Nonparallel Hyperplane Support Vector Machine for PU Learning

Ying Zhang
School of Mathematical Sciences
University of Chinese Academy of Sciences
Research Center on Fictitious
Economy and Data Science, CAS
Beijing, China 100190

XuChan Ju
School of Mathematical Sciences
University of Chinese Academy of Sciences
Research Center on Fictitious
Economy and Data Science, CAS
Beijing, China 100190

YingJie Tian
Research Center on Fictitious
Economy and Data Science, CAS
Beijing, China 100190
corresponding author

***Abstract*—In this paper, we propose to apply the nonparallel support vector machine (NPSVM) for positive and unlabeled learning problem(PU learning problem) in which only a small positive examples and a large unlabeled examples can be used . Like Biased-SVM, NPSVM treats the unlabeled set as the negative set with noise, while NPSVM is modified so that, the first primal problem is constructed such that all the positive points make the contribution to the positive proximal hyperplane; for the second primal problem, only a part of negative points makes the contribution to the negative proximal hyperplane. And we also give the suggestion on the parameters selection. Experimental results show the efficiency of our method for PU learning problem.**

I. Introduction

The standard classification task is to construct a classifier based on a training set, which consists of labeled positive examples and labeled negative examples. Differently, the PU problem where the training set consists of a few Positive examples and a large collection of Unlabeled examples, appearing in many applications such as information retrieval or gene ranking, is now gaining more and more attention [1]–[6].

Obviously, traditional classification techniques are not suitable for the PU problem, although we can label some negative examples manually, it is labor-intensive and very time consuming. Currently, there are several kinds of methods for solving PU learning problem. The first kind dealing with the situation is ignoring the unlabeled examples, which means only positive examples can be used. Joachims [1] ranked the unlabeled examples by decreasing similarity to the mean positive examples, Maneviz and Yousef [2], De Bie [3], Geurts [4], Wenkai Li [5] provided and improved the one-class SVM [2] to solve this problem, while the numerical results showed that their performances are poorer to the learning methods that take advantage of the unlabeled examples. Some methods estimate statistical queries over positive and unlabeled examples. Denis et al [6] modified the naive Bayesian to solve this problem. The PNCT method [7] extended this idea to the Co-Training setting [8]. Topic-sensitive PLSA was used in Ke Zhou [9]

The second kind is the two-step strategy which attempts to select negative examples from the unlabeled data set. It has given a number of specific algorithms including NB [10], S-EM [11], PEBL [12], and Roc-SVM [13]. Two-step methods are iteratively built consisting of step 1 (Identifying the reliable negative examples from the unlabeled set) and step 2 (Building a set of classifiers by iteratively applying a classification algorithm and then selecting a good classifier from the set). Some different algorithms such as the PNLH (Positive and Negative examples Labeling Heuristic) method [14] go further along this way, and extract not only negative examples but also positive examples from the unlabeled set.

The third kind is the one-step strategy, which converts the PU problem into an unbalance binary classification problem, i.e., it treats the unlabeled set as the negative set with noise. Lee and Liu [15] performed logistic regression after weighting the examples to handle noise rates of greater than a half. Biased-PrTFIDF [16] used a simple probabilistic approach to learning from positive and unlabeled examples. The Biased-SVM [17] gave bigger weights to the positive examples and small weights to the unlabeled examples. The p-norm Biased-SVM [18] improved the performance of Biased-SVM and can select relevant features automatically. Recently, the idea of bagging learning was used to assign labels to unlabeled examples in [19] and [20]. Besides, Graph-based semi-supervised learning methods also attracts great attention. By adding the dissimilarity term, GBTPU model [21] can extract some negative examples from unlabeled examples so as to change PU learning to a classical binary classification problem.

Recently, a branch of SVM, nonparallel hyperplane SVM, is developed and has attracted many interests. The representative algorithms include the generalized eigenvalue proximal support vector machine (GEPSVM) [22] and the twin support vector machine (TWSVM) [23]. For TWSVM, it seeks two nonparallel proximal hyperplanes such that each hyperplane is closer to one of the two classes and is at least one distance from the other. It is implemented by solving two smaller quadratic programming problems (QPPs) instead of a larger one, which increases the TWSVM training speed by approximately fourfold compared to that of standard SVC. TWSVMs have been studied extensively [24]–[39].

Among the extensions of TWSVMs, most dwelled on the original TWSVM for different problems, while the nonparallel support vector machine (NPSVM) [37], [38] are superior

978-1-4799-5151-2/14/$31.00 ©2014 IEEE

theoretically and overcomes several drawbacks of the existing TWSVMs. NPSVM combines ε-support vector regression (ε-SVR) and C-support vector classification (C-SVC) together into one model and therefore has several advantages. In this paper, we propose to apply NPSVM for the PU problem. Like Biased-SVM, we treat the unlabeled set as the negative set with noise. We build different models for positive data set and unlabeled data set. For the positive data set, all the positive examples make the contribution to the positive proximal hyperplane. For the unlabeled data set, only a part of unlabeled examples make the contribution to the negative proximal hyperplane.

