Cost-Sensitive Semi-Supervised Support Vector Machine

Yu-Feng Li[1] **James T. Kwok**[2] **Zhi-Hua Zhou**[1*]
[1] National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China
[2] Department of Computer Science & Engineering, Hong Kong University of Science and Technology, Hong Kong
liyf@lamda.nju.edu.cn jamesk@cse.ust.hk zhouzh@lamda.nju.edu.cn

Abstract

In this paper, we study cost-sensitive semi-supervised learning where many of the training examples are unlabeled and different misclassification errors are associated with unequal costs. This scenario occurs in many real-world applications. For example, in some disease diagnosis, the cost of erroneously diagnosing a patient as healthy is much higher than that of diagnosing a healthy person as a patient. Also, the acquisition of labeled data requires medical diagnosis which is expensive, while the collection of unlabeled data such as basic health information is much cheaper. We propose the CS4VM (Cost-Sensitive Semi-Supervised Support Vector Machine) to address this problem. We show that the CS4VM, when given the *label means* of the unlabeled data, closely approximates the supervised cost-sensitive SVM that has access to the ground-truth labels of all the unlabeled data. This observation leads to an efficient algorithm which first estimates the label means and then trains the CS4VM with the plug-in label means by an efficient SVM solver. Experiments on a broad range of data sets show that the proposed method is capable of reducing the total cost and is computationally efficient.

Introduction

In many real-world applications, different misclassifications are often associated with unequal costs. For example, in medical diagnosis, the cost of erroneously diagnosing a patient as healthy may be much higher than that of diagnosing a healthy person as a patient. Another example is fraud detection where the cost of missing a fraud is much larger than a false alarm. On the other hand, obtaining labeled data is usually expensive while gathering unlabeled data is much cheaper. For example, in medical diagnosis the cost of medical tests and analysis is much higher than the collection of basic health information, while in fraud detection the labeling of a fraud is often costly since domain experts are required. Consequently, many of the training examples may remain unlabeled.

Cost-sensitive learning (Domingos 1999; Fan et al. 1999; Elkan 2001; Ting 2002; Zadrozny, Langford, and Abe 2003; Zhou and Liu 2006b; 2006a; Masnadi-Shirazi and Vasconcelos 2007; Lozano and Abe 2008) aims to make the optimal decision minimizing the total cost. Semi-supervised learning aims to improve the generalization performance by appropriately exploiting the unlabeled data. Over the past decade, these two learning paradigms have attracted growing attention and many techniques have been developed (Chapelle, Schölkopf, and Zien 2006; Zhu 2007; Zhou and Li 2010). However, existing cost-sensitive learning methods mainly focus on the supervised learning setting, while semi-supervised learning methods are usually cost-insensitive.

To deal with scenarios where unequal misclassification cost occur while the exploitation of unlabeled data is necessary, we study in this paper cost-sensitive semi-supervised learning. We propose the CS4VM (Cost-Sensitive Semi-Supervised Support Vector Machine) to address such problem. We show that the CS4VM, when given the *label means* of the unlabeled data, is closely related to the supervised cost-sensitive SVM that has ground-truth labels for all the unlabeled data. Based on this observation, we propose an efficient algorithm that first estimates the label means of the unlabeled examples, and then use these plug-in estimates to solve the CS4VM with an efficient SMO algorithm. Experimental results on a broad range of data sets validate our proposal.

The rest of this paper is organized as follows. We start by a brief introduction of some related work. Then we propose C4SVM and report on our experiments, which is followed by the conclusion.

Related Work

Learning process may encounter many types of costs, such as the testing cost, teacher cost, intervention cost, etc. (Turney 2000), among which the most important type is the misclassification cost. There are two kinds of misclassification cost. The first one is class-dependent cost, where the costs of classifying any examples in class A to class B are the same. The second one is example-dependent cost, where the costs of classifying different examples in class A to class

*This research was supported by the National Fundamental Research Program of China (2010CB327903), the National Science Foundation of China (60635030, 60903103), the Jiangsu Science Foundation (BK2008018) and the Research Grants Council of the Hong Kong Special Administrative Region (614508).

B are different. Generally the costs of different kinds of misclassifications are given by the user, and in practice it is much easier for the user to give class-dependent cost than example-dependent cost. So, the former occurs more often in real applications and has attracted more attention.

In this paper, we will focus on class-dependent cost. Existing methods for handling class-dependent costs mainly fall into two categories. The first one is geared towards particular classifiers, like decision trees (Ting 2002), neural networks (Kukar and Kononenko 1998), AdaBoost (Fan et al. 1999), etc. The second one is a general approach, which rescales (Elkan 2001; Zhou and Liu 2006a) the classes such that the influences of different classes are proportional to their costs. It can be realized in different ways, such as instance weighting (Ting 2002; Zhou and Liu 2006b), sampling (Elkan 2001; Zhou and Liu 2006b), threshold moving (Domingos 1999; Zhou and Liu 2006b), etc. Cost-sensitive support vector machines have also been studied (Morik, Brockhausen, and Joachims 1999; Brefeld, Geibel, and Wysotzki 2003; Lee, Lin, and Wahba 2004).

Many semi-supervised learning methods have been proposed (Chapelle, Schölkopf, and Zien 2006; Zhu 2007; Zhou and Li 2010). A particularly interesting method is the S3VM (Semi-Supervised Support Vector Machine) (Bennett and Demiriz 1999; Joachims 1999). It is built on the cluster assumption and regularizes the decision boundary by exploiting the unlabeled data. Specifically, it favors decision boundaries that go cross the low-density regions (Chapelle and Zien 2005). The effect of its objective has been well studied in (Chapelle, Sindhwani, and Keerthi 2008). Due to the high complexity in solving the S3VM, many efforts have been devoted to speeding up the optimization. Examples include local search (Joachims 1999), concave convex procedure (Collobert et al. 2006) and many other optimization techniques (Chapelle, Sindhwani, and Keerthi 2008). Recently, (Li, Kwok, and Zhou 2009) revisits the formulation of S3VM and shows that when given the class (label) means of the unlabeled data, the S3VM is closely related to a supervised SVM that is provided with the unknown ground-truth labels of all the unlabeled training data. This indicates that the label means of the unlabeled data, which is a simpler statistic than the set of labels of all the unlabeled patterns, can be very useful in semi-supervised learning.

