Research Journal of Applied Sciences, Engineering and Technology 5(3): 944-949, 2013
ISSN: 2040-7459; E-ISSN: 2040-7467

Submitted: June 20, 2012 Accepted: July 23, 2012 Published: January 21, 2013

Imbalanced Classification Based on Active Learning SMOTE

Ying Mi
Foundation Department, Dalian Vocational and Technical College, Dalian 116035, China

Abstract: In real-world problems, the data sets are typically imbalanced. Imbalance has a serious impact on the performance of classifiers. SMOTE is a typical over-sampling technique which can effectively balance the imbalanced data. However, it brings noise and other problems affecting the classification accuracy. To solve this problem, this study introduces the classification performance of support vector machine and presents an approach based on active learning SMOTE to classify the imbalanced data. Experimental results show that the proposed method has higher Area under the ROC Curve, F-measure and G-mean values than many existing class imbalance learning methods.

Keywords: Active learning, imbalanced data set, SMOTE, support vector machine

INTRODUCTION

For classification problem, the training data will significantly influence the classification accuracy (Yen and Lee, 2009). However, the data in real-world applications often are imbalanced class distribution, that is, most of the data are in majority class and little data are in minority class. The level of imbalance, namely the ratio of size of the majority class to minority class, can be as huge as 10^6 (Wu *et al.*, 2008). In this case, if all the data are used to be the training data, the classifier tends to predict that most of the incoming data belongs to the majority class. Hence, it is important to select the suitable training data for classification in the imbalanced class distribution problem.

Imbalanced data sets are very common in real-world, such as medical diagnosis, oil blowout detection, financial fraud detection, network intrusion detection, spam detection, text classification, etc. They have a common characteristic that the minority class information is a focus. Traditional machine learning methods are mostly based on balanced data sets and lead to a high overall accuracy. Unfortunately, such a strategy is not useful for identifying the class of interest. These classifiers generally perform poorly for class imbalance problem and often achieve low accuracy on the minority class. Further, the cost of misclassifying the minority class is usually much higher than the cost of other misclassifications. It may be much more costly, for example, to fail to identify a case of financial fraud which leads to lose a significant amount of money than to misclassify an innocent case as fraud.

Currently, solutions to the imbalanced data classification problem generally fall into 1 of 2 categories (He and Garcia, 2009). First, re-sampling is a common method, including random re-sample, over-sampling and under-sampling. Second, algorithmic solutions have been proposed, including integrated approach, cost-sensitive learning, feature selection and single-class learning and so on. Each of these categories of solutions has advantages and disadvantages. For example, re-sampling techniques have the advantage that they can be used with any base learner, such as support vector machine, C4.5, Naïve bayes classifier etc., to address the class imbalance problem. Under-sampling techniques result in a smaller training dataset and allow classifier construction to proceed more rapidly. SMOTE (Chawla *et al.*, 2002) is an intelligent over-sampling method. Due to synthetic samples, SMOTE method avoids over-fitting largely and achieves a good performance in the imbalanced data classification problem. However, SMOTE brings new noise and other problems. Cost-sensitive learning deals with class imbalance by incurring different costs for the 2 classes and is considered as an important class of methods to handle class imbalance. The difficulty with cost-sensitive classification is that costs of misclassification are often unknown.

Active learning is a kind of learning strategies which actively selects the best samples to learn. It can select more valuable samples and abandon the samples which have less information, so as to improve the classification performance. However, it causes classifier skewing for class imbalance distribution when only used active learning.

Although the existing imbalance-learning methods applied for normal SVMs can solve the problem of class imbalance, they can still suffer from the problem of outliers and noise. This study introduces the

classification performance of support vector machine and presents an approach based on active learning SMOTE to classify the imbalanced data.

