On Generalizable Low False-Positive Learning Using Asymmetric Support Vector Machines

Shan-Hung Wu, Keng-Pei Lin, Hao-Heng Chien, Chung-Min Chen, and Ming-Syan Chen, *Fellow*, *IEEE*

Abstract—The Support Vector Machines (SVMs) have been widely used for classification due to its ability to give low generalization error. In many practical applications of classification, however, the wrong prediction of a certain class is much severer than that of the other classes, making the original SVM unsatisfactory. In this paper, we propose the notion of Asymmetric Support Vector Machine (ASVM), an asymmetric extension of the SVM, for these applications. Different from the existing SVM extensions such as thresholding and parameter tuning, ASVM employs a new objective that models the imbalance between the costs of false predictions from different classes in a novel way such that user tolerance on false-positive rate can be explicitly specified. Such a new objective formulation allows us of obtaining a lower false-positive rate without much degradation of the prediction accuracy or increase in training time. Furthermore, we show that the generalization ability is preserved with the new objective. We also study the effects of the parameters in ASVM objective and address some implementation issues related to the Sequential Minimal Optimization (SMO) to cope with large-scale data. An extensive simulation is conducted and shows that ASVM is able to yield either noticeable improvement in performance or reduction in training time as compared to the previous arts.

Index Terms—Support Vector Machine, classification, low false-positive learning

1 Introduction

False-positive is a critical concern in many real-world classification problems in which the wrong prediction of a certain class is much severer than the total prediction accuracy. For instance, in spam mail filtering, users are highly sensitive the misclassification of a good mail into the spam because it may result in the loss of important messages [1], [7], [17]. In computer-aided disease diagnosis such as the early detection of cancer, wrongly predicting a malignant tumor as benign may cause the loss of a life because the success of treating the cancer depends on how early the discovery of the disease. The detection in the very first stage can lead to longer survival of the patient [13], [40]. In facial image recognition [37] and network intrusion detection [3], costly but wrong decision may follow if a false match or alarm is fired. In these applications, users are particularly sensitive to the misclassification on a certain class. Therefore, when applied to these applications, the classifier must result in very low false-positive (or negative) rate.

- *S.-H. Wu and H.-H. Chien are with the Department of Computer Science, National Tsing Hua University, No. 101, Section 2, Kuang-Fu Road, Hsinchu, Taiwan 30013, R.O.C. E-mail: shwu@cs.nthu.edu.tw, hhchien@netdb.cs.nthu.edu.tw.*
- *K.-P. Lin is with the Department of Information Management, National Sun Yat-sen University, Kaohsiung, Taiwan. E-mail: kplin@mis.nsysu.edu.tw.*
- *C.-M. Chen is with Telcordia Technologies Inc., Piscataway, NJ, and Telecordia Applied Research Center, Taipei 115, Taiwan, R.O.C. E-mail: chungmin@research.telcordia.com.*
- *M.-S. Chen is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, and the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. E-mail: mschen@cc.ee.ntu.edu.tw.*

Manuscript received 6 Apr. 2011; revised 4 Dec. 2011; accepted 17 Feb. 2012; published online 2 Mar. 2012.
Recommended for acceptance by Z.-H. Zhou.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2011-04-0181.
Digital Object Identifier no. 10.1109/TKDE.2012.46.

There have been many research works made upon different classification algorithms to reduce the false-positive rate [1], [8], [11], [16], [19], [21], [22], [24], [28], [29], [34], [39]. The SVM is a statistically robust classification algorithm which shows state-of-the-art performance [10], [15], [16], [35]. However, there are relatively few studies concerning on reducing the false-positive rate in utilizing the SVM [16], [21], [32]. Common techniques in SVM for low false-positive learning include parameter tuning [12], [21] and thresholding [16], [32]. The parameter tuning technique tackles the low false-positive classification problems by applying different misclassification costs among classes in training the SVM. This technique either incurs time-consuming exhaustive search for the appropriate parameter combination [12] or requires the domain-specific knowledge of the pattern contents (e.g., the relation between different mail categories in spam filtering [21]) since the generated classifier may not generalize well to testing data due to the heuristic nature in setting the cost parameters. The thresholding technique applies to the generated SVM classifier by establishing a larger than zero threshold on the Receiver Opeating Characteristic (ROC) curve of the testing data. The patterns must have the prediction scores higher than the threshold to be classified as positive. As the threshold moves high, there will be fewer patterns to be predicted as positive. Hence, the false-positive rate is lowered. However, the true-positive rate is also lowered at the same time. This results in an unwanted tradeoff between minimizing the false-positive rate and maximizing true-positive rate.

The traditional SVM optimizes the margin between two classes of data to achieve high classification performance such as testing accuracy or the area under the ROC curve (AUC). However, a classifier with high accuracy does not necessarily result in correct predictions to the sensitive class. For example, consider a data set which are highly imbalanced in class distribution. The prediction accuracy

will be dominated by the majority class. At an extreme, misclassifying all instances in a minority class can still result in high accuracy. Furthermore, for the classification problem which is sensitive to a particular class of wrong prediction, keeping the resultant false-positive rate under a user-specified tolerance is usually in a higher priority than obtaining high accuracy or AUC. For example, although a spam mail filter can identify most of the spam mails, users will be satisfied with it only when no (or very few) good mail is wrongly classified to spam [39]. Another example is that a hospital will tolerate only a certain number of misses in disease screening, if its insurance company pays for only a fixed number of compensation cases annually. Hence, there is a basic need for a new SVM which seeks high classification performance only when the false-positive rate meets the user tolerance.

In this paper, we propose the Asymmetric Support Vector Machine (ASVM), an asymmetric variant of the support vector learning algorithm, which incorporates the user-specified tolerance of false-positive rate into the objective formulation. The ASVM is asymmetric in the sense that the margin maximized in the algorithm is between the negative class and the *core* [6], which is a high confidence subset, of the positive class. Given a user-specified tolerance, we are able to determine a core in proper size which ensures a satisfactory false-positive rate to the user. Basically, the smaller the core is (i.e., the higher the confidence), the less chance the false-positive will occur. In reducing the false-positive rate by maximizing the core-margin (i.e., shrinking the core), the class-margin is maximized at the same time to yield good classification performance. We bridge the core-margin and the *structural risk minimization* [35] framework which provides the generalization to cover the unseen data. The false-positive rate can be controlled by the ASVM parameters, and an upper bound is derived for the generalized false-positive rate given by the ASVM. Therefore, the ASVM avoids the tradeoff between the false-positive and true-positive rates in the thresholding technique, and does not require prior or domain-specific knowledge as in the parameter tuning.

To the best of our knowledge, this is the first work which exploits the asymmetry in the objective of the SVM to control the false-positive rate. Our contributions are summarized as follows: first, we propose the notion and the formulation of the ASVM, the asymmetric support vector machine. The core-margin is maximized in the objective function in addition to the class-margin of the traditional SVM to realize the asymmetry. The ASVM respects to the structural risk minimization to ensure the generalization ability. Second, the effects of the ASVM parameters are studied in detail, and we observe their linkage to the empirical measure of the portion of outliers. Thus, the ASVM is able to incorporate the prior knowledge when the fraction of noises is known in advance or there exist low-confidence patterns in the data set. Third, we address the issues of implementing the ASVM, and propose a Sequential Minimal Optimization (SMO) [20], [26], [32] like bitraining technique to scale the ASVM to large data sets. Finally, we conduct extensive simulations on both the synthetic and real-world data sets [2], [27]. The experimental results show that the ASVM can reach about 6.4 percent improvement over the thresholding technique in AUC when a user-specified tolerance of the false-positive rate is required to meet, and in the ROC convex hull, the ASVM dominates in the low false-positive region. Compared to the parameter tuning technique, the ASVM is able to achieve a comparable performance while consuming an order less training time.

The rest of this paper is organized as follows: in Section 2, we give the preliminaries of this work by reviewing some related works and briefly explaining the rationales of the SVM. Section 3 introduces the ASVM and examines the effects of each parameter in the ASVM formulation. Then the generalization ability of the ASVM is analyzed in Section 4. In Section 5, we conduct the experiments to evaluate the performance of the ASVM, and discuss some implementation and training issues to cope with large-scale data sets. Section 6 concludes the paper.

2 Preliminaries

In this section, we briefly review related studies, and give preliminaries of SVMs. We specify some terminologies and assumptions that will be used throughout the text.

2.1 Related Works

Recent studies for cost-sensitive learning and class-imbalance learning include the techniques with utility [9], [21], cascaded classifiers [38], [39], ensemble [24], [41], boosting [11], [25], [37], compression [8], and the Bayesian classifiers [1], [28], [29]. Androutsopoulos et al. [1], Sahami et al. [28], and Schneider [29] adjusted the parameters of the probability model of Bayesian approaches to associate the positive predictions with high confidence. Studies in [9] and [21] employed the utilities, sometimes called stratifications, to change the prior of a decision tree or costs of SVM slacks. The work of [11] induced a decision tree that is able to give confidence-rated predictions by following the AdaBoost algorithm. Bratko et al. [8] derived two compression models for the positive and negative classes, respectively, and assigned the label of a pattern to the class having higher compression rate. These compression models are adaptive so the false-positive rate may be controlled. Yih et al. [39] proposed a two-stage cascaded classifier. Patterns reported as positive in the first stage are further validated in the second to reduce the false-positive rate. Lynam et al. [24] merges different classifiers (those submitted to TREC 2005 Spam Evaluation Track [14]) and combines their outputs using the log-odd average to achieve low false-positive rate.

Zhou and Liu [41] studied on applying sampling and thresholding as well as the ensemble of the two techniques in cost-sensitive neural network learning. The sampling technique causes unbalance between classes by duplicating instances of the high-cost class or decreasing the instances of the low-cost class. The thresholding which moves the decision boundary toward the low misclassifying cost side is similar to the thresholding of traditional SVMs [16], [32]. Furthermore, the neural network algorithm converges to locally optimal solutions (in back-propagation). The ASVM, on the other hand, finds the global optimum. When the Gaussian kernels are used, the resulted ASVM corresponds to an Radial Basis Function (RBF) network with Gaussian radial basis functions, and the size of the hidden layer, which is controlled manually in a neural network, can be obtained automatically during the ASVM training procedure.

There are also several cost-sensitive methods designed for the popular AdaBoost classification algorithm, mostly for the face detection problems where there are only a few face patterns but a large number of nonface patterns in an image. Viola and Jones [37] proposed an asymmetric AdaBoost algorithm in which the weight update in the iteration of AdaBoost learning procedure is modified to asymmetry by increasing the weights of face patterns and decreasing the weights of nonface patterns. Masnadi-Shirazi and Vasconcelos [25] also proposed an asymmetric AdaBoost. It is based on a statistical framework by viewing the boosting algorithm as a stage-wise fitting of additive logistic regression model, and then applying an asymmetric cost-sensitive logistic regression in the boosting. The false-positive rate in the asymmetric AdaBoost is lowered by sequentially minimizing an exponential loss function. Instead, our ASVM employs the hinge loss, which less penalizes instances with slacks, and therefore is more robust to outliers.

Wu et al. [38] tackle the face detection problem by utilizing a cascade of classifiers to stage-wise reject the nonface patterns where the classifiers in the cascade are learned by asymmetric AdaBoost. Rather than learning a single classifier with low false-positive rate, it cascades multiple base learners. Only those patterns pass all the stages will be classified as faces, so the cascade produces a lower false-positive rate. The concept of cascade in [38] is orthogonal to our work, and the ASVM can be used as a base learner in the cascade.

