Direct Estimation of Leave-One-Out Error in SVMs

Chenlei Leng

April 22, 2003

Abstract

We propose a direct estimate of the Leave-Out-One (LOO) error of Support Vector Machines (SVMs) in classification cases.

1 Introduction

Suppose we have training set $\{(y_i, x_i), i = 1, ..., n\}$ where $y_i \in \{-1, 1\}$ and $x_i \in R^d$. The SVM finds the minimizer of

$$\frac{1}{n}\sum_{i=1}^{n}(1 - y_i f(x_i)))_+ + \lambda ||h||^2_{\mathcal{H}_K}, \tag{1.1}$$

where $f(x) = h(x) + b$ and $h \in \mathcal{H}_K$, the Reproducing Kernel Hilbert Space with Reproducing Kernel K. And $(\tau)_+ = \tau, \tau > 0; = 0$ otherwise.

The solution to (1.1) is known to be of the form $f(x) = \sum_{i=1}^{n} c_i K(x_i, x) + b$. Denote Y as the $n \times n$ diagonal matrix with y_i in the $ii$the position. The coefficients c_i and d can be solved using a dual formulation of (1.1) after introducing Lagrange multipliers $\alpha = (\alpha_1, ..., \alpha_n)^T$

$$\begin{aligned} &\max L(\alpha) = -\frac{1}{2}\alpha^T H\alpha + e'\alpha \\ &\text{subject to } \; 0 \le \alpha_i \le 1; \; y'\alpha = 0. \end{aligned} \tag{1.2}$$

where $H = \frac{1}{2n\lambda}YKY$. $C = \frac{1}{2n\lambda}Y\alpha$ after α is solved and b can be solved as a solution of

$$\min_b e'(e - Y(KY\alpha + b))_+. \tag{1.3}$$

The tuning parameter λ can be determined by using n-fold cross validation or more ambitiously leave-one-out cross validation. Wahba, Lin, and Zhang (2000) proposed GACV using a Taylor expansion to approximate LOO error. Joachims (2000) used an upper bound on LOO error to find optimal λ, which is called $\xi\alpha$ estimator. An interesting link between the two methods are pointed out by Wahba, Lin, Lee, and Zhang (2001). However, although such approximation gives fairly good estimate, it is also been observed GACV and $\xi\alpha$ tend to overestimate λ.

2 Reformulation of SVMs

We will deal with a variant of SVMs, which will be called VSVMs,

$$\frac{1}{n}\sum_{i=1}^{n}(1 - y_i f(x_i)))_+ + \lambda ||f||^2_{\mathcal{H}_K \oplus \{1\}}. \tag{2.1}$$

Notice we penalize the constant b as well as h in f, where (1.1) only penalizes h. It can be shown when $n \to \infty$ and $\lambda \to 0$ at certain rate, the minimizer of SVMs and VSVMs goes to $E(1 - Yf(x))_+$.

Theorem 2.1 (VSVM and Bayes rule). *The minimizer of* $E(1 - Yf(x))_+$ *is* $sign(2P(Y = 1|X = x) - 1)$.

Proof. See Lin (2002) for a proof. □

The dual of (2.1) is the following

$$\begin{aligned} \max L(\alpha) &= -\frac{1}{2}\alpha^T H\alpha + e'\alpha \\ \text{subject to } & \ 0 \le \alpha_i \le 1. \end{aligned} \tag{2.2}$$

where $H = \frac{1}{2n\lambda}Y(K + ee')Y$. $C = \frac{1}{2n\lambda}Y\alpha$ and $b = e'Y\alpha$ after α is solved. Notice that b can be directly solved using VSVM. The dual has no linear constraint. Similiar formulation has appeared in Mangasarian and Musicant (2001).

