Hindawi Publishing Corporation
Advances in Artificial Neural Systems
Volume 2011, Article ID 302572, 6 pages
doi:10.1155/2011/302572

Research Article

Cross-Validation, Bootstrap, and Support Vector Machines

Masaaki Tsujitani[1, 2] **and Yusuke Tanaka**[1, 2]

[1] *Division of Informatics and Computer Sciences, Graduate School of Engineering, Osaka Electro-Communication University, Osaka 572-8530, Japan*
[2] *Biometrics Department, Statistics Analysis Division, EPS Co., Ltd., 3-4-30 Miyahara, Yodogawa-ku, Osaka 532-0003, Japan*

Correspondence should be addressed to Masaaki Tsujitani, ekaaf900@ricv.zaq.ne.jp

Received 2 April 2011; Accepted 7 June 2011

Academic Editor: Tomasz G. Smolinski

This paper considers the applications of resampling methods to support vector machines (SVMs). We take into account the leaving-one-out cross-validation (CV) when determining the optimum tuning parameters and bootstrapping the deviance in order to summarize the measure of goodness-of-fit in SVMs. The leaving-one-out CV is also adapted in order to provide estimates of the bias of the excess error in a prediction rule constructed with training samples. We analyze the data from a mackerel-egg survey and a liver-disease study.

1. Introduction

In recent years, support vector machines (SVMs) have been intensively studied and applied to practical problems in many fields of science and engineering [1–3]. SVMs have many merits that distinguish them from many other machine learning algorithms, including the nonexistence of local minima, the speed of calculation, and the use of only two tuning parameters. There are at least two reasons to use a leaving-one-out cross-validation (CV) [4]. First, the criterion based on the method is demonstrated to be favorable when determining the tuning parameters. Second, the method can estimate the bias of the excess error in prediction. No standard procedures exist by which to assess the overall goodness-of-fit of the model based on SVM. By introducing the maximum likelihood principle, the deviance allows us to test the goodness-of-fit of the model. Since no adequate distribution theory exists for the deviance, we provide bootstrapping on the null distribution of the deviance for the model having optimum tuning parameters for SVM with a specified significance level [5–8].

The remainder of this paper is organized as follows. In Section 2, using the leaving-one-out CV, we focus on the determination of the tuning parameters and the evaluation of the overall goodness-of-fit with the optimum tuning parameters based on bootstrapping. The leaving-one-out CV is also adapted in order to provide estimates of the bias of the excess error in a prediction rule constructed with training samples [9]. In Section 3, the one-against-one method is used to estimate a vector of multiclass probabilities for each pair of classes and then to couple the estimates together [3, 10]. In Section 4, the methods are illustrated using mackerel-egg survey and liver-disease data. We discuss the relative merits and limitations of the methods in Section 5.

2. Support Vector Machines and Resampling Methods

2.1. Support Vector Machines. Given n training pairs $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$, where $\mathbf{x}_i$ is an input vector and $y_i \in \{-1, +1\}$, the SVM solves the following primal problem:

$$\begin{aligned} \min_{\boldsymbol{\beta}, b} \quad & \frac{1}{2}\boldsymbol{\beta}^T \boldsymbol{\beta} + C\sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & y_i \left\{ \boldsymbol{\beta}^T \phi(\mathbf{x}_i) + b \right\} \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, n, \end{aligned} \tag{1}$$

where $\boldsymbol{\beta}$ is a unit vector (i.e., $\|\boldsymbol{\beta}\| = 1$), T denotes the transposition of the matrix, $K(\mathbf{x}, \mathbf{x}_i) \equiv \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_i)$ is the kernel function, C is the tuning parameter denoting the tradeoff between the margin width and the training data error, and $\xi_i \geq 0$ are slack variables. For an unknown input pattern $\mathbf{x}$, we have the decision function

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b, \tag{2}$$

where $\{\alpha_i,\ i = 1, 2, \ldots, n;\ \alpha_i \geq 0\}$ are the Lagrange multipliers. We employ the Gaussian radial basis function as the kernel function [3, 11, 12]

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2\right), \tag{3}$$

where $\gamma > 0$ is a fixed parameter, and

$$\|\mathbf{x} - \mathbf{x}'\|^2 = \langle \mathbf{x} - \mathbf{x}', \mathbf{x} - \mathbf{x}' \rangle. \tag{4}$$

