WILEY InterScience® DISCOVER SOMETHING GREAT

PROTEINS STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

PREDICTION REPORT

Protein classification with imbalanced data

Xing-Ming Zhao,[1,2,3] Xin Li,[4] Luonan Chen,[1,3,5,6]* and Kazuyuki Aihara[1,3]*

[1] ERATO Aihara Complexity Modelling Project, JST, Tokyo 151-0064, Japan

[2] Intelligent Computing Lab, Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, Anhui, 230031, China

[3] Institute of Industrial Science, The University of Tokyo, Tokyo 153-8505, Japan

[4] Department of Computer Science, Hong Kong Baptist University, Hong Kong

[5] Department of Electrical Engineering and Electronics, Osaka Sangyo University, Osaka 574-8530, Japan

[6] Institute of Systems Biology, Shanghai University, Shanghai 200444, China

ABSTRACT

Generally, protein classification is a multi-class classification problem and can be reduced to a set of binary classification problems, where one classifier is designed for each class. The proteins in one class are seen as positive examples while those outside the class are seen as negative examples. However, the imbalanced problem will arise in this case because the number of proteins in one class is usually much smaller than that of the proteins outside the class. As a result, the imbalanced data cause classifiers to tend to overfit and to perform poorly in particular on the minority class.

This article presents a new technique for protein classification with imbalanced data. First, we propose a new algorithm to overcome the imbalanced problem in protein classification with a new sampling technique and a committee of classifiers. Then, classifiers trained in different feature spaces are combined together to further improve the accuracy of protein classification. The numerical experiments on benchmark datasets show promising results, which confirms the effectiveness of the proposed method in terms of accuracy. The Matlab code and supplementary materials are available at http://eserver2.sat.iis.u-tokyo.ac.jp/~xmzhao/proteins.html.

Proteins 2008; 70:1125–1132.

Key words: ensemble classifier; feature extraction; hybrid sampling; multi-class classification; rebalancing technique.

INTRODUCTION

Annotating new sequenced proteins with structural and functional features is one of the core problems in computational biology. One way to approach this problem is to classify a new protein into some certain known protein class so that the structural and functional features of the query protein can be easily identified. This problem is usually referred to as protein classification. In the literature, a variety of methods, e.g., PSI-BLAST,[1] position-specific weight matrices,[2] and Hidden Marked Models (HMM),[3] have been developed for protein classification. Recently, machine learning techniques, e.g., Support Vector Machines (SVMs)[4–7] and neural networks,[8] have also been successfully applied to protein classification and shown the superiority to other methods.

Generally, protein classification is a multi-class classification problem and usually reduced to a set of binary classification problems, where one classifier is designed for each class. The proteins in one class are seen as positive examples whereas those outside the class are seen as negative examples.[4,5] However, the imbalanced problem will arise in this case because the number of proteins in one class is usually much smaller than that of the proteins outside the class. As a result, the imbalanced data cause classifiers to tend to overfit and to perform poorly on the mi-

The Supplementary Material referred to in this article can be found online at http://www.interscience.wiley.com/jpages/0887-3585/suppmat/

Grant sponsor: National High Technology Research and Development Program of China; Grant number: 2006AA02Z309.

*Correspondence to: Luonan Chen, Department of Electronics, Information and Communication Engineering, Osaka Sangyo University, Nakagaito 3-1-1, Daito, Osaka 574-8530, Japan. E–mail: chen@eic.osaka-sandai.ac.jp or Kazuyuki Aihara, Room Ce601, Institute of Industrial Science, University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan. E–mail: aihara@sat.t.u-tokyo.ac.jp

Received 18 April 2007; Revised 30 August 2007; Accepted 1 October 2007

Published online 12 December 2007 in Wiley InterScience (www.interscience.wiley.com).

DOI: 10.1002/prot.21870

nority class. Despite the considerable success, existing methods tend to misclassify the examples in the minority class (positive examples in this case) especially when the data are highly imbalanced. Recently, the imbalanced problem has attracted much attention in machine learning community. A number of solutions have been proposed at both algorithmic and data levels.[9] At the algorithmic level, one-class learning algorithm[10,11] and feature selection[12,13] methods have been proposed for the imbalanced problem. In the one-class learning algorithm, a classifier is trained only on the examples of the target class. The one-class learning algorithm has been shown to be particularly useful when used on extremely imbalanced data in a high dimensional noisy feature space.[14] On the other hand, the feature selection methods select the features that lead to better separability between two classes and these features can capture the high skew in the class distribution.[9] At the data level, various variants of resampling techniques have been proposed, including oversampling,[15] undersampling,[16,17] and the combinations of these two methods.[18] Furthermore, boosting algorithms, which utilize the multiple classifier system and resampling, have also been applied to the imbalanced problem.[19] In this article, inspired by the methods presented in Ref. 18 and Ref. 20, we proposed a new hybrid sampling algorithm, which integrates both ensemble classifier and over-sampling techniques, to overcome the imbalanced problem in protein classification.