The solution to (1.2) involves a bounded quadratic programming problem. A working set algorithm using a loose convergence threshold has been shown in Lee and C.-J. Lin (2000). Mangasarian and Musicant (1999) proposes a fast algorithm successive overralaxation (SOR) without using a quadratic programming solver. If we define $Q = L + E + L'$, where E is diagonal and L is lower triangular, a necessary and sufficient optimality condition for (3.3) is

$$\alpha = (\alpha - \omega E^{-1}(Q\alpha - e))_{\#}, \ \omega \in (0, 2) \tag{2.3}$$

where $((\tau))_{\#} = 0, \tau \le 0; \tau, 0 < \tau < 1; 1, \tau > 1$. It can be implemnted as

$$\alpha_j^{i+1} = (\alpha_j^i - \omega E_{jj}^{-1}(\sum_{l=1}^{j-1} Q_{jl}\alpha_l^{i+1} + \sum_{l=j}^{n} Q_{jl}\alpha_l^i - 1))_{\#}. \tag{2.4}$$

We will denote the solution to (3.3) as $\hat{\alpha}$.

3 Leave-Out-One error

Let $f^{[-i]}$ be the solution to (3.3) with ith sample removed. We define

$$\begin{aligned} MISC(\lambda) &= E_{true}\frac{1}{n}\sum_{i=1}^{n}(1 - y_i f_i)_* \\ GCKL(\lambda) &= E_{true}\frac{1}{n}\sum_{i=1}^{n}(1 - y_i f_i)_+ \end{aligned} \tag{3.1}$$

where $(\tau)_* = 1, \tau > 0; = 0, \tau \le 0$.

Define corresponding LOO estimate of $MISC$ and $GCKL$ as

$$\begin{aligned} LOOM(\lambda) &= \frac{1}{n}\sum_{i=1}^{n}(1 - y_i f_i^{[-i]})_* \\ LOOG(\lambda) &= \frac{1}{n}\sum_{i=1}^{n}(1 - y_i f_i^{[-i]})_+ \end{aligned} \tag{3.2}$$

We know removing a non support vector does not change the solution. The LOO estimates only need to evaluate the number of support vector times. $\hat{\alpha}$ is already a solution of (3.3). The solution $\alpha^{[-i]}$ with $\alpha_i = 0$ to the problem

$$\begin{aligned} \max L(\alpha) &= -\frac{1}{2}\alpha^T H\alpha + e'\alpha \\ \text{subject to } & 0 \le \alpha_j \le 1, \alpha_i = 0 \end{aligned} \tag{3.3}$$

should be near the optimal $\hat{\alpha}$. We can use one step SOR with α_i fixed at zero to solve for $\alpha^{[-i]}$. In fact, if $\alpha_{[-i]} = \hat{\alpha}$ with $\alpha_i = 0$, Jaakkola and Haussler (1999) shows $sum(-y_i f_i^{[-i]})_*$ is an upper bound of misclassification rate. But this bound tends to be loose.

4 Simulation

We will use Gaussian kernel $K(x, x') = exp(-||x - x'||/(2\gamma^2))$. 200 samples x_i are drawn uniformly from R^2. $y_i = 1$ with 0.95 probabilty if $||X||_2^2 < 0.5$, 0.05 probabilty if $||X||_2^2 > 0.64$ and 0.5 probabilty if in between. This is the same example used by Wahba, Lin, and Zhang (2000). Figure 1 shows out proposed methods approximate the truth pretty well. When there are multiple minima, we randomly choose one to plot the classification boundary. Figure 6 shows the upper bound in Jaakkola and Haussler (1999) fails to find a minimum close to the theoretical minimum.

References

Jaakkola, T. S. and D. Haussler (1999). Probabilistic kernel regression models. In *Proceedings of the 1999 Conference on AI and Statistics.*

Joachims, T. (2000). Estimating the generalization performance of a SVM efficiently. In *Proceedings of the International Conference on Machine Learning*, San Francisco. Morgan Kaufman.

Lee, J.-H. and . C.-J. Lin (2000). Automatic model selection for support vector machines. Technical report, National Taiwan Univ.

Lin, Y. (2002). Support vector machines and the bayes rule in classification. *Data Mining and Knowledge Discovery 6*, 259–275.

Mangasarian, O. L. and D. R. Musicant (1999). Successive overrelaxation for support vector machines. *IEEE Transactions on Neural Networks, 10, 1999, 1032-1037 10*, 1032–1037.

