

# Adjusted Support Vector Machines Based on A New Loss Function

Shuchun Wang<sup>†</sup>, Wei Jiang<sup>‡</sup>, Kwok-Leung Tsui<sup>†</sup>

<sup>†</sup>*School of Industrial and Systems Engineering  
Georgia Institute of Technology, Atlanta, GA*

<sup>‡</sup>*Department of Systems Engineering and Engineering Management,  
Stevens Institute of Technology, Hoboken, NJ*

September 1, 2006

Support vector machine (SVM) has attracted considerable attentions recently due to its successful applications in various domains. However, by maximizing the margin of separation between the two classes in a binary classification problem, the SVM solution often suffers two serious drawbacks. First, SVM separating hyperplane is usually very sensitive to training samples since it strongly depends on support vectors which are *only* a few points located near the boundary. Second, the separating hyperplane is equidistant to the two classes which are considered equally important when optimizing the separating hyperplane location regardless the number of training data points and their dispersion in each class. In this paper, we propose a new SVM solution, adjusted support vector machine (ASVM), based on a new loss function to adjust the SVM solution taking into account the sample sizes and dispersions of the two classes. Numerical experiments show that the ASVM outperforms the conventional SVM, especially when the two classes have large differences in sample size and dispersion.

*Key words:* Classification Error; Cross Validation; Dispersion; Sampling Bias

## 1 Introduction

Support Vector Machine (SVM) has attracted considerable attentions due to its successful applications for learning classification and regression rules in various domains. Take a two-class classification problem as an example. Given the training data  $\{x_i, y_i, i = 1, 2, \dots, n\}$  with  $x_i \in \mathbb{R}^m$  and  $y_i \in \{-1, +1\}$ , a classification rule can be characterized by a separating hyperplane  $f(x)$  such that a new observation  $x$  is classified as  $+1$  if  $f(x) > 0$  and  $-1$  if  $f(x) < 0$ . SVM searches for the optimal separating hyperplane, i.e. the hyperplane  $f(x) = x^T w + b$  that is, in a sense, equidistant to the two classes and maximizes the margin of separation between classes  $-1$  and  $+1$  (Burges 1998). Mathematically, this can be achieved by solving the following optimization problem

$$\min_{w,b} \frac{1}{2} \|w\|^2 + C_1 \cdot \sum_{i=1}^n [1 - y_i(x_i^T w + b)]_+, \quad (1)$$

where  $C_1 > 0$  is a penalty parameter, and  $[z]_+$  equals to  $z$  when  $z > 0$  and 0 otherwise. Let  $w^*$  and  $b^*$  be the optimal solution. Then, the plane represented by  $x^T w^* + b^* = 0$  is the maximum-margin separating hyperplane, and  $x^T w^* + b^* = -1$  and  $x^T w^* + b^* = +1$  are the margin boundaries for classes  $-1$  and  $+1$  respectively.

It is interesting to note that, when there exist mis-classifications, SVM uses a loss function of the form

$$L(yf(x)) = [1 - yf(x)]_+ = \begin{cases} 1 - yf(x) & \text{if } yf(x) < 1 \\ 0 & \text{if } yf(x) \geq 1 \end{cases} \quad (2)$$

to penalize only points that fall on the wrong side of the corresponding boundaries and the penalties are proportional to the distances between the mis-classified points and the corresponding boundaries. This loss function is often referred to as the “hinge loss” and the points on the wrong side of their corresponding boundaries are referred to as “support vectors,” therefore, the name “support vector machine.”

However, there are several critical concerns with the conventional SVM. First, the fact that the hinge loss in (2) penalizes *only* support vectors leaves the SVM solution depending strongly on those support vectors. As support vectors are often just a few points located near the boundary, it is very likely that they vary dramatically with different training samples and make the SVM solution unstable in practice, especially when the sample size is small. Second, since the SVM solution (i.e., the maximum-margin separating hyperplane) takes an equal distance from the two boundaries of the margin regardless the number of training points and the dispersion of training points in each class. In many cases, this may result in very poor performance in situations when there is sampling bias as illustrated in Figure 1, where the conventional SVM solution is located to the right of the Bayes solution and may generalize to poor overall performance.