Moreover, in this article, classifiers trained in different feature spaces introduced by different feature extraction methods are combined together to further improve the accuracy of protein classification. Generally, an ensemble of classifiers can obtain better results compared with any component classifier in the ensemble.[21] Recently, the ensemble classifier has been applied to protein classification and shown promising results.[22–24] In these methods, different classifiers trained in the same feature space are combined together to improve the performance, which is nothing more than the comparison of classifiers. On the other hand, the success of protein classification depends largely on the construction of the feature space that a classifier will work in, and it has been found that the same classifier working in different feature spaces gives different results for the same classification problem.[5,4,25] Therefore, in addition to the new hybrid sampling algorithm, we further develop a new technique for protein classification by combining classifiers working in different feature spaces so that the classifiers from different feature spaces can complement each other. The numerical experiments on benchmark datasets show promising results and confirm the effectiveness of the proposed method. The Matlab code is available at http://eserver2.sat.iis.u-tokyo.ac.jp/~xmzhao/proteins.html.

METHODS

Feature extraction

To utilize machine learning techniques for protein classification, each protein sequence needs to be converted into a fixed length feature vector. In literature, many methods have been proposed for extracting features from protein sequences, such as hydrophobicity,[7] sequence similarity,[5] and mismatch alignment kernel.[4] In this work, the pairwise sequence similarity that has been successfully applied to protein homology detection[5] was employed here to convert protein sequences into feature vectors. Three methods, namely local alignment (LA) kernel,[25] Smith-Waterman (SW) algorithm,[26] and Needleman-Wunsch (NW) algorithm,[27] were adopted to measure the similarity between a pair of protein sequences. The LA kernel measures the similarity between a pair of protein sequences by constructing a kernel function, while SW and NW are two different sequence alignment algorithms. SW is a local alignment algorithm and NW is a global alignment algorithm, and both can be utilized to measure the similarity among protein sequences. The details of LA, NW, and SW can be found in,[25,27,26] respectively.

With the three methods described earlier, we can get three similarity matrices, where each element in the matrices represents the similarity between a pair of proteins. Therefore, each protein sequence S_i can be expressed as a vector:

$$f = [\mathrm{sim}(S_i, S_1), \ldots, \mathrm{sim}(S_i, S_n)] \quad (1)$$

where $\mathrm{sim}(S_i,S_j)$ denotes the pairwise similarity between protein S_i and protein S_j in the training set, n is the total number of proteins in the training set, and the pairwise similarity here means the similarity measured by any methods described above. Furthermore, all the vectors are standardized to have mean 0 and variance 1. Since different feature extraction methods introduce different feature spaces, we can obtain three kinds of classifiers working in three different feature spaces by LA, NW, and SW, respectively, in this article.

Ensemble classifier for protein classification

In this section, we present a new ensemble classifier for protein classification. In this work, protein classification is reduced to a set of binary classification problems, where proteins in one class are seen as positive examples while those outside the class are seen as negative examples, i.e., "one vs rest." Figure 1 shows the schematic flowchart of the proposed method. First, a new technique, namely EnClassifier, is utilized to handle the imbalanced problem, where we set a threshold of three because the imbalanced problem generally arises when the ratio of sizes between majority class and minority class is larger than three. Subsequently, an ensemble of n classifiers is trained on the processed data and applied to protein classification. Since there are three different ways for feature extraction in this article, the number of classi-