Mangasarian, O. L. and D. R. Musicant (2001). Data discrimination via nonlinear generalized support vector machines. In O. L. M. M. C. Ferris and J.-S. Pang (Eds.), *Complementarity: Applications, Algorithms and Extensions*, pp. 233–251. Kluwer Academic Publishers.

Wahba, G., Y. Lin, Y. Lee, and H. Zhang (2001). On the relation between the gacv and joachims' xa method for tuning support vector machines, with extension to the nonstandard case. Technical Report 1039, UW Madison, Dept of Statistics.

Wahba, G., Y. Lin, and H. Zhang (2000). Generalized approximate cross validation for support vector machines, or, another way to look at margin-like quantities. In S. Smola, Bartlett and Schurmans (Eds.), *Advances in Large Margin Classifiers*, pp. 297–309. MIT Press.

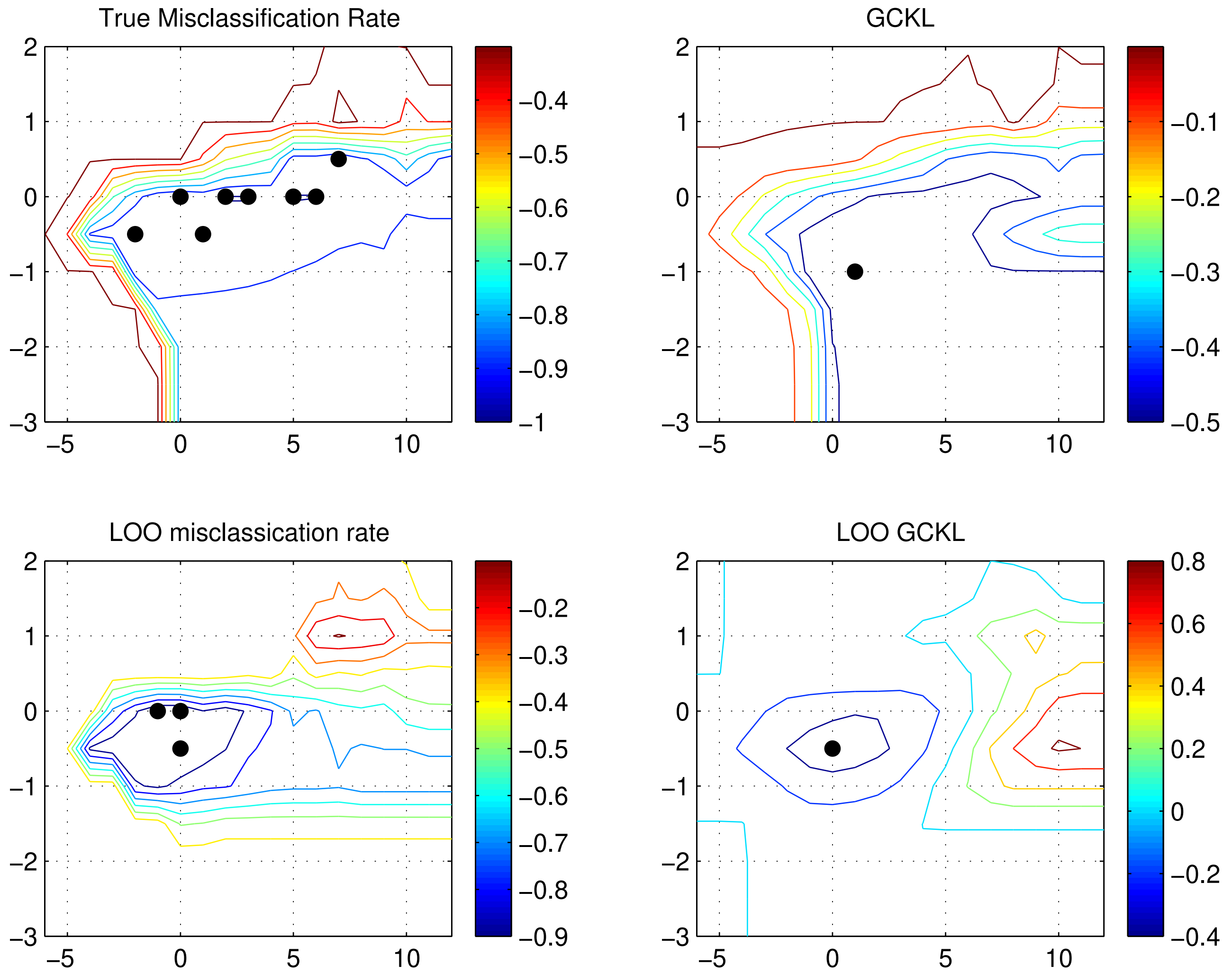

Figure 4.1: Contour plot of scores in log 10 sclae w.r.t. $log10(\gamma)$ and $-log2(2n\lambda)$. The minima are plotted as black dots.

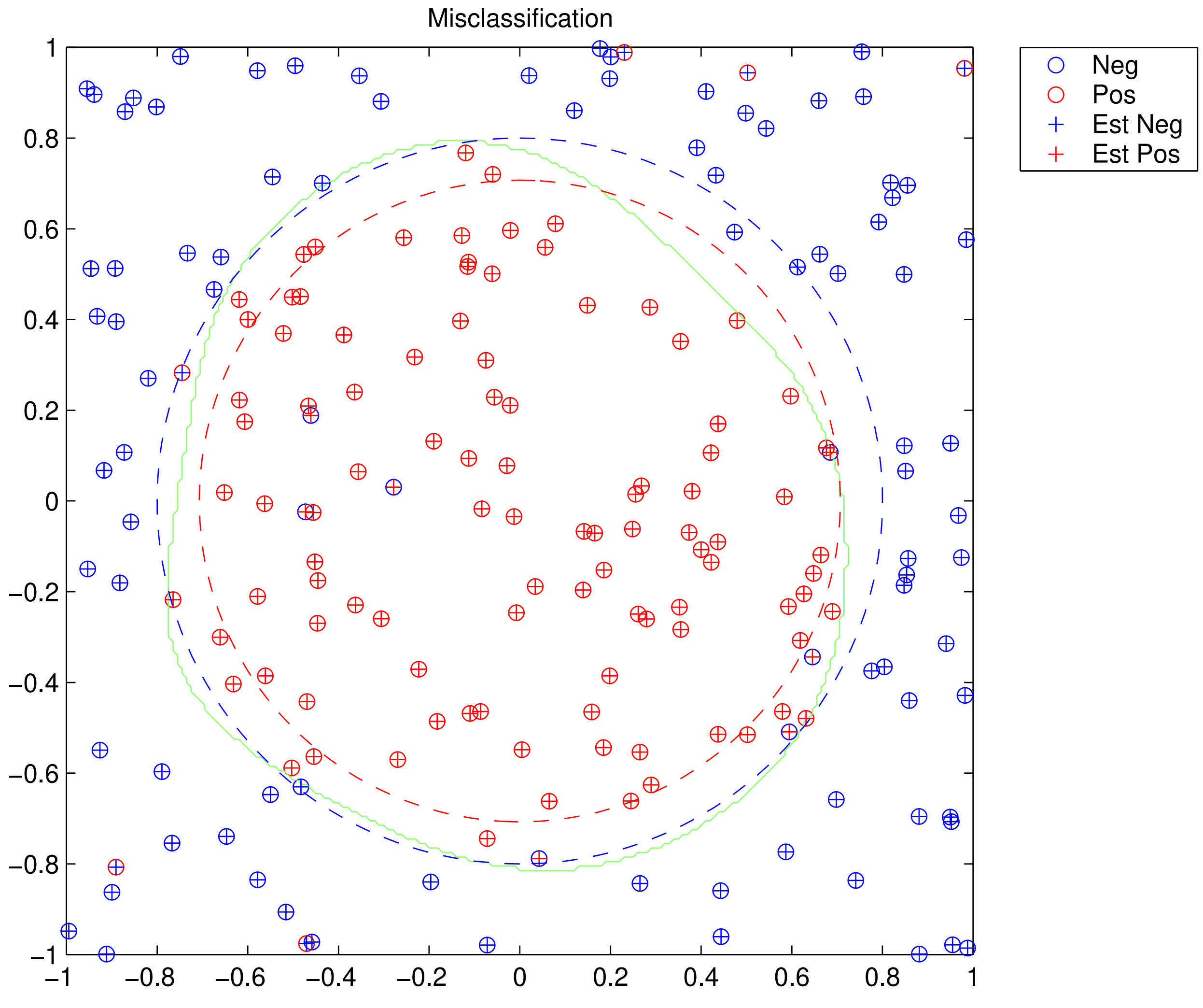

Figure 4.2: Best model by using MISC. The green line is the lassification boundary. A point with different color is a misclassified point for the training set.