This paper is organized as follows. Section 2 dwells on the Biased-SVM and NPSVM. Section 3 proposes our NPSVM for PU learning problem. Section 4 deals with experimental results and Section 5 contains concluding remarks.

II. BACKGROUND

In this section, we briefly introduce the Biased SVM and the nonparallel classifier-NPSVM. First we give the definition of the PU binary classification problem as follows:

PU binary classification problem: Given a training set

$$T = \{(x_1, y_1), \cdots, (x_p, y_p)\} \cup \{x_{p+1}, \cdots, x_{p+q}\}, \tag{1}$$

where $x_i \in R^n, y_i = 1$, i.e. x_i is a positive input, $i = 1, \cdots, p$; $x_i \in R^n$, i.e. x_i is an unlabeled input known to belong to one of the two classes, $i = p+1, \cdots, p+q$, find a real function $g(x)$ in R^n such that the value of y for any x can be predicted by

$$f(x) = \mathrm{sgn}(g(x)). \tag{2}$$

A. Biased SVM

The above PU problem can be changed into an unbalance binary classification problem by supposing that the unlabeled examples in T are negative examples, i.e., the unlabeled $x_i \in R^n, i = p+1, \cdots, p+q$ are supposed to be labeled -1. Therefore, the training set T turns to be

$$\begin{gathered} T^* = \{(x_1, +1), (x_2, +1), \ldots, (x_p, +1)\} \cup \\ \{(x_{(p+1)}, -1), (x_{(p+2)}, -1), \ldots, (x_{(p+q)}, -1)\} \end{gathered} \tag{3}$$

For the training set T^*, Biased-SVM gives larger weights to the positive examples and smaller weights to the unlabeled examples:

$$\min_{w,b,\xi} \ \frac{1}{2}\|w\|^2 + C_+ \sum_{i=1}^{p} \xi_i + C_- \sum_{i=p+1}^{p+q} \xi_i \tag{4}$$

$$s.t. \ \ y_i((w \cdot x_i) + b) \geqslant 1 - \xi_i, i = 1, \cdots, p+q, \tag{5}$$

$$\xi_i \geqslant 0, i = 1, \cdots, p+q. \tag{6}$$

Intuitively, we give a bigger value for C_+ and a smaller value for C_- because the unlabeled set, which is assumed to be negative, contains positive inputs.

B. NPSVM

Consider the binary classification problem with the training set

$$T = \{(x_1, +1), \cdots, (x_p, +1), (x_{p+1}, -1), \cdots, (x_{p+q}, -1)\}, \tag{7}$$

where $x_i \in \mathcal{R}^n, i = 1, \cdots, p+q$.

For the linear case, NPSVM [37] seeks two nonparallel hyperplanes

$$(w_+ \cdot x) + b_+ = 0 \text{ and } (w_- \cdot x) + b_- = 0 \tag{8}$$

by solving two quadratic programming problems (QPPs)

$$\begin{aligned} \min_{w_+, b_+, \eta_+^{(*)}, \xi_-} \quad & \frac{1}{2}\|w_+\|^2 + C_1 \sum_{i=1}^{p} (\eta_i + \eta_i^*) + C_2 \sum_{j=p+1}^{p+q} \xi_j, \\ \text{s.t.} \quad & (w_+ \cdot x_i) + b_+ \leqslant \varepsilon + \eta_i, \ i = 1, \cdots, p, \\ & -(w_+ \cdot x_i) - b_+ \leqslant \varepsilon + \eta_i^*, \ i = 1, \cdots, p, \\ & (w_+ \cdot x_j) + b_+ \leqslant -1 + \xi_j, \\ & \qquad j = p+1, \cdots, p+q, \\ & \eta_i, \eta_i^* \geqslant 0, \ i = 1, \cdots, p, \\ & \xi_j \geqslant 0, \ j = p+1, \cdots, p+q. \end{aligned} \tag{9}$$

and

$$\begin{aligned} \min_{w_-, b_-, \eta_-, \xi_+} \quad & \frac{1}{2}\|w_-\|^2 + C_3 \sum_{i=p+1}^{p+q} (\eta_i + \eta_i^*) + C_4 \sum_{j=1}^{p} \xi_j, \\ \text{s.t.} \quad & (w_- \cdot x_i) + b_- \leqslant \varepsilon + \eta_i, \\ & \qquad i = p+1, \cdots, p+q, \\ & -(w_- \cdot x_i) - b_- \leqslant \varepsilon + \eta_i^*, \\ & \qquad i = p+1, \cdots, p+q, \\ & (w_- \cdot x_j) + b_- \geqslant 1 - \xi_j, \ j = 1, \cdots, p, \\ & \eta_i, \eta_i^* \geqslant 0, \ i = p+1, \cdots, p+q, \\ & \xi_j \geqslant 0, \ j = 1, \cdots, p. \end{aligned} \tag{10}$$

