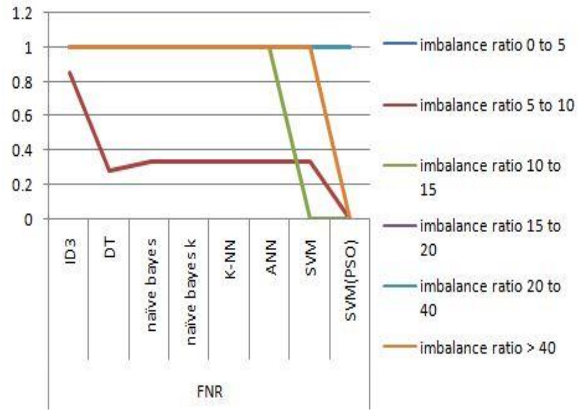

Figure 5. False negative rate (FNR) versus imbalance ratio using machine learners with AdaBoost

However, there is not much difference in the results shown by bagging and boosting fig 5, still maximum times AdaBoost showed good performance measures with less number of false negatives.

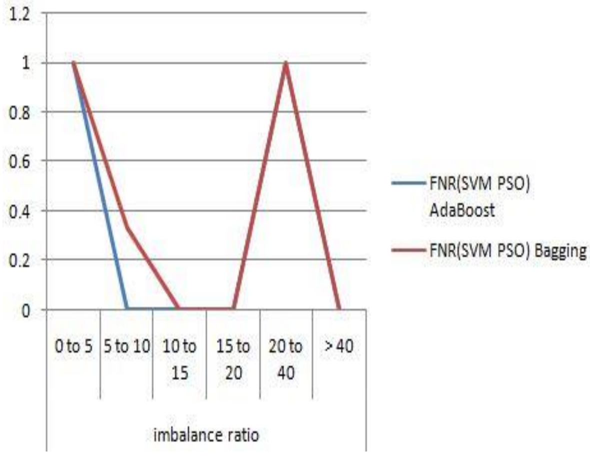

Figure 6. SVM(PSO) false negative rate (FNR) of AdaBoost versus Bagging

OVO pair wise learning showed some good results than OVA. In OVA often the negatives get misclassified as positive and positives as negative which was seen very little when OVO was evaluated.

- When imbalance ratio was high between 4 and 100, SVM(PSO) showed good results. It is because of its evolutionary nature that it remembers its previous state and at every iteration it tries to find an optimal solution.
- AdaBoost proved better than bagging because bagging depends on the votes of the series of the learners used. So votes can not sometimes lead to the optimal solution as they can go wrong where as in AdaBoost at every iteration gives more importance to the misclassified samples by adding weights to them. Hence AdaBoost works better than Bagging
- In OVA method there is more chance of increase in the imabalance ratio and noisy data. Thus accuracy gets affected when OVA learning is considered while in OVO instance or data set gets reduced thereby decreasing the load on the learner.

V. CONCLUSION AND FUTURE WORK

From the comparative experimental study, it was found that evolutionary optimized svm works good when ova pair wise learning was evaluated with ensemble method (AdaBoost). Our aim was to check the effect of various machine learners on a multi-imbalanced dat sets without sampling. Sampling either leads to overfiting or removal of important data. Although many researches have been done to avoid overfitting and information loss due to sampling, but still there is a much need to make the classification and prediction more accurate. Much Much work is needed to overcome the limitation of sampling and to make it work in a better way. In future we will study the prediction pattern using all the combinations of preprocessing (feature selection and sampling) techniques with some ensemble and hybrid methods. A direct method is needed for the multi class data problem that can prove better than all other techniques.

REFERNCES

[1] Chawla, Nitesh V. "Data mining for imbalanced datasets: An overview." *Data mining and knowledge discovery handbook*. Springer US, 2005. 853-867.

[2] Galar, Mikel, et al. "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.4 (2012): 463-484.

[3] Ali, Aida, Siti Mariyam Shamsuddin, and Anca L. Ralescu. "Classification with class imbalance problem: A Review." *Int. J. Advance Soft Compu. Appl* 7.3 (2015).

[4] Fernández, Alberto, María José del Jesus, and Francisco Herrera. "On the 2-tuples based genetic tuning performance for fuzzy rule based classification systems in imbalanced data-sets." *Information Sciences* 180.8 (2010): 1268-1291.

[5] Sun, Yanmin, Andrew KC Wong, and Mohamed S. Kamel. "Classification of imbalanced data: A review." *International Journal of Pattern Recognition and Artificial Intelligence* 23.04 (2009): 687-719.

[6] Fernández, Alberto, Mara José Del Jesus, and Francisco Herrera. "Multi-class imbalanced data-sets with linguistic fuzzy rule based classification systems based on pairwise learning." *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer Berlin Heidelberg, 2010.