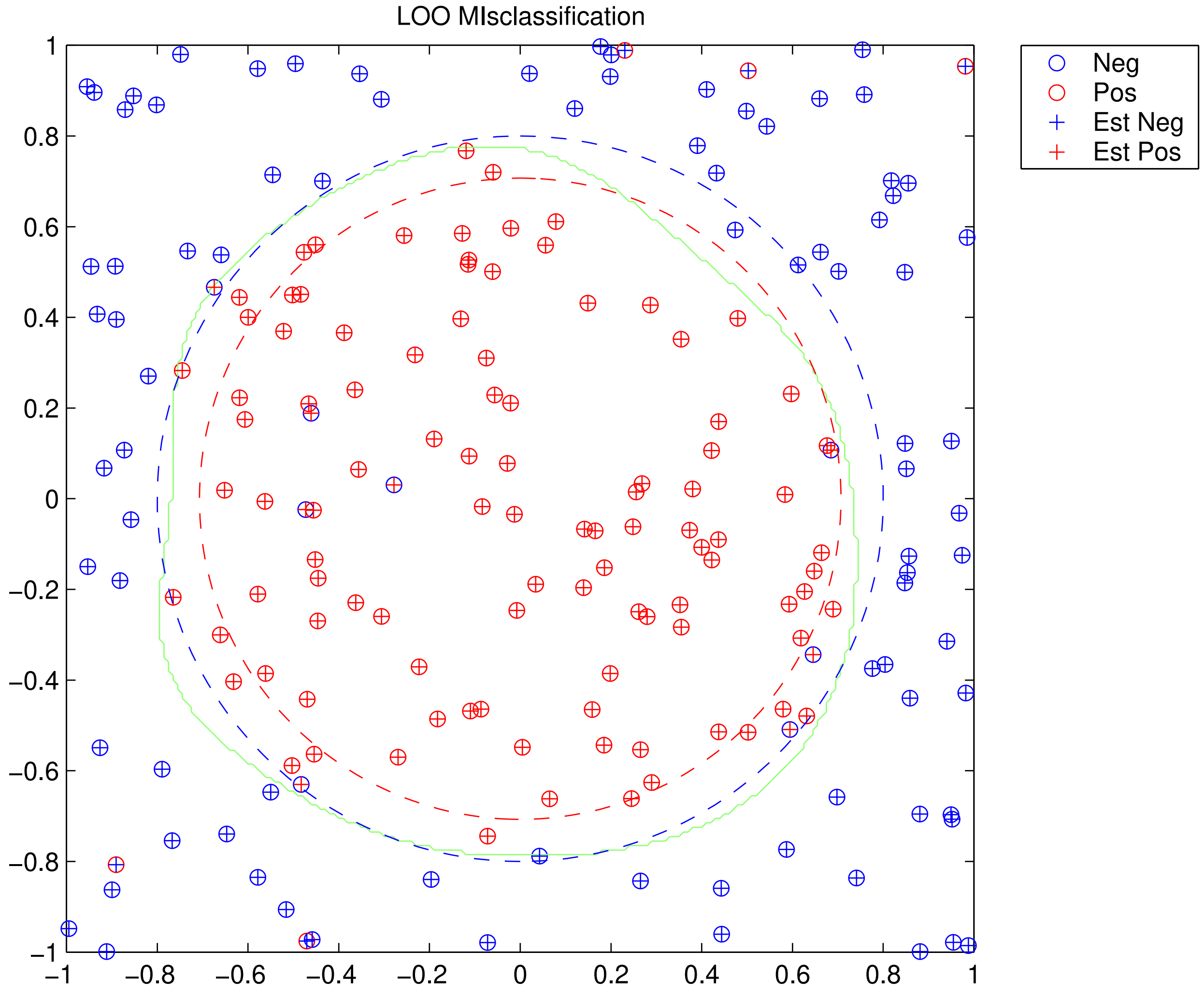

Figure 4.3: Best model by using LOOM. The green line is the lassification boundary.