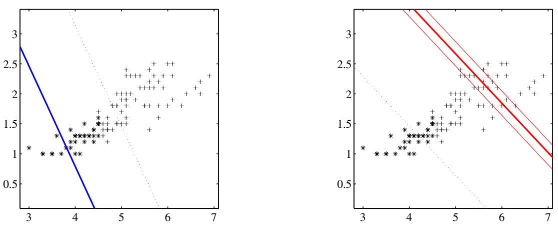
where $C_i \geqslant 0, i = 1, \cdots, 4$ are penalty parameters, ε is the parameter controlling the sparsity, $\xi_+ = (\xi_1, \cdots, \xi_p)^\top$, $\xi_- = (\xi_{p+1}, \cdots, \xi_{p+q})^\top$, $\eta_+^{(*)} = (\eta_+^\top, \eta_+^{*\top})^\top = (\eta_1, \cdots, \eta_p, \eta_1^*, \cdots, \eta_p^*)^\top$, $\eta_-^{(*)} = (\eta_-^\top, \eta_-^{*\top})^\top = (\eta_{p+1}, \cdots, \eta_{p+q}, \eta_{p+1}^*, \cdots, \eta_{p+q}^*)^\top$ are slack variables.

III. NPSVM FOR PU LEARNING PROBLEM

In this section, we will apply NPSVM to solve the PU problem.

A. linear case

Like Biased SVM, NPSVM first converts the PU learning problem by let the training set T (1) to be the training set T^* (3). Then NPSVM is modified based on the two problems



(a) (b)

Fig. 1. Geometrical illustration for two primal Problems

(9) and (10) separately: For the positive data set, problem (9) changes to be

$$
\begin{aligned}
\min_{w_+,b_+,\eta_+,\xi_-} \quad & \frac{1}{2}\|w_+\|^2 + C_1\sum_{i=1}^{p}\eta_i + C_2\sum_{j=p+1}^{p+q}\xi_j, \\
\text{s.t.} \quad & (w_+\cdot x_i) + b_+ \leqslant \eta_i,\ i=1,\cdots,p, \\
& -(w_+\cdot x_i) - b_+ \leqslant \eta_i,\ i=1,\cdots,p, \\
& (w_+\cdot x_j) + b_+ \leqslant -1+\xi_j, \\
& \qquad\qquad j=p+1,\cdots,p+q, \\
& \xi_j \geqslant 0,\ j=p+1,\cdots,p+q.
\end{aligned} \tag{11}
$$

and the negative data set, problem (10) keeps almost the same

$$
\begin{aligned}
\min_{w_-,b_-,\eta_-,\xi_+} \quad & \frac{1}{2}\|w_-\|^2 + C_3\sum_{i=p+1}^{p+q}\eta_i + C_4\sum_{j=1}^{p}\xi_j, \\
\text{s.t.} \quad & (w_-\cdot x_i) + b_- \leqslant \varepsilon+\eta_i, \\
& \qquad\qquad i=p+1,\cdots,p+q, \\
& -(w_-\cdot x_i) - b_- \leqslant \varepsilon+\eta_i, \\
& \qquad\qquad i=p+1,\cdots,p+q, \\
& (w_-\cdot x_j) + b_- \geqslant 1-\xi_j,\ j=1,\cdots,p, \\
& \eta_i \geqslant 0,\ i=p+1,\cdots,p+q, \\
& \xi_j \geqslant 0,\ j=1,\cdots,p.
\end{aligned} \tag{12}
$$

where $C_i \geqslant 0, i=1,\cdots,4$ are penalty parameters, ε is the parameter controlling the sparsity of the negative data set, $\xi_+ = (\xi_1,\cdots,\xi_p)^\top$, $\xi_- = (\xi_{p+1},\cdots,\xi_{p+q})^\top$, $\eta_+^{(*)} = (\eta_+^\top,\eta_+^{*\top})^\top = (\eta_1,\cdots,\eta_p,\eta_1^*,\cdots,\eta_p^*)^\top$, $\eta_-^{(*)} = (\eta_-^\top,\eta_-^{*\top})^\top = (\eta_{p+1},\cdots,\eta_{p+q},\eta_{p+1}^*,\cdots,\eta_{p+q}^*)^\top$ are slack variables.