The use of unlabeled data in cost-sensitive learning has been considered in a few studies (Greiner, Grove, and Roth 2002; Margineantu 2005; Liu, Jun, and Ghosh 2009; Qin et al. 2008), most of which try to involve human feedback on informative unlabeled instances and then refine the cost-sensitive model using the queried labels. In this paper, we focus on using SVM to address unequal costs and utilize unlabeled data simultaneously, by extending the approach of (Li, Kwok, and Zhou 2009) to the cost-sensitive setting.

CS4VM

In this section, we first present the formulation of CS4VM and show the usefulness of the label means in this context. Then, an efficient learning algorithm will be introduced.

Formulation

In cost-sensitive semi-supervised learning, we are given a set of labeled data $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_l, y_l)\}$ and a set of unlabeled data $\{\mathbf{x}_{l+1}, \cdots, \mathbf{x}_{l+u}\}$, where $y \in \{\pm 1\}$, l and u are the numbers of labeled and unlabeled instances, respectively. Let $\mathcal{I}_l = \{1, \cdots, l\}$ and $\mathcal{I}_u = \{l+1, \cdots, l+u\}$ be the sets of indices for the labeled and unlabeled data, respectively. Moreover, suppose the cost of misclassifying a positive (or negative) instance is $c(+1)$ (or $c(-1)$).

We first consider the simpler supervised learning setting. Suppose that for each unlabeled pattern $\mathbf{x}_i$ ($i \in \mathcal{I}_u$), we are given the corresponding label $\hat{y}_i$. Then we can derive the supervised cost-sensitive SVM (CS-SVM) (Morik, Brockhausen, and Joachims 1999) which finds a decision function $f(\mathbf{x})$ by minimizing the following functional:

$$\min_f \frac{1}{2}\|f\|_{\mathcal{H}}^2 + C_1 \sum_{i \in \mathcal{I}_l} \ell(y_i, f(\mathbf{x}_i)) + C_2 \sum_{i \in \mathcal{I}_u} \ell(\hat{y}_i, f(\mathbf{x}_i)) \ , \tag{1}$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space (RKHS) induced by a kernel k and $\ell(y, f(\mathbf{x})) = c(y)\max\{0, 1 - yf(\mathbf{x})\}$ is the weighted hinge loss, C_1 and C_2 are regularization parameters trading off the complexity and empirical errors on the labeled and unlabeled data. The relation between Eq. 1 and the Bayes rule has been discussed in (Brefeld, Geibel, and Wysotzki 2003).

In semi-supervised setting, the labels $\hat{\mathbf{y}} = [\hat{y}_i; i \in \mathcal{I}_u]$ of the unlabeled data are unknown, and so need to be optimized as well. This leads to the CS4VM:

$$\min_{\hat{\mathbf{y}} \in \mathcal{B}} \min_f \frac{1}{2}\|f\|_{\mathcal{H}}^2 + C_1 \sum_{i \in \mathcal{I}_l} \ell(y_i, f(\mathbf{x}_i)) + C_2 \sum_{i \in \mathcal{I}_u} \ell(\hat{y}_i, f(\mathbf{x}_i)), \tag{2}$$

where $\mathcal{B} = \{\hat{\mathbf{y}} | \hat{y}_i \in \{\pm 1\}, \hat{\mathbf{y}}'\mathbf{1} = r\}$, $\mathbf{1}$ is the all-one vector, and $\hat{\mathbf{y}}'\mathbf{1} = r$ (with the user-defined parameter r) is the balance constraint which avoids the trivial solution that assigns all the unlabeled instances to the same class. Note that the label $\hat{y}_i$ should be as same as the sign [1] of the prediction $f(\mathbf{x}_i)$, i.e., $\hat{y}_i = \text{sgn}(f(\mathbf{x}_i))$. Substituting this into Eq. 2, we obtain the following optimization problem which no longer involves the additional variable $\hat{\mathbf{y}}$:

$$\begin{aligned} \min_f \ & \frac{1}{2}\|f\|_{\mathcal{H}}^2 + C_1 \sum_{i \in \mathcal{I}_l} \ell(y_i, f(\mathbf{x}_i)) + C_2 \sum_{i \in \mathcal{I}_u} \ell(\hat{y}_i, f(\mathbf{x}_i)) \\ \text{s.t.} \ & \sum_{i \in \mathcal{I}_u} \text{sgn}(f(\mathbf{x}_i)) = r, \hat{y}_i = \text{sgn}(f(\mathbf{x}_i)), \forall i \in \mathcal{I}_u \end{aligned} \tag{3}$$

Figure 1 shows the loss function used for the unlabeled data. When $c(1) = c(-1)$, it becomes the standard symmetric hinge loss and CS4VM degenerates to TSVM (Joachims 1999). When $c(1) \neq c(-1)$, however, the loss is no longer continuous and many optimization techniques (Chapelle and Zien 2005) could not be applied.

Label Means for CS4VM

Note that Eq. 3 involves the estimation of labels of all the unlabeled instances, which will be computationally inefficient

[1] Here, we assume that $\text{sgn}(0) = 1$.

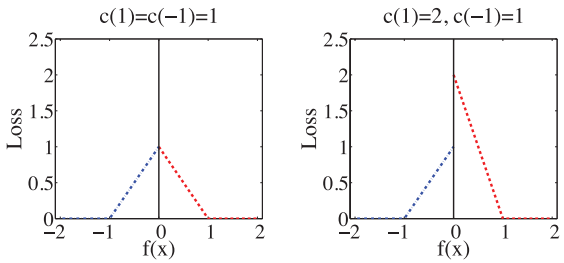

Figure 1: Loss function for the unlabeled data.

when the number of unlabeled instances is large. Motivated by the observation in (Li, Kwok, and Zhou 2009) that the label means offer a simpler statistic than the set of labels on the unlabeled data, we will extend this observation to the CS4VM. Moreover, we will see that the label means naturally decouple the cost and prediction, and the estimation is also efficient.

