A Novel Nonparallel Plane Proximal SVM for Imbalance Data Classification

Bing Yang and Ling Jing*
Department of Applied Mathematics, College of Science, China Agricultural University, Beijing, P.R. China
E-mail address: 434915607@qq.com, jingling@cau.edu.cn

***Abstract*—The research of imbalance data classification is the hot point in the field of data mining. Conventional classifiers are not suitable to the imbalanced learning tasks since they tend to classify the instances to the majority class which is the less important class. This paper pays close attention to the uniqueness of uneven data distribution in imbalance classification problems. Without change the original imbalance training data, this paper indicated the advantages of proximal classifier for imbalance data classification. In order to improve the accuracy of classification, this paper proposed a new model named LS-NPPC, based the classical proximal SVM models which find two nonparallel planes for data classification. The LS-NPPC model is applied to six UCI datasets and one real application. The results indicate the effectiveness of the proposed model for imbalanced data classification problems.**

***Index Terms*—Class imbalance learning, Twin support vector machine, Nonparallel plane, Proximal classifier, least square one class support vector machine (LS-OCSVM)**

I. INTRODUCTION

Learning from imbalanced data sets is an important issue in machine learning research. The class imbalance problem corresponds to domains for which one class is represented by a large number of instances while the other is represented by only a few. It occurs frequently in datasets from many real-world applications. Traditional classifiers that seek accuracy over a full range of instances are not suitable to deal with imbalanced learning tasks, because they tend to be overwhelmed by the majority class which is usually the less important class. This paper addresses the challenge on Nonparallel Plane Proximal Support Vector Machine and proposes a novel LS-NPPC model for class imbalance data classification.

On the basis of statistical learning theory, Support Vector Machine (SVM) was proposed as computationally powerful tools for supervised learning including both classification and regression [1-4, 36-37]. SVM have established themselves as a successful approach for various machine learning tasks. But as for the extremely imbalanced datasets, the decision boundary of SVM obtained from the training data is largely biased toward the minority class. A large number of studies [10-16] have been conducted for investigating the impact of class imbalance on supervised machine learning. There are roughly two types of approaches to deal with imbalanced data classification problems. One is to instance data, either randomly or intelligently, to obtain an altered class distribution. These approaches consist of under-sampling the majority class or over-sampling the minority class, such as randomly under-sampling, randomly over-sampling, one sided selection, cluster-based over-sampling, Wilson's editing, SMOTE, and borderline-SMOTE [17-21]. The other is to modify the standard learning algorithms, such as cost-sensitive methods [12-13], margin calibration method [14], unsupervised self-organizing method [15], minimax probability machines [16,17], and one-class support vector machine [18].TWSVM [2, 3, 32-35] was proposed by the proximity of patterns to one of the two nonparallel planes. Instead of finding a single hyper-plane, TWSVM finds two nonparallel hyper-planes such that each plane is clustered around one particular class data. For this purpose it solves two smaller sized quadratic programming problems (QPPs) instead of solving large one as in the standard SVM [1]. As standard SVM, TWSVM achieves good classification accuracy.

In this paper, based on the idea of nonparallel plane classifier, we propose a novel nonparallel plane proximal classifier, called LS-NPPC, to build two decision hyper-planes for minority class and majority class respectively. The new method enhances utilization of the minority class data. The experimental results on benchmark data sets show the prediction accuracy of the minority class is apparently improved.

The paper is organized as follows. The backgrounds including the classical SVM and TWSVM are introduced in Section 2. In Section 3, we discussed the advantage of nonparallel plane classifier for imbalance data classification and built a novel LS-NPPC model. Following that, we evaluate LS-NPPC model on a series of benchmark data experiments and a real-world application in Section 4. Section 5 is the conclusion.

II. BACKGROUND

Consider the traditional classification problem with the training set $T = \{(x_1, y_1),...,(x_l, y_l)\}$, where

*Corresponding author
Manuscript received Jan 19, 2014; revised Mar 7, 2014; accepted March 25, 2014

doi:10.4304/jsw.9.9.2384-2392

$x_i \in R^n$ (i=1,…,l) are input, $y_i \in Y = \{-1,+1\}$ (i=1,…,l) are output.

A. Support Vector Machine (SVM)

The classical SVM method searches for such a hyperplane: $g(x) = w^T x + b$ that maximizes the margin between the training instance points for class +1 and class -1:

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i \\ &s.t. \qquad y_i[(w^T \cdot x_i)+b] \geq 1-\xi_i \qquad (2\text{-}1\text{-}1) \\ &\qquad \xi_i \geq 0, i = 1,\cdots,l \end{aligned}$$

Where $w \in R^n, b \in R$. $C > 0$ is a parameter used to tune the tradeoff between the maximum margin and the minimum training error, and ξ_i are slack variables to deal with the linearly non-separable problem.

The Lagrangian of (2-2-1) is:

$$L(w,b,\xi,\alpha,\beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i - \sum_{i=1}^{l}\alpha_i\left[y_i(w^T x_i + b) - 1 + \xi_i\right] - \sum_{i=1}^{l}\beta_i\xi_i \qquad (2\text{-}1\text{-}2)$$

Where $\alpha_i \geq 0$ and $\beta_i \geq 0$ are Lagrange multipliers.