Now, we discuss the primal problem (11) geometrically in Fig.1(a). First, we hope that the positive class (marked by "$*$") locates as close as possible to the hyperplane $(w_+\cdot x)+b_+=0$ (blue thick solid line), the errors are measured by $\eta_i, i=1,\cdots,p$; Second, we need to push the negative class from the hyperplane $(w_+\cdot x)+b_+=-1$ (blue thin dotted line) as far as possible, the errors are measured by $\xi_j, j=p+1,\cdots,p+q$; Third, we hope to maximize the margin between the hyperplanes $(w_+\cdot x)+b_+=-1$ and $(w_+\cdot x)+b_+=0$, which can be expressed by $\frac{1}{\|w\|}$. Obviously, we can see that the only difference between the problems (11) and (9) is that $\varepsilon=0$ is set in the problem (11). We will see later that for the positive data set, almost all the positive examples are support vectors, which means that they all make the contribution to the proximal hyperplane $(w_+\cdot x)+b_+=0$.

It is easy to get the dual problem of the primal problem (11) as follows

$$
\begin{aligned}
\min_{\tilde{\pi}} \quad & \frac{1}{2}\tilde{\pi}^\top\tilde{\Lambda}\tilde{\pi} + \tilde{\kappa}^\top\tilde{\pi}, \\
\text{s.t.} \quad & \tilde{e}^\top\tilde{\pi} = C_1 p, \\
& 0 \leqslant \tilde{\pi} \leqslant \tilde{C}.
\end{aligned} \tag{13}
$$

Where

$$
\begin{aligned}
\tilde{\pi} &= (\alpha_+^\top,\beta_-^\top)^\top, && (14)\\
\tilde{\kappa} &= (C_1 e_+^\top AA^\top, e_+^\top BA^\top + e_-^\top)^\top, && (15)\\
\tilde{e} &= (2e_+^\top, e_-^\top)^\top, && (16)\\
\tilde{C} &= (C_1 e_+^\top, C_2 e_-^\top)^\top. && (17)
\end{aligned}
$$

$$
\tilde{\Lambda} = \begin{pmatrix} 4AA^\top & 2AB^\top \\ 2BA^\top & BB^\top \end{pmatrix}. \tag{18}
$$

and $A=(x_1,\cdots,x_p)^\top \in \mathcal{R}^{p\times n}$, $B=(x_{p+1},\cdots,x_{p+q}) \in \mathcal{R}^{q\times n}$.

Now for the primal problem (12), its geometrical explaination can be referred to Fig.1(b). Since it is almost the same with the problem (10) except that the variables η_i and η_i^* are combined to be η_i, its dual problem is also easily derived and formulated as

$$
\begin{aligned}
\min_{\tilde{\pi}} \quad & \frac{1}{2}\tilde{\pi}^\top\tilde{\Lambda}\tilde{\pi} + \tilde{\kappa}^\top\tilde{\pi}, \\
\text{s.t.} \quad & \tilde{e}^\top\tilde{\pi} = 0, \\
& 0 \leqslant \tilde{r}^\top\tilde{\pi} \leqslant C_3 e_+, \\
& 0 \leqslant \tilde{\pi} \leqslant \tilde{C}.
\end{aligned} \tag{19}
$$

where

$$
\begin{aligned}
\hat{\pi} &= (\alpha_-^{*\top},\alpha_-^\top,\beta_+^\top)^\top, && (20)\\
\hat{\kappa} &= (0e_-^\top, 0e_-^\top, -e_+^\top)^\top, && (21)\\
\hat{e} &= (-e_-^\top, e_-^\top, -e_+^\top)^\top, && (22)\\
\hat{r} &= (e_-^\top, e_-^\top, 0e_+^\top)^\top, && (23)\\
\hat{C} &= (0e_-^\top, 0e_-^\top, C_4 e_+^\top)^\top, && (24)\\
& && (25)
\end{aligned}
$$

and

$$
\hat{\Lambda} = \begin{pmatrix} Q_1 & Q_2 \\ Q_2^\top & Q_3 \end{pmatrix}, Q_1 = \begin{pmatrix} BB^\top & -BB^\top \\ -BB^\top & BB^\top \end{pmatrix},
\quad Q_2 = \begin{pmatrix} BA^\top \\ -BA^\top \end{pmatrix}, Q_3 = AA^\top. \tag{26}
$$

B. Nonlinear case

In this section, we consider the nonlinear case. By introducing the kernel function $K(x, x') = (\Phi(x) \cdot \Phi(x'))$ and the corresponding transformation

$$\mathrm{x} = \Phi(x), \tag{27}$$

where $\mathrm{x} \in \mathcal{H}$, $\mathcal{H}$ is the Hilbert space, two corresponding dual problems (13) and (19) are almost the same only the the inner products, such as $AA^\top, BA^\top, AB^\top, BB^\top$, are replaced by $K(A, A^\top), K(B, A^\top), K(A, B^\top), K(B, B^\top)$.