LITERATURE REVIEW

Since many real applications have the imbalanced class distribution problem, class imbalance learning has recently received considerable attention in machine learning as current algorithms do not provide satisfactory classification performance. Standard algorithms are overwhelmed by majority examples while minority examples contribute very little. A number of improved algorithms have been proposed in the literature, where considerations have been made at the data level and algorithm level.

Sampling technology: At the data level, proposed methods mainly include re-sampling. Sampling is 1 of techniques for adjusting the size of a training dataset (Diamantini and Potena, 2009). In general, it can be distinguished into over-sampling approach (Chawla *et al.*, 2002; Japkowicz, 2001) and under-sampling approach (Chyi, 2003; Zhang and Mani, 2003). The over-sampling approach increases the number of minority class samples to reduce the degree of imbalanced distribution. In addition, it is efficient in term of time complexity when handling a large volume of data. Under-sampling uses only a part of major category data, so the sample may not represent the characteristics of the whole major category.

SMOTE (Chawla *et al.*, 2002) added new synthetic minority class examples by randomly interpolating pairs of closest neighbors in the minority class. SMOTE boost algorithm (Chawla *et al.*, 2003) combines SMOTE technique and the standard boosting procedure. It utilizes SMOTE for improving the accuracy over the minority classes and utilizes boosting to not sacrifice accuracy over the entire data set. Instead of changing the distribution of training data by updating the weights associated with each example, SMOTE boost alters the distribution by adding new minority-class examples using the SMOTE algorithm.

Jo and Japkowicz (2004) present a cluster-based over-sampling technique. The technique first clusters the minority samples and the majority samples independently and performs random over-sampling with replacement separately for each cluster. After clustering, each of the clusters of majority class samples, except for the largest 1, are randomly over-sampled until they have the same number of samples as the largest majority class cluster. The samples in each minority class cluster are then over-sampled with replacement until each minority class cluster has corresponding samples.

Stefanowski and Wilk (2007) propose an effective approach for selectively filtering the majority class while strengthening relevant minority class examples.

Yen and Lee (2009) propose a cluster-based under-sampling approach for selecting the representative data as training data to improve the classification accuracy for minority class and investigate the effect of under-sampling methods in the imbalanced class distribution environment. The experimental results show that our cluster-based under-sampling approaches outperform the other under-sampling techniques in the previous studies.

Modified algorithmic solutions: At the algorithmic level, developed methods mainly include cost-sensitive learning (Drummond and Holte, 2003; Elkan, 2001) and modified algorithms. Cost-sensitive learning approach assumes the misclassification costs are known in a classification problem. A cost-sensitive classifier tries to learn more characteristics of samples with the minority class by setting a high cost to the misclassification of a minority class sample. However, misclassification costs are often unknown and a cost-sensitive classifier may result in over fitting training. Reported works in cost-sensitive learning fall into 3 main categories: weighting the data space, making a specific classifier learning algorithm cost sensitive and using Bayes risk theory to assign each sample to its lowest risk class.

Sun *et al*. (2007) investigate cost-sensitive boosting algorithms for advancing the classification of imbalanced data and propose 3 cost-sensitive boosting algorithms by introducing cost items into the learning framework of Ada boost.

In term of modified algorithms, several specific attempts using SVMs have been made at improving their class prediction accuracy in the case of class imbalances (Wang and Japkowicz, 2010; Akbani *et al*., 2004). The results obtained with such methods show that SVMs have the particular advantage of being able to solve the problem of skewed vector spaces, without introducing noise.

In addition, Wang and Japkowicz (2010) combine modifying the data distribution approach and modifying the classifier approach in class imbalance problem and use support vector machines with soft margins as the base classifier to solve the skewed vector spaces problem.