The necessary conditions for the optimality are:

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{l}\alpha_i y_i x_i \\ \dfrac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{l}\alpha_i y_i = 0 \\ \dfrac{\partial L}{\partial \xi_i} = 0 \Rightarrow C - \alpha_i - \beta_i = 0 \end{cases} \qquad (2\text{-}1\text{-}3)$$

Based on Eq. (2-2-3), the optimization problem (2-2-1) could be converted into the following dual problem:

$$\begin{aligned} &\max_{\alpha} \quad -\frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j \left\langle x_i, x_j \right\rangle + \sum_{i=1}^{l}\alpha_i \\ &s.t. \quad \sum_{i=1}^{l}\alpha_i y_i = 0 \qquad (2\text{-}1\text{-}4) \\ &\qquad C \geq \alpha_i \geq 0, i = 1,\cdots,l \end{aligned}$$

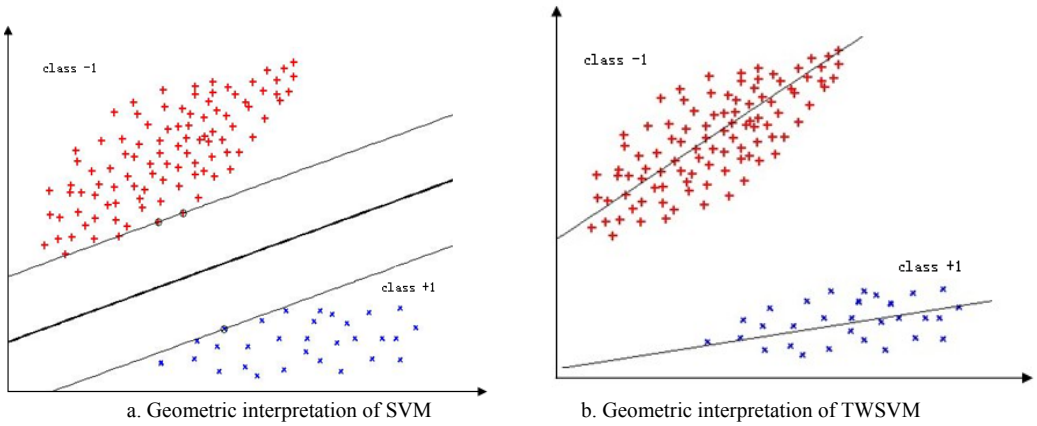
To predict a new instance x's class, the decision function is given as follows:

$$f(x) = \mathrm{sgn}(g(x)) = \mathrm{sgn}\left(\sum_{i=1}^{l}\alpha_i y_i \left\langle x_i, x \right\rangle + b\right) \qquad (2\text{-}1\text{-}5)$$

B. Twin Support Vector Machine and Nonparallel Plane Proximal Classifier

The essential idea of classical SVM is to search a linear separating hyperplane which maximizes the distance between two classes of data to create a classifier [2]. For classifying a new instance, the classifier will judge which half space the new instance points located (Fig.1a).

Instead of finding a single hyper-plane in classical SVM, Twin Support vector Machine (TWSVM) [2] is proposed, which finds two nonparallel hyper-planes such that each plane is clustered around one particular class data (Fig.1b).



a. Geometric interpretation of SVM b. Geometric interpretation of TWSVM

Figure 2-1: For a binary classification problem, (a): the principle of classical SVM, the decision plane (thick line) is closely related to a few support vectors (circled points) and it classifies points by assigning them to one of two disjoint half-plane. (b): TWSVM, It can find two nonparallel planes that are pushed apart as far as possible, and points are classified depending on which of the two planes they lies closest to.

The model of TWSVM is following:

(TWSVM1)

$$\underset{(w_1,b_1,\xi_1)}{Min} \quad \frac{1}{2}\left\|Aw_1+e_1b_1\right\|^2+c_1e_2^T\xi_2$$
$$s.t. \qquad -(Bw_1+e_2b_1)+\xi_2\geq e_2,\xi_2\geq 0$$

And (2-2-1)

(TWSVM2)

$$\underset{(w_2,b_2,\xi_2)}{Min} \quad \frac{1}{2}\left\|Bw_2+e_2b_2\right\|^2+c_2e_1^T\xi_1$$
$$s.t. \qquad (Aw_2+e_1b_2)+\xi_1\geq e_1,\xi_1\geq 0$$

Where metrics $A\in\Re^{m_1\times n}$ and $B\in\Re^{m_2\times n}$ represent the instances of class +1 and -1 respectively, $w_1,w_2\in\Re^n$ is weight vector, $b_1,b_2\in\Re$ is the bias terms of respective planes, c_1,c_2,c_3 and $c_4>0$ is parameter, $e_1\in\Re^{m_1}$ and $e_2\in\Re^{m_2}$ is one vector, $\xi_1\in\Re^{m_1},\xi_2\in\Re^{m_2}$ is slack variable.

