# Posterior Probability Estimation using SVM with Noisy Targets

CS281b Final Project, Prof. M. Jordan

Shawn Chang
SID: 13736897
<shawnc@icsi.berkeley.edu>

14 May 2001

**Abstract**

Posterior probability estimation is desired in many practical applications but is not produced directly by standard SVM. Sigmoid-fitting is an effective way to convert SVM outputs into posterior probability estimates. In many real world problems such as speech recognition, training targets are often imperfect. This project demonstrates how patterns of distortion in SVM posterior estimates relate to the randomness in training targets and differences in data class prior distributions. A prior compensation scheme is derived to improve the quality of posterior estimates and shown to be effective on a binary classification problem.

## 1 Introduction

Support Vector Machine (SVM) has becoming increasingly popular in the machine learning community with many successful applications. However, standard SVM approach does not produce a posterior probability $P(class|input)$ estimate, which is often useful and required in many practical recognition applications. For example, in pattern recognition, we are interested in finding a model with maximum posterior probability for a given input observation, which leads to Bayes optimal decision based on equal loss assumption. Several methods have been proposed to modify the standard SVM to produce posterior probabilities, such as the regularized likelihood method by Wahba [4] and sigmoid-fitting method by Platt [1]. The first part of this project implements the sigmoid-fitting method and tests it on a simple binary classification problem.

In many real classification problems, the target labels are not perfect. For example, in speech phoneme recognition, frames of speech are labelled manually at phonetic level by human transcribers. However, although transcribers are highly trained linguistically, the ambiguous nature of phonetic identity and segmentation leads to a fair amount of arbitrariness in labelling especially for frames around transitions between different phones. It is not uncommon to see a 10 to 20 percent disagreement among transcribers on realistic speech corpora. Training classifiers with these imperfect targets certainly gives less than optimal performance and distorted posterior probability estimates. From past experience, it is interesting to notice that the pattern of distortion in posterior probability estimate is related to the prior distributions of different classes in training data. In the second part of this project, I will explore such distortion patterns and how to improve SVM posterior probability estimation with noisy training targets.

I will first describe the sigmoid-fitting method and experiment result in the next section. Section 3 will discuss what happens when randomness is introduced in training targets. Section 4 proposes a method for

improving SVM posterior probability estimation along with some experiment results. I will conclude with discussion in the last section.

# 2 Fitting-Sigmoid Method for SVM Posterior Estimation

In [1], Platt proposed a sigmoid-fitting method to post-process standard SVM output (unthresholded) into posterior probability estimates. Given $f(x)$ , the unthresholded output of an SVM, one fits a parametric sigmoid model to approximate posterior probability of a class:

$$P(y=1|f) = \frac{1}{1+exp(Af+B)} \tag{1}$$

where $A$ and $B$ are parameters to be determined. It was shown [1] that fitting sigmoid produces good posterior estimates while retains the sparseness of SVM solution. To fit the sigmoid, one can use a maximum likelihood method on a training set $(f_i, y_i)$. Since SVM target $y_i$ are $+1$ or $-1$, we first transform it by:

$$t_i = \frac{y_i+1}{2} \tag{2}$$

Then, parameters $A$ and $B$ are found by minimizing the negative log likelihood of the training data, which is a cross-entropy error function:

$$min - \sum_i t_i \log(p_i) + (1-t_i)\log(1-p_i) \tag{3}$$

where

$$p_i = \frac{1}{1+exp(Af_i+B)} \tag{4}$$

A robust method for solving this two-parameter optimization problem is to use a model-trust algorithm based on the Levenberg-Marquardt algorithm [2].