THE PROPOSED ALGORITHM BASED ON ACTIVE LEARNING AND SMOTE

SMOTE (Synthetic Minority Oversampling Technique, SMOTE) algorithm is a kind of typical

over-sampling method proposed by Chawla *et al.* (2002). SMOTE add some new and artificial minority samples by extrapolating between pre-existing minority instances rather than simply sampling with replacement. The newly created samples cause the minority regions of the feature-space to be more substantial and more general. SMOTE first finds the *k* nearest neighbors of the minority class for each minority sample. The artificial samples are then generated in the direction of some or all of the nearest neighbors, depending on the amount of over-sampling desired. SMOTE technique causes a classifier to learn a larger and more general decision region in the feature-space.

In this study, the SMOTE method is adapted for advancing the classification of imbalanced data. The proposed method is developed by introducing Support Vector Machine (SVM) into the learning framework of SMOTE for class-imbalance learning. The proposed algorithm first uses the most efficient equalization to the imbalanced data sets and then uses the SVM algorithm to process the imbalanced data classification. Based on description above, the proposed algorithm is described as follows:

Step 1: Suppose the training set is *A*, the total of samples is *n*. Divide *A* into *e* portions randomly, labeled as B_i (i = 1, 2,…, e)

Step 2: Extract the minority class sample set *C* from the first training set B_1 and the minority class number is m_1

Step 3: Use SMOTE method to oversample according to the proportion of majority class samples to minority class samples $(n/e - m)/m$ and obtain synthetic sample set *D*

Step 4: Merge the synthetic samples *D* to B_1, get a new training set *F*

Step 5: Take SVM classification on *F* and get the first separate hyperplane l_1

Step 6: Repeat for the rest *e*-1 datasets

Step 6.1: According to the distance formula $d=|w*x+b|$, find the nearest sample set *E* to hyperplane in training set B_i ($i = 1, 2,…, e$)

Step 6.2: Extract the number of minority class samples m_i and minority class sample set *P* from the training set B_i

Step 6.3: Extract m_i samples from the former majority class samples in *E* and get majority class sample set *G*

Step 6.4: Merge *P* and *G* to the training set *F* of step 5

Step 6.5: Classify data set *F* using SVM and get the i^{th} separate hyperplane

Until $i = e$

Table 1: Data sets

Data set	Total samples	N_{ma}	N_{mi}	Target	Ratio
Cmc	1473	1140	333	class 2	3.4
Haberman	306	225	81	class 2	2.8
Abalone	4177	3786	391	class 7	9.7
Housing	506	400	106	[20,23]	3.8
Pima	768	500	268	class 1	1.9
Satimage	6435	5809	626	class 4	9.3

Table 2: Confusion matrix

	Predicted positive class	Predict negative class
Actual positive class	TP (True Positives)	FN (False Negatives)
Actual negative class	FP (False Positives)	TN (True Negatives)

EXPERIMENTAL EVALUATION METRICS

In this section, we list data sets in our experiments, and present some performance evaluation metrics:

Data sets: We experimented on six different datasets from UCI machine learning repository (http://archive.ics.uci.edu/ml/). These datasets are summarized in Table 1. These datasets vary extensively in their sizes and class proportions. N_{ma} and N_{mi} represent the sample number of majority class and the sample number of minority class, respectively. The class proportion is the ratios of N_{ma} to N_{mi} as shown in Table 1. In this experiment, we take the minority class as the target class, all the other categories as majority class.

Performance evaluation metrics: Evaluation metrics play a crucial role in both assessing the classification performance and guiding the classifier modeling. The traditional evaluation standard uses accuracy for these purposes. However, for classification with the class imbalance problem, accuracy is no longer a proper measure since the minority class has very little impact on accuracy as compared to the majority class (Joshi *et al.*, 2001). For example, in a problem where a minority class is represented by only 1% of the training data, a simple strategy can be to predict the majority class label for every example. It can achieve a high accuracy of 99%. However, this measurement is meaningless to some applications where the learning concern is the identification of the rare cases.