Binary classification is performed by using the decision function $f(\mathbf{x})$: the input $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^T$ is assigned to the positive class if $f(\mathbf{x}) \geq 0$, and to the negative class otherwise. Platt [13] proposed one method for producing probabilistic outputs from a decision function by using logistic link function

$$p(y = +1 \mid f) = \frac{1}{1 + e^{Af+B}}, \tag{5}$$

where $f_i = f(\mathbf{x}_i)$ and y_i represent the output of the SVM and the target value for the sample, respectively [14]. This is equivalent to fitting a logistic regression model to the estimated decision values. The unknown parameters A, B in (5) can be estimated by minimizing the cross-entropy

$$\underset{A,B}{\operatorname{Min}} \left[-\sum_{i=1}^{n} \{ y_i \ln p_i + (1 - y_i) \ln(1 - p_i) \} \right], \tag{6}$$

where

$$p_i = \frac{1}{1 + e^{Af_i + B}}. \tag{7}$$

Putting

$$t_i = \frac{y_i + 1}{2} = \begin{cases} 0: & y_i = -1, \\ 1: & y_i = +1, \end{cases} \tag{8}$$

from (6), (7), and (8), we obtain

$$\begin{aligned} &- \{ t_i \ln p_i + (1 - y_i) \ln(1 - t_i) \} \\ &\quad = t_i (Af_i + B) + \ln\{1 + \exp(-Af_i - B)\}. \end{aligned} \tag{9}$$

Lin et al. [15] observed that the problem of ln(0) never occurs for (9).

2.2. Leaving-One-Out Cross-Validation

2.2.1. CV Score. We must determine the optimum values of tuning parameters C and γ in (1) and (3), respectively. This can be done by means of the leaving-one-out CV; a byproduct is that the excess error rate of incorrectly predicting the outcome is estimated.

Let the initial sample $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{i-1}, \mathbf{X}_i, \mathbf{X}_{i+1}, \ldots, \mathbf{X}_n\}$ with $\mathbf{X}_i = (\mathbf{x}_i, t_i)$ be independently distributed according to an unknown distribution. The leaving-one-out CV algorithm is then given as follows (see, e.g., [5]).

Step 1. From the initial sample $\mathbf{X}$, $\mathbf{X}_i$ are deleted in order to form the training sample $\mathbf{X}_{[i]} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{i-1}, \mathbf{X}_{i+1}, \ldots, \mathbf{X}_n\}$.

Step 2. Using each training sample, fit an SVM and predict the decision value $\widehat{f}_{[i]}$ for $\mathbf{X}_i$.

Step 3. From the decision value $\widehat{f}_{[i]}$, we can predict $\widehat{p}_{[i]}$ for the deleted ith sample using (7) and calculate the predicted log-likelihood $t_{[i]} \ln \widehat{p}_{[i]} + (1 - t_{[i]}) \ln(1 - \widehat{p}_{[i]})$.

Step 4. Steps 1 to 3 are repeated for $i = 1, 2, \ldots, n$.

Step 5. The CV score (i.e., averaged predicted log-likelihood) is given by

$$\mathrm{CV} = -2 \sum_{i=1}^{n} \{ t_{[i]} \ln \widehat{p}_{[i]} + (1 - t_{[i]}) \ln(1 - \widehat{p}_{[i]}) \}. \tag{10}$$

Step 6. Carry out a grid search over tuning parameters C and γ, taking the tuning parameters with minimum CV as optimal. It should be noted that the CV score is asymptotically equivalent to AIC (akaike information criterion) and EIC (extended information criterion) [16–18].

2.2.2. Excess Error Estimation. Let the *actual error rate* be the probability of incorrectly predicting the outcome of a new observation, given a discriminant rule on initial sample $\mathbf{X}$; this is useful for performance assessment of a discriminant rule. Given a discriminant rule based on the initial sample, the error rates of discrimination are also of interest. As the same observations are used for forming and assessing the discriminant rule, this proportion of errors, called the *apparent error rate*, underestimates the actual error rate. The estimate of the error rate is seriously biased when the initial sample is small. This bias for a given discriminant rule is called the *excess error* of that rule. To correct this bias and estimate the error rates, we provide the bias correction of the apparent error rate associated with a discriminant rule, which is constructed by fitting to the training sample in the SVM.

By applying a discriminant rule to the initial sample $\mathbf{X}$, we can form the realized discriminant rule $\eta_{\mathbf{X}}$. Let $\eta_{\mathbf{X}_{[i]}}$ be the discrimination rule based on $\mathbf{X}_{[i]}$. Given a subject with $\mathbf{x}_i$, we predict the response by $\eta_{\mathbf{X}_{[i]}}(\mathbf{x}_i)$. The algorithm for leaving-one-out CV that estimates the excess error rate when fitting a SVM is given as follows [9].