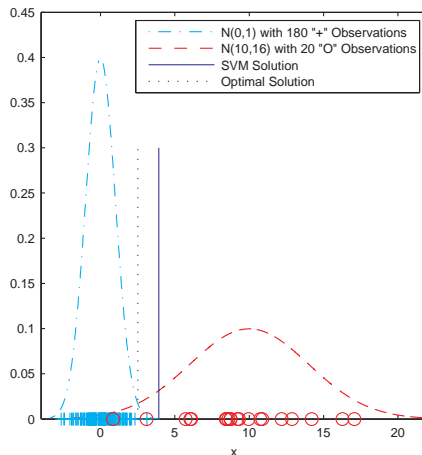


Figure 1: A One-Dimensional Example

This paper proposes a modification of the conventional SVM based on a new loss function that implicitly considers both the number of training points and their dispersion by penalizing all training data. Using this new loss function, we expect to adjust the separating hyperplane toward the class with a larger number of and less dispersed training observations (e.g., class “+” in Figure 1) and consequently achieve good generalization performance for both classes. We prove that the solution based on the new loss asymptotically approaches the Bayes solution. To simplify the implementation of the new SVM, we further develop an adjustment of the conventional SVM, namely Adjusted Support Vector Machine (ASVM), which is shown to have better generalization performance with a negligible increase in computational time compared to SVM.

The rest of this paper is organized as follows. Section 2 introduces the new loss function and develops the asymptotic behavior of SVM based on the new loss function. Section 3 develops the adjusted SVM based on the solution of the conventional SVM. Section 4 presents several numerical experiments to compare the generalization performance of ASVM and the conventional SVM. Section 5 concludes the paper and highlights several future research issues.

## 2 A New Loss Function for SVM

A number of loss functions have been proposed for the variants of SVM to improve the generalization performance, to increase the speed of optimization problem solving, or to account for certain nonstandard situations (Lee and Mangasarian 2001, Shen et al. 2003, Lin et al. 2002). All of the above loss functions penalize only points with  $yf(x) < 1$ . As a result, they all have the same problems as SVM, namely, they are sensitive to training samples and perform poorly under situations as illustrated in Figure 1. To robustify the SVM solution, we propose a new loss function that penalizes all training points instead of just a few support vectors. The new loss function has a form

$$L(yf(x)) = \begin{cases} 1 - yf(x), & \text{if } yf(x) < 0, \\ 1/(1 + yf(x)), & \text{if } yf(x) \geq 0 \end{cases} \quad (3)$$

to penalize all training points according to the sign of  $yf(x)$ . It is apparent that the new loss function looks similar to the hinge loss but is smooth and strictly monotone decreasing with respect to  $yf(x)$ . For  $yf(x) < 0$ , the two losses are equal; and for  $yf(x) \geq 0$ , the new loss is slightly greater than the hinge loss.

Under the new loss, the optimization problem in (1) becomes

$$\min_{w,b} \frac{1}{2} \|w\|^2 + C_1 \cdot \sum_{i=1}^n (1 - y_i f(x_i)) \cdot I\{y_i f(x_i) < 0\} + \frac{1}{1 + y_i f(x_i)} \cdot I\{y_i f(x_i) \geq 0\} \quad (4)$$

where  $I\{\cdot\}$  is an indicator function. As the new loss penalizes all training points, it is expected to be robust against the ‘‘support vectors’’ while keeping the sensitivity of the conventional SVM in classification. In fact, it can be proved that, under certain regularity conditions (Vapnik 1998), the solution of the optimization problem in (4) approaches to the Bayes classifier as  $n$  approaches infinity. Due to space limit, the proof is omitted here.

## 3 Adjusted SVM

To utilize the new loss function while at the same time maintain the nice properties of the conventional SVM, we here propose a simple adjustment based on the conventional SVM. Denote the solution of the conventional SVM by  $(w^0, b^0)$ , the adjustment optimizes the following loss function regarding  $b$

$$\min_b \sum_{i=1}^n ((1 - d_i) \cdot I\{d_i < 0\} + C_2 \cdot \frac{1}{1 + d_i} \cdot I\{d_i \geq 0\}), \quad (5)$$

subject to

$$y_i(x_i^T w^0 + b) = d_i, \forall i, \quad (6a)$$

$$b^0 - 1 \leq b \leq b^0 \text{ or } b^0 \leq b \leq b^0 + 1. \quad (6b)$$

Which constraint,  $b^0 - 1 \leq b \leq b^0$  or  $b^0 \leq b \leq b^0 + 1$ , to use depends on at which side of the maximum-margin separating hyperplane the class with smaller variance is located. That is, ASVM adjusts the location of the maximum-margin separating hyperplane toward the class with smaller variance.