[7] Liu, Xu-Ying, Qian-Qian Li, and Zhi-Hua Zhou. "Learning imbalanced multi-class data with optimal dichotomy weights." *Data Mining (ICDM), 2013 IEEE 13th International Conference on*. IEEE, 2013.

[8] Du, Li-min, Yang Xu, and Hua Zhu. "Feature Selection for Multi-Class Imbalanced Data Sets Based on Genetic Algorithm." *Annals of Data Science* 2.3 (2015): 293-300.

[9] Peng, Lizhi, et al. "A new approach for imbalanced data classification based on data gravitation." *Information Sciences* 288 (2014): 347-373.

[10] Wang, Shuo, and Xin Yao. "Multiclass imbalance problems: Analysis and potential solutions." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 42.4 (2012): 1119-1130.

[11] Cieslak, David, and Nitesh Chawla. "Learning decision trees for unbalanced data." *Machine learning and knowledge discovery in databases* (2008): 241-256.

[12] Hoens, T., et al. "Building decision trees for the multi-class imbalance problem." *Advances in knowledge discovery and data mining* (2012): 122-134.

[13] Sun, Yanmin, Mohamed S. Kamel, and Yang Wang. "Boosting for learning multiple classes with imbalanced class distribution." *Data Mining, 2006. ICDM'06. Sixth International Conference on*. IEEE, 2006.

[14] Phoungphol, Piyaphol, Yanqing Zhang, and Yichuan Zhao. "Robust multiclass classification for learning from imbalanced biomedical data." *Tsinghua Science and Technology* 17.6 (2012): 619-628.

		Id3				Decision tree				Naïve bayes				Naïve bayes kernel				k-nn				ann				svm				Svm(pso)			
		Adaboost		bagging		adaboost		bagging		adaboost		bagging		adaboost		bagging		adaboost		bagging		Adab00st		bagging		adaboost		bagging		adaboost		bagging	
		ACC	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**	**ACC**	**FNR**
Cp versus	im	31.82	1	31.82	1	81.82	0.26	96.97	0.04	93.94	0.17	93.94	0.17	80.3	0.08	90.91	0.08	87.88	0.28	87.88	0.13	92.42	0.17	90.91	0.17	87.88	0.11	87.88	0.13	**98.48**	0.02	**98.48**	0.02
	ims	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	**97.67**	0	96.32	0
	iml	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	95.35	1	**97.67**	0	96.32	0
	imu	96.23	0.22	96.23	0.22	**98.11**	0.11	83.02	1	94.34	0.11	94.34	0.11	94.34	0.11	94.34	0.11	**98.11**	1	92.45	0.44	**98.11**	0.11	**98.11**	0.11	**98.11**	0.11	**98.11**	0.11	90.57	0.11	**98.11**	0.11
	om	87.76	0.85	87.76	0.85	95.92	0.28	85.71	1	97.96	0.14	95.92	0.20	97.96	0.14	95.92	0.20	97.96	0.14	85.71	1	97.96	0.14	97.96	0.14	97.96	0.14	97.96	0.14	**99.9**	0	**99.90**	0
	oml	99.13	0	99.13	0	90.91	1	90.91	1	99.13	0	93.18	0	99.13	0	93.18	0	99.13	0	93.18	0.75	99.23	0	99.23	0	99.41	0	99.41	0	**99.18**	0	95.45	0
	pp	75.86	0.28	74.14	0.28	87.93	0.14	93.10	0	93.10	0	93.10	0.07	94.83	0.07	94.83	0.07	94.83	0.07	94.83	0.07	94.83	0	**94.83**	0	**94.83**	0	**94.83**	0	87.93	0	91.38	0
Im versus	ims	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	**95.83**	**1**	**95.83**	1
	iml	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	95.83	1	**95.83**	**1**	**95.83**	1
	imu	67.65	0.72	67.65	0.72	79.41	0.45	**82.35**	0.45	64.71	1	64.71	1	64.71	0.45	75.53	0.36	76.47	0.27	76.47	o.27	79.41	0.27	76.47	0.36	76.47	0.36	67.65	1	79.41	0.45	73.53	0.36
	om	79.31	0.66	79.31	0.66	96.55	0	96.55	0.99	99.10	0	99.10	0	99.10	0	99.10	0	99.10	0	99.10	0	99.10	o	99.10	0	99.35	0	99.35	0	**99.90**	0	**99.90**	0
	oml	96.00	0	96.00	0	96.00	0	96.00	0	96.00	0	96.00	0	96.00	0	96.00	0	98.00	0	98.00	0	96.00	0	96.00	0	99.10	0	99.10	0	98.00	0	**99.99**	0
	pp	87.18	0.25	87.18	0.25	89.74	0	92.31	0.16	97.44	0	94.87	0.16	94.87	0.08	94.87	0.08	97.44	0	97.44	0	94.87	0	94.87	0.08	94.87	0	97.44	0	**99.99**	0	**99.99**	0
Ims versus	iml	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0
	imu	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0	99	0
	om	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	85.71	1	14.29	0	14.29	0	85.71	0	85.71	0
	oml	**99.10**	0	**99.10**	0	50.00	1	50.00	1	50.00	1	50.00	1	50.00	1	50.00	1	99.10	0	99.10	0	50.00	1	50.00	1	50.00	1	50.00	1	50.00	1	50.00	1
	pp	87.50	1	87.50	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	1	93.75	0	6.25	0	6.25	0	**93.75**	0	**93.75**	0
Imu versus	om	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	75.00	0.66	99.10	0	99.10	0	**99.99**	0	**99.99**	0
	oml	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0.33	91.67	0	91.67	0.33
	pp	96.15	0.11	96.15	0.11	84.62	0	84.62	0	88.64	0	88.64	0	96.15	0.11	96.15	0	99.10	0	99.10	0	96.15	0	96.15	0	38.46	0	38.46	0	**99.99**	0	**99.99**	0
Om versus	oml	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0	85.71	0
	pp	86.36	1	86.36	1	99.01	0	99.01	0	99.10	0	99.10	0	99.10	0	99.10	0	99.01	0	99.01	0	**99.10**	0	**99.10**	0	**99.10**	0	**99.10**	0	95.45	0	95.45	0
Oml versus	pp	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.01	0	99.57	0	99.57	0	**99.99**	0	**99.99**	0