Now we establish the NPSVM for PU as follows:

Algorithm III.1 (NPSVM for PU)

(1) For the training set (1)*, change it to be the training set* T^* (3)*;*

(2) Choose appropriate kernels $K(x, x')$*, appropriate parameters* $\varepsilon > 0$*,* $C_i > 0, i = 1, 2, 3, 4$*;*

(3) Construct and solve the two kernel QPPs (13) *and* (19) *separately;*

(4) Construct the decision functions

$$f_+(x) = \sum_{i=1}^{p}(\alpha_i^* - \alpha_i)K(x_i, x) - \sum_{j=p+1}^{p+q}\beta_j K(x_j, x) + b_+, \tag{28}$$

and

$$f_-(x) = \sum_{i=p+1}^{p+q}(\alpha_i^* - \alpha_i)K(x_i, x) + \sum_{j=1}^{p}\beta_j K(x_j, x) + b_-, \tag{29}$$

separately;

(5) For any new input x*, assign it to the class* $k (k = -, +)$ *by*

$$\arg\min_{k=-,+} |f_k(x)|. \tag{30}$$

IV. Experimental Results

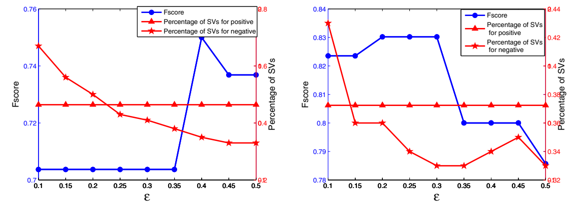
In this section, in order to validate the performance of our NPSVM for PU, we will compare it with some PU methods on two types of datasets. All methods are implemented in MATLAB 2010 on a PC with an Intel Core I5 processor and 2 GB RAM. We use the popular F-score on the positive class as the evaluation measure. F-score takes into account of both recall (r) and precision (p),

$$F = \frac{2pr}{p + r} \tag{31}$$

F-score cannot to be calculated on the validation set during the training process because there are no negative examples. An approximated method is used to evaluate the performance by

$$F = \frac{r^2}{Pr[f(x) = 1]} \tag{32}$$

where x is an input vector, $Pr[f(x) = 1]$ is the probability of this input example x is classified as positive.



(a) (b)

Fig. 4. (a) Linear case, (b) Kernel case.

A. Illustrative data

First, we apply NPSVM to the iris data set [43], which contains three classes *(Setosa, Versilcolor, Viginica)* and four attributes for an iris. Here, the two classes *(Versilcolor, Viginica)* and two features (the petal length and the petal width) are used. For this data set, we treat δ percent of *Versilcolor* as the positive examples, the rest of *Versilcolor* and *Viginica* are used as the unlabeled examples, δ ranges in $\{0.2, 0.4, 0.6, 0.8\}$, thus in this way, we created a series of PU problems. In order to deal with the imbalance, for the problem (11) with the positive hyperplane, we set $C_1/C_2 = q/p$, $C_1 = 10$; for the problem (12) with the negative hyperplane, we set $C_3 = C_4 = 10$ and ε varies in the range $\{0.1, 0.2, 0.3, 0.4\}$.

First, we give the classification performance using the linear kernel with fixed $\varepsilon = 0.1$ and varying δ (Fig.2). In order to verify our model, we also apply C-SVM to get the optimal separating hyperplane for the original problem, i.e., the balanced two classes *(Versilcolor, Viginica)*.

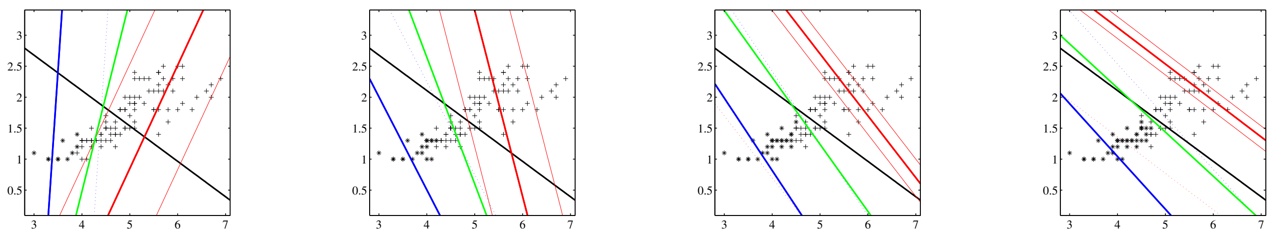
From Fig.2, we can see that for the $\varepsilon = 0.1$, the best result is the $\delta = 0.8$ (green line coincides with black line), which can be roughly understood that: $\varepsilon = 0.1$ means that the negative training points in the $\varepsilon = 0.1$ band between two negative ε-bounded lines $f_-(x) = \pm\varepsilon$ contributes nothing to the negative proximal line $f_-(x) = 0$, which are about $2*\varepsilon = 0.2$ percent of the negative set, therefore we are dealing a balanced problem with 0.8 percent positive points and 0.8 percent negative points.