## 4 Experimental Comparisons

To compare the generalization errors of the proposed ASVM and conventional SVM, we now use a simple one-dimensional example to illustrate that ASVM gives a solution closer to the Bayes

classifier than SVM does, especially when the two classes have large differences in variation and the class with larger variance has a smaller number of training data points. In this example, the training data contain  $n_1$  class 1 observations randomly drawn from  $N(0, \sigma_1^2)$  and  $n_2$  class 2 observations randomly drawn from  $N(10, \sigma_2^2)$ . To investigate the combination effects of different parameters, we assume  $\sigma_1 = 1$  and vary  $\sigma_2$  from  $[1, 2, 3, \dots, 15]$ . The total sample size is assumed fixed,  $n_1 + n_2 = 200$ , with  $n_2 = [20, 30, \dots, 90, 100]$ . The tuning parameters  $C_1$  and  $C_2$  are chosen from  $[2^{-16}, 2^{-14}, \dots, 2^8, 2^{10}]$  and  $[2^{-16}, 2^{-14}, \dots, 2^2, 2^0]$  respectively using a hold-out sample of size 10000 (5000 from class 1 and 5000 from class 2). The testing sample contains 20000 observations with 10000 from class 1 and 10000 from class 2. For each  $\sigma_2$  and  $n_2$ , we run 40 simulations and calculate the average and standard deviation of the error rates.

Figure 2 shows their comparisons. In general, when the variances of the two classes are comparable, the performance of ASVM is very competitive to that of the conventional SVM. However, when there are considerable differences between the variances of the two classes, it is apparent that ASVM outperforms the conventional SVM no matter sampling bias exists or not. In general, the larger the difference between  $\sigma_1$  and  $\sigma_2$ , the better ASVM performs than the conventional SVM does, especially when sampling bias is severe. For example, if  $\sigma_2 = 15\sigma_1$ , ASVM outperforms the conventional SVM by 23% when  $n_1 = n_2 = 100$  (i.e., no sampling bias in the training sample) and by 34% when  $n_1 = 180$  and  $n_2 = 20$  (heavily biased training sample). It is interesting to note that, although the error rate of the conventional SVM increases from .21 to .24 if  $\sigma_2 = 15\sigma_1$  when the training sample becomes more biased (from  $n_1 = 100$  to  $n_1 = 180$ ), that of ASVM almost remains the same. This indicates a nice robust property of ASVM against sampling bias in practice.

It is important to point out that, the conventional SVM is much more unstable compared with the proposed ASVM if there is a considerable difference between the variances of the two classes. In the case that  $\sigma_2 = 15\sigma_1$ , the standard deviation of the classification error for ASVM is often less than 1/3 of that for the conventional SVM regardless the sampling bias. In fact, the latter (i.e., the conventional SVM) becomes more unstable when there is a sizeable sampling bias. For example, when  $n_1 = 180$  and  $n_2 = 20$ , the error rate of the conventional SVM very likely falls in the interval  $[.24 \pm .15]$ , while that of ASVM is about  $[.16 \pm .01]$  (3 standard deviations).

In summary, both the sampling bias and difference between the variabilities of the two classes have profound impacts on the performance of the conventional SVM classifier. The proposed ASVM classifier is much more efficient and robust against different nonstandard situations. Moreover, the proposed ASVM can also allow the kernel trick if necessary. Although only a one-dimensional example is illustrated here, it is not difficult to generalize the comparison to high-dimensional cases since ASVM is just a simple adjustment of the solution of the conventional SVM. Nonetheless, two real examples, **Wisconsin Breast Cancer** and **PIMA Indian Diabetes**, which are taken from UCI Machine Learning Repository, also show the benefits of the ASVM classifier for high-dimensional data in real life.

## 5 Conclusions and Future Research

Despite of many advantages of support vector machine in different domain applications, this paper identifies that the conventional SVM is unstable and non-efficient when the training sample is unbalanced and there are considerable differences between the variances of the two classes. By proposing a new loss function, this paper develops a new SVM which assigns losses to both mis- and correct-classified training observations. The new SVM is shown to approach to the Bayes classifier's efficiency asymptotically.