 DOI 10.1002/prot

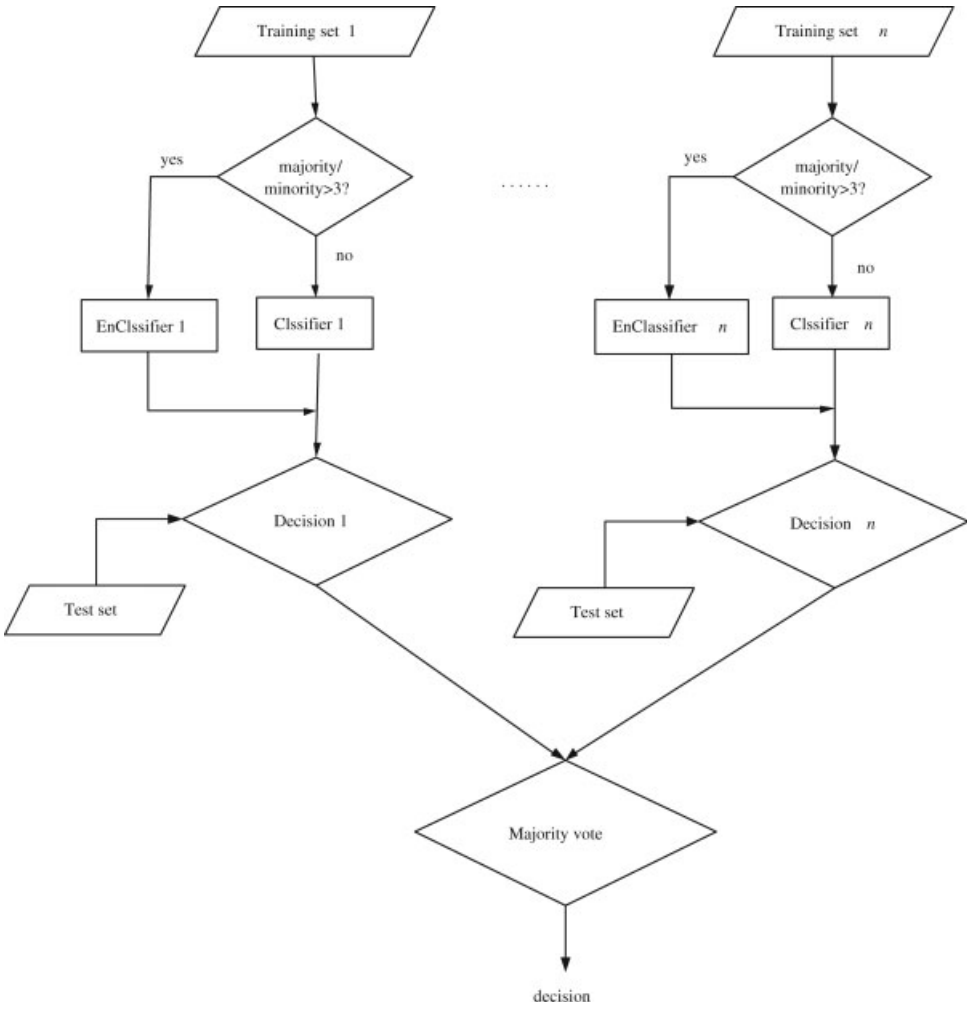

Figure 1

Schematic flowchart of the ensemble classifier, where "majority" denotes the majority class, and "minority" stands for the minority class.

fiers in Figure 1 is three, i.e., $n = 3$, where the EnClassifier is seen as a single classifier. Next, we explain the proposed method in details.

Rebalancing imbalanced dataset

After getting the feature vectors for the protein sequences, one classifier can be designed for each protein class, and the new protein sequence can be classified into the class with the biggest decision value. However, as described previously, the imbalanced problem will arise in this case. To overcome this problem, a new sampling method which integrates both under-sampling and over-sampling techniques, is proposed here. To further boost the performance, a committee of classifiers is employed in this article. That is, one classifier is trained on each dataset, and the final result is obtained by fusing the results from the base classifiers.

Figure 2 presents the schematic flowchart of the proposed method for rebalancing the imbalanced dataset. In this method, the majority class is first under-sampled and split into m groups, where each group has the same or similar size. On the other hand, the SMOTE algorithm[18] is used to over-sample the minority class so that the minority class has nearly the same size as each group generated from the majority class. After the sampling procedure, we get m new datasets from the majority class and one augmented minority class, where each of m datasets from the majority class has the similar size as the augmented minority class. The minority class and each group from the majority class will form a new training set, and consequently we get m new training sets, i.e., {majority class group 1, minority class}, . . . , {majority class group m, minority class}. With the newly generated datasets, we train m classifiers with one for each training set. Given a new test example, the predic-