There is also a work considering the cost-sensitive cases for semi-supervised SVM [23]. Its goal is also orthogonal to our work since we focus on supervised learning with the SVM. The assumption of the ASVM is different that there is no massive unlabeled data available.

In the following, we briefly review the objective formulations of the SVM for the preliminary of our study.

2.2 Support Vector Machines

Given a sample $\mathbf{Z}_m = ((\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m))$ of m training instances drawn i.i.d. from $\mathcal{X} \times \{\pm 1\}$, where $\mathbf{x}_i \in \mathcal{X}$ denotes a pattern and $y_i \in \{\pm 1\}$ is a class label. Our goal for classification is to find a real value function f such that $\forall(\mathbf{x}, y) \in \mathcal{X} \times \{\pm 1\}$, $f(\mathbf{x}) \geq 0$, if $y = 1$; $f(\mathbf{x}) < 0$ otherwise. The value $f(\mathbf{x})$ is called the *decision value*.

The SVM classifier. The SVM [10], [15], [35] is a statistically robust learning method with state-of-the-art performance on classification. The SVM trains a classifier by finding an optimal separating hyperplane which maximizes the margin between two classes of data. The optimal separating hyperplane $\mathbf{w} \cdot \Phi(\mathbf{x}) + b = 0$ is found by solving the quadratic programming optimization problem in a high-dimensional Reproducing Kernel Hilbert Space (RKHS), $\mathcal{H}$, with a mapping function Φ

$$\begin{aligned} &\arg\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \\ &\text{subject to } y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, m. \end{aligned} \tag{1}$$

Minimizing $\frac{1}{2}\|\mathbf{w}\|^2$ in the objective function means maximizing the margin between two classes of data. Each slack variable ξ_i denotes the extent of $\mathbf{x}_i$ falling into the erroneous region, and $C > 0$ is the cost parameter which controls the tradeoff between maximizing the margin and minimizing the slacks. Studies [4], [33], [35] show that the large margin can actually lead to better generalization performance in prediction. The decision function is $f(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b$, and the testing instance $\mathbf{x}$ is classified by $sgn(f(\mathbf{x}))$ to determine which side of the optimal separating hyperplane it falls into.

The SVM's optimization problem is usually solved in dual form to apply the *kernel trick*

$$\arg\min_{\alpha} \frac{1}{2}\alpha^T Q\alpha - \sum_{i=1}^{m}\alpha_i \tag{2}$$

$$\text{subject to } \sum_{i=1}^{m}\alpha_i y_i = 0, 0 \leq \alpha_i \leq C, i = 1, \ldots, m, \tag{3}$$

where Q is called *kernel matrix* with $Q_{i,j} = y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$, $i = 1, \ldots, m$, $j = 1, \ldots, m$. The function $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle$ is called *kernel function*, which implicitly maps $\mathbf{x}_i$ and $\mathbf{x}_j$ into a high-dimensional feature space and computes their dot product there. By applying the kernel trick, the SVM implicitly maps data into the kernel induced high-dimensional space to find an optimal separating hyperplane which prevents the high-dimensional mapping of function Φ. Commonly used kernel functions include Gaussian Radial Basis Function kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-q\|\mathbf{x} - \mathbf{y}\|^2)$ with $q > 0$, polynomial kernel $k(\mathbf{x}, \mathbf{y}) = (q\mathbf{x} \cdot \mathbf{y} + r)^d$ with $q > 0$, and the neural network kernel $k(\mathbf{x}, \mathbf{y}) = \tanh(q\mathbf{x} \cdot \mathbf{y} + r)$, where q, r, and d are kernel parameters. The original dot product is called linear kernel $k(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{y}$. The corresponding decision function of the dual form SVM is

$$f(\mathbf{x}) = \sum_{i=1}^{m}\alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b, \tag{4}$$

where α_i, $i = 1, \ldots, m$ are called supports, which denote the weights of each instance to compose the optimal separating hyperplane in the feature space.

One-class SVM. There is another type of SVM [6], [30] that aims at distinguishing the regular patterns from outliers. Given a sample $\mathbf{X}_m = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$ of m unlabeled patterns drawn i.i.d. from $\mathcal{X}$ with distribution D, the one-class SVM searches for the smallest ball that encloses the *support* of D. When data are mapped to an RKHS, finding the smallest ball is equivalent to searching a hyperplane that approaches the data set as close as possible from the origin [30]. Let $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho : \Phi(\mathbf{x}) \in \mathcal{H}\}$, $\rho \in \mathbb{R}$, be the hyperplane, the objective of one-class SVM is formulated as follows:

$$\begin{aligned} &\arg\min_{\mathbf{w},\rho,\xi} \frac{1}{2}\|\mathbf{w}\|^2 - \rho + C\sum_{i=1}^{m}\xi_i, \\ &\text{subject to } \langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle \geq \rho - \xi_i \text{ and } \xi_i \geq 0, \end{aligned} \tag{5}$$

for all $i = 1, \ldots, m$. The above objective puts all instances $\mathbf{x}_i$ at the upper side of the hyperplane $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho \geq 0 : \Phi(\mathbf{x}) \in \mathcal{H}\}$ and let the boundary $\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho = 0$ approach the elements of $\mathbf{X}_m$ by maximizing its margin from the origin (i.e., $\rho/\|\mathbf{w}\|$). Patterns $\mathbf{x}_i$ falling outside the region $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho \geq 0 : \Phi(\mathbf{x}) \in \mathcal{H}\}$ are called outliers and have $\xi_i > 0$. The parameter C controls the tradeoff between maximizing the margin (i.e., $\rho/\|\mathbf{w}\|$) and minimizing the

training error (i.e., $\sum_{i=1}^{m}\xi_i$). Solving (5), the function $\mathrm{sgn}(\langle \mathbf{w}, \Phi(\mathbf{x}')\rangle - \rho)$ can be used to indicate whether a testing pattern $\mathbf{x}'$ belongs to the support or not.

To reduce the false-positive rate of the SVM classifier, current solutions either set a threshold [16], [32] or differentiate the cost C of the slack variables [12], [21]. In thresholding [16], [32], a testing instance $\mathbf{x}'$ may be predicted as positive only if $\langle \mathbf{w}, \Phi(\mathbf{x}')\rangle + b \geq t$, where $t > 0$ is a threshold whose value is determined from the ROC curve. Clearly, the larger the value of t, the less chance a false-positive occurs in a prediction. However, fewer true-positives can be identified. The latter approach [12], [21] associates different costs C_i to different slacks ξ_i in (1). This approach is time consuming as it requires either human interaction [21] or extra searches [12] to obtain proper values of C_i.

3 ASVM

In this section, we introduce the Asymmetric Support Vector Machine and its rationale. We also show how ASVM can incorporate the user tolerance to achieve low false-positive learning.[1]

3.1 Notation

$\mathcal{X}$	the pattern domain
$\mathbf{x}$	a pattern
y	a class label, $y \in \{\pm 1\}$
$\mathbf{Z}_m$	a sample of m training instances $((\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m))$
$\mathcal{Z}_m$	the domain of samples of size m
$\mathcal{H}$	Reproducing Kernel Hilbert Space
Φ	the feature map
k	a positive definite kernel
$\mathcal{F}$	a class of functions
f	a real value or $\{\pm 1\}$ function
$\mathcal{A}$	a class of events
A	an event
D	the distribution of $\mathcal{X} \times \{\pm 1\}$
$\lvert A\rvert$	the cardinality of a set (event) A
$\Pr\{A\}$	the probability of a set (event) A
fa	the false alarm rate $D\{(\mathbf{x}, -1) : f(\mathbf{x}) > \rho - \frac{\gamma}{2}\}$,
er	the misclassification rate $D\{(\mathbf{x}, y) : f(\mathbf{x}) \neq y\}$
$\mathbf{s}^+$ ($\mathbf{s}^-$)	positive (negative) in-bound support vectors
$\mathbf{o}^+$ ($\mathbf{o}^-$)	positive (negative) outliers
ρ	the core-margin
γ	the class-margin
ξ	the slack variable
α, β, η	Lagrange multipliers
μ, τ	ASVM parameters
q	the parameter of Gaussian RBF kernel
$\Pr^{emp}$	the empirical probability
t	the user tolerance

3.2 An Asymmetric Formulation

Recall that in traditional SVM classifier, the margin are maximized between the positive and negative classes described by the training (noisy) instances. To lower the false-positive rate, we aim at searching for a better described positive class that is able to catch a higher confidence area among the positive training patterns. Note changing the value of C in (1) to identify more outliers from the positive patterns may not lead to a better description since by definition the outliers do not reflect the low confidence points in the underlying data distribution. One naive solution is to adopt two one-class SVMs, with different values of C in (5), to estimate proper borders of the two classes and let the decision boundary sit in the middle of the two balls. However, the balls are independent of each other. This approach does not take into account the interaction (e.g., overlap, margin) between the two classes, and the accuracy of predictions is expected to be low from the statistical learning theory [35] point of view.

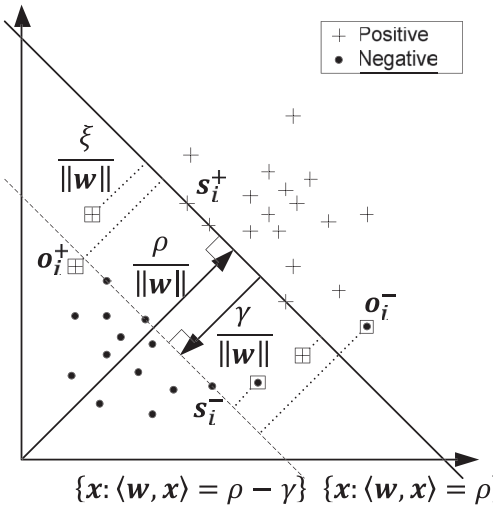

Fig. 1. A logic view of ASVM in RKHS. Two margins, the core-margin ($\rho/\|\mathbf{w}\|$) and class-margin ($\gamma/\|\mathbf{w}\|$), are maximized simultaneously to allow classifying the negative class and the *core* of the positive class.

We formulate the objective of ASVM as follows:

$$\begin{aligned} &\arg\min_{\mathbf{w},\rho,\gamma,\xi} \frac{1}{2}\|\mathbf{w}\|^2 - \rho - \frac{\mu}{\tau}\gamma + \frac{1}{\tau m}\sum_{i=1}^{m}\xi_i, \\ &\text{subject to } y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle - \rho) + \frac{1}{2}(y_i - 1)\gamma \geq -\xi_i, \\ &\xi_i \geq 0, \text{ and } \gamma \geq 0, \end{aligned} \tag{6}$$

for $i = 1, \ldots, m$, where μ and τ are constants. The concept of (6) is illustrated in Fig. 1. Note we use the shorthand $\mathbf{x}$ for $\Phi(\mathbf{x})$. Consider two parallel hyperplanes $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho : \Phi(\mathbf{x}) \in \mathcal{H}\}$ and $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho + \gamma : \Phi(\mathbf{x}) \in \mathcal{H}\}$. The above objective puts the positive patterns at the upper side of the first plane $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle \geq \rho : \Phi(\mathbf{x}) \in \mathcal{H}\}$; and the negative ones at the lower side of the second $\{\langle \mathbf{w}, \Phi(\mathbf{x})\rangle \leq \rho - \gamma : \Phi(\mathbf{x}) \in \mathcal{H}\}$. Instances falling outside their corresponding regions are called slacks and have positive penalties $\xi_i > 0$. We set $f(\mathbf{x}) = \langle \mathbf{w}, \Phi(\mathbf{x})\rangle - \rho + \frac{\gamma}{2}$, and predict the label of a testing instance $\mathbf{x}'$ by $\mathrm{sgn}(f(\mathbf{x}'))$.