Second, we give the classification performance using the linear kernel with fixed $\delta = 0.4$ and varying ε (Fig. 3). From Fig.3, we can see that for the $\delta = 0.4$, the result $\varepsilon = 0.2$ or $\varepsilon = 0.3$ are better(green line coincides with black line), which also validate that $\delta \approx 1 - 2 * \varepsilon$.

We also give the relations between the ε and F-score, between the ε and the percentage of SVs in Fig.4, in which we can found that with the increasing ε, the percentage of SVs decreases for both linear case and kernel case, and the F-score get the optimal value when $\varepsilon = 0.4$ for the linear case, and $\varepsilon = 0.2, 0.25, 0.3$ for the kernel case.

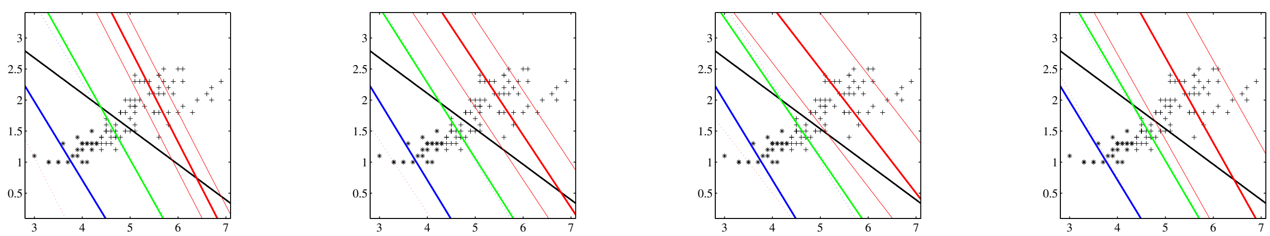
B. Text Categorization

In this section, we evaluates our method using the Reuters-21578 text collection [41]. The Reuters-21578 collection contains 21578 text documents, here only the most populous 10 out of the 135 topic categories are used, and each category is



(a) (b) (c) (d)

Fig. 2. (a) $\delta = 0.2$, (b) $\delta = 0.4$, (c) $\delta = 0.6$, (d) $\delta = 0.8$. Positive proximal line $f_+(x) = 0$ (blue thick solid line), negative proximal line $f_-(x) = 0$ (red thick solid line), two negative ε-bounded lines $f_-(x) = \pm\varepsilon$ (red thin solid lines), two margin lines $f_+(x) = -1$ (blue thin dotted line). Final decision line (green thick solid line). Optimal separating line by C-SVC (black thick solid line).



(a) (b) (c) (d)

Fig. 3. (a) $\delta = 0.1$, (b) $\delta = 0.2$, (c) $\delta = 0.3$, (d) $\delta = 0.4$. Lines are colored as Fig.2.

taken as the positive class, and the rest of the categories as the negative class.

We aim to select the positive documents from the unlabeled data set. We construct each data set for our experiment as follows: δ percent of the examples from the positive class are first selected as the positive set P. The rest of the positive examples and negative documents are used as unlabeled set U. The classifier selection criterion (32) is used to select the optimal parameters and the F-score is used to evaluate the classification performance.

We choose $\delta = 0.15$ and $\delta = 0.45$ separately, and conduct the numerical experiments on the datasets by comparing S-EM [11], PEBL [12], Roc-SVM [42]. The experiment results are given in Table I and Table II separately. The results of other methods are reported in [42]. We can see that our NPSVM performs better for most of the datasets. For the two values of δ, the mean of the F scores of NPSVM is larger than other three methods.

TABLE I
$\delta = 0.15$ AVERAGE F SCORES ON REUTERS COLLECTION

class	S-EM	PEBL	ROC-SVM	NPSVM
acq	**0.876**	0.001	0.846	0.851
corn	0.452	0	**0.804**	0.8
crude	0.82	0	0.782	**0.8571**
earn	**0.947**	0	0.858	0.833
grain	0.807	0.02	**0.845**	0.8
interest	0.648	0	0.704	**0.8**
money	0.793	0	0.768	**0.833**
ship	**0.742**	0.008	0.578	0.667
trade	0.698	0	0.759	**0.8**
wheat	0.611	0	0.834	**1**
mean	0.7394	0.0029	0.7778	**0.8241**