Introducing additional variables $\mathbf{p}^+ = [p^+_{\ell+1}, \ldots, p^+_{\ell+u}]'$ and $\mathbf{p}^- = [p^-_{\ell+1}, \ldots, p^-_{\ell+u}]'$, we can rewrite the CS4VM as the following:

$$\begin{aligned}\min_{f,\mathbf{p}^+,\mathbf{p}^-} \ & \frac{1}{2}\|f\|^2_{\mathcal{H}} + C_1 \sum_{i\in\mathcal{I}_l} \ell(y_i, f(\mathbf{x}_i)) + C_2 \sum_{i\in\mathcal{I}_u} (p_i^+ \\ & + p_i^- - c(\text{sgn}(f(\mathbf{x}_i)))(\text{sgn}(f(\mathbf{x}_i)f(\mathbf{x}_i) - 1))) \\ \text{s.t.} \quad & c(+1)f(\mathbf{x}_i) - c(+1) \le p_i^+, \\ & -c(-1)f(\mathbf{x}_i) - c(-1) \le p_i^-, \\ & p_i^+, p_i^- \ge 0, \forall i \in \mathcal{I}_u; \sum_{i\in\mathcal{I}_u} \text{sgn}(f(\mathbf{x}_i)) = r\ . \end{aligned} \tag{4}$$

Proposition 1. *Eq. 4 is equivalent to the CS4VM.*

Proof. When $0 \le f(\mathbf{x}_i) \le 1$, both $p_i^{\pm}$ are zero and the loss of $\mathbf{x}_i$ is $-c(+1)(f(\mathbf{x}_i) - 1)$, which is equal to $c(+1)(1 - f(\mathbf{x}_i))$ in Eq. 3. When $f(\mathbf{x}_i) \ge 1$, $p_i^- = 0$ and $p_i^+ = c(+1)(f(\mathbf{x}_i) - 1)$, and thus the overall loss is zero, which is equal to CS4VM. A similar proof holds for $f(\mathbf{x}_i) < 0$. □

Let $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$, where $\phi(\cdot)$ is the feature mapping induced by the kernel k. As in (Li, Kwok, and Zhou 2009), by using the balance constraint, the number of positive (resp. negative) instances in the unlabeled data can be obtained as $u_+ = \frac{r+u}{2}$ (resp. $u_- = \frac{u-r}{2}$). Suppose that the ground-truth label of the unlabeled instance $\mathbf{x}_i$ is y_i^*. The label means of the unlabeled data are then $\mathbf{m}_+ = \frac{1}{u_+}\sum_{y_i^*=1}\phi(\mathbf{x}_i)$ and $\mathbf{m}_- = \frac{1}{u_-}\sum_{y_i^*=-1}\phi(\mathbf{x}_i)$, respectively. Let $n_1 = c(1)u_+ + c(-1)u_-$ and $n_2 = c(1)u_+ - c(-1)u_-$. We have

$$\begin{aligned} & \sum_{i\in\mathcal{I}_u} c(\text{sgn}(f(\mathbf{x}_i)))(\text{sgn}(f(\mathbf{x}_i)f(\mathbf{x}_i) - 1)) + n_1 \qquad (5) \\ & = c(1) \sum_{f(\mathbf{x}_i)\ge 0, i\in\mathcal{I}_u} f(\mathbf{x}_i) + c(-1) \sum_{f(\mathbf{x}_i)<0, i\in\mathcal{I}_u} -f(\mathbf{x}_i) \\ & = \mathbf{w}'\Big(u_+c(+1)\hat{\mathbf{m}}_+ - u_-c(-1)\hat{\mathbf{m}}_-\Big) + n_2 b, \end{aligned}$$

where $\hat{\mathbf{m}}_+ = \frac{1}{u_+}\sum_{i\in\mathcal{I}_u, f(\mathbf{x}_i)\ge 0}\phi(\mathbf{x}_i)$ (resp. $\hat{\mathbf{m}}_- = \frac{1}{u_-}\sum_{i\in\mathcal{I}_u, f(\mathbf{x}_i)<0}\phi(\mathbf{x}_i)$) is an estimate of $\mathbf{m}_+$ (resp. $\mathbf{m}_-$).

Eq.5 implies the objective in Eq.4 is only related to label means. If we substitute the true label means $\mathbf{m}_\pm$ into Eq. 4, we have

$$\begin{aligned} \min_{\mathbf{w},b,\mathbf{p}^\pm} \ & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i\in\mathcal{I}_l} \ell(y_i, \mathbf{w}'\mathbf{x}_i + b) + C_2(\mathbf{1}'\mathbf{p}^+ + \mathbf{1}'\mathbf{p}^-) \\ & - C_2\Big(\mathbf{w}'\Big(u_+c(+1)\mathbf{m}_+ - u_-c(-1)\mathbf{m}_-\Big) - n_1 + n_2 b\Big) \\ \text{s.t.} \quad & \text{constraints in Eq. 4.} \end{aligned} \tag{6}$$

Eq. 6 is the CS4VM with known label means of the unlabeled data. The relation between Eq. 6 and the supervised CS-SVM is stated by the following theorem.

Theorem 1. *Suppose that f^* is the optimal solution of Eq. 6. When all the unlabeled data do not suffer from large loss, i.e., $y_i^* f^*(\mathbf{x}_i) \ge -1, \forall i \in \mathcal{I}_u$, Eq. 6 is equivalent to the CS-SVM. Otherwise, let $\hat{\ell}(\mathbf{x}_i)$ be the loss for the unlabeled instance $\mathbf{x}_i$ in Eq. 6. Then, $\hat{\ell}(\mathbf{x}_i) \le \frac{c(1)+c(-1)}{c(y_i^*)}\ell(y_i^*, f(\mathbf{x}_i))$.*

It is notable that Theorem 1 reduces to the results in (Li, Kwok, and Zhou 2009) when $c(1) = c(-1)$. The proof is similar to (Li, Kwok, and Zhou 2009), and so will be omitted here. Figure 2 compares the loss in Eq.6 and the cost-sensitive hinge loss in CS-SVM on positive and negative examples.