ASVM maximizes two margins, the *core-margin* (i.e., $\rho/\|\mathbf{w}\|$) and the traditional class-margin (i.e., $\gamma/\|\mathbf{w}\|$) as in SVM. The rationale behind is that, by enlarging the core-margin, we are able to enclose the *core* [6] (i.e., high confidence description) of the positive class in a set $\{\Phi(\mathbf{x}) : \langle \mathbf{w}, \Phi(\mathbf{x})\rangle \geq \rho\}$. At the same time, the class-margin

1. Due to the space limitation, we focus ourselves on the two-class classification problem. The ASVM objective proposed in this paper can be easily extended to the multiclass problem.

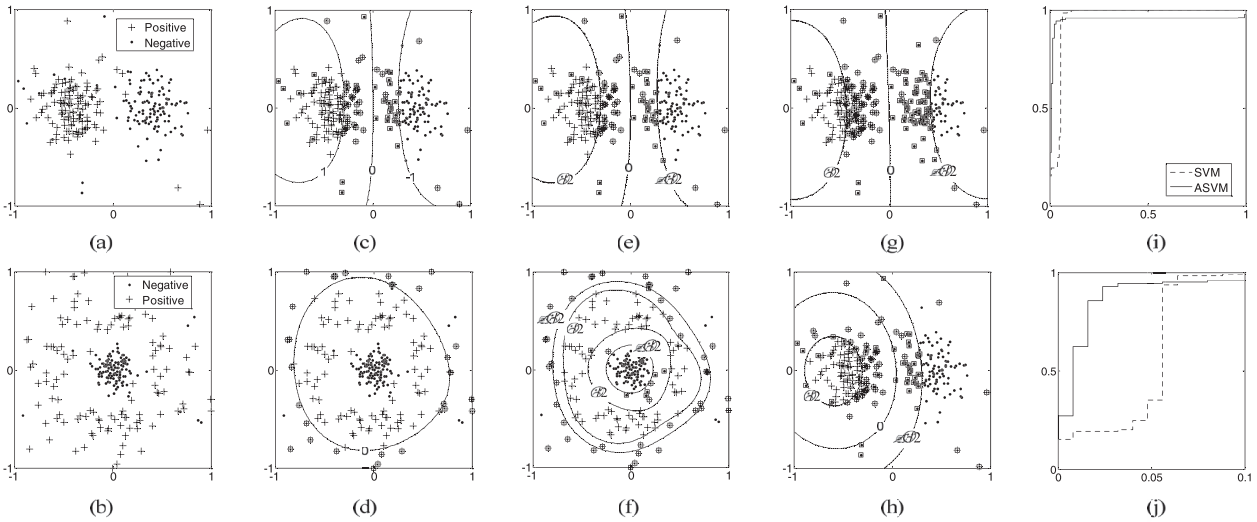

Fig. 2. Toy examples. (a) and (b) Distributions of the first and second data sets. (c) Decision boundary given by the SVM classifier. (d) Enclosing ball of the positive class returned by the one-class SVM. (e) Decision boundary given by the ASVM. (f) Enclosing balls returned by ASVM. (g) Increasing μ of ASVM results in a larger class-margin. (h) Obtaining a high confidence region of the positive class by increasing τ. (i) The ROCs achieved by the SVM output in (c) and the ASVM output in (h). (j) The areas under respective ROCs that meet a user tolerance 0.1 to the false-positive rate.

is maximized between the negative class and this core to achieve high accuracy in prediction as well as its generalization. The false-positive rate is expected to be lowered when ρ increases. Note ASVM is orthogonal to most previous studies described in Section 2, and can be readily integrated with the techniques like thresholding [16], [32], utility/cost tuning [9], [21], cascading [39], and ensemble [24].

We may transform (6) by using the Lagrangian into the following dual objective:

$$\begin{aligned} &\arg\max_{\alpha} \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j k(\mathbf{x}_i,\mathbf{x}_j) \\ &\text{subject to } \sum_{i=1}^{m}\alpha_i \geq 2\frac{\mu}{\tau}+1, \sum_{i=1}^{m}\alpha_i y_i = 1, \\ &\text{and } 0 \leq \alpha_i \leq \frac{1}{\tau m}. \end{aligned} \tag{7}$$

The details can be found in Appendix. We will discuss how to solve this problem efficiently later.

Learning under the user tolerance. Consider two toy data sets shown in Figs. 2a and 2b. Fig. 2c depicts the margin (with decision values ± 1) and the decision line $\{\Phi(\mathbf{x}) : \langle \mathbf{w}, \Phi(\mathbf{x})\rangle + b = 0\}$ returned by the SVM classifier given the cost parameters $C = 1$ and the Gaussian RBF kernel parameter $q = 0.5$. The parameters are found using the cross-validation [18]. We mark the slacks with squares. Fig. 2d depicts an enclosing ball of the positive class returned by the one-class SVM with parameters $C = 0.25$, $q = 1$. The outputs of ASVM for these two data sets are shown in Figs. 2e and 2f with parameters $\mu = 0.15$, $\tau \approx 0$, $q = 0.5$, and $\mu = 0.15$, $\tau = 0.0225$, $q = 1.5$, respectively. Comparing Figs. 2c and 2e, we can see that ASVM behaves similarly to the SVM classifier when τ is close to 0.

By increasing μ, we are able to obtain a larger margin, as depicted in Fig. 2g ($\mu = 0.3$, $\tau \approx 0$, $q = 0.5$). The effect of μ is analogous to that of C in SVM. On the other hand, as illustrated in Fig. 2h, we are able to capture the dense region of the positive classes by increasing τ ($\mu = 0.15$, $\tau = 0.05$, $q = 0.5$) since the core-margin grows as τ increases. The dense region, unlike those captured by one-class SVM, are *antagonistic* to the negative class since by (6) it aims at excluding as many negative instances as possible. We may see this clearly by comparing Figs. 2d and 2f. Note, we omit the decision line in Fig. 2f for simplicity. The captured dense region may reasonably represent the high confidence area of the positive class due to its high density, purity (in class label), and long distance to the negative class.

ASVM is useful in the situations where a user given tolerance t to the false-positive rate must be met. Fig. 2i shows two typical ROC curves resulted by the SVM and ASVM classifiers in Figs. 2c and 2h, respectively. Both SVM and ASVM achieve 95 percent accuracy in prediction. The AUC given by ASVM is 0.95, which is slightly lower than that (0.96) achieved by SVM. However, benefiting from a better description of the positive class, ASVM can significantly reduce the chance that a false-positive occurs from an instance with high decision value. Denote t-AUC the area under the ROC curve in y-axis and t in x-axis. Suppose $t = 0.1$, Fig. 2j depicts the performance of SVM and ASVM when the false-positive rate must be less then 0.1. In such a case, the 0.1-AUC given by ASVM is $0.86t$, which is about 56 percent higher than that ($0.55t$) given by SVM.

3.3 The Effects of Parameters

Although we have seen by Fig. 2 the relations between the parameters, μ and τ, and the margins, the values of these parameters are still unintuitive to users. In this section, we show that the effects of μ and τ can actually be quantified in terms of the portion of outliers.

Let m^+ (resp. m^-) be the number of the positive (resp. negative) instances in $\mathbf{Z}_m$. Denote $\mathbf{s}_i^+$ (resp. $\mathbf{s}_i^-$) the positive (resp. negative) in-bound support vectors, i.e., instances $(\mathbf{x}_i, 1)$ (resp. $(\mathbf{x}_i, -1)$) having $0 < \alpha_i < \frac{1}{\tau m}$; and $\mathbf{o}_i^+$ (resp. $\mathbf{o}_i^-$) the positive (resp. negative) outliers, i.e., instances $(\mathbf{x}_i, 1)$ (resp. $(\mathbf{x}_i, -1)$) having $\alpha_i = \frac{1}{\tau m}$, as depicted in Fig. 1. Let $\Pr^{emp}(\mathbf{s}_i^+) = \frac{1}{m}|\{\mathbf{s}_i^+\}|$ (resp. $\Pr^{emp}(\mathbf{s}_i^-)$) and $\Pr^{emp}(\mathbf{o}_i^+) = \frac{1}{m}|\{\mathbf{o}_i^+\}|$ (resp. $\Pr^{emp}(\mathbf{o}_i^-)$) be the portions of the positive

(resp. negative) in-bound support vectors and the outliers among $\mathbf{Z}_m$, respectively.

Theorem 1. *Assume* $\rho > 0$ *and* $\gamma > 0$, *then* $\Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-)$ *is upper bounded by* $\tau + \Pr^{emp}(\mathbf{s}_i^-)$.

Proof. At KKT complementarity conditions, $\gamma > 0$ implies $\eta = 0$ (see Appendix). Therefore, the term $\sum_{i=1}^{m} \alpha_i \geq 2\frac{\mu}{\tau} + 1$ in (7) becomes an equation. We have

$$\begin{cases} \sum_{i=1}^{m^+} \alpha_i + \sum_{i=1}^{m^-} \alpha_i = 2\frac{\mu}{\tau} + 1, \\ \sum_{i=1}^{m^+} \alpha_i - \sum_{i=1}^{m^-} \alpha_i = 1. \end{cases}$$

Summing the above two equations, we have $\sum_{i=1}^{m^+} \alpha_i = \frac{\mu}{\tau} + 1$, $0 \leq \alpha_i \leq \frac{1}{\tau m}$. There exist at most $(\frac{\mu}{\tau} + 1)(\frac{1}{\tau m})$ positive instances that have $\alpha_i = \frac{1}{\tau m}$. Since the outliers have $\alpha_i = \frac{1}{\tau m}$, we obtain

$$\Pr^{emp}(\mathbf{o}_i^+) \leq \frac{(\mu + \tau)m}{m} = \mu + \tau. \tag{8}$$

Now subtract the above two equations. We have $\sum_{i=1}^{m^-} \alpha_i = \frac{\mu}{\tau}$, $0 \leq \alpha_i \leq \frac{1}{\tau m}$. Since each α_i can contribute at most $\frac{1}{\tau m}$, there exist at least $(\frac{\mu}{\tau})(\frac{1}{\tau m}) = \mu m$ negative instances that have $\alpha_i \geq 0$. This implies that $\Pr^{emp}(\mathbf{s}_i^-) + \Pr^{emp}(\mathbf{o}_i^-) \geq \frac{\mu m}{m} = \mu$, and therefore

$$\Pr^{emp}(\mathbf{o}_i^-) \geq \mu - \Pr^{emp}(\mathbf{s}_i^-). \tag{9}$$

Combining (8) and (9), we obtain

$$\begin{aligned} \Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-) &\leq (\mu + \tau) - (\mu - \Pr^{emp}(\mathbf{s}_i^-)) \\ &= \tau + \Pr^{emp}(\mathbf{s}_i^-). \end{aligned}$$

□

Theorem 2. *Assume* $\rho > 0$ *and* $\gamma > 0$, *then* $\Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-)$ *is lower bounded by* $\tau - \Pr^{emp}(\mathbf{s}_i^+)$.