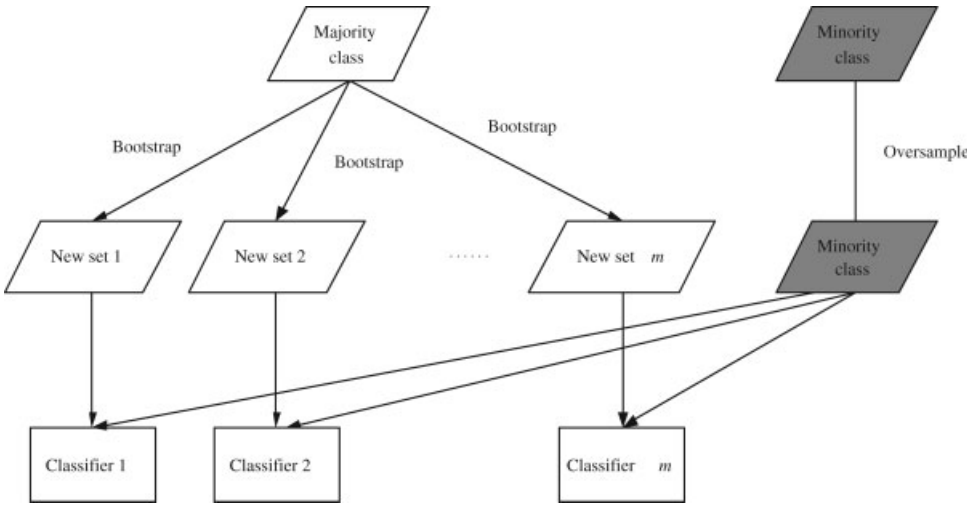

Figure 2
Schematic for EnClassifier.

tion result is obtained by fusing the outputs from the m classifiers. The technique of the ensemble classifier presented in this subsection is denoted as EnClassifier hereinafter.

Protein classification with ensemble classifier

With different feature extraction methods described earlier, each protein is described in a different way. It has been shown that different descriptions for proteins can lead to different results.[5,25] Generally, there is no guarantee that one single method can always outperform other methods in any cases. On the other hand, these methods may complement each other, and the combination of these methods may lead to better results.[22–24]

In this work, we combine classifiers trained in different feature spaces introduced by different feature extraction methods. As shown in Figure 1, each classifier has inputs with feature descriptions that are different from those to the other classifiers, where the ensemble classifier obtained in subsection 2.2.1 is seen as a single classifier. Consequently, n classifiers can be constructed if there are n different ways to describe the protein sequences. For a new test example, the combination of outputs from the n classifiers is the final decision. In this article, the simple majority voting method is adopted both here and in EnClassifier.

EXPERIMENTAL RESULTS

In this section, the proposed method was applied to classify proteins of the benchmark datasets downloaded from http://hydra.icgeb.trieste.it/benchmark, which is a repository of benchmark datasets for protein classification.[28] In this work, the proposed method was applied to four datasets collected from SCOP[29] and CATH databases[30] with different specificity. These four datasets include SCOP40_Minidatabase (Accession number: PCB00019), SCOP95_Superfamily_Family_Filtered (Accession number: PCB00020), SCOP95_Fold_Superfamily_Filtered (Accession number: PCB00022), and CATH95_Topology_Homology_Filtered (Accession number: PCB00028). Table I summarizes the data used in this study, where Min denotes the minimum number of samples and Max the maximum number of samples. Table I shows that the ratio between some positive samples and negative samples in the training set is about 1:60, which evidently demonstrates that the class imbalance exists in the data.

To evaluate the performance of the proposed method, the AUC (area under ROC curve) score was adopted in this study. Before applying the proposed method to protein classification, we need to determine the value of m in Figure 2, i.e., the split number of the majority class. In this article, the 10-fold cross-validation was utilized to

Table I
Summary of Data Descriptions

		Positive samples		Negative samples		
Data sets	# Samples	Min	Max	Min	Max	Classification tasks
PCB00019	1357	10	168	592	670	55
PCB00020	11,944	10	771	566	587	246
PCB00022	11,944	10	1013	555	587	191
PCB00028	11,373	10	1301	503	573	199