Therefore, some evaluation standards are put forward to the classification with the class imbalance problem, including ROC (Receiver Operating Characteristic Curve), *F-measure* and *G-mean* (Fawcett, 2003). Here for a binary classification problem, we usually take minority class as positive class for high identification importance and take the

other as the negative class. Samples can be categorized into 4 groups after a classification process as denoted in the confusion matrix presented in Table 2.

Several measures can be derived using the confusion matrix:

$$\text{True Positive Rate: TPR} = \frac{TP}{TP+FN}$$

$$\text{False Positive Rate: FPR} = \frac{FP}{TN+FP}$$

$$\text{True Negative Rate: TNR} = \frac{TN}{TN+FP}$$

$$\text{False Negative Rate: FNR} = \frac{FN}{TP+FN}$$

for different evaluation criteria, several measures are devised, including *recall*, *precision*, *F-meansure*, *G-mean* and *ROC*.

Recall: In information retrieval, True Positive Rate is defined as *recall* denoting the percentage of retrieved objects that are relevant:

$$Recall = \frac{TP}{TP+FN} = TPR \quad (1)$$

Precision: *Precision* is defined as the percentage of relevant objects that are identified for retrieval:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

F-meansure: *F-measure* is suggested in Lewis and Gale (1998) to integrate these 2 measures as an average:

$$F-measure = \frac{2\times Precision\times Recall}{Precision+Recall} = \frac{2}{1/Precision+1/Recall} \quad (3)$$

A high *F-measure* value ensures that both recall and precision are reasonably high from Eq. (3).

G-mean: When the performance of both classes is concerned, both True Positive Rate (TPR) and True Negative Rate (TNR) are expected to be high simultaneously. *G-mean* is defined as:

$$G-mean = \sqrt{TPR\times TNR} \quad (4)$$

G-mean measures the balanced performance of a learning algorithm between these two classes.

ROC curve and AUC: (Fawcett, 2003)**.** Some classifiers, such as Bayesian network inference or some neural networks, assign a probabilistic score to its prediction. Class prediction can be changed by varying the score threshold. Each threshold value generates a pair of measurements of (FPR, TPR). By linking these measurements with the False Positive Rate (FPR) on the *X*-axis and the True Positive Rate (TPR) on the *Y*-axis, a ROC graph is plotted. ROC curves can be thought of as representing the family of best decision boundaries for relative costs of TP and FP. The ideal point on the ROC curve would be (0, 1), that is all positive examples are classified correctly and no negative examples are misclassified as positive.

A ROC graph depicts relative trade-offs between benefits (true positives) and costs (false positives) across a range of thresholds of a classification model. A ROC curve gives a good summary of the performance of a classification model. To compare several classification models by comparing ROC curves, it is hard to claim a winner unless 1 curve clearly dominates the others over the entire space. The Area Under a ROC Curve (AUC) provides a single measure of a classifier's performance for evaluating which model is better on average. It integrates performance of the classification method over all possible values of FPR and is proved to be a reliable performance measure for imbalanced and cost-sensitive problems (Lewis and Gale, 1998).

EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify the effectiveness of our proposed method, we compared it to random under-sampling, Adaboost algorithm, SMOTE algorithm and Active Learning SVM algorithm (ALSVM). Based on MATLAB 7.0, we tested these methods on 6 UCI datasets. Information about these datasets is summarized in Table 1.

For every dataset, we perform a tenfold stratified cross validation. Within each fold, the classification method is repeated 10 times considering that the sampling of subsets introduces randomness. The AUC, *F-measure* and *G-mean* of this cross-validation process are averaged from these 10 runs. The whole cross-validation process is repeated for 5 times and the final values from this method are the averages of these five cross-validation runs.

First, we developed these classifiers on the imbalanced datasets and evaluated their performance by using *F-measure* values. The average *F-measure* values of the compared methods are summarized in Table 3. From these results, we can clearly observe that for all the datasets, the proposed algorithm yielded the highest results on *F-measure* values.