Then two hyperplanes $w_1^Tx+b_1=0$ and $w_2^Tx+b_2=0$ can be obtained from the solution of TWSVM1 and TWSVM2. A new instance $x\in\Re^n$ is assigned to class +1 or -1 depending on which of the two hyperplanes lies closest to the point in terms of perpendicular distance. Finally, the decision function can be written as:

$$Class\,k=\min_{k=1,2}\left|x^{\mathrm{T}}\,w_k+b_k\right| \qquad (2\text{-}2\text{-}2)$$

Nonparallel Plane Proximal Classifier (NPPC) [3,32] is an improvement formulation of TWSVM. The formulation of NPPC for binary data classification is based on two identical mean square error (MSE) optimization problems which lead to solving two small systems of linear equations in input space. Thus it eliminates the need of any specialized software for solving the quadratic programming problems [3]. The model of NPPC is following:

(NPPC1)

$$\underset{(w_1,b_1,\xi_1)}{Min} \quad \frac{1}{2}\left\|Aw_1+e_1b_1\right\|^2+c_1e_2^T\xi_2+\frac{c_2}{2}\xi_2^T\xi_2$$
$$s.t. \qquad -(Bw_1+e_2b_1)+\xi_2=e_2$$

And (2-2-3)

(NPPC2)

$$\underset{(w_2,b_2,\xi_2)}{Min} \quad \frac{1}{2}\left\|Bw_2+e_2b_2\right\|^2+c_3e_1^T\xi_1+\frac{c_4}{2}\xi_1^T\xi_1$$
$$s.t. \qquad (Aw_2+e_1b_2)+\xi_1=e_1$$

Where metrics $A\in\Re^{m_1\times n}$ and $B\in\Re^{m_2\times n}$ represent the instances of class +1 and -1 respectively, $w_1,w_2\in\Re^n$ is weight vector, $b_1,b_2\in\Re$ is the bias terms of respective planes, c_1,c_2,c_3 and $c_4>0$ is parameter, $e_1\in\Re^{m_1}$ and $e_2\in\Re^{m_2}$ is one vector, $\xi_1\in\Re^{m_1}$, $\xi_2\in\Re^{m_2}$ is slack variable. Same as TWSVM, for a new instance $x\in\Re^n$, the decision function can be written as:

$$Class\,k=\min_{k=1,2}\left|x^{\mathrm{T}}\,w_k+b_k\right| \qquad (2\text{-}2\text{-}4)$$

III. METHOD

A. *Motivations*

To this day, a large number of studies proposed all kinds of methods to deal with imbalanced data classification. But, whether sampling the imbalanced data, including over-sampling and under-sampling, or modifying the classical learning algorithms, there are still some difficulties in the imbalanced data learning:

- There are only a few minority class data.

The lack of minority class data will lead to the decision boundary is largely biased toward the minority class. It greatly reduced the prediction accuracy of the minority class.

- The problem of data fragmentation

Some divide-and-compute algorithms, such as under-sampling method, are leading to the problem of data fragmentation.

- Poor parameters will hurt the performance of method.

In some classical imbalanced data learning algorithms, such as cost-sensitive methods, the bias of imbalance two classes will be reflected in the parameters of the model.

- The affect of noises.

In the imbalanced data learning, because of the lack of minority class data, it is difficult to discriminate one instance is belong to minority class instance or it is noises instance. Therefore noises bring great influence to classify minority class data. It is an important challenge that how to suppress noise affects on the minority class data.

Based above challenges in imbalanced data learning, we find proximal based SVM has inherent advantages in imbalanced data learning. (1) Different from classical SVM, proximal based SVM build the decision hyper-plane by all the data. Although lost the decision hyper-plane are not sparse, but improved the utilization of data, especially data of minority class, can be boosting the prediction accuracy of the minority class. (2) Compared with some divide-and-compute algorithms, proximal based SVM considering all the data by one model, the problem of data fragmentation can be avoid. It can be more extensive application with different distributions of data. (3) The bias of imbalance two classes is natural reflected in the proximal based SVM model. It is not

affected by the alteration of the parameters from learning process. (4) Obviously, the influence of noise on proximal based classifier is greatly reduced, since all training data is working on the final decision hyper-plane.

Based above reason, we will consider proposed a proximal based SVM for imbalanced data classification. In order to improving the generalization ability for imbalanced data classification and cost less time, we made some improvements on the NPPC model and proposed LS-NPPC formulation as following.

B. The LS-NPPC Formulation

For the NPPC model, a large amount of instance from the majority class will be used in the minority class decision hyper-plane which built by NPPC1 model. In order to avoid the largely biased of the decision boundary toward the minority class, as it can greatly impact the performance of the classifier, we only use the instance of minority and removed the instance of majority class in NPPC1. This can improve the utilization rate of minority class instance, thus boosting the prediction accuracy of the minority class. Here we transfer NPPC1 model to a novel optimization problem which find the proximal hyper-plane of minority class. Because the least square method is used, the new model is called LS-NPPC. The new LS-NPPC model as following:

(LS-NPPC1)

$$\underset{(w_1,b_1,\xi_1)}{Min} \quad \frac{1}{2}\left\|w_1\right\|^2 + b_1 + \frac{c_1}{2}\xi_1^T\xi_1$$
$$s.t. \qquad (Aw_1 + e_1b_1) + \xi_1 = 0$$

And (3-2-1)