Proof. Consider $\sum_{i=1}^{m^+} \alpha_i = \frac{\mu}{\tau} + 1$, $0 \leq \alpha_i \leq \frac{1}{\nu m}$. Since each α_i can contribute at most $\frac{1}{\tau m}$, there exist at least $(\frac{\mu}{\tau} + 1)(\frac{1}{\tau m}) = (\mu + \tau)m$ positive instances that have $\alpha_i \geq 0$. Hence, we obtain $\Pr^{emp}(\mathbf{s}_i^+) + \Pr^{emp}(\mathbf{o}_i^+) \geq \frac{(\mu+\tau)m}{m} = \mu + \tau$; that is,

$$\Pr^{emp}(\mathbf{o}_i^+) \geq \mu + \tau - \Pr^{emp}(\mathbf{s}_i^+). \tag{10}$$

Now consider $\sum_{i=1}^{m^-} \alpha_i = \frac{\mu}{\tau}$, $0 \leq \alpha_i \leq \frac{1}{\tau m}$. There exist at most $(\frac{\mu}{\tau})(\frac{1}{\tau m}) = \mu m$ negative instances that have $\alpha_i = \frac{1}{\tau m}$. We have

$$\Pr^{emp}(\mathbf{o}_i^-) \leq \frac{\mu m}{m} = \mu. \tag{11}$$

Combining (10) and (11), we obtain

$$\begin{aligned} \Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-) &\geq (\mu + \tau - \Pr^{emp}(\mathbf{s}_i^+)) - \mu \\ &= \tau - \Pr^{emp}(\mathbf{s}_i^+). \end{aligned}$$

□

Theorem 3. *Assume* $\rho > 0$ *and* $\gamma > 0$. *Suppose the instances in* $\mathbf{Z}_m$ *are generated i.i.d. from a distribution* D *that is continuous with respect to* $\mathbf{x}$. *Suppose, moreover, the kernel is analytic and nonconstant. The difference* $\Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-)$ *converges almost surely to* τ, *i.e.*,

$$\Pr(\lim_{m \to \infty} (\Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-)) = \tau) = 1.$$

Proof. With Theorems 1 and 2, this can be proofed intuitively by claiming that, when $m \to \infty$, both $\Pr^{emp}(\mathbf{s}_i^+) \to 0$ and $\Pr^{emp}(\mathbf{s}_i^-) \to 0$ [31]. □

We can see that the parameter τ controls the difference between the outliers from the positive and negative classes. As a byproduct, we can see from (8) and (10) that

$$(\mu + \tau) - \Pr^{emp}(\mathbf{s}_i^+) \leq \Pr^{emp}(\mathbf{o}_i^+) \leq (\mu + \tau), \tag{12}$$

and from (9) and (11) that

$$\mu - \Pr^{emp}(\mathbf{s}_i^-) \leq \Pr^{emp}(\mathbf{o}_i^-) \leq \mu. \tag{13}$$

The parameter μ controls the basic portion of the outliers from each class. Note the effect of μ in ASVM is similar to that of the parameter ν in ν-SVM classifier [31]. Using the above conclusions, ASVM may incorporate with the prior knowledge (in portion of the outliers) to obtain a more sophisticated and high confidence area.

4 Generalization Performance

As shown in Section 3, we can lower the false-positive rate of ASVM by increasing τ (and therefore the core-margin). However, the false-positive rate is estimated *empirically* based on testing data and may not generalize well to the other data sets. In this section, we give a Probably Approximately Correct (PAC) analysis of the ASVM generalization performance[2] and show that given an unseen data drawn from the same distribution with $\mathbf{Z}_m$, the false-positive rate can indeed be lowered by increasing τ.

Consider two samples $\mathbf{Z}_m = ((\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m))$ and $\widetilde{\mathbf{Z}}_m = ((\widetilde{\mathbf{x}}_1, \widetilde{y}_1), (\widetilde{\mathbf{x}}_2, \widetilde{y}_2), \ldots, (\widetilde{\mathbf{x}}_m, \widetilde{y}_m))$, where the respective instances $(\mathbf{x}_i, y_i)$ and $(\widetilde{\mathbf{x}}_i, \widetilde{y}_i)$ are drawn i.i.d. from $\mathcal{X} \times \{\pm 1\}$ with distribution D. Define the true (i.e., expected) false-positive rate

$$\mathrm{fa}(f) = D\Big\{(\mathbf{x}, -1) : f(\mathbf{x}) > \rho - \frac{\gamma}{2}\Big\},$$

where $\mathbf{x} \in \mathcal{X}$ and $0 < \gamma < \rho$. We first generalize a standard *symmetrization* technique [31], [36].

Lemma 4. *Consider two classes of events* $\mathcal{A}$ *and* $\mathcal{A}'$ *measurable with respect to* D. *Given* $\mathbf{Z}_m$ *and* $\widetilde{\mathbf{Z}}_m$, *define* $\Pr_A^{emp}(\mathbf{Z}_m) = \frac{1}{m}|\mathbf{Z}_m \cap A|$. *Let* $\varphi = \sup_{A \in \mathcal{A}, A' \in \mathcal{A}'} |\Pr(A) - \Pr_{A'}^{emp}(\mathbf{Z}_m)|$ *and* $\psi = \sup_{A \in \mathcal{A}, A' \in \mathcal{A}'} |\Pr_A^{emp}(\widetilde{\mathbf{Z}}_m) - \Pr_{A'}^{emp}(\mathbf{Z}_m)|$. *For* $\epsilon^2 m \geq 2$,

$$D^m\{\mathbf{Z}_m : \varphi > \epsilon\} \leq 2D^{2m}\Big\{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m : \psi > \frac{\epsilon}{2}\Big\}, \tag{14}$$

$\mathbf{Z}_m\widetilde{\mathbf{Z}}_m = ((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m), (\widetilde{\mathbf{x}}_1, \widetilde{y}_1), \ldots, (\widetilde{\mathbf{x}}_m, \widetilde{y}_m))$.

Proof. The proof can be found in Appendix. □

Theorem 5. *Suppose* $\gamma > 0$. *With probability at least* $1 - \delta$, *every* $f \in \mathcal{F}$ *has*

2. Note ASVM is a large margin classifier, so previous analyses [4], [33], [35] can be easily adapted to obtain the generalization performance of ASVM in classification. Here, we concentrate on the false-positive analysis.

$$fa(f) < \Pr^{emp}(\mathbf{o}_i^-) + \sqrt{\frac{2}{m}\left(\frac{1024}{\gamma^2}\ln\left(\frac{17em\gamma^2}{512}\right)\log_2(578m) + \ln\frac{4}{\delta}\right)}. \tag{15}$$

Proof. We bound the target probability $D^m\{\mathbf{Z}_m : \exists f \in \mathcal{F}, \text{fa}(f) > \Pr^{emp}(\mathbf{o}_i^-) + \epsilon\}$.

Step 1: Define a threshold function

$$\pi(x) = \begin{cases} \rho - \frac{\gamma}{2}, & \text{if } x \geq \rho - \frac{\gamma}{2}, \\ \rho - \gamma, & \text{if } x \leq \rho - \gamma, \\ x, & \text{otherwise.} \end{cases}$$

Denote $\pi(\mathcal{F}) = \{\pi \circ f : f \in \mathcal{F}\}$. It is clear that $\text{fa}(f) \leq D\{(\mathbf{x}, y) : y = -1 \text{ and } \pi(f(\mathbf{x})) \geq \rho - \gamma/2\}$. Also, $f(\mathbf{x}_i) > \rho - \gamma$ if and only if $\pi(f(\mathbf{x}_i)) \neq \rho - \gamma$. We may rewrite the target probability as

$$D^m\{\mathbf{Z}_m : \exists f \in \mathcal{F}, D\{(\mathbf{x}, -1) : \pi(f(\mathbf{x})) \geq \rho - \gamma/2\} > \frac{1}{m}|\{(\mathbf{x}_i, -1) : \pi(f(\mathbf{x}_i)) \neq \rho - \gamma\}| + \epsilon\Big\}.$$

By Lemma 4, the probability above is no more than

$$2D^{2m}\Big\{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m : \exists f \in \mathcal{F}, \frac{1}{m}|\{(\widetilde{\mathbf{x}}_i, -1) : \pi(f(\widetilde{\mathbf{x}}_i)) \geq \rho - \gamma/2\}| > \frac{1}{m}|\{(\mathbf{x}_i, -1) : \pi(f(\mathbf{x}_i)) \neq \rho - \gamma\}| + \epsilon\Big\}, \tag{16}$$

as long as $\epsilon^2 m \geq 2$.

Step 2: Consider a $\frac{\gamma}{4}$-cover of $\pi(\mathcal{F})$, $\mathcal{G}$ (with respect to the pseudometric $l_\infty^{\mathbf{X}_m\widetilde{\mathbf{X}}_m}$). For every $\pi \circ f \in \pi(\mathcal{F})$, there exists $g \in \mathcal{G}$ such that $|g(\mathbf{x}_j) - \pi(f(\mathbf{x}_j))| < \gamma/4$, $\forall(\mathbf{x}_j, -1) \in \mathbf{Z}_m\widetilde{\mathbf{Z}}_m$. Note

$$\pi(f(\widetilde{\mathbf{x}}_i)) \geq \rho - \gamma/2$$

implies $g(\widetilde{\mathbf{x}}_i) \geq \rho - 3\gamma/4$, i.e.,

$$\{(\widetilde{\mathbf{x}}_i, -1) : \pi(f(\widetilde{\mathbf{x}}_i)) \geq \rho - \gamma/2\} \subseteq \{(\widetilde{\mathbf{x}}_i, -1) : g(\widetilde{\mathbf{x}}_i) \geq \rho - 3\gamma/4\}.$$

On the other hand, $g(\mathbf{x}_i) \geq \rho - 3\gamma/4$ implies $\pi(f(\mathbf{x}_i)) \neq \rho - \gamma$, i.e.,

$$\{(\mathbf{x}_i, -1) : g(\mathbf{x}_i) \geq \rho - 3\gamma/4\} \subseteq \{(\mathbf{x}_i, -1) : \pi(f(\mathbf{x}_i)) \neq \rho - \gamma\}.$$

Hence, (16) is no more than

$$2D^{2m}\Big\{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m : \exists g \in \mathcal{G}, \frac{1}{m}|\{(\widetilde{\mathbf{x}}_i, -1) : g(\widetilde{\mathbf{x}}_i) \geq \rho - 3\gamma/4\}| > \frac{1}{m}|\{(\mathbf{x}_i, -1) : g(\mathbf{x}_i) \geq \rho - 3\gamma/4\}| + \epsilon\Big\}. \tag{17}$$

Step 3: Next, consider a class of permutations Λ over $\{1, 2, \ldots, 2m\}$ in which each permutation σ randomly swaps the corresponding elements of the first and second half, i.e., $\{\sigma(i), \sigma(m+i)\} = \{i, m+i\}$, $\forall 1 \leq i \leq m$. Since a sample $\mathbf{Z}_m\widetilde{\mathbf{Z}}_m$ is drawn from a product probability measure D^{2m}, (17) will not be affected if we apply a permutation σ to the element index of $\mathbf{Z}_m\widetilde{\mathbf{Z}}_m$. Suppose σ is selected from Λ by following an uniform distribution U, we may rewrite (17) as

$$2D^{2m}\Big\{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m : \exists g \in \mathcal{G}, U\Big\{\sigma : \frac{1}{m}\sum_{i=1}^{m}(\theta_{\sigma(m+i)} - \theta_{\sigma(i)}) > \epsilon\Big\}\Big\}, \tag{18}$$

where $\theta_j \in \{0, 1\}$, $\theta_j = 1$ if and only if $g(\mathbf{x}_j) \geq \rho - \frac{3\gamma}{4}$, $\forall(\mathbf{x}_j, -1) \in \mathbf{Z}_m\widetilde{\mathbf{Z}}_m$. By the union bound, (18) is no more than

$$\begin{aligned} &2 \sup_{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m} |\mathcal{G}| \sup_{g \in \mathcal{G}} U\Big\{\sigma : \frac{1}{m}\sum_{i=1}^{m}(\theta_{\sigma(m+i)} - \theta_{\sigma(i)}) > \epsilon\Big\} \\ &= 2 \sup_{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m} |\mathcal{G}| \sup_{g \in \mathcal{G}} U\Big\{\sigma : \frac{1}{m}\sum_{i=1}^{m}(\theta_{m+i} - \theta_i)u_i > \epsilon\Big\} \\ &\leq 2|\mathcal{G}| \sup_{\mathbf{a},\mathbf{b}} U\Big\{\sigma : \frac{1}{m}\sum_{i=1}^{m}(a_i - b_i)u_i > \epsilon\Big\}, \end{aligned}$$

where u_i are chosen independently and uniformly from $\{\pm 1\}$. By Hoeffding's inequality, the above probability is no more than

$$2|\mathcal{G}|\exp(-\epsilon^2 m/2) \leq 2\mathcal{N}(\gamma/4, \pi(\mathcal{F}), 2m)\exp(-\epsilon^2 m/2).$$