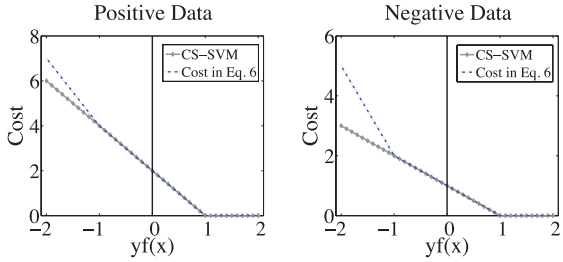

Figure 2: Loss in Eq. 6 and the cost-sensitive hinge loss in CS-SVM. Here $c(1) = 2$ and $c(-1) = 1$.

Learning Algorithm

Analysis in the above section suggests that the label means are useful for cost-sensitive semi-supervised support vector machines. This motivates us to first estimate the label means and then solve Eq. 6 with the estimated label means.

Estimating Label Means To estimate the label means, we employ the large margin principle (Li, Kwok, and Zhou 2009) consistent with the CS4VM, i.e., maximizing the margin between the means, which is also interpretable by Hilbert space embedding of distributions (Gretton et al. 2006). Mathematically,

$$\begin{aligned} \min_{\mathbf{d}\in\Delta} \min_{\mathbf{w},b,\rho} \ & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i\in\mathcal{I}_l} \ell(y_i, \mathbf{w}'\mathbf{x}_i + b) - C_2\rho \\ \text{s.t.} \quad & \mathbf{w}'\left(\frac{\sum_{i\in I_u} d_i\phi(\mathbf{x}_i)}{u_+}\right) + b \ge c(+1)\rho, \qquad (7) \\ & \mathbf{w}'\left(\frac{\sum_{i\in I_u}(1-d_i)\phi(\mathbf{x}_i)}{u_-}\right) + b \le -c(-1)\rho\ , \qquad (8) \end{aligned}$$

where $\mathbf{d} = [d_i; i \in \mathcal{I}_u]$ and $\Delta = \{\mathbf{d}|d_i \in \{0,1\}, \mathbf{d}'\mathbf{1} = u_+\}$. Note from Eqs. 7 and 8 that the class with the larger

Table 1: Comparison of total costs ((mean $\pm$ std.) $\times 10^3$) in the first series of experiments ($c(1)$ is randomly sampled from an interval). The best performance (paired t-tests at 95% significance level) and its comparable results are bolded. The last line shows the win/tie/loss counts of CS4VM versus other methods.

Data set	Supervised CS-SVM	Laplacian SVM	TSVM	CS4VM	GT CS-SVM
Heart-Statlog	9.745 ± 6.906	$\mathbf{1.640 \pm 2.708}$	10.28 ± 6.985	6.261 ± 4.920	1.894 ± 1.836
Ionosphere	17.02 ± 12.84	27.19 ± 17.03	11.98 ± 7.749	$\mathbf{7.811 \pm 5.130}$	5.036 ± 3.604
Live Disorder	$\mathbf{0.178 \pm 0.388}$	11.37 ± 17.29	12.01 ± 7.844	0.507 ± 1.018	0.100 ± 0.005
Echocardiogram	3.955 ± 2.609	$\mathbf{1.314 \pm 2.305}$	4.129 ± 2.610	3.576 ± 2.391	0.982 ± 1.301
Spectf	6.022 ± 6.451	$\mathbf{2.974 \pm 5.514}$	12.52 ± 8.384	$\mathbf{2.873 \pm 2.533}$	0.940 ± 1.056
Australian	23.63 ± 19.06	25.01 ± 27.15	24.80 ± 19.00	$\mathbf{15.98 \pm 11.86}$	3.263 ± 3.191
Clean1	17.96 ± 13.44	20.63 ± 14.88	21.97 ± 14.26	$\mathbf{13.47 \pm 9.942}$	3.618 ± 2.631
Diabetes	$\mathbf{5.772 \pm 10.84}$	$\mathbf{6.162 \pm 14.11}$	32.08 ± 19.30	10.01 ± 8.946	0.249 ± 0.007
German Credit	30.17 ± 22.28	30.54 ± 26.16	26.48 ± 18.83	$\mathbf{18.63 \pm 13.30}$	6.686 ± 5.029
House Votes	8.594 ± 7.187	9.693 ± 8.515	12.50 ± 8.551	$\mathbf{6.206 \pm 4.644}$	1.804 ± 1.694
Krvskp	144.9 ± 87.03	131.5 ± 81.30	158.0 ± 90.43	$\mathbf{92.42 \pm 52.09}$	1.865 ± 1.789
Ethn	$\mathbf{9.919 \pm 16.25}$	119.3 ± 85.15	74.90 ± 64.07	16.14 ± 11.84	1.131 ± 0.973
Heart	0.615 ± 1.188	1.962 ± 6.346	6.908 ± 4.770	$\mathbf{0.127 \pm 0.205}$	0.472 ± 0.852
Texture	4.094 ± 6.755	5.748 ± 6.489	2.512 ± 4.668	$\mathbf{0.045 \pm 0.205}$	0.000 ± 0.000
House	1.760 ± 1.505	1.325 ± 1.415	1.458 ± 1.479	$\mathbf{0.935 \pm 1.061}$	0.465 ± 0.618
Isolet	4.976 ± 4.218	7.207 ± 6.382	0.943 ± 1.394	$\mathbf{0.420 \pm 0.670}$	0.198 ± 0.368
Optdigits	6.642 ± 6.881	4.025 ± 4.177	$\mathbf{1.097 \pm 1.951}$	$\mathbf{0.773 \pm 1.197}$	0.222 ± 0.538
Vehicle	1.978 ± 3.812	18.70 ± 26.50	7.191 ± 7.800	$\mathbf{1.002 \pm 1.667}$	0.378 ± 0.432
Wdbc	$\mathbf{0.127 \pm 0.125}$	32.92 ± 38.52	11.33 ± 8.367	0.264 ± 0.415	0.251 ± 0.479
Sat	$\mathbf{3.404 \pm 7.363}$	6.968 ± 10.01	$\mathbf{2.122 \pm 9.839}$	$\mathbf{2.521 \pm 9.407}$	0.646 ± 0.729
CS4VM: W/T/L	14/2/4	16/1/3	17/3/0	-	