(LS-NPPC2)

$$\underset{(w_2,b_2,\xi_2)}{Min} \frac{1}{2}\left\|Bw_2 + e_2b_2\right\|^2 + c_2e_1^T\xi_2 + \frac{c_3}{2}\xi_2^T\xi_2$$
$$s.t. \qquad (Aw_2 + e_1b_2) + \xi_2 = e_1$$

Where metrics $A \in \Re^{m_1 \times n}$ and $B \in \Re^{m_2 \times n}$ represent the instances of minority class and majority class respectively, $w_1, w_2 \in \Re^n$ is weight vector, $b_1, b_2 \in \Re$ is the bias terms of respective planes, c_1, c_2, c_3 and $c_4 > 0$ is parameter, $e_1 \in \Re^{m_1}$ and $e_2 \in \Re^{m_2}$ is one vector, $\xi_1 \in \Re^{m_1}$, $\xi_2 \in \Re^{m_2}$ is slack variable.

It is noteworthy that the LS-NPPC1 problem can be consider as a version of least square one class support vector machine (LS-OCSVM) [4]. Compared with LS-OCSVM, the advantage of LS-NPPC1 is that: LS-OCSVM can't be used in classification problem, since it cans only ranking the new instance and can't determine the new instance's class. But LS-NPPC1 can be classifying the new instance with LS-NPPC2.

The LS-NPPC1 problem in (3-2-1) can be solved as follows. By introducing Lagrangian multipliers α_1, the corresponding objective function can be written as the following.

$$L_1(w_1, b_1, \xi_1, \alpha_1) = \frac{1}{2}\left\|w_1\right\|^2 + b_1 + \frac{c_1}{2}\xi_1^T\xi_1 - \alpha_1^T[(Aw_1 + e_1b_1) + \xi_1] \qquad (3\text{-}2\text{-}2)$$

The Karush–Kuhn–Tucker (KKT) optimality conditions [5] for LS-NPPC1 are obtained by equating the stationary points of (3-2-2) to zero as follows:

$$\frac{\partial L}{\partial w_1} = w_1 - A^T\alpha_1 = 0$$
$$\frac{\partial L}{\partial b_1} = e_1^T\alpha_1 = 1$$
$$\frac{\partial L}{\partial \xi_1} = c_1\xi_1 - \alpha_1 = 0 \qquad (3\text{-}2\text{-}3)$$

And

$$(Aw_1 + e_1b_1) + \xi_1 = 0$$

Combination of above expressions leads to:

$$\xi_1 + (AA^T\alpha_1 + e_1b_1) = 0 \qquad (3\text{-}2\text{-}4)$$
$$(AA^T + \frac{I}{c_1})\alpha_1 + e_1b_1 = 0$$

With the linear constraint $e_1^T\alpha_1 = 1$, equations in (3-2-4) reduce to the following set of linear equations to solve:

$$\begin{bmatrix} 0 & e_1^T \\ e & AA^T + \frac{I}{c_1} \end{bmatrix}\begin{bmatrix} b_1 \\ \alpha_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad (3\text{-}2\text{-}5)$$

Using the block matrix inversion [6,7], one can easily obtain b_1, α_1 and w_1 as follows.

$$b_1 = 1 / e^T(\frac{I}{c_1} + AA^T)^{-1}e$$
$$\alpha_1 = ((\frac{I}{c_1} + AA^T)^{-1}e) / (e^T(\frac{I}{c_1} + AA^T)^{-1}e) \qquad (3\text{-}2\text{-}6)$$
$$w_1 = A^T\alpha_1$$

Note that α_1 satisfies the linear constraint $e^T\alpha_1 = 1$. The hyper-plane for LS-NPPC1 obtained from (3-2-6) can be written in a vector form as follows:

$$f_1(x) = w_1^T x + b_1 \qquad (3\text{-}2\text{-}7)$$

Meanwhile the LS-NPPC2 problem in (3-2-1) also can be transfer to linear equations to solve [3]. And we can get its analytic solution as follow:

$$L_2(w_2, b_2, \xi_2, \alpha_2) = \frac{1}{2}\left\|Bw_2 + e_2 b_2\right\|^2 + c_2 e_1^T \xi_2 + \frac{c_3}{2}\xi_2^T \xi_2 - \alpha_2^T[(Aw_2 + e_2 b_2) + \xi_2 - e_2] \quad (3\text{-}2\text{-}8)$$

Where L_2 is the Lagrangian of LS-NPPC2, and α_2 is the vector of Lagrange multipliers

Let $H = [A \quad e_1]$, $u = [w_1 \quad b_1]$

$$\begin{bmatrix} w_2 \\ b_2 \end{bmatrix} = (G^T G)^{-1} H^T \alpha_2$$

$$\alpha_2 = c_3\left\{I - H(G^T G)^{-1}\left[\frac{I}{c_3} + H^T H (G^T G)^{-1}\right]^{-1} H^T\right\}(\frac{c_2}{c_3} + 1)e_1 \quad (3\text{-}2\text{-}9)$$

The hyper-plane for LS-NPPC2 obtained from (3-2-9) can be written in a vector form as follows:

$$f_2(x) = w_2^T x + b_2 \quad (3\text{-}2\text{-}10)$$

After two hyper-planes $w_1^T x + b_1 = 0$ and $w_2^T x + b_2 = 0$ are obtained from the solution of LS-NPPC1 and LS-NPPC2. A new instance data $x \in R^n$ is assigned to class +1 or -1 depending on which of the two hyper-planes lies closest to the point in terms of perpendicular distance. Finally, the decision function can be written as:

$$Class\, k = \min_{k=1,2}\left|\mathrm{x}^{\mathrm{T}} w_k + \mathrm{b}_k\right| \quad (3\text{-}2\text{-}11)$$

We now cite our algorithm for implementation of LS-NPPC as follow.