TABLE II
$\delta = 0.45$ AVERAGE F SCORES ON REUTERS COLLECTION

class	S-EM	PEBL	ROC-SVM	NPSVM
acq	0.891	0.891	0.905	**0.933**
corn	0.517	0.663	0.635	**0.8**
crude	0.85	0.798	0.811	**0.933**
earn	0.95	**0.956**	0.886	0.8301
grain	0.772	0.9	**0.903**	0.878
interest	0.614	0.77	0.614	**0.858**
money	0.76	0.714	0.76	**0.833**
ship	0.806	0.672	**0.829**	0.8
trade	0.693	0.728	0.728	**0.889**
wheat	0.595	0.783	0.779	**1**
mean	0.7448	0.7875	0.785	**0.875**

V. CONCLUSION

In this paper, we proposed to apply the modified NPSVM for PU learning problem, in which the positive primal problem is constructed by set $\varepsilon = 0$ such that all the positive points will be the support vectors; for the unlabeled primal problem, ε is chosen nonzero such that only a part of negative points makes the contribution to the negative proximal hyperplane. Based on this idea, we treated the unbalanced problem to be a balanced problem. Numerical experiments proved the efficiency of our method, and we also give some suggestions on choosing the parameter ε.

VI. ACKNOWLEDGMENT

This work has been partially supported by grants from National Natural Science Foundation of China (No.11271361, No.71331005), Major International (Regional) Joint Research Project (No.71110107026), and the Ministry of water resources' special funds for scientific research on public causes (No.201301094).

REFERENCES

[1] T. Joachims, F. Informatik, B. str, "A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization," ICML '97 Proceedings of the Fourteenth International Conference on Machine Learning, pp.143-151, 1996.

[2] L. M. Manevitz and M. Yousef, "One-class svms for document classification," Machine Learning Research, vol. 2, pp. 139-154, 2002.

[3] T. D. Bie, L. C. Tranchevent, L. M. M. van Oeffelen and Y. Moreau, "Kernel-based data fusion for gene prioritization," Bioinformatics, vol. 23, no. 13, pp. 125-132, 2007.

[4] P. Geurts, "Learning from positive and unlabeled examples by enforcing statistical significance," Int. Conf. Arti. Intel. Stat., vol. 15, pp. 305-314, 2011.

[5] W. K. Li, Q. H. Guo and C. Elkan, "A positive and unlabeled learning algorithm for one-class classification of remote-sensing data," IEEE Trans. Geo. Remo. Sens., vol. 49, pp. 717-725, 2011.

[6] F. Denis, R. Gilleron and M. Tommasi, "Text classification from positive and unlabeled examples," Int. Conf. Infor. Proces. Manag. Uncer., pp. 1927-1934, 2002.

[7] F. Denis, A. Laurent, R. Gilleron and M. Tommasi, "Text classification and co-training from positive and unlabeled examples," ICML, pp. 80-87, 2003.

[8] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," Conf. Comput. learning theory, pp. 92-100, 1998.

[9] K. Zhou, G. R. Xue, Q. Yang and Y. Yu, "Learning with positive and unlabeled examples using topic-sensitive PLSA," Knowledge and Data Engineering, vol. 22, no. 1, pp. 46-58, 2010.

[10] B. Liu, Y. Dai, X. Li, W. S. Lee and P. S. Yu, "Building text classifiers using positive and unlabeled examples," Int. Conf. Data Mining, pp. 179-186, 2003.

[11] B. Liu, W. S. Lee, P. S. Yu and X. L. Li, "Partially supervised classification of text documents," in Proc. Int. Conf. Mach. Learn., pp. 387-394, 2002.

[12] H. Yu, J. W. Han and K. C. C. Chang, "PEBL: Positive example based learning for web page classification using SVM,"Knowledge discovery and data mining, pp. 239-248, 2002.

[13] S. T. Li, J. T. Kwok, H. L.Zhu and Y. N. Wang, "Texture classification using the support vector machines," Pattern Recognition, vol. 36, no. 12, pp. 2883-2893, 2003.

[14] G. P. C. Fung, J. X. Yu, H. J. Lu and P. S. Yu, "Text classification without negative examples revisit," IEEE Trans. Know. Data Engineering, vol. 18, pp. 6-20, 2006.

[15] W. S. Lee and B. Liu, "Learning with positive and unlabeled examples using weighted logistic regression," Int. Conf. Mach. Learn., vol. 20. 2003.

[16] D. Zhang and W.S. Lee, "A simple probabilistic approach to learning from positive and unlabeled examples," pp. 817-854, 2005.

[17] B. Liu, "Web data mining: Exploring hyperplinks, contents, and usage data," Springer, 2006.

[18] J. Y. Tan, L. Zhen, N. Y. Deng and C. H. Zhang, "A new support vector machine for the classification of positive and unlabeled examples," Operations Research and its Applications in Engineering, Technology and Management 2013 (ISORA 2013), 11th International Symposium on, pp. 1-8, 2013.