misclassification cost is given a larger weight in margin computation. Moreover, unlike the formulation of Eq. 3, the optimization problem is now much easier since there are only two constraints corresponding to the unlabeled data, and the misclassification costs do not couple with the signs of the predictions. Indeed, as in (Li, Kwok, and Zhou 2009), it can be solved by an iterative procedure that alternates between two steps. We first fix $\mathbf{d}$ and solve for $\{\mathbf{w}, b, \rho\}$ via standard SVM training, and then fix $\{\mathbf{w}, b, \rho\}$ and solve for $\mathbf{d}$ via a linear program.

Solving Eq. 6 with Estimated Label Means After obtaining $\mathbf{d}$, the label means can be estimated as $\mathbf{m}_+ = \frac{1}{u_+}\sum_{i\in I_u} d_i\phi(\mathbf{x}_i)$ and $\mathbf{m}_- = \frac{1}{u_-}\sum_{i\in I_u}(1-d_i)\phi(\mathbf{x}_i)$. Note that in Eq. 6, the constraints are linear and the objective is convex, and thus it is a convex optimization problem. By introducing Lagrange multipliers $\boldsymbol{\alpha} = [\alpha_i; i \in \mathcal{I}_l]$ and $\boldsymbol{\beta}^{\pm} = [\beta_i^{\pm}; i \in \mathcal{I}_u]$ for the constraints in Eq. 6, its dual can be written as

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}^{\pm}} \sum_{i\in\mathcal{I}_l}\alpha_i - \sum_{i\in\mathcal{I}_u}(c(1)\beta_i^+ + c(-1)\beta_i^-) - \frac{1}{2}\|\mathbf{w}\|^2$$

$$\text{s.t.} \sum_{i\in\mathcal{I}_l}\alpha_i y_i - \sum_{i\in I_u}(c(1)\beta_i^+ - c(-1)\beta_i^-) = 0, \qquad (9)$$

$$0 \le \alpha_i \le c(y_i)C_1, \forall i \in \mathcal{I}_l; 0 \le \beta_i^{\pm} \le C_2, \forall i \in \mathcal{I}_u\ ,$$

where $\mathbf{w} = u_+c(1)\mathbf{m}_+ - u_-c(-1)\mathbf{m}_- + \sum_{i\in\mathcal{I}_l}\alpha_i y_i\mathbf{x}_i + \sum_{i\in\mathcal{I}_u}(-c(1)\beta_i^+ + c(-1)\beta_i^-)\mathbf{x}_i$. Eq. 9 is a convex quadratic program (QP) with one linear equality constraint. This is similar to the dual of standard SVM and can be efficiently handled by state-of-the-art SVM solvers, like LIBSVM using the SMO algorithm.

Experiments

In this section, we empirically evaluate the performance of the proposed CS4VM. A collection of twenty UCI data sets are used in the experiments. Each data set is split into two equal halves, one for training and the other for testing. Each training set contains ten labeled examples. The linear kernel is always used. Moreover, since there are too few labeled examples for reliable model selection, all the experiments are performed with fixed parameters: C_1 and C_2 are fixed at 1 and 0.1, respectively, while u_+ and u_- are obtained as the ratios of positive and negative examples in the labeled training data.

Two different cost setups are considered:

1. $c(-1)$ is fixed at 1 while $c(1)$ is chosen randomly from a uniform distribution on the interval $[0, 1000]$. Each experiment is repeated for 100 times and then the average results are reported. From this series of experiments we can see how an approach is robust to different costs.
2. $c(-1)$ is fixed at 1 while $c(1)$ is set to 2, 5 and 10, respectively. For each value of $c(1)$, the experiment is repeated for 30 times and then the average results are reported. From this series of experiments we can see how the performance of an approach changes as the cost varies.

We compare CS4VM with the following approaches: 1) A supervised CS-SVM using only the labeled training examples; 2) A supervised CS-SVM (denoted by GT CS-SVM) using the labeled training examples and all the unlabeled data with ground-truth labels; 3) Two state-of-the-art semi-supervised learning methods, that is, the Laplacian SVM (Belkin, Niyogi, and Sindhwani 2006) and TSVM (Joachims 1999). These two methods are cost-blind, and in the experiments we extend them for cost-sensitive learning by incorporating the misclassification cost for each labeled training example as in the CS-SVM, while the costs of the unlabeled data are difficult to incorporate. All the compared approaches are implemented in MATLAB 7.6, and experiments are run on a 2GHz Xeon©2 Duo PC with 4GB mem-

Table 2: Comparison of total costs (mean $\pm$ std.) in the second series of experiments ($c(1)$ is set to different fixed values). The best performance (paired t-tests at 95% significance level) and its comparable results are bolded. The last line shows the win/tie/loss counts of CS4VM versus other methods.