Algorithm 1 Least Square Nonparallel Plane Proximal Classifier

Input: *Training data m_1 and m_2 is training instances of classes +1 and -1 respectively, in n-dimensional space represented by matrices A and B.*

(1): *Compute the Lagrange multipliers α_1 and α_2 from (3-2-6) and (3-2-9) with some positive values of c_1, c_2, c_3. Typically these values are chosen by means of a tuning set.*

(2): *Determine the vectors w_1, w_2 and b_1, b_2 from (3-2-6) and (3-2-9) respectively, to obtain the two decision nonparallel hyper-plane (3-2-7) and (3-2-10).*

(3): *Classify a new instance $x \in R^n$ by using (3-2-11)*

IV. Experiment and Result

In this section, the LS-NPPC model will be applied to six UCI datasets and one real application. The effectiveness of the proposed model for imbalanced data classification problems will be tested.

A. UCI Dataset

Six UCI imbalanced datasets [9] are used in our experimental study to test our proposed method, including Glass, Segment, Abalone, Satalog, Letter4 and Letter1. The numbers in the parentheses indicate which classes are selected as minority (positive) class, and all others are used as majority (negative) class. These datasets are often appeared in related works about imbalanced research. The basic information about these datasets is summarized in Table 4-1 including the size of every dataset, the number of features, the number of positive instances and negative instances in each dataset respectively. These datasets are seriously selected to vary in data size (from several hundreds to tens of thousands) and imbalance ratio; the first two datasets are mildly imbalanced, while the rest ones are highly imbalanced as less than 10% instances are positive.

TABLE 4-1
CHARACTERISTICS OF 6 UCI DATASETS USED IN THIS PAPER

Dataset	Instances	Features	Pos/Neg	Ratio of Pos (%)
Glass	214	9	29/185	13.55
Segment	2310	19	330/2280	14.28
Abalone	4177	7	391/3786	9.37
Satalog	6435	36	626/5809	9.73
Letter4	20000	16	805/19185	4.03
Letter1	20000	16	789/19211	3.95

We employ g-means, sensitivity, and specificity [10] which are the popular measures for imbalanced learning to evaluate the performance of algorithms in our experiments.

(1) g-means = $acc^+ \times acc^-$

(2) sensitivity = TP/(TP + FN) (4-1-1)

(3) specificity = TN/(TN + FP)

Where acc^+ indicates the sensitivity and acc^- the specificity; TP and TN denote true positives and true negatives, respectively. FN and FP denote false negatives and false positives, respectively. Sensitivity is defined as the accuracy on the positive class and specificity is the accuracy on the negative class. The value of g-means is high when both acc^+ and acc^- are high as well as the difference between acc^+ and acc^- is small. Obviously, if a classifier is highly biased toward one class, the G-mean value would be low. Hence G-mean is used to compare classification performance of models with imbalanced datasets in our experiments.

In the following, LS-NPPC algorithm is compared with four other popular methods, which are SVM, Under-Sampling, SMOTE and Cost-sensitive learning. In our experiments we use 10-fold cross-validation to train our classifier since it provides more realistic results. Each dataset in Table 4-1 is trained respectively by all methods, while finding the optimal parameters the final 10-fold cross-validation results can be obtained. SVM is implemented in LIBSVM [9], and LS-NPPC is implemented by writing procedure in Matlab, but the results of other three methods on different datasets are referred to different literatures [17-21] as shown in Table 4-2. Our G-mean measure is gained by running the experiment 10 times with different train and test data sets, besides the measurement of G-mean, the standard error of the G-mean are also given to our classifier. Table 4-2 also illustrates the comparison of the G-mean value by our method and the rest ones.

TABLE 4-2
G-MEAN OF FIVE METHODS OVER ALL DATASETS

Dataset	SVM	Under-sampling	SMOTE	Cost sensitive learning	LS-NPPC
Glass	0.8666	0.8801	0.8771	0.9199	**0.9265**
Segment	0.9792	0.9918	0.9765	**0.9950**	0.9743
Abalone	0	0.765	0.742	0.412	**0.7855**
Statlog	0.7678	**0.871**	0.862	0.761	0.8648
Letter1	0.9712	0.994	**0.995**	0.988	**0.9950**
Letter4	0.8876	0.5354	0.7078	0.8890	**0.9060**
Mean	0.7454	0.8396	0.86	0.8247	**0.9087**

Table 4-2 shows obviously that the performance of LS-NPPC algorithm is better than previously proposed methods, the G-mean value of LS-NPPC is higher than other methods on most of experiment datasets. There are only two datasets on which LS-NPPC does not perform best, but the G-mean value of LS-NPPC is much close to the best result. The last line of Table 4-2 displays the average G-mean values over all datasets of each methods, the result of LS-NPPC is higher overwhelmingly than other methods, and the standard errors show that LS-NPPC is stabile too.