Setting this to δ and solving ϵ gives

$$\epsilon = \sqrt{\frac{2}{m}\ln\frac{2\mathcal{N}(\gamma/4, \pi(\mathcal{F}), 2m)}{\delta}}.$$

Step 4: The study [5] shows that $\log_2\mathcal{N}(\gamma/4, \pi(\mathcal{F}), 2m) < 1 + d\log_2(34em/d)\log_2(578m)$ provided

$$m \geq 1 + d\ln(34em/d),$$

where $d = fat_{\mathcal{F}}(\gamma/32)$. In addition, study [4] shows that $d \leq 1024/\gamma^2$ if the RBF kernel is used. Applying ϵ and these terms to the target probability, we obtain the proof. □

Back to Fig. 1, consider a fixed μ and we increase τ. Since $\Pr^{emp}(\mathbf{o}_i^-)$ does not grow with τ by (13), the first term at the right hand side of (15) remains the same (or close to its original value). On the other hand, $\Pr^{emp}(\mathbf{o}_i^+)$ increases proportionally to τ by (12). The core-margin extends directly. Since $\Pr^{emp}(\mathbf{o}_i^-)$ is fixed, this effectively broadens the class-margin toward the positive side, and the value of γ is increased meanwhile. Notice that the square root at the right-hand side of (15) decreases when γ grows. Therefore, we may obtain a lowered true false positive rate by setting a higher τ.

5 Performance Evaluation

In this section, we evaluate the performance of ASVM. We also study the scalability of ASVM and discuss some implementation issues to cope with large-scale data.

5.1 Metrics and Settings

We implement ASVM based on LIBSVM [12]. To evaluate the performance of ASVM, we consider several public real-world data sets obtained from the UCI machine learning repository [2] and IJCNN 2001 competition [27]. We control a 1:9 ratio between the positive and negative instances by either resampling (for two-class data sets) or merging the class labels (for multiclass data sets) [37]. Users under such a ratio are sensitive to the false-positives since any increment in the false-positive rate may seriously affect the positive predictions. In each data set the training and testing instances are split according to a 5:1 ratio. We use tenfold cross validation in each training process.

This paper focuses on low false-positive learning. In particular, we are interested in the performance of a classifier *provided that a user tolerance* t, $0 \leq t \leq 1$, *to the false-positive rate must be met*. We focus on the t-ROC space, i.e., an ROC space with the axis of false-positive rate ranging from 0 to t. We use the following metrics in our performance evaluation:

- **Slopes in t-ROC space.** This metrics is useful to investigate the tradeoff between different classifiers when the isoperformance line varies.
- **t-AUC.** This metrics demonstrates the *discriminability* of a classifier in t-ROC space. We let each classifier maximize this metrics in training time.

We compare ASVM with the ThresHolding (TH) [16], [32] and the Parameter Tuning (PT) [12], [21] techniques, which are both available in LIBSVM by default. Note that since we focus on a general purpose classifier, no prior knowledge, such as that used in [21], is assumed. In thresholding, the standard SVM classifier is used and has two parameters, C and q, as we have seen in Section 2.2, which need to be determined during the training time. We adopt a 2D grid search [18] for the optimal combination of these two parameters that maximizes t-AUC. In parameter tuning, we differentiate the parameter C of a standard SVM between the positive (C^+) and negative (C^-) classes, and employ a 3D grid search for the optimal combination of C^+, C^-, and q maximizing t-AUC. In ASVM, there are three parameters, μ, τ, and q, as we have seen in Section 2.2 and Section 3. Rather than adopting a 3D grid search directly, we first fix a very small τ (to simulate the conventional SVM classifier) and apply a 2D grid search for the optimal combination of μ and q that maximizes t-AUC. After proper μ and q are obtained, we perform a linear search (i.e., 1D grid search) for τ maximizing the t-AUC further. In addition to the SVM variants, we also compare ASVM with the Asymmetric Boosting (AB) [38], as it has a large margin interpretation which is similar to that employed in the objective of SVMs. In the Asymmetric Boosting, we use decision stumps as the weak learners and allow the algorithm to run for 50 iterations. There are two parameters, C_1 and C_2, which control the cost of false negatives and positives respectively. We set $C_2 = 1$ and adopt a 1D grid search for the optimal value of C_1 maximizing t-AUC.

5.2 SMO Implementation

For better scalability, we reduce the ASVM dual to the Sequential Minimal Optimization [26] problem. In order to match the SMO input, we need to rewrite the constraint $\sum_{i=1}^{m} \alpha_i \geq 2\frac{\mu}{\tau} + 1$ in (7) as $\sum_{i=1}^{m} \alpha_i = 2\frac{\mu}{\tau} + 1$. Doing this can effectively relax the constraint $\gamma \geq 0$ in the ASVM primal (see (6)) and therefore a special care is needed when selecting μ in the training time to prevent a negative class-margin γ. One easy way is to check whether $\gamma < 0$ during each iteration of a grid search and skip the corresponding candidates. Another way is to train an auxiliary hyperplane with γ always equal to 0 in (6) first during each iteration of the grid search. We are able to estimate the *basic portion of zero* by calculating the portions of the negative instances falling across the auxiliary hyperplane. Following (12) and (13), we can see that $\gamma \geq 0$ as long as

$$\mu \geq basic\ portion\ of\ zero.$$

This approach, called *bitraining*, is particularly useful to those cases, such as online training, where the grid-search technique is infeasible. In this paper, we adopt the former approach for the sake of simplicity. Fig. 3 shows the scalability of ASVM. Currently, we are able to handle about 20,000 instances within a minute.

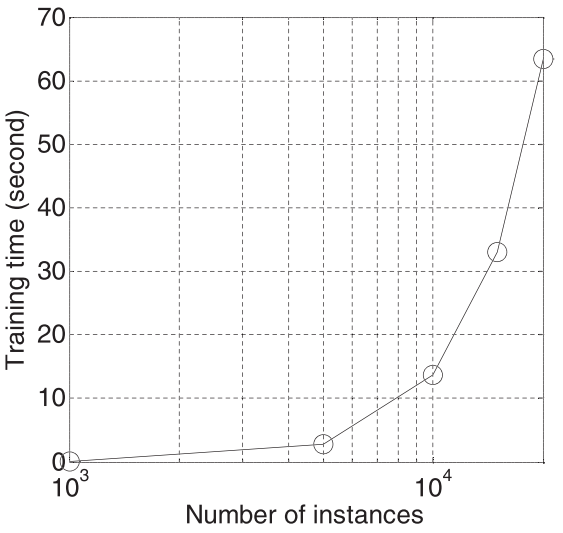

Fig. 3. The scalability of ASVM based on the SMO implementation.

5.3 Comparison with Thresholding

In this section, we compare the testing results of ASVM with those of ThresHolding. TH is based on traditional SVM classifier. As mentioned in Section 3, ASVM is also compatible to this technique and therefore we consider setting up different thresholds for ASVM's positive predictions as well. The resultant performance of both the classifiers can be easily arranged and shown in an ROC space, where each point on an ROC curve presents a tradeoff between the true- and false-positive rates given a certain threshold (not necessarily larger than 0 in this case).

We use data sets including Pima Indian Diabetes, Statlog German, Wisconsin Breast Cancer, Ionosphere, Statlog Australian, Covertype, and IJCNN in our experiments. We consider $t = 1$ and 0.1 for each data set in the training phase. For larger data sets such as Covertype and IJCNN, we consider $t = 0.05$ additionally since under such a configuration the training instances are still sufficient to apply the learned model to the testing data. Note that since the ratio between the positive and negative instances is 1:9, we differentiate the parameter C in TH between the positive

TABLE 1
Performance Comparison between the ThresHolding and ASVM in Terms of t-AUC, Where $t = 1$, 0.1, and 0.05 Are Given in Training Time

	ThresHolding (TH)			ASVM			Improvement		
Training target	1-AUC	0.1-AUC	0.05-AUC	1-AUC	0.1-AUC	0.05-AUC	%	%	%
Diabetes	**0.828618**	0.031508	–	0.828550	**0.040713**	–	-8.2e-5	29.2	–
Statlog German	**0.767854**	0.019167	–	0.763777	**0.019292**	–	-0.5	0.7	–
Breast Cancer	0.995638	0.095638	–	**0.995895**	**0.095895**	–	2.6e-2	0.3	–
Ionosphere	0.987302	0.087302	–	**0.996825**	**0.096825**	–	1.0	10.9	–
Australian	**0.948124**	**0.066397**	–	0.925197	0.065787	–	-2.4	-0.9	–
Covertype	**0.982942**	**0.083836**	0.0345622	0.982186	0.083080	**0.0348064**	-7.7e-2	-0.9	0.7
IJCNN	0.959185	0.078075	0.0349608	**0.976187**	**0.083115**	**0.0387323**	1.8	6.5	10.8

(C^+) and negative (C^-) classes and set C^+:$C^- = 9$:1 to compensate for the skew data distribution.[3]

Table 1 shows the maximal t-AUCs achieved by TH and ASVM, respectively. As we can see, for Diabetes the 1-AUCs given by TH and ASVM are very close to each other. By comparing the 1-AUCs of the rest data sets, we can see that, generally, ASVM gives similar performance as SVM in classification. When focusing on 0.1-AUCs, however, we observe that ASVM is able to give 33 percent improvement over TH. The other data sets based on which ASVM can make noticeable improvement include Ionosphere (10.9 percent for 0.1-AUC) and IJCNN (5.1 percent for 0.1-AUC, 3.8 percent for 0.05-AUC). We believe this is mainly because that ASVM successfully obtain a high confidence area of the positive class in these data sets. Overall, ASVM gives about 6.4 percent improvement in t-AUC when $t \leq 0.1$.

In the Ionosphere data set, the targets of the radar data are free electrons in the ionosphere. The label indicates if the signal shows evidence of some type of structure in the ionosphere. The ASVM seems to be able to find the dense region of the specific structure. The Pima Indian diabetes data set are medical records of female Pima Indian heritage, which are used to learn classifiers to predict if a patient is subject to diabetes. Although this data set is highly overlapped, some of the positive cases have apparent characteristics hence the ASVM can find an apparent core region. In Statlog German and Australian, people are labeled in terms of good or bad credit risks according to their credit data. Most attributes are indicator variables. It seems that these indicator variables prevent instances from forming a core region tightly, because the weight of each attribute is missing during data normalization. The breast cancer data set, which contains clinical cases of breast cancer detection, is gathered for predicting whether the organization is benign or malignant. Since the TH is already able to separate it very well, the ability of the ASVM in finding the core seems to do little help.

Notice that in the Statlog Australian data set, the advantage of ASVM does not help a better performance. We believe this is because that the classes are separable in RKHS. Under such a case, SVM is good enough to make low false-positive predictions.