[19] F. Mordelet and J. P. Vert, "A bagging svm to learn from positive and unlabeled examples," Pattern Recognition Letters, vol. 37, pp. 201-209, 2014.

[20] J. Qiang, "Positive and unlabeled learning research based on posterior probability," pp. 92-100, 2012.

[21] T. Ke, J. Y. Tan, B. Yang, L. J. Song, Y. Li and L. Jing, "A novel graph-based approach for transductive positive and unlabeled learning," Computational Information Systems, vol. 10, no. 4, pp. 1439-1447, 2014.

[22] O. L. Mangasarian and E. W. Wild, "Multisurface proximal support vector classification via generalized eigenvalues," IEEE Trans. Pattern Analysis and Machine Intellegence, vol. 28, no. 1, pp. 69-74, 2006.

[23] Jayadeva, R. Khemchandani and S. Chandra, "Twin support vector machines for pattern classification," IEEE Trans. Pattern Anal. Mach. Intell, vol. 29, no. 5, pp. 905-910, 2007.

[24] M. A. Kumar and M. Gopal, "Application of smoothing technique on twin support vector machines," Pattern Recognition Letters, vol. 29, no. 13, pp. 1842-1848, 2008.

[25] R. Khemchandani, Jayadeva and S. Chandra, "Optimal kernel selection in twin support vector machines," Optim. Lett., vol. 3, no.1, pp. 77-88, 2009.

[26] M. A. Kumar and M. Gopal, "Least squares twin support vector machines for pattern classification," Expert systems with applications, vol. 36, no. 4, pp. 7535-7543, 2009.

[27] Y. H. Shao, C. H. Zhang, X. B. Wang and N. Y. Deng, "Improvements on twin support vector machines," IEEE Trans. Neural Networks, vol. 22, no. 6, pp. 962- 968, 2011.

[28] Y. H. Shao and N.Y. Deng, "A novel margin-based twin support vector machine with unity norm hyperplanes," Neural Computing and Applications, vol. 22, pp. 1627-1635, 2013.

[29] Y. H. Shao and N. Y. Deng "A coordinate descent margin based-twin support vector machine for classification," IEEE Trans. Neural networks, vol. 25, pp. 114-121, 2012.

[30] X. J. Peng, "TSVR: An efficient twin support vector machine for regression," Neural Networks, vol. 23, no. 3, pp. 365-372, 2010.

[31] X.B. Chen, J.Yang, Q. L. Ye, and J. Liang, "Recursive projection twin support vector machine via within-class variance minimization," Pattern Recognition, vol. 44, pp. 2643-2655, 2011.

[32] Z. Q. Qi, Y. J. Tian and Y. Shi, "Laplacian twin support vector machine for semi-supervised classification," Neural Networks, vol. 35, pp. 46-53, 2012.

[33] Z. Q. Qi, Y. J. Tian and Y. Shi, "Twin support vector machine with universum data," Neural Networks, vol. 36, pp. 112-119, 2012.

[34] Z. Q. Qi, Y. J. Tian, and Y. Shi, "Robust twin support vector machine for pattern classification," Pattern Recognition, vol. 46, pp. 305-316, 2013.

[35] Z. Q. Qi, Y. J. Tian, and Y. Shi, "Structural twin support vector machine for classification," Knowledge-Based Systems, vol. 43, pp. 74-81, 2013.

[36] Y. J. Tian, Y. Shi, and X. H. Liu, "Recent advances on support vector machines research," Technological and Economic Development of Economy, vol. 18, pp. 5-33, 2012.

[37] Y. J. Tian, X. C. Ju, and Z. Q. Qi, "Efficient sparse nonparallel support vector machines for classification," Neural Computing and Applications, vol. 24, pp. 1089-1099, 2014.

[38] Y.J. Tian, Z. Q. Qi, X. C. Ju, Y. Shi, and X. H. Liu, "Nonparallel support vector machines for pattern classification," IEEE Trans. Cybernetics, vol.44, pp. 1067-1079, 2013.

[39] Y. J. Tian, X. C. Ju, Z. Q. Qi, and Y. Shi, "Improved twin support vector machine," Science China Mathematics, vol. 57, pp. 417-432, 2014.

[40] O. L. Mangasarian, "Nonlinear programming," Philadelphia, PA: SIAM, 1994.

[41] Reuters-21578. [online]. Available: http://www.daviddlewis.com/resources /testcollections/reuters21578/. 2007.

[42] X. L. Li and B. Liu, "Learning to classify text using positive and unlabled data," Proceeding IJCAI'03 Proceedings of the 18th international joint conference on Artificial intelligence, pp. 587-592, 2003.

[43] C. L. Blake and C. J. Merz. UCI Repositioy for machine learning database [Online]. Available: http://www.ics.uci.edu/ mlearn/MLRepository.html /testcollections/reuters21578/. 2007.