Appendix A. accuracy and false negative rate (FNR) of OVO using AdaBoost and Bagging with given learners

AdaBoost																
	CP versus All		IM versus all		IMS versus all		IML versus all		IMU versus all		OM versus all		OML versus all		PP versus all	
	Accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR
ID3	78.22	0.36	81.19	0.34	99.19	1	99.19	1	89.11	0.54	95.05	0.66	97.03	0	87.13	0.56
Decision tree	99.01	0.09	91.09	0.21	99.09	1	99.09	1	94.06	0.45	95.05	0.33	98.02	0	93.07	0.37
K-NN	95.05	0.06	86.14	0.39	99.9	1	99.9	1	94.06	0.27	97.03	0.16	99.01	0	93.07	0.12
Naïve bayes	96.04	0.09	70.30	0.04	99.9	1	99.9	1	90.10	0.09	82.18	0.16	98.02	0	85.15	0
Naïve bayes kernel	97.03	0.06	92.08	0.21	99.9	1	99.9	1	93.07	0.45	97.03	0.16	99.01	0	94.06	0.31
ANN	98.02	0.04	90.10	0.17	99.9	1	99.9	1	93.07	0.54	97.03	0	99.01	0	94.06	0.25
SVM	99.01	0.02	89.11	0.26	99.9	1	99.9	1	92.08	0.63	99.9	0	98.10	0	91.09	0.43
SVM(PSO)	99.01	0.02	81.19	0	67.33	0	67.33	0	71.29	0	67.33	0	66.34	0	89.11	0

Appendix B. OVA accuracy and FNR results using Adaboost with different base classifiers

BAGGING																
	CP versus All		IM versus all		IMS versus all		IML versus all		IMU versus all		OM versus all		OML versus all		PP versus all	
	Accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR	accuracy	FNR
ID3	80.20	0.09	81.19	0.39	99.9	1	99.9	1	86.14	0.90	95.05	0.66	97.03	0	87.13	0.56
Decision tree	96.04	0.04	92.08	0.17	99.9	1	99.9	1	94.06	0.54	98.02	0.16	97.03	0	93.07	0.37
K-NN	95.05	0.06	86.14	0.39	99.9	1	99.9	1	94.06	0.27	97.03	0.16	99.01	0	93.07	0.12
Naïve bayes	96.04	0.09	70.30	0.04	98.02	1	98.02	1	88.12	0	84.16	0.16	98.02	0	85.15	0
Naïve bayes kernel	97.03	0.06	86.14	0.13	99.9	1	99.9	1	93.07	0	97.03	0.16	99.01	0	94.06	0.12
ANN	98.02	0.04	93.07	0.17	99.9	1	99.9	1	95.05	0.36	96.04	0	99.01	0	94.06	0.12
SVM	99.01	0.02	87.13	0.30	99.9	1	99.9	1	93.07	0.63	99.01	0	97.03	0	91.09	0.43
SVM(PSO)	99.01	0.04	83.17	0.08	68.32	0	68.32	0	73.20	0	66.34	0	63.37	0	89.11	0

Appendix C. OVA accuracy and FNR results using Bagging with different base classifiers