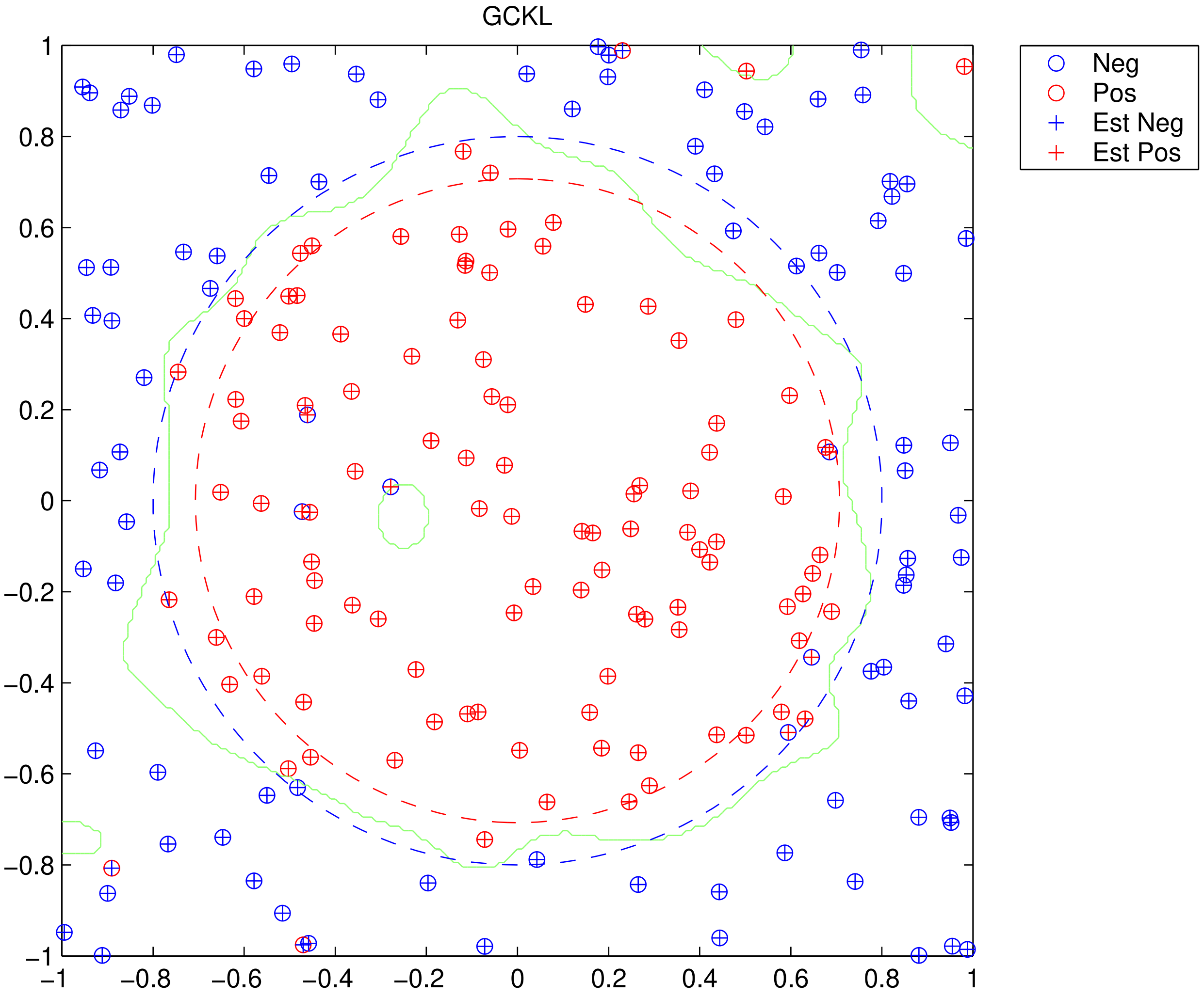

Figure 4.4: Best model by using GCKL. The green line is the lassification boundary.