DOI 10.1002/prot

Table II
The Results by the Four Classifiers on Balanced and Imbalanced Data

Dataset	Classifier	LA_IB	LA_B	NW_IB	NW_B	SW_IB	SW_B
PCB00019	1NN	0.77	0.82	0.78	0.85	0.80	0.86
	SVM	0.75	0.78	0.79	0.82	0.79	0.82
	Logreg	0.76	0.83	0.68	0.88	0.63	0.83
	C4.5	0.69	0.76	0.67	0.73	0.68	0.75
PCB00020	1NN	0.75	0.78	0.75	0.78	0.61	0.65
	SVM	0.71	0.75	0.72	0.77	0.63	0.66
	Logreg	0.71	0.80	0.60	0.81	0.52	0.78
	C4.5	0.62	0.69	0.63	0.69	0.57	0.63
PCB00022	1NN	0.62	0.66	0.61	0.65	0.62	0.66
	SVM	0.64	0.67	0.66	0.70	0.64	0.69
	Logreg	0.60	0.70	0.52	0.67	0.50	0.64
	C4.5	0.58	0.60	0.57	0.61	0.58	0.61
PCB00028	1NN	0.65	0.68	0.63	0.66	0.63	0.67
	SVM	0.60	0.62	0.65	0.68	0.65	0.67
	Logreg	0.57	0.63	0.53	0.61	0.52	0.59
	C4.5	0.57	0.60	0.56	0.58	0.55	0.56

determine the value of m. Furthermore, a series of classifiers were employed to demonstrate the performance of the proposed rebalancing technique, and these classifiers include the nearest neighbor classifier (1NN), Support Vector Machines (SVMs),[31] C4.5 decision tree,[32] and logistic regression (logreg).[33] The details of implementation of the classifiers are described as follows.

For 1NN classifier, instead of using the "biological" 1NN as described in Ref. 28, the 1NN classifier was performed in the euclidean space with feature vectors generated by feature extraction. For the C4.5 decision tree, the fraction of incorrectly assigned samples at a node is set to 0.05. The 1NN classifier and C4.5 decision tree implemented in the MATLAB Classification Toolbox[34] were employed here. For SVMs, we used the LIBSVM library,[35] where the RBF kernel was adopted with the capacity parameter C set to 100 and the width parameter set to be the median euclidean distance from any positive training example to the nearest negative example.[28] The logistic regression model from the MATLAB statistical toolbox was employed in this study. The parameters for all the classifiers were fixed in the sequel.

Results on balanced versus imbalanced data

In this part, we investigated whether the proposed rebalancing technique improves the performance of the classifiers or not. First, the 1NN classifier was used as the baseline classifier to generate new samples for the minority class so as to make the data balanced. Sequentially, the imbalanced and balanced data were respectively used as the inputs to all the four classifiers (i.e., 1NN, SVMs, C4.5, and logreg) to compare the performance of the classifiers on imbalanced versus balanced data. Note that the parameters used for all the classifiers were the same ones with or without rebalancing. Therefore, the comparison of performance of classifiers before and after rebalancing can demonstrate the effectiveness of the proposed rebalancing techniques. Table II shows the results obtained via the four classifiers before and after rebalancing, by averaging over all the classification tasks for each dataset, where LA_IB denotes the imbalanced dataset by LA, LA_B denotes the balanced dataset by LA, and so on. The detailed results can be found in the Supplementary materials (S19.txt for PCB00019 dataset, and the same for the other three datasets). It can be seen from Table II that the results on the balanced data are better than those on the imbalanced data for any classifier tested here. The results in Table II confirm that the proposed rebalancing technique can indeed improve the accuracy of protein classification. On the other hand, the results also imply that the imbalanced dataset can really degrade the performance of classifiers.

Results by ensemble classifier versus single classifier

Furthermore, we investigated whether or not the ensemble of classifiers can improve the accuracy of protein classification compared against the component classifiers in the ensemble. Table III shows the results by the ensemble classifier and the component classifiers, where the results were averaged over all the classification tasks. The detailed results by the ensemble classifier can be found in the Supplementary materials (S_ensemble.txt). The results in Table III clearly show that the ensemble classifier outperforms any component classifiers, and also prove that the information obtained in different ways can complement each other.

In addition, to demonstrate the performance of the proposed method, we countered the number of classes versus a AUC score threshold for all the methods employed in this experiment, which has been widely used to compare

Table III
The Performance of the Ensemble Classifier Versus Single Classifiers on the Balanced Data