Cost ratio	Data set	Supervised CS-SVM	Laplacian SVM	TSVM	CS4VM	GT CS-SVM
2	House votes	**39.76 ± 10.90**	47.47 ± 20.77	58.04 ± 15.41	**37.96 ± 11.63**	21.80 ± 4.160
	Clean1	**131.6 ± 21.08**	141.2 ± 20.70	144.7 ± 23.35	**132.1 ± 21.43**	57.56 ± 7.620
	Australian	**136.0 ± 44.93**	170.4 ± 63.79	**132.3 ± 53.56**	**120.8 ± 40.37**	82.63 ± 9.752
	German Credit	**276.7 ± 27.39**	286.4 ± 31.74	**268.2 ± 45.57**	**275.0 ± 28.82**	196.6 ± 8.840
	Krvskp	**848.4 ± 172.6**	856.3 ± 141.2	936.3 ± 188.7	**794.7 ± 143.8**	59.10 ± 8.601
	Heart-statlog	**54.23 ± 12.36**	**59.33 ± 11.71**	**54.96 ± 14.57**	**54.43 ± 13.21**	33.86 ± 5.912
	Diabetes	236.3 ± 30.57	241.9 ± 17.60	220.4 ± 45.40	**196.4 ± 43.38**	136.3 ± 8.844
	Ionosphere	**79.96 ± 23.11**	112.5 ± 14.76	**71.40 ± 23.47**	**68.40 ± 23.79**	39.36 ± 6.262
	Liver Disorders	**99.83 ± 4.653**	114.0 ± 21.31	114.9 ± 14.81	105.2 ± 11.56	96.86 ± 5.811
	Echocardiogram	23.76 ± 4.513	44.76 ± 11.69	**20.90 ± 4.932**	23.86 ± 4.383	21.66 ± 5.282
	Spectf	**88.90 ± 14.82**	104.0 ± 11.97	**82.10 ± 16.56**	**86.03 ± 16.12**	50.06 ± 6.103
	Heart	67.46 ± 12.58	69.10 ± 12.27	**48.69 ± 14.30**	**49.83 ± 12.08**	34.60 ± 4.432
	House	**14.06 ± 4.945**	**12.98 ± 6.523**	**12.78 ± 5.772**	**12.06 ± 5.782**	5.102 ± 1.931
	Wdbc	102.6 ± 8.082	139.6 ± 101.5	**57.65 ± 14.37**	**52.13 ± 10.24**	20.13 ± 4.522
	Isolet	34.10 ± 17.07	37.64 ± 18.77	**4.322 ± 4.373**	**5.000 ± 4.202**	1.864 ± 1.473
	Optdigits	44.30 ± 22.93	29.64 ± 16.52	**6.592 ± 7.123**	14.96 ± 15.13	1.900 ± 1.903
	Texture	15.76 ± 17.61	23.81 ± 18.96	9.582 ± 13.15	**0.262 ± 0.980**	0.000 ± 0.000
	Vehicle	78.00 ± 17.85	96.45 ± 51.04	**47.43 ± 26.61**	**53.00 ± 18.07**	12.86 ± 3.414
	Ethn	654.9 ± 44.58	612.3 ± 129.5	**525.1 ± 299.4**	**499.3 ± 105.0**	64.23 ± 8.137
	Sat	44.23 ± 75.91	30.99 ± 33.22	**12.85 ± 36.43**	**32.83 ± 76.55**	4.363 ± 2.203
CS4VM: W/T/L		8/11/1	15/5/0	6/14/0	-	
5	House-votes	**85.00 ± 28.66**	**90.46 ± 49.69**	127.2 ± 34.13	**72.03 ± 25.43**	34.60 ± 9.742
	Clean1	**231.4 ± 56.70**	257.8 ± 46.66	269.5 ± 53.75	**209.6 ± 41.24**	83.13 ± 12.39
	Australian	282.3 ± 106.6	292.4 ± 133.4	**251.8 ± 86.41**	**231.0 ± 76.42**	117.9 ± 15.86
	German Credit	464.4 ± 76.69	488.4 ± 120.6	462.9 ± 85.49	**406.4 ± 63.33**	294.4 ± 21.13
	Krvskp	1717. ± 481.4	1714. ± 423.4	1757. ± 345.2	**1364. ± 282.2**	73.23 ± 15.83
	Heart-statlog	112.0 ± 31.17	**67.33 ± 17.26**	115.3 ± 32.11	96.26 ± 28.45	59.50 ± 11.25
	Diabetes	**264.0 ± 49.27**	313.9 ± 119.2	431.0 ± 115.4	338.9 ± 108.9	178.0 ± 10.94
	Ionosphere	176.7 ± 62.76	277.3 ± 35.80	151.0 ± 50.13	**128.3 ± 32.24**	82.72 ± 12.64
	Liver Disorders	**100.3 ± 5.472**	184.0 ± 116.6	220.0 ± 28.21	**105.6 ± 16.21**	99.60 ± 4.432
	Echocardiogram	**51.26 ± 13.85**	50.76 ± 8.492	**45.26 ± 11.18**	**49.90 ± 13.58**	31.90 ± 6.213
	Spectf	**112.7 ± 40.97**	122.9 ± 27.43	152.5 ± 37.51	**104.4 ± 32.86**	62.10 ± 11.18
	Heart	**73.00 ± 10.77**	80.77 ± 32.86	92.24 ± 33.03	**68.60 ± 7.612**	54.43 ± 9.004
	House	**24.66 ± 11.89**	**20.92 ± 13.31**	**20.77 ± 11.98**	**19.26 ± 10.81**	8.463 ± 4.025
	Wdbc	103.1 ± 8.332	323.0 ± 273.2	123.5 ± 38.57	**70.96 ± 12.59**	27.93 ± 5.154
	Isolet	73.00 ± 45.70	83.81 ± 47.96	**9.612 ± 10.06**	16.23 ± 15.03	3.462 ± 3.334
	Optdigits	81.52 ± 39.44	51.66 ± 27.75	**13.22 ± 15.80**	28.90 ± 25.11	3.264 ± 4.362
	Texture	34.96 ± 44.35	59.42 ± 47.51	23.37 ± 33.06	**1.000 ± 3.472**	0.000 ± 0.000
	Vehicle	89.33 ± 21.33	194.2 ± 151.8	**92.49 ± 65.53**	**74.56 ± 23.29**	19.36 ± 3.743
	Ethn	**716.4 ± 116.8**	1331. ± 394.2	964.4 ± 615.7	**717.7 ± 142.2**	82.33 ± 12.52
	Sat	61.83 ± 95.10	72.00 ± 84.99	**22.97 ± 72.02**	**59.36 ± 131.9**	9.736 ± 5.952
CS4VM: W/T/L		10/9/1	14/5/1	13/5/2	-	
10	House-votes	159.0 ± 58.43	175.5 ± 98.15	248.6 ± 67.70	**127.7 ± 45.23**	49.50 ± 13.52
	Clean1	397.8 ± 120.5	454.3 ± 97.13	479.2 ± 101.1	**340.3 ± 76.87**	116.2 ± 26.84
	Australian	526.2 ± 213.0	519.6 ± 278.7	478.4 ± 159.7	**395.3 ± 128.7**	158.0 ± 34.83
	German Credit	777.2 ± 174.6	818.4 ± 314.5	767.6 ± 157.4	**600.7 ± 119.3**	389.3 ± 56.62
	Krvskp	3167. ± 1006.	3096. ± 899.9	3193. ± 628.5	**2264. ± 524.2**	88.66 ± 23.40
	Heart-statlog	207.9 ± 63.73	**80.66 ± 33.70**	214.3 ± 63.80	160.5 ± 55.56	76.20 ± 22.79
	Diabetes	**314.8 ± 120.5**	**433.9 ± 319.3**	776.2 ± 210.0	489.4 ± 181.0	247.2 ± 12.74
	Ionosphere	340.9 ± 130.5	552.0 ± 72.84	284.0 ± 93.76	**219.2 ± 57.97**	130.7 ± 30.01
	Liver Disorders	**101.2 ± 7.602**	300.7 ± 276.4	363.6 ± 60.11	**109.5 ± 23.97**	99.60 ± 4.432
	Echocardiogram	87.30 ± 28.80	**60.76 ± 21.91**	84.36 ± 25.79	83.96 ± 27.76	48.70 ± 12.06
	Spectf	181.2 ± 92.48	**154.4 ± 75.98**	280.4 ± 72.72	**130.0 ± 40.54**	67.50 ± 13.84
	Heart	**79.00 ± 19.27**	**91.66 ± 46.03**	147.6 ± 52.63	**75.53 ± 6.982**	71.53 ± 8.523
	House	42.33 ± 24.38	**32.46 ± 30.19**	**35.26 ± 17.56**	**29.20 ± 15.87**	13.13 ± 8.062
	Wdbc	103.1 ± 8.332	627.2 ± 530.2	208.0 ± 75.69	**80.23 ± 12.58**	34.93 ± 9.226
	Isolet	137.8 ± 94.81	176.1 ± 96.36	**20.83 ± 24.49**	**32.56 ± 21.77**	6.302 ± 6.693
	Optdigits	143.5 ± 77.71	88.03 ± 48.97	**28.62 ± 40.95**	**44.73 ± 30.55**	5.464 ± 8.763
	Texture	66.96 ± 90.05	109.7 ± 99.07	28.53 ± 37.92	**2.402 ± 6.093**	0.000 ± 0.000
	Vehicle	108.0 ± 39.93	347.0 ± 345.6	178.7 ± 122.1	**88.10 ± 33.03**	26.83 ± 5.722
	Ethn	**829.1 ± 327.2**	2580. ± 828.5	1841. ± 1125.	**925.0 ± 183.7**	102.8 ± 16.59
	Sat	91.16 ± 136.9	140.8 ± 173.7	**20.06 ± 11.55**	97.40 ± 205.2	17.86 ± 11.71
CS4VM: W/T/L		14/5/1	13/5/2	15/4/1	-	