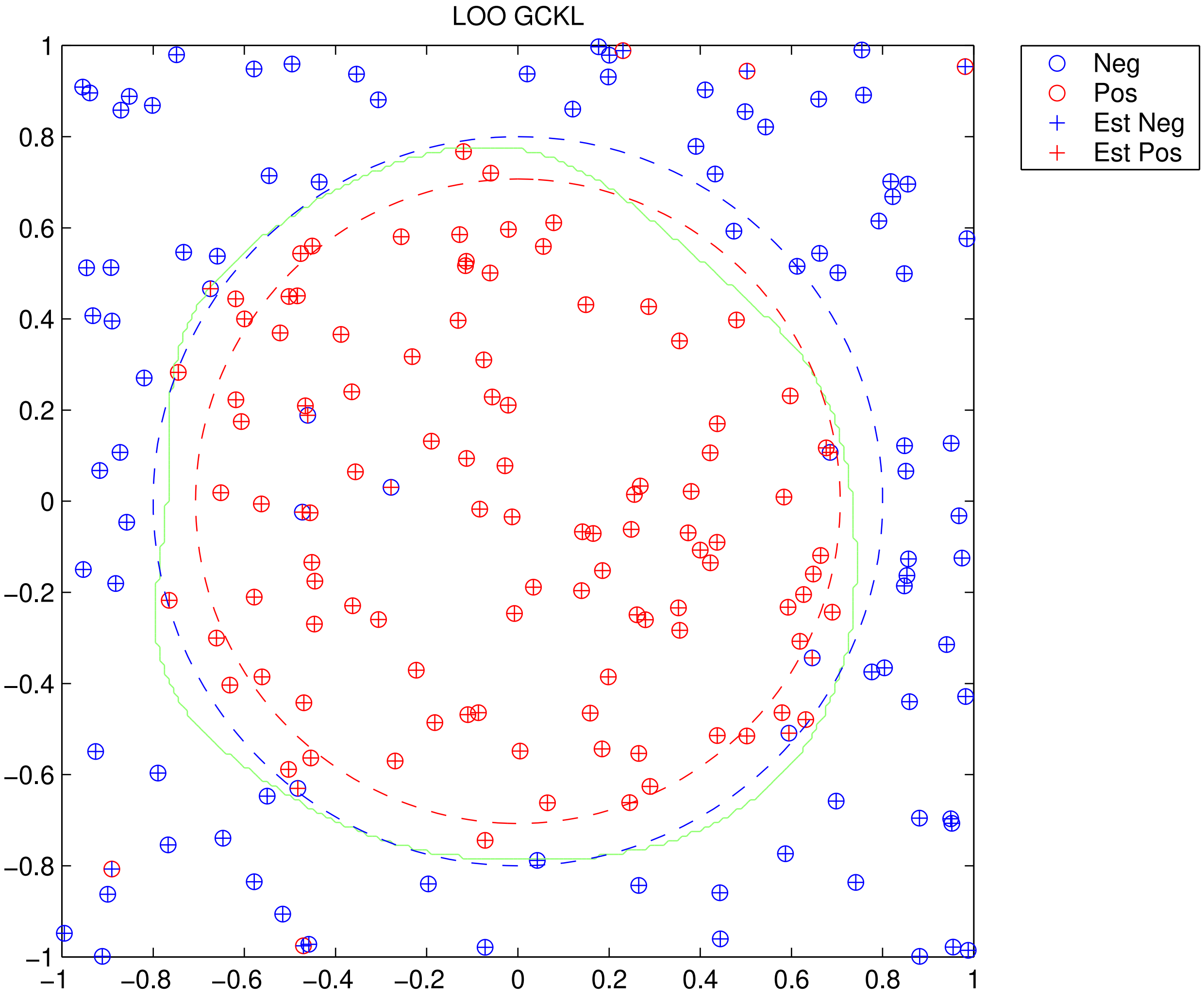

Figure 4.5: Best model by using LOOG. The green line is the lassification boundary.