Dataset	Classifier	LA_B	NW_B	SW_B	Ensemble classifier
PCB00019	1NN	0.82	0.85	0.86	0.86
	SVM	0.78	0.82	0.82	0.82
	Logreg	0.83	0.88	0.83	0.91
	C4.5	0.76	0.73	0.75	0.80
PCB00020	1NN	0.78	0.78	0.65	0.78
	SVM	0.75	0.77	0.66	0.75
	Logreg	0.80	0.81	0.78	0.86
	C4.5	0.69	0.69	0.63	0.70
PCB00022	1NN	0.66	0.65	0.66	0.68
	SVM	0.67	0.70	0.69	0.71
	Logreg	0.70	0.67	0.64	0.73
	C4.5	0.60	0.61	0.61	0.65
PCB00028	1NN	0.68	0.66	0.67	0.70
	SVM	0.62	0.68	0.67	0.69
	Logreg	0.63	0.61	0.59	0.67
	C4.5	0.60	0.58	0.56	0.63

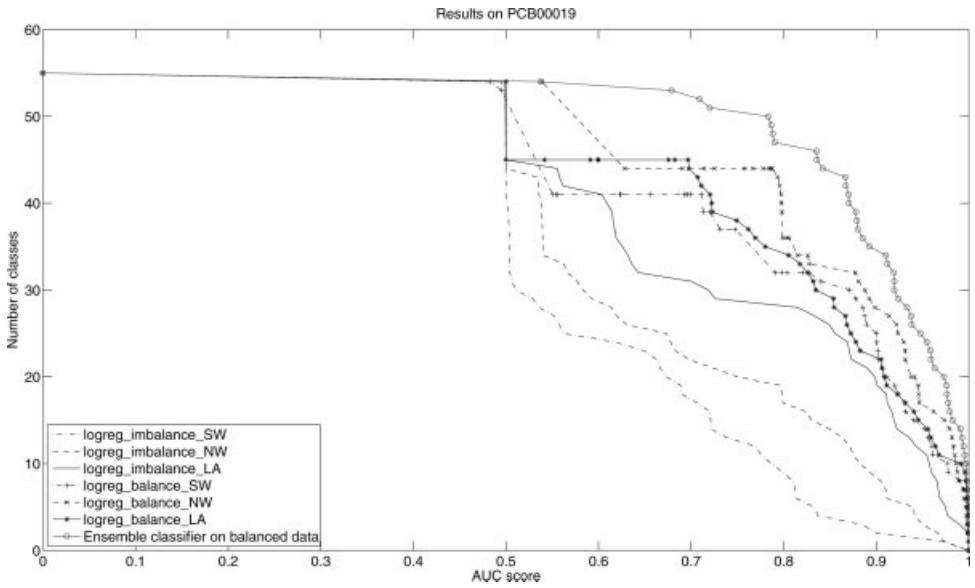

Figure 3

The comparison of the performance of the logistic regression and the ensemble classifiers on balanced or imbalanced data for PCB00019. The curves in the figure show the number of classes versus a AUC score threshold.

the performance of different protein classification methods.[4,5] For clarity, Figure 3 only shows the results by logistic regression on PCB00019, where a higher curve means a more accurate result. The results by other classifiers on other datasets can be found in the Supplementary materials (S_fig1 for logistic regression, S_fig2 for SVMs, S_fig3 for 1NN, and S_fig4 for C4.5). From Figure 3, we can readily see that the proposed ensemble classifier outperforms all the other methods and the rebalancing technique can indeed improve the performance of the classi-

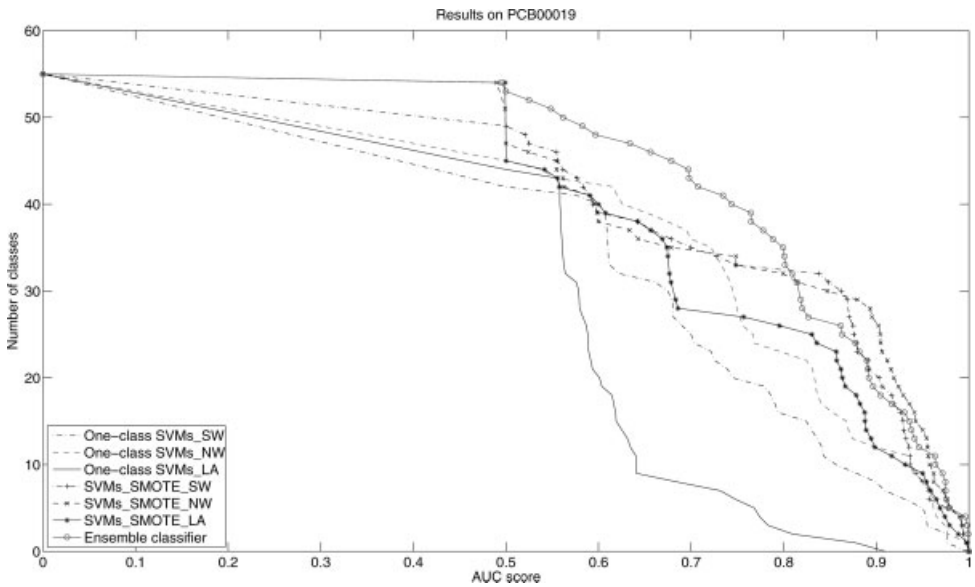

Figure 4

The comparison of the ensemble classifier against one-class SVMs and SMOTE on PCB00019. The curves in the figure show the number of classes versus a AUC score threshold.