## 2.1 Example of SVM Posterior Probability Estimation

The sigmoid-fitting method is tested with a simple binary classification problem consisting of two Gaussian-distributed data classes. Figure 1 (left panel) shows the prior distribution of randomly generated data points with $\mu_1 = -1$, $\mu_2 = 1$ and $\sigma_1^2 = \sigma_2^2 = 0.7$, and priors of the two classes are 0.8 and 0.2. In this experiment, I used 1000 data points for training an SVM with Gaussian kernels using SVMTorch [3]. Another separate set of 1000 data points are used for fitting the sigmoid. All performance results and posterior estimation are based on a held-out test set of 1000 data points from the same distribution. Figure 1 (right panel) shows that the SVM posterior estimates are very accurate. Table 1 shows the confusion matrix of the classification based on the posterior probability estimates.

# 3 Introducing Randomness in Targets

In real applications, training targets are often imperfect. In this section, I analyze the SVM posterior probability estimation using sigmoid-fitting method with some randomness introduced into training targets.

| Ref/Target | class1 | class2 |
|---|---|---|
| class1 | 96.25% | 3.75% |
| class2 | 18.91% | 81.09% |

Table 1: Confusion matrix for binary classification with training with perfect targets.

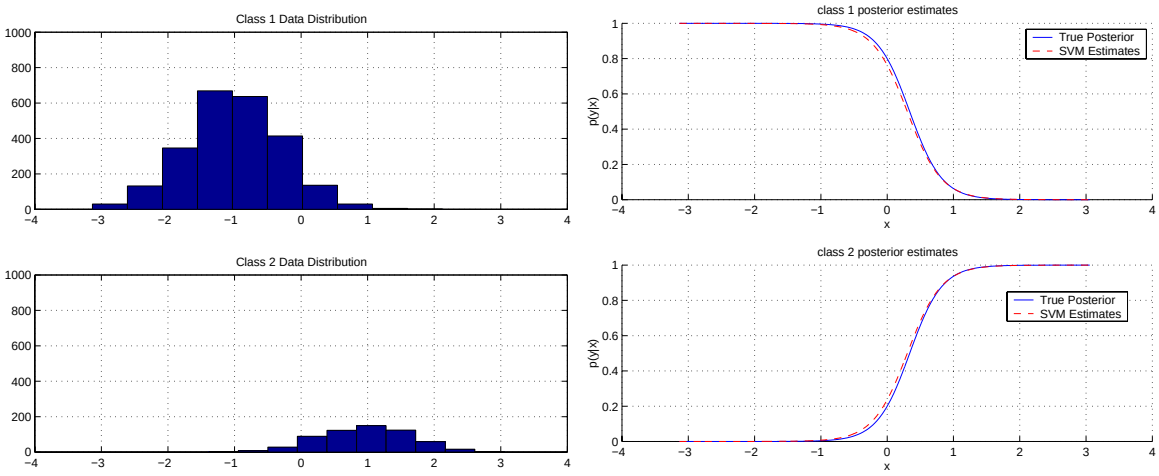


Figure 1: Left: Prior distribution of two classes are 0.8 and 0.2. Right: Posterior probability estimates using SVM output and sigmoid-fitting method.

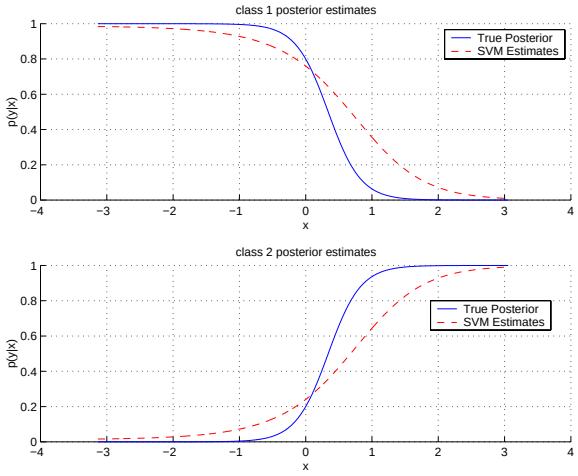


Figure 2: Posterior Probability Estimates using modified training data with $a = 0.8$.