B. Detection of Horizontal Gene Transfer in Bacterial Genomes

In this section, LS-NPPC algorithm will be test in a real application. It is used to detect the horizontally transferred genes.

Horizontal Gene Transfer (HGT), also called Lateral Gene Transfer (LGT), is defined as movement of genetic material between different species, or across broad taxonomic categories. Although most thinking in genetics has focused on the more prevalent vertical transfer, there is a recent awareness that horizontal gene transfer is a significant phenomenon [29].

In our research, we simulated the HGT between bacterial and phage genomes which indeed happens in real nature. For example, Lysogenic phages are able to integrate into the host genome and become part of the genetic material which make up of the host bacterium. Furthermore, transduction is regarded as one of three main mechanisms of gene transfer in prokaryotes, which is a bacteriophage-facilitated transfer of genetic material from one bacterial host to another [30]. Therefore, we used three complete bacterial genomes including Escherichia coli str. K-12, Bacillus cereus E33L and Borrelia burgdorferi B31 as host genomes. Meanwhile we took the overall 1574 gene sequences of 27 phages as donor pool, and randomly sub-selected an appropriate fraction of these genes that were then incorporated into the bacterial host genome [31]. The intention of our experiment was to recover as many as possible of the inserted donor genes. All of these sequence data were downloaded from NCBI/Gen-Bank.

In this study, the training data sets are large and imbalanced for detection of HGT. In artificial simulated experiments, for example, the genome of Bacillus cereus E33L contains 103 positive instances data and 5134 negative instances data in which there are noises. In other words, the training dataset is highly imbalanced and there are some positive data in negative class and we don't know which one is false, also we know the false negative ones may distribute closed to true positive instances data.

For each of the 3 host bacterial genome in turn, we carried out 100 experiments of simulated transfers from a gene pool composed of 1574 gene sequences of 27 phages [31]. In each experiment, the number of added genes was chosen to be a fixed percentage of the number of genes in the host genome, as to the genome of Escherichia coli str. K-12 and Bacillus cereus E33L, the transfer percentages were both 2%, but we chose 8% for

the genome of Borrelia burgdorferi B31 because its genome size was smaller. The transferred genes were selected at random from the donor pool, the objective is to recover as many of the artificially transferred genes as possible, without using any a priori knowledge about the host genome or the donor genes.

For each training procedure, discrimination functions whose coefficients could be calculated were used to discriminate the test data consisting of only phage genes. Consequently, we also used 'hit ratio (HT)' [29-31] to denote the proportion between the number of artificially inserted genes and the number of genes recovered by a particular procedure. HT can be calculated by

$$HT = \frac{1}{100}\sum_{i=1}^{100}\frac{PT_i}{NT_i} \quad (4\text{-}2\text{-}1)$$

Where PT_i and NT_i represent the number of genes and the number of predicted horizontally transferred genes in test dataset respectively. So HT was used to evaluate the reasonability of a method for the detection of HGT.

Furthermore, we compared our methods against other methods including C-SVC and OC-SVM which were used by Aristotelis T. and Isidore R. in 2005 [29,30] and Jiansheng Wu et al. in 2007 [31]. In experiments we used LS-NPPC implemented by writing procedure in Matlab. Table 4-3 and Figure 4-2 show the comparison of the HT by our two methods and previous proposed methods on each bacterial genome. It can be seen from the results that our proposed methods outperform all the other methods in this case.

TABLE 4-3
RESULTS WITH THREE DATASETS OF HT BETWEEN LS-NPPC AND PREVIOUS METHODS

Dataset	Aristotelis*'s method*	*Wu's method*	*C-SVM*	*OC-SVM*	*LS-NPPC*
Escherichia colistr. K-12	0.375	0.539	0.493	0.346	**0.765**
Bacillus cereus E33L	0.541	0.647	0.571	0.566	**0.836**
Borrelia burgdorferi B31	0.758	0.945	0.767	0.846	**0.962**