DOI 10.1002/prot

fiers. The results shown in Figure 3 also verify the efficiency and effectiveness of the proposed method.

Comparison

To demonstrate the performance, we compared the proposed method with the existing methods, namely one-class SVMs and SMOTE, that have been proposed for imbalanced data. For the one-class SVMs, the RBF kernel was adopted and 10-fold cross-validation was employed for searching the optimal parameters for the classifier, which was trained only on the positive training set. For the SMOTE algorithm, the same protocol used in our system was adopted. In the SMOTE algorithm, the minority class was augmented to have the similar size as that of the majority class. For comparison, the SVMs were used in both the proposed method and the SMOTE algorithm. The number of classes versus an AUC score threshold was countered for all the methods. Figure 4 shows the results by the proposed method versus the existing methods on PCB0019, where the one-class SVMs perform worst because of the few number of positive training samples. The results on other datasets can be found in Supplementary materials. From Figure 4, clearly the proposed method outperforms all the existing methods, thereby proving the effectiveness of the proposed method for protein classification. On the other hand, the better performance of the proposed method than the SMOTE and one-class methods also demonstrates the usefulness of the proposed method for imbalanced data in protein classification.

DISCUSSIONS AND CONCLUSIONS

In this work, a new technique for protein classification with the imbalanced data was presented. Imbalanced problem exists in most of the protein classification tasks, where the number of proteins in one class is usually much smaller than that of those outside the class. Generally, the imbalanced data make the classifier tend to overfit, thereby degrading the performance of the classifiers on the minority class. To overcome this problem, we proposed a new algorithm in this article with a new hybrid sampling technique and a committee of classifiers. Experimental results show that the imbalanced data can really degrade the performance of the classifier, and the proposed technique can alleviate that problem by rebalancing the imbalanced data. Furthermore, classifiers trained with different features are combined together to improve the accuracy of protein classification in this article. The numerical experiments on benchmark datasets show the effectiveness of the proposed method. On the other hand, the results indicate that it is more effective to improve the accuracy by the rebalancing technique rather than the ensemble classifier. The method proposed in this work can also be applied to other fields in bioinformatics because the imbalanced problem exists in many of the biological datasets. Note that we only utilized four kinds of classifiers in this work. In fact, other classifiers can be used instead, by combining with the proposed method. Furthermore, the proposed method was only applied to proteins described by the pairwise similarity in this work. It can be applied to any other forms of descriptions for proteins, such as the predicted secondary structure, hydrophobicity, van der Waals volume, and polarity.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive suggestions to improve the work.