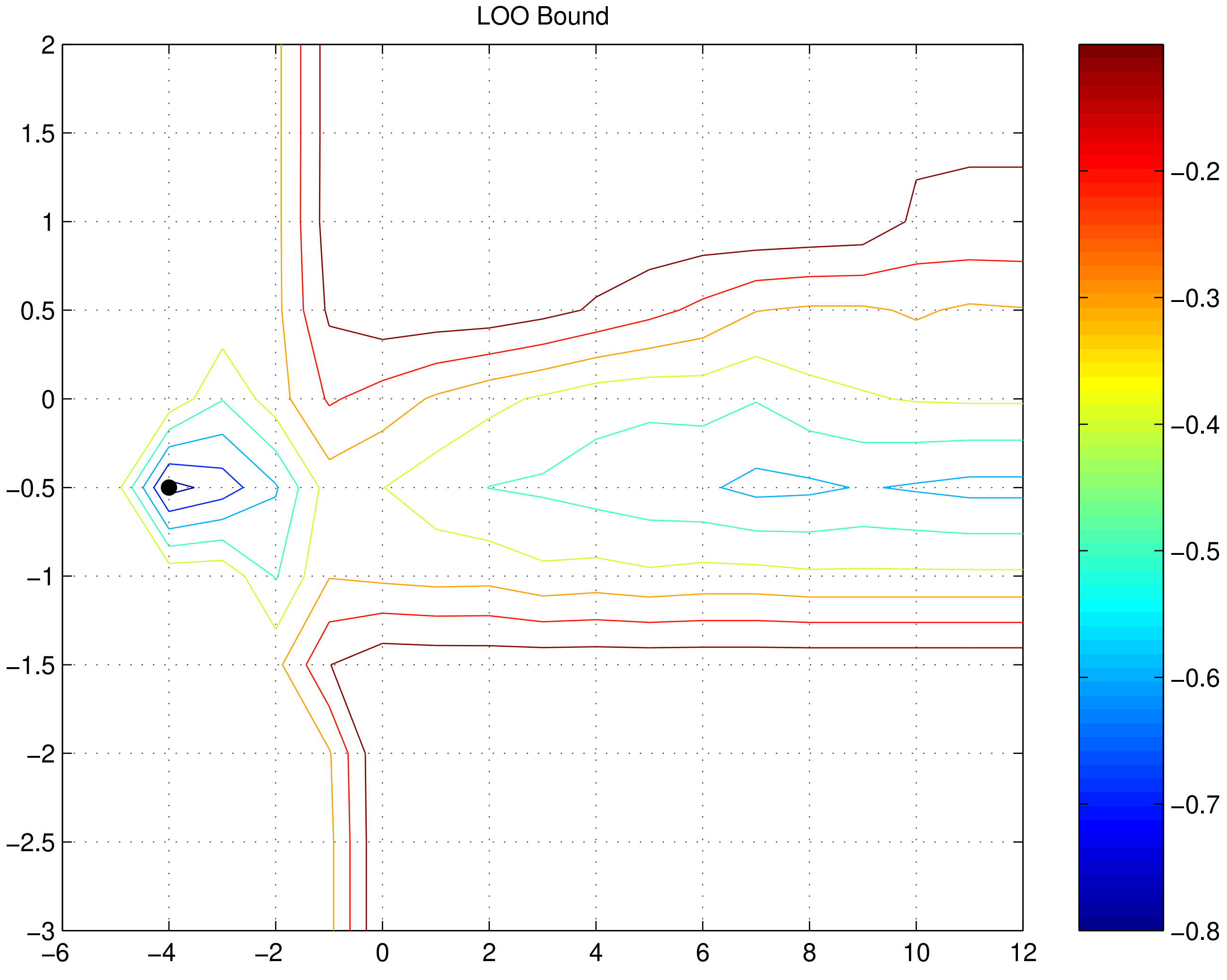

Figure 4.6: Contour plot of upper bound.