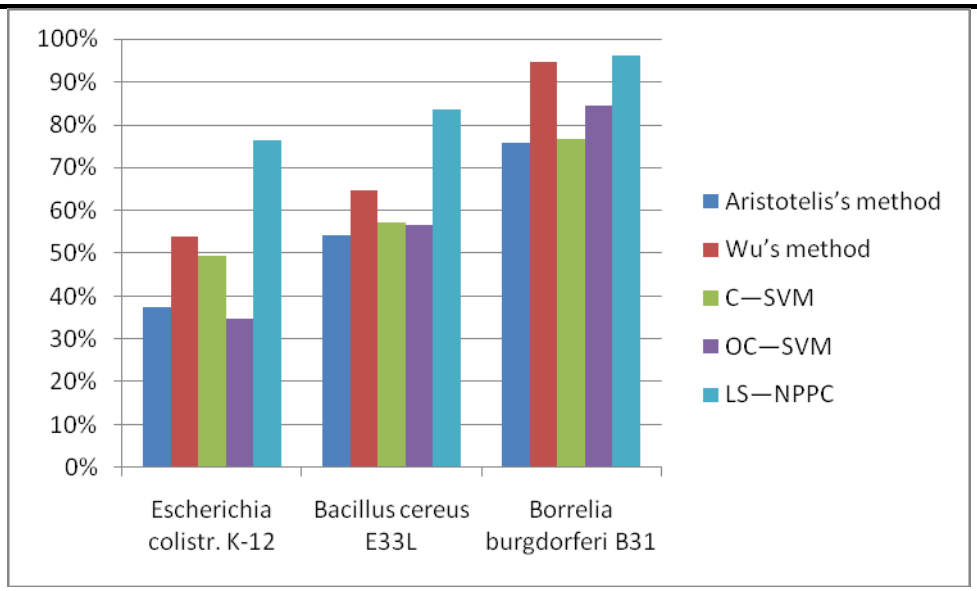


Figure 4-2: The percent of the result of HT from the result of Table 4-3.

The results of first four columns in Table 4-3 are obtained from [31], the last one is the result of LS-NPPC. Obviously, HT of LS-NPPC is much higher than previous proposed methods. It also can be observed that proposed methods get higher relative improvements on Escherichia coli str. K-12 and Bacillus cereus E33L genomes than on Borrelia burgdorferi B31 genome, the reason is that the codon usage in Borrelia burgdorferiB31 genome is strongly biased which is related to the translation and codon choice of a gene [31]. It proves that LS-NPPC algorithm is suitable for the detection of HGT.

V. CONCLUSON

In order to improve the prediction accuracy of imbalance data classification problem, this paper pays close attention to the uniqueness of uneven data distribution in imbalance classification problems. Without change the original imbalance training data, this paper indicated the advantages of proximal classifier for imbalance data classification. Based on nonparallel plane proximal classifier model, we proposed the new SVM model, LS-NPPC method. The new method can get better prediction accuracy and cost less time. The advantages of new method can be mainly summarized as the following:

(1) Because the enhancing of the minority class data utilization rate, the prediction accuracy of the minority class can be apparently improve.

(2) Considering all the data by one classification model, the problem of data fragmentation, which produced by using the divide and compute method classification algorithms, can be avoid. New method can be more extensive application with different distributions of data.

(3) The bias of imbalance two classes is natural reflected in the LS-NPPC model. It is not affected by the alteration of the parameters from learning process.

(4) The LS-NPPC method construct decision hyper plane by all training data. The influence of noise on classifier is greatly reduced.

In experiment studies, the proposed method was applied to six UCI datasets and one real application, and the results confirmed its better performance than previously proposed methods for imbalance data classification problem. Further research is required for discussing the relationship between the performance of the algorithm and the parameter in the model, and developing the parameter search algorithm to search for the best setting. Meanwhile the linear kernel function just is used in LS-NPPC model. How to handle the non-linear kernel function in LS-NPPC model is another area we would like to further investigate.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (11071252, 11371365, 11301535), Chinese Universities Scientific Fund (2013YJ010).