REFERENCES

1. Altschul S, Madden T, Schafer A, Zhang J, Zhang Z, Miller W, Lipman D. Gapped blast and psi-blast: a new generation of protein data. Nucleic Acids Res 1997;25:3389–3402.
2. Henikoff S, Henikoff J. Position-based sequence weights. J Mol Biol 1994;243:574–578.
3. Krogh A, Brown M, Mian I, Sjolander K, Haussler D. Hidden markov models in computational biology: applications to protein modeling. J Mol Biol 1994;235:1501–1531.
4. Leslie C, Eskin E, Cohen A, Weston J, Noble W. Mismatch string kernels for discriminative protein classification. Bioinformatics 2004; 20:467–476.
5. Liao L, Noble W. Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships. J Comput Biol 2003;10:857–868.
6. Zhao X, Cheung Y, Huang D. A novel approach to extracting features from motif content and protein composition for protein sequence classification. Neural Networks 2005;18:1019–1028.
7. Huang D, Zhao X, Huang G, Cheung Y. Classifying protein sequences using hydropathy blocks. Pattern Recog 2006;39:2293–2300.
8. Ding CH, Dubchak I. Multi-class protein fold recognition using support vector machines and neural networks. Bioinformatics 2001;17:349–358.
9. Chawla NV, Japkowicz N, Kotcz A. Editorial: special issue on learning from imbalanced data sets. SIGKDD Explor Newslett 2004;6:1–6.
10. Scholkopf B, Platt JC, Shawe-Taylor JC, Smola AJ, Williamson RC. Estimating the support of a high-dimensional distribution. Neural Comput 2001;13:1443–1471.
11. Tax D. One-class classification. PhD thesis. Delft University of Technology, Delft, The Netherlands; 2001.
12. Zheng Z, Wu X, Srihari R. Feature selection for text categorization on imbalanced data. ACM SIGKDD Explor Newslett 2004;6:80–89.
13. Mladenic D, Grobelnik M. Feature selection for unbalanced class distribution and naive bayes. In the Proceedings of the Sixteenth International Conference on machine learning, Bled, Slovenia; June 27–30, 1999. pp 258–267.
14. Raskutti B, Kowalczyk A. Extreme re-balancing for SVMs: a case study. ACM SIGKDD Explor Newslett 2004;6:60–69.
15. Guo H, Viktor HL. Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. SIGKDD Explor 2004;6:30–39.
16. Kubat M, Matwin S.Addressing the curse of imbalanced training sets: one-sided selection. In the Proceedings of the 14th International Conference on machine learning, Nashville, TN; 1997. pp 179–186.
17. Wilson DR, Martinez TR. Reduction techniques for instance-based learning algorithms. Mach Learn 2000;38:257–286.
18. Chawla N, Bowyer K, Hall L, Kegelmeyer W. Smote: synthetic minority over-sampling technique. J Artif Intell Res 2002;16:321–357.

19. Chawla NV, Lazarevic A, Hall L, Bowyer KW. SMOTEBoost: improving prediction of the minority class in boosting. In the Proceedings of the 7th European Conf on principles and practice of knowledge discovery in databases (PKDD), Cavtat, Croatia; 2003. pp 107–119.
20. Liu Y, An A, Huang X.Boosting prediction accuracy on imbalanced datasets with SVM ensembles. In the Proceedings of the 10th Pacific-Asia Conference on knowledge discovery and data mining (PAKDD'06), Singapore, 2006. p 107–118.
21. Dietterich TG. Machine learning research: four current directions. AI Magazine 1998;18:97–136.
22. Shen H, Chou K. Ensemble classifier for protein fold pattern recognition. Bioinformatics 2006;22:1717–1722.
23. Tan A, Gilbert D, Deville Y. Multi-class protein fold classification using a new ensemble machine learning approach. Genome Informatics 2003;14:206–217.
24. Can T, Camoglu O, Singh A, Wang Y. Automated protein classification using consensus decision. Computational Systems Bioinformatics Conference, Stanford, CA; 2004. pp 224–235.
25. Saigo H, Vert JP, Ueda N, Akutsu T. Protein homology detection using string alignment kernels. Bioinformatics 2004;20:1682–1689.
26. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–197.
27. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 1970;48:443–453.
28. Sonego P, Pacurar M, Dhir S, Kertsz-Farkas A, Kocsor A, Gspri Z, Leunissen JA, Pongor S. A Protein Classification Benchmark collection for machine learning. Nucleic Acids Res 2007;35 (Database issue):D232–D236.
29. Andreeva A, Howorth D, Brenner SE, Hubbard TJ, Chothia C, Murzin AG. SCOP database in 2004: refinements integrate structure and sequence family data. Nucleic Acids Res 2004;32:D226–D229.
30. Pearl F, Todd A, Sillitoe I, Dibley M, Redfern O, Lewis T, Bennett C, Marsden R, Grant A, Lee D, Akpor A, Maibaum M, Harrison A, Dallman T, Reeves G, Diboun I, Addou S, Lise S, Johnston C, Sillero A, Thornton J, Orengo C. The CATH Domain Structure Database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis. Nucleic Acids Res 1005;33:D247–D251.
31. Vapnik VN. Statistical learning theory. New York: Wiley; 1998.
32. Quinlan JR. C4.5: Programs for machine learning. San Francisco, CA: Morgan Kaufmann Publishers; 1993.
33. Hosmer DW, Stanley L. Applied logistic regression, 2nd ed. New York: Wiley; 2000.
34. Stork DG, Yom-Tov E. Computer manual in MATLAB to accompany pattern classification, 2nd edition. Hoboken, NJ: Wiley; 2004.
35. Chang CC, Lin CJ. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

 DOI 10.1002/prot