Generally, under and Adaboost methods are not performing well with *F-measure*. Their corresponding

Table 3: F-measure values of the compared methods

F-measure	Cmc	Haberman	Abalone	Housing	Pima	Satimage	Avg.
Under	0.4307	0.4397	0.3473	0.5240	0.5902	0.5477	0.480
Adaboost	0.3808	0.3806	0.2363	0.4695	0.5873	0.6276	0.447
SMOTE	0.4568	0.4509	0.3780	0.5306	0.6032	0.6159	0.506
ALSVM	0.5190	0.5697	0.5847	0.5455	0.5971	0.5468	0.560
Proposed algorithm	0.5490	0.6229	0.5946	0.5500	0.6173	0.6415	0.596

Table 4: G-mean values of the compared methods

G-mean	Cmc	Haberman	Abalone	Housing	Pima	Satimage	Avg.
Under	0.6240	0.5337	0.7689	0.7024	0.6534	0.8307	0.686
Adaboost	0.5469	0.5184	0.3952	0.6156	0.6458	0.7524	0.579
SMOTE	0.6540	0.5687	0.7493	0.7104	0.6645	0.8475	0.699
ALSVM	0.6991	0.6897	0.8160	0.6908	0.6042	0.5531	0.675
Proposed algorithm	0.7295	0.7566	0.8854	0.7416	0.6353	0.7741	0.754

Table 5: AUC values of the compared methods

AUC	Cmc	Haberman	Abalone	Housing	Pima	Satimage	Avg.
Under	0.6770	0.6312	0.8330	0.8057	0.7299	0.9322	0.768
Adaboost	0.4961	0.5952	0.8187	0.8046	0.7450	0.9537	0.736
SMOTE	0.6807	0.6352	0.8371	0.8154	0.7554	0.9436	0.778
ALSVM	0.7546	0.6486	0.9642	0.8341	0.5574	0.9177	0.780
Proposed algorithm	0.9016	0.7371	0.9785	0.8682	0.7222	0.9386	0.858

average *F-measures* on six datasets are 0.480 and 0.447, respectively. They are lower than those of SMOTE, ALSVM and our proposed algorithm.

Next, we evaluated their performance by using *G-mean* values. The average *G-mean* values of the compared methods are summarized in Table 4. From these results, SMOTE and our proposed method obtained better performance. For *Pima and sati mage* datasets, SMOTE method yielded the highest results, while for other datasets, *cmc*, *haberman*, *abalone* and *housing*, our proposed algorithm yielded the highest results.

Shown as Table 4, Adaboost method is not performing well with *G-mean*. Its average *G-mean* is lower than the other compared methods. ALSVM is comparable to or slightly lower than those of under and SMOTE and they are lower than that of our proposed method.

Then, we evaluated their performance by using AUC values. The average AUC values of the compared methods are summarized in Table 5. The results show Adaboost method has the highest AUC on *sati mage* among these compared methods, while SMOTE method has the highest AUC on *Pima*. Except 2 datasets above, our proposed method has higher AUC on the other datasets, including *cmc*, *haberman*, *abalone* and *housing*.

Similarly, Adaboost method is not performing well with AUC from the results shown as Table 5. Its average AUC is only 0.736 and is slightly lower than those of Under, SMOTE and ALSVM. Our proposed method attains the highest average AUC among these compared methods.

CONCLUSION

Classification is an important task of Knowledge Discovery in Databases (KDD) and data mining. However, reports from both academia and industry indicate that imbalanced class distribution of a data set has posed a serious difficulty to most classifier learning algorithms, which assume a relatively balanced distribution. In this study, the SMOTE method is adapted for advancing the classification of imbalanced data. Our proposed method is developed by introducing SVM into the learning framework of SMOTE for class-imbalance learning. The proposed method uses active learning SMOTE to classify the imbalanced data. Experiment results show that the proposed method has higher *F-measure*, *G-mean* and AUC than almost all other compared methods, including Under, Adaboost, SMOTE and ALSVM.