Distribution of the data is modified as follows: let $a$ be the proportion of data points that have correct class labels; let $b = 1 - a$ be the proportion of data points whose class labels are randomly re-assigned according to the prior distribution of the two classes. Thus, the prior distribution of the modified data set is the same as before. Using the modified data, an SVM is trained and a sigmoid is fitted. Figure 2 gives posterior probability estimates using the modified training data with $a = 0.8$. Clearly, the posterior probability estimates are much worse than that using perfect data. More interestingly, notice that posterior estimates now are highly favoring the data class with larger prior. This makes sense intuitively: when data is less correlated with class labels (i.e. data contains less information for determining class labels), it is better off for the classifier to favor the class with larger prior. In the limit when all targets are completely random, a classifier should assign everything to the class with larger prior to achieve minimum classification error. Next, I will characterize the effect of prior differences quantitatively and propose a method for compensating for the prior differences.

# 4 Compensating for Prior Differences

Consider the problem of minimizing an expected cross-entropy function for estimating posterior probabilities

$$\Delta = -E\{t \log g(x) + (1-t) \log(1 - g(x))\} \tag{5}$$

where $t$ is a 0-or-1 target and $g(x)$ is some function of input $x$, which is in our case the output of posterior probability estimation. Now consider a new training set having proportion $a$ perfect targets and $b = 1 - a$ random targets as described in the previous section. Let $d$ be the correct training target and $r$ be the randomly assigned target according to prior distribution.

$$\Delta = -E\{a[d \log g(x) + (1-d) \log(1 - g(x))] + b[r \log g(x) + (1-r) \log(1 - g(x))]\} \tag{6}$$

$$= -\int p(x) d_x \sum_{k=0}^{1} \{a[d_k \log g(x) + (1-d_k) \log(1 - g(x))] + \tag{7}$$

| Ref/Target | class1 | class2 |
|---|---|---|
| class1 | 98.87% | 1.13% |
| class2 | 36.32% | 63.68% |

| Ref/Target | class1 | class2 |
|---|---|---|
| class1 | 96.25% | 1.50% |
| class2 | 28.36% | 71.64% |

Table 2: Confusion matrices for binary classification with modified targets at $a = 0.8$. Left: before prior compensation, Right: after prior compensation.

$$
\begin{aligned}
& b[r_k \log(g(x)) + (1 - r_k) \log(1 - g(x))\} p(q_k|x) && (8)\\
= & -\int p(x) d_x \{ap(q|x) \log g(x) + (1 - p(q|x)) \log(1 - g(x))] + b[r \log g(x) + (1 - r) \log(1 - g(x))]\} && (9)\\
= & -\int p(x) d_x \{ap(q|x) \log g(x) + a(1 - p(q|x)) \log(1 - g(x)) + br \log g(x) + b(1 - r) \log(1 - g(x)) && (10)\\
& +[(ap(q|x) + br) \log p(q|x) - (ap(q|x) + br) \log p(q|x)] && (11)\\
& +[a(1 - p(q|x)) + b(1 - r)] \log(1 - p(q|x)) - [a(1 - p(q|x)) + b(1 - r)] \log(1 - p(q|x))]\} && (12)\\
= & -\int p(x) d_x \{(ap(q|x) + br) \log \frac{g(x)}{p(q|x)} + [a(1 - p(q|x)) + b(1 - r)] \log \frac{1 - g(x)}{1 - p(q|x)}\} + Constant && (13)
\end{aligned}
$$

Take the first derivative of the expression in the integrand in (13) with respect to $g$ and set to 0. We get:

$$
\begin{aligned}
& p(q|x) \frac{p(q|x)}{g(x)} \frac{1}{p(q|x)} + 1 - p(q|x) \frac{1 - p(q|x)}{1 - g(x)} \frac{-1}{1 - p(q|x)} = 0 && (14)\\
\Rightarrow & \quad \frac{ap(q|x) + br}{g(x)} = \frac{a - ap(q|x) + b - br}{1 - g(x)} && (15)\\
\Rightarrow & \quad g(x) = ap(q|x) + br && (16)\\
\Rightarrow & \quad Eg(x) = ap(q|x) + bp(q) && (17)
\end{aligned}
$$

Thus, the expected optimizing solution is not exactly the posterior probability, but a weighted sum of posterior probability and prior probability with $a$ and $b$ as weights. To compensate for prior differences, one gets:

$$p(q|x) = \frac{g(x) - bp(q)}{a} \tag{18}$$

The proposed prior compensation scheme is tested on the binary classification problem with different values for $a$. Figure 3 shows the posterior estimates for $a = 0.8$ and $a = 0.9$. In both cases, prior compensation improves posterior probability estimation. If posterior probability estimates are used to perform classification, performance are also improved and are more balanced for the two classes when prior compensation is used (c.f. Table 2).