Next, we study the detailed performance of ASVM and TH within the 0.1- and 0.05-ROC space. Our observation shows that ASVM is usually the best classifier at the very first segment of the false-positive rate (starting from 0). This is true even for the Covertype data set, despite the fact that ASVM does not achieve the highest 0.1-AUC in Table 1. Fig. 4a illustrates the ROC curves returned by ASVM and TH using $t = 0.1$ in training time. As we can see, ASVM is the best classifier when the false-positive rate ranges from 0 to 0.019 and gives the sharpest range of slope, $[15.129, \infty]$, along the ROC Convex Hull. The true-positive rate is 0.774 at the point of false-positive rate 0.019. Fig. 4b illustrates the ROC curves when $t = 0.05$ is used. Again, ASVM is the best classifier when the false-positive rate is above 0 and under 0.002. It also gives the sharpest slopes ranging from 32.780 to ∞ along the ROC Convex Hull. The true-positive rate is 0.387 at the point of false-positive rate 0.002. ASVM is useful in the situations that the cost of the false-positives is high (or, the slope of the isoperformance line is sharp).

5.4 Comparison with Parameter Tuning

In this section, we compare the testing results of ASVM with those of Parameter Tuning. Although both PT and ASVM

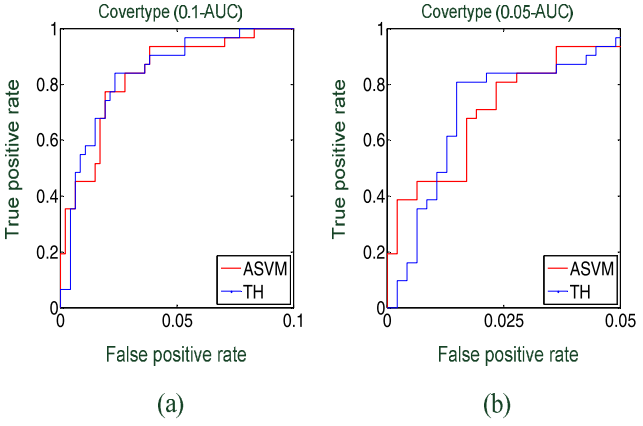

Fig. 4. The ROC curves of TH and ASVM given $t = 0.1$ and 0.05 in training time.

3. This is suggested in LIBSVM [12].

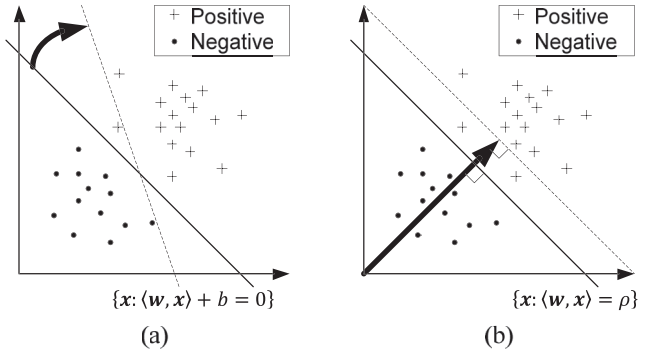

Fig. 5. Decision planes in RKHS. (a) In PT, the movement of a decision plane is unpredictable when the values of C^+ and C^- are changed. (b) In ASVM, changing the value of τ effectively shifts the decision boundary toward the positive class.

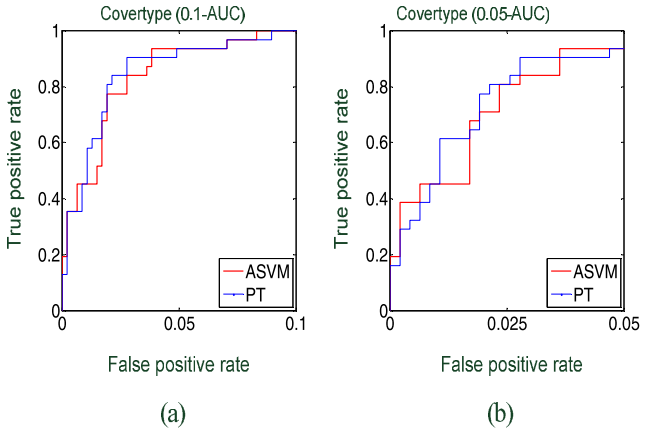

Fig. 6. The ROC curves of PT and ASVM given $t = 0.1$ and 0.05 in training time.

have three parameters (C^+, C^-, q and μ, τ, q, respectively), they are trained in different way. In PT, the effects of C^+ and C^- are correlated. Changing any value of C^+, C^-, and q may result in movement of a decision boundary as well as its margin, as shown in Fig. 5a. Under such a case, we need to search the entire 3D space for the best combination of C^+, C^-, and q. In ASVM, on the other hand, we can see from Fig. 5b that given μ and q, increasing the value of τ effectively shifts the decision boundary toward the positive class. The class margin is enlarged, but its placement, which is determined by μ and q, is not affected by τ. Based on this observation, we adopt a heuristic training method aiming at reducing the training times of a 3D grid search. As mentioned before, we first apply a 2D grid search for τ and q to determine a proper placement of the decision boundary when $\tau \approx 0$, and then increase τ to obtain a high confidence area of the positive class.

The maximal t-AUCs achieved by PT and ASVM are summarized in Table 2. As we can see, the difference between the results of ASVM and PT is not significant, ranging between $\pm 3\%$.

To see the detailed performance of ASVM and PT within the 0.1- and 0.05-ROC space, let's consider again the Covertype data set. Fig. 6a illustrates the ROC curves returned by ASVM and PT using $t = 0.1$ in training time. As we can see, ASVM is the best classifier when the false-positive rate ranges from 0 to 0.002 and gives the sharpest range of slope, $[26.476, \infty]$, along the ROC Convex Hull. The true-positive rate is 0.355 at the point of false-positive rate 0.002. Fig. 6b illustrates the ROC curves when $t = 0.05$ is used. In this case, ASVM remains the best in the range $[0, 0.002]$ of the false-positive rate. It also gives the sharpest range of slope $[26.476, \infty]$ along the ROC Convex Hull. The true-positive rate is 0.387 at the point of false-positive rate 0.002. Generally, ASVM is able to give comparable performance against PT in terms of either t-AUC, $t \leq 0.1$, or slopes.

Next, we compare the number of training times required in the grid searches adopted by ASVM and PT, respectively. The results are depicted in Fig. 7 whose x-axis denotes the granularity, i.e., the number that a search range in each dimension is divided into. As we can see, ASVM requires an order less training times than PT. This is because we perform only a 2D search (for μ and q) with one extra linear search (for τ) rather than a 3D search as PT does. From the above discussions, ASVM is able to give comparable performance as compared with PT while significantly reducing the total training times.

5.5 Comparison with Asymmetric Boosting

Table 3 shows the maximal t-AUCs achieved by the Asymmetric Boosting and ASVM, respectively. As we can see, ASVM outperforms AB in most data sets across different values of t. In average, ASVM is able to give 1.5, 7.0, and 21.9 percent improvement when $t = 1$, 0.1, and 0.05, respectively. We believe this is because that ASVM minimizes the hinge loss function as in traditional SVMs, which penalizes instances with positive slacks less than the exponential loss function used by AB does. Therefore, ASVM is more robust to the noises and outliers. We also notice that AB achieves better performance given the Covertype data set when $t = 0.1$ and 0.5. In such a case, AB outperforms all SVM-based techniques, including TH and PT. This may be due to the fact that the margin of AB can increase during the iterations in the training process even

TABLE 2
Performance Comparison between the Parameter Tuning and ASVM in Terms of t-AUC, Where $t = 1$, 0.1, and 0.05 Are Given in Training Time

	Param. Tuning (PT)			ASVM			Improvement		
Training target	1-AUC	0.1-AUC	0.05-AUC	1-AUC	0.1-AUC	0.05-AUC	%	%	%
Covertype	**0.984043**	**0.084180**	**0.0358381**	0.982186	0.083080	0.0348064	-0.2	-1.3	-2.9
IJCNN	**0.981917**	0.079808	0.0354446	0.976187	**0.083115**	**0.0387323**	-0.6	4.1	9.3

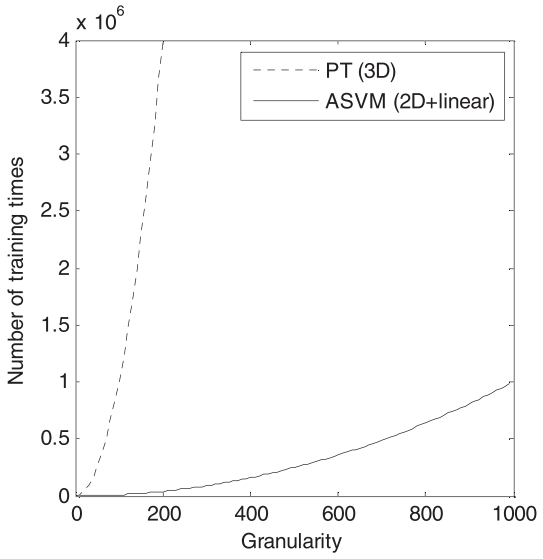

Fig. 7. Number of iterations required to complete a grid search.

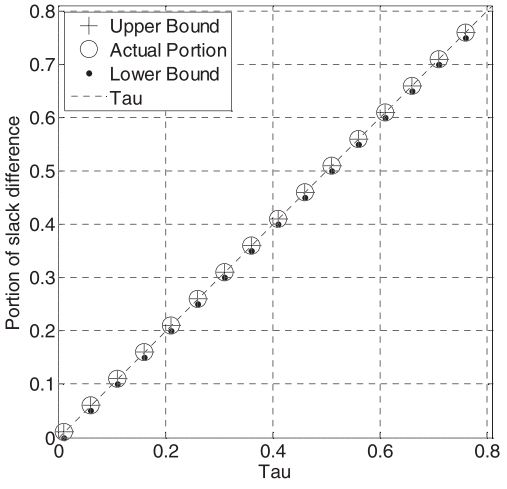

Fig. 8. The asymptotic property of τ.

after training error becomes zero. However, this advantage is not prominent universally, since in practice the data sets are usually noisy, and the margin may not grow fast enough to be satisfactory within a fixed number of iterations.

5.6 Asymptotic Property of τ

Another advantage of ASVM is that it is able to give more insight into the data set. In Section 3, we showed that there is an asymptotic relationship on the difference of the portion of the outliers between two classes. In order to give a more comprehensive view, we test the asymptotic property of τ in a synthetic data set with 90 positive labeled and 10 negative labeled instances. Fig. 8 shows the experimental results and compares the difference derived theoretically with that obtained in the simulation under different values of τ. Note the dotted line along the diagonal depicts the values of τ.

As we can see, the actual portion of outliers lies within the theoretical upper and lower bounds. Actually, these three lines will converge to a single when the number of training data increases. From above, the relation between the difference of the portion of outliers and τ is justified.

Recall in Section 4, we showed that the true false-positive rate can be lowered by increasing the parameter τ. Although there is no way to measure the true false-positive rate directly, the relationship between τ and the true false-positive rate holds as long as the relationship between τ and $\Pr^{emp}(\mathbf{o}_i^+)$, and the relationship between τ and $\Pr^{emp}(\mathbf{o}_i^-)$ follow (12) and (13), respectively. Note, these two equations can be combined as

$$\tau - \Pr^{emp}(\mathbf{s}_i^+) \le \Pr^{emp}(\mathbf{o}_i^+) - \Pr^{emp}(\mathbf{o}_i^-) \le \tau + \Pr^{emp}(\mathbf{s}_i^-),$$

which is verified by Fig. 8. Therefore, the relationship between τ and the true false-positive rate is also justified.