ACKNOWLEDGMENT

This study is supported by China Postdoctoral Science Foundation (No. 20110491530), Science Research Plan of Liaoning Education Bureau (No. L2011186), and Dalian Science and Technology Planning Project of China (No. 2010J21DW019).

REFERENCES

Akbani, R., S. Kwek and N. Japkowicz, 2004. Applying support vector machines to imbalanced datasets. Proceedings of the 2004 European Conference on Machine Learning (ECML'2004).

Chawla, N.V., K. Bowyer, L. Hall and W. Kegelmeyer, 2002. SMOTE: Synthetic minority over-sampling technique. J. Artificial Intell. Res., 16: 231-357.

Chawla, N.V., A. Lazarevic, L.O. Hall and K.W. Bowyer, 2003. Smoteboost: Improving prediction of the minority class in boosting. Lect. Notes Artif. Int., 2838-2003: 107-119.

Chyi, Y.M., 2003. Classification analysis techniques for skewed class distribution problems. MA Thesis, Department of Information Management, National Sun Yat-Sen University, China-Taiwan.

Diamantini, C. and D. Potena, 2009. Bayes vector quantizer for class-imbalance problem. IEEE T. Knowl. Data En., 21(5): 638-651.

Drummond, C. and R.C. Holte, 2003. C4.5, class imbalance and cost sensitivity: Why under-sampling beats over-sampling. Proceedings of the ICML'03 Workshop on Learning from Imbalanced Datasets, pp: 1-8.

Elkan, C., 2001. The foundations of cost-sensitive learning. Proceedings of the 17th International Joint Conference on Artificial Intelligence, pp: 973-978.

Fawcett, T., 2003. ROC graphs: Notes and practical considerations for researchers. Technology Report, HP Labs.

He, H.B. and E.A. Garcia, 2009. Learning from imbalanced data. IEEE T. Knowl. Data En., 21(9): 1263-1284.

Japkowicz, N., 2001. Concept-learning in the presence of between-class and withinclass imbalances. Proceedings of the 14th Conference of the Canadian Society for Computational Studies of Intelligence, pp: 67-77.

Jo, T. and N. Japkowicz, 2004. Class imbalances versus small disjuncts. SIGKDD Explorations, 6(1): 40-49.

Joshi, M.V., V. Kumar and R.C. Agarwal, 2001. Evaluating boosting algorithms to classify rare classes: Comparison and improvements. ICDM'01, Proceedings of the 1st IEEE International Conference on Data Mining, pp: 257-264.

Lewis, D. and W. Gale, 1998. Training text classifiers by uncertainty sampling. Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information, pp: 73-79.

Stefanowski, J. and S. Wilk, 2007. Improving rule-based classifiers induced by MODLEM by selective preprocessing of imbalanced data. Proceedings of the RSKD at ECML/PKDD, Warsaw, pp: 54-65.

Sun, Y., M.S. Kamela, A.K.C. Wong and Y. Wang, 2007. Cost-sensitive boosting for classification of imbalanced data. Pattern Recogn., 40: 3358-378.

Wang, B.X. and N. Japkowicz, 2010. Boosting support vector machines for imbalanced data sets. Knowl. Inf. Syst., 25(1): 1-20.

Wu, J., S.C. Brubaker, M.D. Mullin and J.M. Rehg, 2008. Fast asymmetric learning for cascade face detection. IEEE Trans. Pattern Anal. Mach. Intell., 30(3): 369-382.

Yen, S.J. and Y.S. Lee, 2009. Cluster-based under-sampling approaches for imbalanced data distributions. Expert Syst. Appl., 36: 5718-5727.

Zhang, J. and I. Mani, 2003. KNN approach to unbalanced data distributions: A case study involving information extraction. Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Datasets, II, pp: 42-48.