To implement the new SVM under the proposed loss function, this paper also proposes an adjusted SVM based on the solution of the conventional SVM. The adjustment consists of two steps. First, run the conventional SVM to find the maximum-margin separating hyperplane and the margin boundaries. Then, adjust the location of the separating hyperplane within the boundaries by solving an optimization problem based on the new loss function. This adjusted SVM

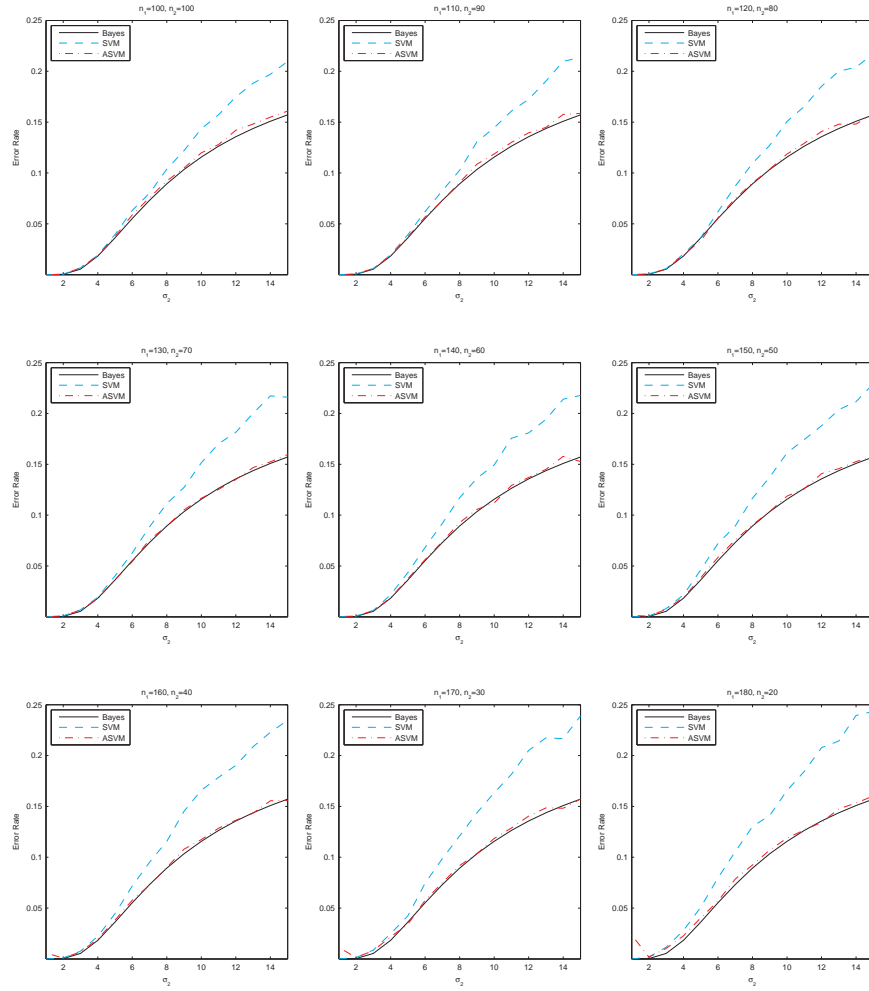


Figure 2: Error rate as a function of  $\sigma_2^2$  under different sampling bias

not only outperforms the conventional SVM, but also allows the kernel trick in implementation.

There are several research directions that need to be further pursued. First, note that, the proposed loss function is smooth which makes the first-order derivatives easy to compute. The explicit solution of (4) should be analyzed and compared with the ad hoc adjustment based on the conventional SVM. More importantly, generalization to nonlinear kernel functions needs to be investigated.

## Acknowledgement

This work is supported by National Science Foundation Grants #DMI-0200224 and #IIS-0542881.

## References

- [1] C. J. C. Burges (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Knowledge Discovery and Data Mining*, 2(2), 121-167.
- [2] Y. Lee and O. L. Mangasarian (2001). SSVM: Smooth Support Vector Machine for Classification. *Computational Optimization and Applications*, 20(1), 5-22.
- [3] Y. Lin (2002a). Support Vector Machines and The Bayes Rule in Classification. *Data Mining and Knowledge Discovery*, 6(3), 259-275.
- [4] Y. Lin (2002b). A Note on Margin-based Loss Function in Classification. *Technical Report*, Department of Statistics, University of Wisconsin.
- [5] Y. Lin, Y. Lee, and G. Wahba (2002) Support Vector Machines for Classification in Non-standard Situations. *Machine Learning*, 46, 191-202.
- [6] X. Shen, G.C. Tseng, X. Zhang, and W. H. Wong (2003). On  $\psi$ -Learning. *Journal of American Statistical Association*, 98, 724-734.
- [7] V. Vapnik (1998). *Statistical Learning Theory*. John Wiley & Sons.