5.7 Imbalanced Data Sets

In this section, we evaluate the performance of ASVM over imbalanced data set. We consider the cases where the number of positive (resp. negative) instances is much more than the number of instances in the other class. This usually happens in applications such as transaction approval screening, credit risk analysis, and spam email detection. We vary the ratio between the positive and negative instances from 0.05 to 20. The data sets are retrieved by resampling the Statlog German, which labels people in terms of good or bad credit risks according to their credit

TABLE 3
Performance Comparison between the Asymmetric Boosting and ASVM in Terms of t-AUC, Where $t = 1$, 0.1, and 0.05 Are Given in Training Time

	Asym. Boosting (AB)			ASVM			Improvement		
Training target	1-AUC	0.1-AUC	0.05-AUC	1-AUC	0.1-AUC	0.05-AUC	%	%	%
Diabetes	**0.829362**	0.039149	–	0.828550	**0.040713**	–	-9.8e-4	4.0	–
Statlog German	0.741588	0.017729	–	**0.763777**	**0.019292**	–	3.0	8.8	–
Breast Cancer	0.992645	0.095424	–	**0.995895**	**0.095895**	–	0.3	0.5	–
Ionosphere	0.988571	0.089841	–	**0.996825**	**0.096825**	–	0.8	7.8	–
Australian	**0.940713**	0.064937	–	0.925197	**0.065787**	–	-1.6	1.3	–
Covertype	0.973244	**0.093390**	**0.0468464**	**0.982186**	0.083080	0.0348064	0.9	-11.0	-25.7
IJCNN	0.913250	0.060344	0.0228594	**0.976187**	**0.083115**	**0.0387323**	6.9	37.7	69.4

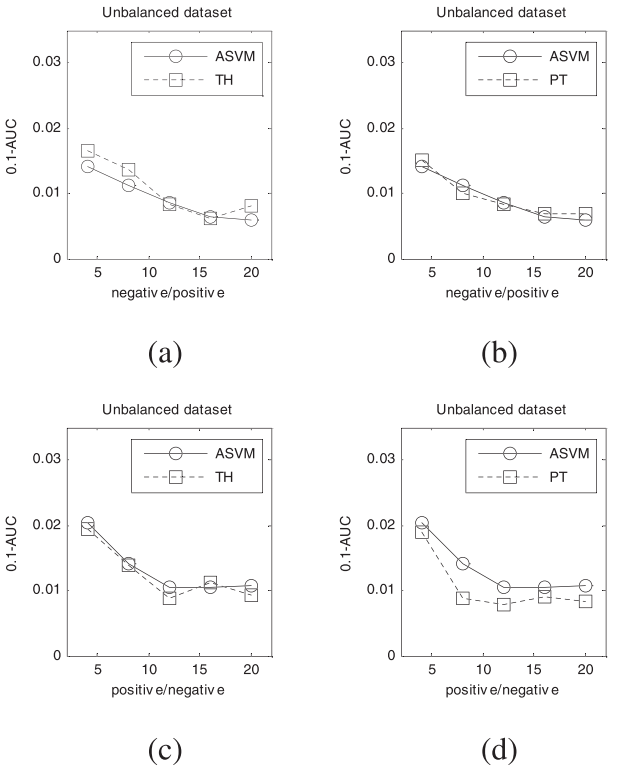

Fig. 9. The t-AUC achieved by TH, PT, and ASVM given imbalanced data sets. (a) TH versus ASVM when there are more negative instances. (b) PT versus ASVM when there are more negative instances. (c) TH versus ASVM when there are more positive instances. (d) PT versus ASVM when there are more positive instances.

data. We compare the ASVM with TH and PT, and train each of them using the 0.1-AUC metric plus fivefold cross validation. The results are averaged from 10 resampled data sets and shown in Fig. 9.

Generally, the higher the imbalance, the worse the results, as fewer training instances in one class leads to higher generalization error. It is interesting to see that TH outperforms PT in most cases. We believe this is because that TH has fewer parameters than PT, and therefore lower model complexity. Although giving bias, TH has lower variance and is less affected by the lack of training instances (in either class). The ASVM offers similar advantages over PT since it is trained using a 2D grid search rather than 3D. However, ASVM is outperformed by TH when there are fewer positive instances (Fig. 9a). On the other hand, it gives better results provided more positive instances (Fig. 9c). This implies that ASVM requires sufficient positive instances in order to find a good core. Also, it is less sensitive to the lack of negative instances.

6 Conclusions

In this paper, we propose the ASVM, an asymmetric variant of the support vector machine which takes into account the user tolerance of false-positive rate by maximizing the margin between the negative class and the core of the positive class. We show that the low false-positive rate achieved by the ASVM over the training data is generalizable, and quantitate the effects of μ and τ in terms of the portion of outliers. Thus, we are able to raise the confidence in predicting the positives and obtain a lower false-positive rate. The experimental results showed that ASVM achieves 6.4 percent improvement in AUC and dominates in the low false-positive region of the ROC Convex Hull as compared to the thresholding, and can result in a significant reduction in training time as compared to the parameter tuning.

Appendix A

A.1 Derivation of the ASVM Dual

To solve (6), we introduce a Lagrangian

$$\begin{aligned} L = \frac{1}{2}\|\mathbf{w}\|^2 - \rho + \frac{1}{\tau m}\sum_{i=1}^{m}\xi_i - \frac{\mu}{\tau}\gamma \\ - \sum_{i=1}^{m}\alpha_i\left(y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle - \rho) + \frac{1}{2}\gamma(y_i - 1) + \xi_i\right) \\ - \sum_{i=1}^{m}\beta_i\xi_i - \eta\gamma, \end{aligned} \tag{19}$$

where α_i, β_i, and η are Lagrange multipliers larger than or equal to 0. The Lagrangian L must be maximized with respect to α_i, β_i, and η, and minimized with respect to $\mathbf{w}$, ρ, γ, and ξ_i. At the Karush-Khun-Tucker (KKT) condition, we have

$$\begin{aligned} \frac{\partial L_P}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{m}\alpha_i y_i \Phi(\mathbf{x}_i) = 0 \\ \Rightarrow \mathbf{w} = \sum_{i=1}^{m}\alpha_i y_i \Phi(\mathbf{x}_i), \end{aligned} \tag{20}$$

$$\frac{\partial L_P}{\partial \rho} = -1 + \sum_{i=1}^{m}\alpha_i y_i = 0 \Rightarrow \sum_{i=1}^{m}\alpha_i y_i = 1, \tag{21}$$

$$\begin{aligned} \frac{\partial L_P}{\partial \gamma} = -\frac{\mu}{\tau} - \frac{1}{2}\sum_{i=1}^{m}\alpha_i(y_i - 1) - \eta = 0 \\ \Rightarrow \sum_{i=1}^{m}\alpha_i \geq 2\frac{\mu}{\tau} + 1, \end{aligned} \tag{22}$$

$$\frac{\partial L_P}{\partial \xi_i} = \frac{1}{\tau m} - \alpha_i - \beta_i = 0 \Rightarrow 0 \leq \alpha_i \leq \frac{1}{\tau m}. \tag{23}$$

Replacing the corresponding terms in (19) by those in (20)-(23) and substituting the kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$ for the dot product $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle$, we obtain the dual objective of ASVM.

Note we may also rewrite $f(\mathbf{x}) = \sum_{i=1}^{m}\alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) - \rho + \frac{\gamma}{2}$. The values of ρ and γ can be recovered using the KKT complementarity conditions. At optimum, we have

$$\begin{aligned} \alpha_i\big(y_i(\langle \mathbf{w}, \Phi(\mathbf{x}_i)\rangle - \rho) + \frac{1}{2}\gamma(y_i - 1) + \xi_i\big) = 0, \\ \beta_i\xi_i = 0, \text{ and } \eta\gamma = 0, \end{aligned} \tag{24}$$

$\forall 1 \leq i \leq m$. For each positive in-bound support vector $\mathbf{s}_i^+$, the second term at the left hand side of (24) must be zero. We have $\rho = \sum_{j=1}^{m}\alpha_j y_j k(\mathbf{x}_j, \mathbf{s}_i^+)$. Furthermore, for each $\mathbf{s}_i^-$, the equation $\gamma = \rho - \sum_{j=1}^{m}\alpha_j y_j k(\mathbf{x}_j, \mathbf{s}_i^-)$ holds.

A.2 Proof of Lemma 4

By definition, we have

$$D^{2m}\Big\{\mathbf{Z}_m\widetilde{\mathbf{Z}}_m : \varphi > \frac{\epsilon}{2}\Big\} = \int_{\mathcal{Z}_{2m}} \theta\Big(\psi - \frac{\epsilon}{2}\Big) dD^{2m}(\mathbf{Z}_m\widetilde{\mathbf{Z}}_m),$$

where $\theta(x) = 1$ if $x \geq 0$; $\theta(x) = 0$ otherwise. Since $\mathbf{Z}_m\widetilde{\mathbf{Z}}_m \in \mathcal{Z}_m \times \mathcal{Z}_m$, by Fubini's theorem the above equation can be reformulated

$$\begin{aligned}&\int_{\mathcal{Z}_m}\Bigg(\int_{\mathcal{Z}_m}\theta\Big(\psi-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\Bigg)dD^m(\mathbf{Z}_m)\\ &\quad\geq\int_{\{\mathbf{Z}_m:\varphi>\epsilon\}}\Bigg(\int_{\mathcal{Z}_m}\theta\Big(\psi-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\Bigg)dD^m(\mathbf{Z}_m).\end{aligned}\tag{25}$$

Note that for each fixed $\mathbf{Z}_m \in \{\mathbf{Z}_m : \varphi > \epsilon\}$, there exist two events $A_0 \in \mathcal{A}$ and $A_0' \in \mathcal{A}'$, such that

$$\Big|\Pr(A_0) - \Pr^{emp}_{A_0'}(\mathbf{Z}_m)\Big| > \epsilon.$$

To satisfy the condition

$$\Big|\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m) - \Pr^{emp}_{A_0'}(\mathbf{Z}_m)\Big| > \frac{\epsilon}{2},$$

it is sufficient to require that $|\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m) - \Pr(A_0)| < \frac{\epsilon}{2}$. We may therefore express the inner integral of (25) as

$$\begin{aligned}&\int_{\mathcal{Z}_m}\theta\Big(\psi-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\\ &\geq\int_{\mathcal{Z}_m}\theta\Big(\big|\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m)-\Pr^{emp}_{A_0'}(\mathbf{Z}_m)\big|-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\\ &\geq\int_{\mathcal{Z}_m}\theta\Big(\frac{\epsilon}{2}-\big|\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m)-\Pr(A_0)\big|\Big)dD^m(\widetilde{\mathbf{Z}}_m)\\ &=1-D^m\Big\{\widetilde{\mathbf{Z}}_m:\big|\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m)-\Pr(A_0)\big|\geq\frac{\epsilon}{2}\Big\}.\end{aligned}$$

Regard $D^m\{\widetilde{\mathbf{Z}}_m : |\widetilde{\mathbf{Z}}_m \cap A_0|\}$ as a sequence of m independent $\{0,1\}$ experiments, each of which yields the outcome 1 with probability $\Pr(A_0)$. The term $D^m\{\{\widetilde{\mathbf{Z}}_m : \Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m)\}$ has a binomial distribution of mean $\Pr(A_0)$ and variance $\frac{(1-\Pr(A_0))\Pr(A_0)}{m}$. By Chebyshev's inequality, we have

$$\begin{aligned}&D^m\Big\{\widetilde{\mathbf{Z}}_m : |\Pr^{emp}_{A_0}(\widetilde{\mathbf{Z}}_m) - \Pr(A_0)| \geq \frac{\epsilon}{2}\Big\}\\ &\quad\leq\frac{4(1-\Pr(A_0))\Pr(A_0)}{\epsilon^2 m}\leq\frac{1}{\epsilon^2 m}.\end{aligned}$$

This implies that, for $\epsilon^2 m \geq 2$,

$$\int_{\mathcal{Z}_m}\theta\Big(\psi-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\geq\frac{1}{2}.$$

Applying this to (25), we obtain

$$\begin{aligned}&\int_{\mathcal{Z}_m}\Bigg(\int_{\mathcal{Z}_m}\theta\Big(\psi-\frac{\epsilon}{2}\Big)dD^m(\widetilde{\mathbf{Z}}_m)\Bigg)dD^m(\mathbf{Z}_m)\\ &\quad\geq\int_{\{\mathbf{Z}_m:\varphi>\epsilon\}}\frac{1}{2}dD^m(\mathbf{Z}_m)\\ &\quad=\frac{1}{2}D^m\{\mathbf{Z}_m:\varphi>\epsilon\}.\end{aligned}$$

The proof follows.