ory and Vista system.

Table 1 summarizes results of the first series of experiments. It can be seen that CS4VM usually outperforms the cost-sensitive extensions of Laplacian SVM and TSVM. Specifically, Paired t-tests at 95% significance level show that CS4VM achieves 16 wins, 1 tie and 3 losses when compared to Laplacian SVM; and 17 wins, 3 ties and 0 loss when compared to TSVM. Wilcoxon sign tests at 95% significance level show that CS4VM is always significantly better than all the other three approaches, while Laplacian SVM

Table 3: Running time in seconds (n is the size of data).

(Data, n)	Laplacian SVM	TSVM	CS4VM
(Heart,270)	0.13 ± 0.23	1.44 ± 0.04	$\mathbf{0.09 \pm 0.04}$
(Wdbc,569)	0.32 ± 0.34	4.69 ± 0.07	$\mathbf{0.20 \pm 0.03}$
(Australian,690)	0.26 ± 0.31	4.27 ± 0.09	$\mathbf{0.12 \pm 0.04}$
(Optdigits,1143)	0.49 ± 0.37	28.39 ± 0.08	$\mathbf{0.18 \pm 0.05}$
(Ethn,2630)	3.16 ± 0.76	46.42 ± 0.44	$\mathbf{0.66 \pm 0.04}$
(Sat,3041)	4.50 ± 1.05	63.73 ± 0.34	$\mathbf{1.01 \pm 0.06}$
(Krvskp,3196)	5.92 ± 1.11	11.76 ± 0.11	$\mathbf{1.01 \pm 0.05}$

and TSVM are not significantly better than the supervised CS-SVM. Moreover, CS4VM often benefits from the use of unlabeled data (on 14 out of the 20 data sets), while Laplacian SVM yields performance improvements on only five data sets and TSVM improves on only three data sets. Even for the few data sets (such as Wdbc) on which the unlabeled data do not help (possibly because the cluster assumption and manifold assumption do not hold), CS4VM does not increase the cost relative to the supervised CS-SVM by three times, while the cost-sensitive versions of Laplacian SVM and TSVM may increase the cost by more than 10 times.

Table 2 summarizes results of the second series of experiments. Similar to results in the first series of experiments, CS4VM usually outperforms the other approaches, and the cost-sensitive extensions of Laplacian SVM and TSVM are not effective in cost reduction. As the cost ratio increases, the advantage of CS4VM becomes more prominent.