# 5 Discussion and Conclusion

This project demonstrated that sigmoid-fitting method for SVM posterior probability estimation works well, at least for the simple binary classification problem presented. There appears to be systematic relationship between prior distribution and pattern of distortion in posterior estimates when randomness is introduced into training targets. A prior compensation scheme is proposed to improve posterior estimation under this

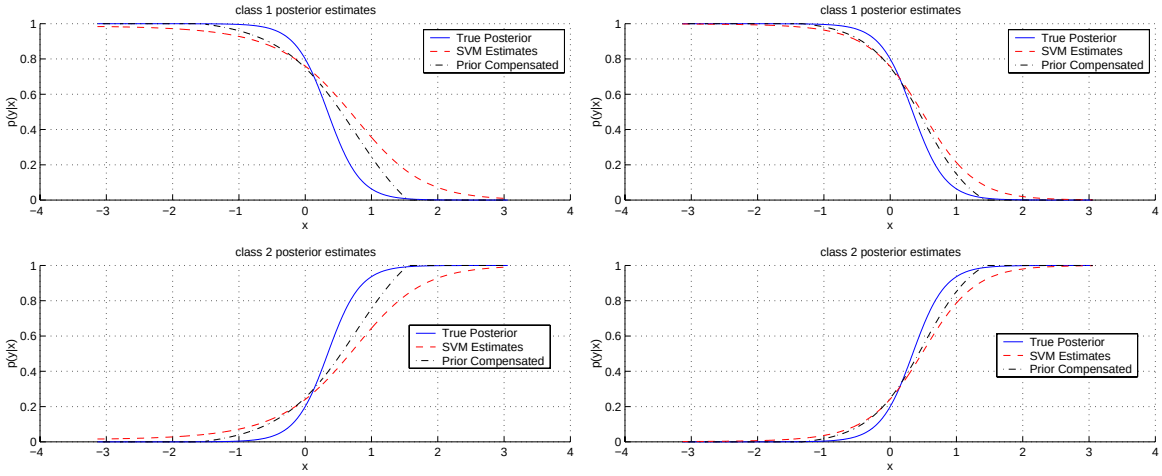


Figure 3: Comparison of posterior estimates with and without the proposed prior compensation scheme. Left: $a = 0.8$, Right: $a = 0.9$.

condition and experimental results support its effectiveness. It will be interesting to apply the SVM based posterior probability estimation to real applications such as various tasks in speech recognition. For simple binary classification problems such as voicing/unvoicing detection, the proposed prior compensation scheme is expected to work well since the assumption of randomness in training targets is more appropriate. For more complex problems such as classification of entire phoneme set, it is not clear how effective prior compensation will be because of the high dependencies among different errors in label targets.

# References

[1] John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, A. J. Smola, P. Bartlett, B. Scholkopf, D. Schuurmans, eds., MIT Press, 1999

[2] W.H. Press, S.A. Teukolsdy, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing (2nd ed.)*. Cambridge University Press, Cambridge, 1992

[3] R. Collobert and S. Bengio. SVMTorch: Support vector machines for large-scale regression problems. In *Journal of Machine Learning Research* 1, 2001, pp. 143-160.

[4] G. Wahba. The bias-variance tradeoff and the randomized GACV. In D. A. Cohn, M. S. Kearns, S. A. Solla, eds. *Advances in Neural Information Processing Systems*, v11. MIT Press, Cambridge, MA, 1999