REFERENCES

[1] V. Vapnik and C. Cortes,. Support vector networks, Machine Learning, 1995 20(3): 273-297.
[2] Jayadeva, Khemchandani, R., Chandra, S., Twin support vector machines for pattern classification, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(5), 905–910.
[3] Santanu Ghorai, Anirban Mukherjee, Pranab K. Dutta, Nonparallel plane proximal classifier, Signal Processing, 2009, 89,510–512.
[4] Young-Sik Choi, Least squares one-class support vector machine, Pattern Recognition Letters 30 (2009) 1236–1240.
[5] M.S. Bazarra, H.D. Sherali, C.M.Shetty, Nonlinear Programming - Theory and Algorithms, second ed., Wiley, 2004 Chapter4, 149–172.
[6] O.L. Mangasarian, E.W. Wild, Multisurface proximal support vector classification via generalized eigen values, IEEETrans. Pattern Anal. Machine Intell. 28(1) (2006)69–74.
[7] K.S. Chua, Efficient computations for large least square support vector machine classifiers, Pattern Recognition Letter 24(2003) 75–80.
[8] P.M. Murphy, D.W. Aha, UCI machine learning database repository, 1985.
[9] http://www.ics.uci.edu/mlearn/MLRepository.html.
[10] Zhuangyuan Zhao, Ping Zhong, Yaohong Zhao, Learning SVM with weighted maximum margin criterion for classification of imbalanced data, Mathematical and Computer Modelling 54 (2011) 1093–1099.
[11] T. Joachims, Text categorization with support vector machines: learning with many relevant features, in: Proceeding of ECML-98 10th European Conference on Machine Learning, 1998.
[12] I. Maglogiannis, E. Zafiropoulos, I. Anagnostopoulos, An intelligent system for automated breast cancer diagnosis and prognosis using SVM based classifiers, Applied Intelligence 30 (2009) 24–36.
[13] Matías Di Martino, Alicia Fernández, Pablo Iturralde, Federico Lecumberry, Novel classifier scheme for imbalanced problems, Pattern Recognition Letters 34 (2013) 1146–1151.
[14] R. Ji, D. Li, L. Chen, W. Yang, Classification and identification of foreign fibers in cotton on the basis of a support vector machine, Mathematical and Computer Modeling 51 (11–12) (2010) 1433–1437.
[15] S. Cai, R. Zhang, L. Liu, D. Zhou, A method of salt-affected soil information extraction based on a support vector machine with texture features, Mathematical and Computer Modeling 51 (11–12) (2010) 1319–1325.
[16] Shaoning Pang, Lei Zhu, Gang Chen, Abdolhossein Sarrafzadeh, Tao Ban, Daisuke Inoue, Dynamic class imbalance learning for incremental LPSVM, Neural Networks 44 (2013) 87–100.
[17] R. Barandela, R.M. Valdovinos, J.S. Sanchez, F.J. Ferri, The imbalanced training sample problem: under or over sampling? in: Joint IAPR International Workshops on Structural, Syntactic, and Statistical Pattern Recognition, SSPR/SPR'04, in: Lecture Notes in Computer Science, vol. 3138, 2004, pp. 806–814.
[18] N. Chawla, K. Bowyer, L. Hall, W. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, Journal of Artificial Intelligence Research 16 (2002) 321–357.
[19] H. Han, W. Wang, B. Mao, Borderline-smote: a new over-sampling method in imbalanced data sets learning, in: International Conference on Intelligent Computing, ICIC'05, in: Lecture Notes in Computer Science, vol. 3644, 2005, pp. 878–887.
[20] T. Jo, N. Japkowicz, Class imbalances versus small disjuncts, SIGKDD Explorations 6 (2004) 40–49.
[21] M. Kubat, S. Matwin, Addressing the curse of imbalanced training sets: one-sided selection, in: Proceeding of the 14th International Conference on Machine Learning, 1997.
[22] D. Margineantu, T.G. Dietterich, Bootstrap methods for the cost-sensitive evaluation of classifiers, in: Proceeding of International Conference on Machine Learning, 2000.
[23] Y. Sun, M. Kamela, A. Wongb, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recognition 40 (2007) 3358–3378.
[24] C. Yang, J. Yang, J. Wang, Margin calibration in SVM class-imbalanced learning, Neurocomputing 73 (1–3) (2009) 397–411.
[25] Y.Y. Nguwi, S.Y. Cho, An unsupervised self-organizing learning with support vector ranking for imbalanced datasets, Expert Systems with Applications 37 (12) (2010) 8303–8312.
[26] G. Lanckriet, L.E. Ghaoui, C. Bhattacharyya, M.I. Jordan, A robust minimax approach to classification, Journal of Machine Learning Research 3 (2002) 555–582.
[27] K. Huang, H. Yang, I. King, et al., The minimum error minimax probability machine, Journal of Machine Learning Research 5 (2004) 1253–1286.
[28] L.M. Manevitz, M. Yousef, One-class SVMs for document classification, Journal of Machine Learning Research 2 (1) (2001) 139–154.
[29] Aristotelis T. and Isidore R., A new computational method for the detection of horizontal gene transfer events, Nucleic Acids Research, 2005, 33(3): 922-933.
[30] Aristotelis T. and Isidore R., A sensitive, support-vector-

machine method for the detection of horizontal gene transfers in viral, archaeal and bacterial genomes, Nucleic Acids Research, 2005, 33(12): 3699–3707.
[31] Jiansheng Wu, Jianming Xie, Tong Zhou, Jianhong Weng, Xiao Sun, Support Vector Machine for Prediction of Horizontal Gene Transfers in Bacteria Genomes, Progress in Biochemistry and Biophysics, 2007, 34(7):724–731.
[32] Y.-H. Shao, W.-J. Chen,N.-Y. Deng. Nonparallel hyperplane support vector machine for binary classification problems. Information Sciences, 2014, 263(1) 2014, 22–35
[33] Y.-H. Shao, N.-Y. Deng, W.-J. Chen. A proximal classifier with consistency. Knowledge-Based Systems,2013, 49:171-178
[34] Y.-H. Shao, C.-H. Chun, X.-B. Wang, N.-Y. Deng. Improvements on Twin Support Vector Machines. IEEE Transactions on Neural Networks, vol.22 no.6 pp. 962-968, 2011
[35] Shifei Ding, Junzhao Yu, Huajuan Huang, Han Zhao. Twin Support Vector Machines Based on Particle Swarm Optimization. Journal of Computers. Vol. 8, no. 9 (2013)
[36] Quanjin Liu, Zhimin Zhao, Ying-xin Li, Xiaolei Yu, Yong Wang. A Novel Method of Feature Selection based on SVM. Journal of Computers. Vol. 8, no. 8 (2013)
[37] Hong-Min Zhu, Chi-Man Pun, Cong Lin. Robust Hand Gesture Recognition Using Machine Learning With Positive and Negative Samples. Journal of Computers. Vol 8, no. 7 (2013): Special Issue: Advances in Internet Technologies and Applications

Bing Yang was born in 1987.4, he was received Bachelor degrees from the China Agricultural University in 2009. Now he is a Ph.D student in College of Mathematics from China Agricultural University. His research interests include optimization methods, machine learning and data mining. He has published many refereed papers on these areas.

Ling Jing was born in 1967.8, she was received Ph.D. degrees from the Beihang University. Currently, she is a professor at College of Science, China Agricultural University. She has wide research interests, mainly including computational methods for optimization, operation research, support vector machine in data mining and bioinformatics. In these areas, she has published over 30 papers in leading international journals or conferences.