Table 3 compares the running time costs of CS4VM and the cost-sensitive extensions of Laplacian SVM and TSVM. Results are averaged over all the settings reported in Tables 1 and 2. Due to the page limit, only results on seven representative data sets are shown. As can be seen, CS4VM is more efficient than the compared approaches.

Conclusion

In this paper, we propose CS4VM (Cost-Sensitive Semi-Supervised Support Vector Machine) which considers unequal misclassification costs and the utilization of unlabeled data simultaneously. This is a cost-sensitive extension of the approach in (Li, Kwok, and Zhou 2009) where an efficient algorithm for exploiting unlabeled data in support vector machine is developed based on the estimation of label means of unlabeled data. Experiments on a broad range of data sets show that CS4VM has encouraging performance, in terms of both the cost reduction and computational efficiency. The current work focuses on two-class problems. Extending CS4VM to multi-class scenario and other types of costs are interesting future issues.

References

Belkin, M.; Niyogi, P.; and Sindhwani, V. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.* 7:2399–2434.

Bennett, K., and Demiriz, A. 1999. Semi-supervised support vector machines. In *Adv. Neural Infor. Process. Syst. 11*. 368–374.

Brefeld, U.; Geibel, P.; and Wysotzki, F. 2003. Support vector machines with example dependent costs. In *Proc. 14th Eur. Conf. Mach. Learn.*, 23–34.

Chapelle, O., and Zien, A. 2005. Semi-supervised learning by low density separation. In *Proc. 10th Int'l Workshop AI and Stat.*, 57–64.

Chapelle, O.; Schölkopf, B.; and Zien, A., eds. 2006. *Semi-Supervised Learning*. Cambridge, MA: MIT Press.

Chapelle, O.; Sindhwani, V.; and Keerthi, S. S. 2008. Optimization techniques for semi-supervised support vector machines. *J. Mach. Learn. Res.* 9:203–233.

Collobert, R.; Sinz, F.; Weston, J.; and Bottou, L. 2006. Large scale transductive SVMs. *J. Mach. Learn. Res.* 7:1687–1712.

Domingos, P. 1999. MetaCost: A general method for making classifiers cost-sensitive. In *Proc. 5th ACM SIGKDD Int'l Conf. Knowl. Disc. Data Min.*, 155–164.

Elkan, C. 2001. The foundations of cost-sensitive learning. In *Proc. 17th Int'l Joint Conf. Artif. Intell.*, 973–978.

Fan, W.; Stolfo, S. J.; Zhang, J.; and Chan, P. K. 1999. AdaCost: Misclassification cost-sensitive boosting. In *Proc. 16th Int'l Conf. Mach. Learn.*, 97–104.

Greiner, R.; Grove, A. J.; and Roth, D. 2002. Learning cost-sensitive active classifiers. *Artif. Intell.* 139(2):137–174.

Gretton, A.; Borgwardt, K. M.; Rasch, M.; Schölkopf, B.; and Smola, A. J. 2006. A kernel method for the two-sample-problem. In *Adv. Neural Infor. Process. Syst. 19*. 513–520.

Joachims, T. 1999. Transductive inference for text classification using support vector machines. In *Proc. 16th Int'l Conf. Mach. Learn.*, 200–209.

Kukar, M., and Kononenko, I. 1998. Cost-sensitive learning with neural networks. In *Proc. 13th Eur. Conf. Artif. Intell.*, 445–449.

Lee, Y.; Lin, Y.; and Wahba, G. 2004. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. *J. Ame. Stat. Assoc.* 99(465):67–82.

Li, Y.-F.; Kwok, J. T.; and Zhou, Z.-H. 2009. Semi-supervised learning using label mean. In *Proc. 26th Int'l Conf. Mach. Learn.*, 633–640.

Liu, A.; Jun, G.; and Ghosh, J. 2009. Spatially cost-sensitive active learning. In *Proc. 9th SIAM Int'l Conf. Data Min.*, 814–825.

Lozano, A. C., and Abe, N. 2008. Multi-class cost-sensitive boosting with p-norm loss functions. In *Proc. 14th ACM SIGKDD Int'l Conf. Knowl. Disc. Data Min.*, 506–514.

Margineantu, D. D. 2005. Active cost-sensitive learning. In *Proc. 19th Int'l Joint Conf. Artif. Intell.*, 1622–1623.

Masnadi-Shirazi, H., and Vasconcelos, N. 2007. Asymmetric boosting. In *Proc. 24th Int'l Conf. Mach. Learn.*, 619–626.

Morik, K.; Brockhausen, P.; and Joachims, T. 1999. Combining statistical learning with a knowledge-based approach: A case study in intensive care monitoring. In *Proc. 16th Int'l Conf. Mach. Learn.*, 268–277.

Qin, Z.; Zhang, S.; Liu, L.; and Wang, T. 2008. Cost-sensitive semi-supervised classification using CS-EM. In *Proc. 8th Int'l Conf. Comput. Infor. Tech.*, 131–136.

Ting, K. M. 2002. An instance-weighting method to induce cost-sensitive trees. *IEEE Trans Knowl. Data Eng.* 14(3):659–665.

Turney, P. 2000. Types of cost in inductive concept learning. In *ICML Workshop Cost-Sensitive Learning*, 15–21.

Zadrozny, B.; Langford, J.; and Abe, N. 2003. Cost-sensitive learning by cost-proportionate example weighting. In *Proc. 3rd IEEE Int'l Conf. Data Min.*, 435–442.

Zhou, Z.-H., and Li, M. 2010. Semi-supervised learning by disagreement. *Knowl. Infor. Syst.*

Zhou, Z.-H., and Liu, X.-Y. 2006a. On multi-class cost-sensitive learning. In *Proc. 21st Nat'l Conf. Artif. Intell.*, 567–572.

Zhou, Z.-H., and Liu, X.-Y. 2006b. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Trans Knowl. Data Eng.* 18(1):63–77.

Zhu, X. 2007. Semi-supervised learning literature survey. Technical report, Department of Computer Science, University of Wisconsin-Madison.