REFERENCES

[1] I. Androutsopoulos, J. Koutsias, K. Chandrinos, and C. Spyropoulos, "An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-Mail Messages," *Proc. 23rd Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, 2000.

[2] A. Asuncion and D. Newman, *UCI Machine Learning Repository*, http://www.ics.uci.edu/ mlearn/MLRepository.html, 2007.

[3] D. Barbara, N. Wu, and S. Jajodia, "Detecting Novel Network Intrusions Using Bayes Estimators," *Proc. First SIAM Conf. Data Mining (SDM)*, 2001.

[4] P. Bartlett and J. Shawe-Taylor, "Generalization Performance of Support Vector Machines and Other Pattern Classifiers," *Advances in Kernel Methods: Support Vector Learning*, MIT Press, 1998.

[5] P. Bartlett, "The Sample Complexity of Pattern Classification with Neuralnetworks: The Size of the Weights is More Important than the Size of Thenetwork," *IEEE Trans. Information Theory*, vol. 44, no. 2, pp. 525-536, Mar. 1998.

[6] A. Ben-Hur, D. Horn, H. Siegelmann, and V. Vapnik, "Support Vector Clustering," *J. Machine Learning Research*, vol. 2, pp. 125-137, 2001.

[7] P. Boykin and V. Roychowdhury, "Leveraging Social Networks to Fight Spam," *Computer*, vol. 38, pp. 61-68, 2005.

[8] A. Bratko, G. Cormack, B. Filipic, T. Lynam, and B. Zupan, "Spam Filtering using Statistical Data Compression Models," *J. Machine Learning Research*, vol. 7, pp. 2673-2698, 2006.

[9] L. Breiman, *Classification and Regression Trees.* Chapman & Hall, 1998.

[10] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998.

[11] X. Carreras and L. Marquez, "Boosting Trees for Anti-Spam Email Filtering," *Proc. Fourth Int'l Conf. Recent Advances in Natural Language Processing*, 2001.

[12] C.-C. Chang and C.-J. Lin, "LIBSVM: A Library for Support Vector Machines," software, http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

[13] H. Cheng, X. Cai, X. Chen, L. Hu, and X. Lou, "Computer-Aided Detection and Classification of Microcalcifications in Mammograms: A Survey," *Pattern Recognition*, vol. 36, no. 12, pp. 2967-2991, 2003.

[14] G. Cormack and T. Lynam, "Overview of the Trec 2005 Spam Evaluation Track," *Proc. 14th Text REtrieval Conf. (TREC '05)*, 2005.

[15] C. Cortes and V. Vapnik, "Support Vector Networks," *Machine Learning*, vol. 20, pp. 273-297, 1995.

[16] H. Drucker, D. Wu, and V. Vapnik, "Support Vector Machines for Spam Categorization," *IEEE Trans. Neural Networks*, vol. 10, no. 5, pp. 1048-1054, Sept. 1999.

[17] J. Goodman, G. Cormack, and D. Heckerman, "Spam and the Ongoing Battle for the Inbox," *Comm. ACM*, vol. 50, no. 2, pp. 24-33, Feb. 2007.

[18] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A Practical Guide to Support Vector Classification," technical report, http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2003.

[19] S. Inalou and S. Kasaei, "Adaboost-Based Face Detection in Color Images with Low False Alarm," *Proc. Second Int'l Conf. Computer Modeling and Simulation*, 2010.

[20] J. Kivinen, A. Smola, and R. Williamson, "Online Learning with Kernels," *Advances in Neural Information Processing Systems*, vol. 14, pp. 785-793, MIT Press, 2002.

[21] A. Kolcz and J. Alspector, "SVM-Based Filtering of E-Mail Spam with Content-Specific Misclassification Costs," *Proc. Workshop Text Mining—IEEE Int'l Conf. Data (TextDM)*, 2001.

[22] H.-Y. Lam and D.-Y. Yeung, "A Learning Approach to Spam Detection Based on Social Networks," *Proc. Fourth Conf. Email and Anti-Spam (CEAS)*, 2007.

[23] Y.-F. Li, J.T. Kwok, and Z.-H. Zhou, "Cost-Sensitive Semi-Supervised Support Vector Machine," *Proc. 24th AAAI Conf. Artificial Intelligence (AAAI)*, 2010.

[24] T. Lynam, G. Cormack, and D. Cheriton, "On-Line Spam Filter Fusion," *Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, pp. 123-130. 2006,

[25] H. Masnadi-Shirazi and N. Vasconcelos, "Asymmetric Boosting," *Proc. 24th Int'l Conf. Machine Learning (ICML)*, 2007.

[26] J. Platt, "Sequenital Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," *Advances in Kernel Methods: Support Vector Learning*, MIT Press, 1998.

[27] D. Prokhorov, *IJCNN 2001 Neural Network Competition,* Ford Research Laboratory, 2001.
[28] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian Approach to Filtering Junk E-Mail," Technical Report WS-98-05, AAAI, 1998.
[29] K. Schneider, "A Comparison of Event Models for Naive Bayes Anti-Spam E-Mail Filtering," *Proc. 11th Conf. the European Chapter of the Assoc. for Computational Linguistics,* 2003.
[30] B. Scholkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R.C. Williamson, "Estimating the Support of a High-Dimensional Distribution," *Neural Computation,* vol. 13, pp. 1443-1471, 2001.
[31] B. Scholkopf and A. Smola, *Learning with Kernels:: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, 2002.
[32] D. Sculley and G. Wachman, "Relaxed Online Support Vector Machines for Spam Filtering," *Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR),* 2007.
[33] J. Shawe-Taylor, P. Bartlett, R. Williamson, and M. Anthony, "Structural Risk Minimization Over Data-Dependent Hierarchies," *IEEE Trans. Information Theory,* vol. 44, no. 5, pp. 1926-1940, Sept. 1998.
[34] D. Song and Y. Xu, "A Low False Negative Filter for Detecting Rare Bird Species from Short Video Segments Using a Probable Observation Data Set-Based EKF Method," *IEEE Trans. Image Processing,* vol. 19, no. 9, pp. 2321-2331, Sept. 2010.
[35] V. Vapnik, *Statistical Learning Theory.* Wiley, 1998.
[36] V. Vapnik and A. Chervonenkis, "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities," *Theory of Probability and Its Applications,* vol. 16, no. 2, pp. 264-280, 1971.
[37] P. Viola and M. Jones, "Fast and Robust Classification Using Asymmetric Adaboost and a Detector Cascade," *Proc. Neural Information Processing Systems Conf. (NIPS),* 2002.
[38] J. Wu, M.D. Mullin, and J.M. Rehg, "Linear Asymmetric Classifier for Cascade Detectors," *Proc. 22nd Int'l Conf. Machine Learning (ICML),* 2005.
[39] W. Yih, J. Goodman, and G. Hulten, "Learning at Low False Positive Rates," *Proc. Third Conf. Email and Anti-Spam (CEAS),* 2006.
[40] B. Zheng, W. Qian, and L. Clarke, "Digital Mammography: Mixed Feature Neural Network with Spectralentropy Decision for Detection of Microcalcifications," *IEEE Trans. Medical Imaging,* vol. 15, no. 5, pp. 589-597, Oct. 1996.
[41] Z.-H. Zhou and X.-Y. Liu, "Training Cost-Sensitive Neural Networks with Methods Addressing the Class Imbalance Problem," *IEEE Trans. Knowledge and Data Eng.,* vol. 18, no. 1, pp. 63-77, Jan. 2006.

Shan-Hung Wu received the PhD degree in electrical engineering from the National Taiwan University, Taiwan, in 2009. He is an assistant professor in the Department of Computer Science, National Tsing Hua University (NTHU), Hsinchu, Taiwan. Before joining NTHU in 2010, he was a senior research scientist at Telcordia Technologies from 2004 to 2010. His research interests include database systems, data mining, and mobile data management.

Keng-Pei Lin received the BS degree in computer science and information engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2005, and the PhD degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 2011. He is currently an assistant professor in the Department of Information Management, National Sun Yat-sen University, Kaohsiung, Taiwan. His research interests include data mining, business intelligence, and machine learning.

Hao-Heng Chien received the BS degree from the Department of Computer Science, National Tsing Hua University (NTHU), Hsinchu, Taiwan, where he is currently a graduate student. His research interests include data mining and social network analysis.

Chung-Min Chen received the BS degree in computer science and information engineering from the National Taiwan University, and the PhD degree in computer science from the University of Maryland, College Park. He is a chief scientist at Telcordia Technologies. His research interests include database systems, mobile networks, and their applications.

Ming-Syan Chen received the BS degree in electrical engineering from National Taiwan University, Taipei, Taiwan, and the MS and PhD degrees in computer, information and control engineering from The University of Michigan, Ann Arbor, Michigan, in 1985 and 1988, respectively. He is now a distinguished research fellow and the director of the Research Center of Information Technology Innovation (CITI) at the Academia Sinica, Taiwan, and is also a distinguished professor jointly appointed by the EE Department, CSIE Department, and Graduate Institute of Communication Engineering (GICE) at National Taiwan University. He was a research staff member at IBM Thomas J. Watson Research Center, Yorktown Heights, New York, the director of GICE, and also the president/CEO of the Institute for Information Industry (III), which is one of the largest organizations for information technology in Taiwan. His research interests include databases, data mining, cloud computing, and multimedia networking, and he has published more than 300 papers in his research areas. In addition to serving as program chairs/vice-chairs and keynote/tutorial speakers in many international conferences, he was an associate editor of *IEEE Transactions on Knowledge and Data Engineering*, *VLDB Journal*, *Knowledge and Information Systems*, and also *Journal of Information Science and Engineering*, is currently the editor-in-chief of the *International Journal of Electrical Engineering (IJEE)*, and is a distinguished visitor of IEEE Computer Society for Asia-Pacific from 1998 to 2000, and also from 2005 to 2007. He is now also serving as the CEO of Networked Communication Program, which is a national program coordinating several primary activities in information and communication technologies in Taiwan. He is a recipient of the Academic Award of the Ministry of Education, the National Science Council (NSC) Distinguished Research Award, Pan Wen Yuan Distinguished Research Award, Teco Award, Honorary Medal of Information, and K.-T. Li Research Breakthrough Award for his research work, and also the Outstanding Innovation Award from IBM Corporate for his contribution to a major database product. He was also elected as a chair professor by National Chung Hsing University. He is a fellow of the IEEE and ACM.