Step 1. Generate the training sample $\mathbf{X}_{[i]}$, and construct the realized discrimination rule $\eta_{\mathbf{X}_{[i]}}$ based on $\mathbf{X}_{[i]}$. Then, define

$$Q\left(t_i; \eta_{\mathbf{X}_{[i]}}(\mathbf{x}_i)\right) = \begin{cases} 1: & \text{incorrect discrimination,} \\ 0: & \text{otherwise.} \end{cases} \tag{11}$$

Then leaving-one-out CV error rate is given by

$$\frac{1}{n}\sum_{i=1}^{n} Q\left(t_i; \eta_{\mathbf{X}_{[i]}}(\mathbf{x}_i)\right). \tag{12}$$

Step 2. Calculate the apparent error

$$\frac{1}{n}\sum_{i=1}^{n} Q(t_i; \eta_{\mathbf{X}}(\mathbf{x}_i)). \tag{13}$$

Step 3. The cross-validation estimator of expected excess error is

$$\hat{r}_{\mathrm{CV}} = \frac{1}{n}\sum_{i=1}^{n} Q(t_i; \eta_{\mathbf{X}}(\mathbf{x}_i)) - \frac{1}{n}\sum_{i=1}^{n} Q\left(t_i; \eta_{\mathbf{X}_{[i]}}(\mathbf{x}_i)\right). \tag{14}$$

2.3. Bootstrapping. Introducing the maximum likelihood principle into the SVM, the deviance allows us to test the goodness-of-fit of the model

$$\begin{aligned} \mathrm{Dev} &= 2\left[\ln \hat{L}_f - \ln \hat{L}_c\right] \\ &= -2\sum_{i=1}^{n}\{t_i \ln \hat{p}_i + (1-t_i)\ln(1-\hat{p}_i)\}, \end{aligned} \tag{15}$$

where $\ln \hat{L}_c$ denotes the maximized log likelihood under some current SVM, and the log likelihood for the saturated model $\ln \hat{L}_f$ is zero. The deviance given by (15) is, however, not even approximately a χ^2 distribution for the case in which ungrouped binary responses are available [19, 20]. The number of degrees of freedom (d.f.) required for the test for significance using the assumed χ^2 distribution for the deviance is a contentious issue. No adequate distribution theory exists for the deviance. The reason for this is somewhat technical (for details, see Section 3.8.3 in [19]). Consequently, the deviance on fitting a model to binary response data cannot be used as a summary measure of the goodness-of-fit test of the model.

Based on the above discussion, the percentile of deviance for goodness-of-fit test can in principle be calculated. However, the calculations are usually too complicated to perform analytically, so Monte Carlo method can be employed [6, 7].

Step 1. Generate B bootstrap samples $\mathbf{X}^*$ from the original sample $\mathbf{X}$. Let $\mathbf{X}_b^*$ denote the bth bootstrap sample.

Step 2. For the bootstrap sample $\mathbf{X}_b^*$, compute the deviance of (15), denoted by $\mathrm{Dev}^*(b)$.

Steps 1 and 2 are repeated independently B times, and the computed values are arranged in ascending order.

Step 3. Take the value of the jth order statistic $\mathrm{Dev}^*(b)$ of the B replications as an estimate of the quantile of order $j/(B+1)$.

Step 4. The estimate of the $100(1-\alpha)$th percentile of $\mathrm{Dev}^*(b)$ is used to test the goodness-of-fit of a model having a specified significance level $\alpha = 1 - j/(B+1)$. The value of the deviance of (15) being greater than the estimate of the percentile indicates that the model fits poorly. Typically, the number of replication B is in the range of $50 \leq B \leq 400$.

2.4. Influential Analysis. Assessing the discrepancies between t_i and $\hat{p}_i$ at the ith observation in (15), the influence measure provides guides and suggestions that may be carefully applied to a SVM [19]. The effect of the ith observation on the deviance can be measured by computing

$$\Delta\mathrm{Dev}_{[i]} = \mathrm{Dev} - \mathrm{Dev}_{[i]}, \tag{16}$$

where $\mathrm{Dev}_{[i]}$ is the deviance with ith observation deleted. The distribution of $\Delta\mathrm{Dev}_{[i]}$ will be approximated by χ^2 with d.f. = 1 when the fitted model is correct. An index plot is a reasonable rule of thumb for graphically presenting the information contained in the values of $\Delta\mathrm{Dev}_{[i]}$. The key idea behind this plot is not to focus on a global measure of goodness-of-fit but rather on local contributions to the fit. An influential observation is one that greatly changes the results of the statistical inference when deleted from the initial sample.

Platt [13] proposed the threefold CV for estimating the decision values in (9). However, the value of $\Delta\mathrm{Dev}_{[i]}$ may be negative because three SVMs are trained on splitted three parts of training pairs $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$. Therefore, in the present paper, we train a single SVM on the training pairs in order to evaluate the decision values $f_i's$ and estimate probabilistic outputs according to [15].

3. Multiclass SVM

We consider the discriminant problem with K classes and n training pairs $(\mathbf{x}_1, \mathbf{t}_1), (\mathbf{x}_2, \mathbf{t}_2), \ldots, (\mathbf{x}_n, \mathbf{t}_n)$, where $\mathbf{x}_i$ is an input vector and $\mathbf{t}_i = (t_{i1}, t_{i2}, \ldots, t_{iK})$ [10, 21, 22]. Let $p_{ii}, p_{i2}, \ldots, p_{iK}$ denote the response probabilities, with $\sum_{k=1}^{K} p_{ik} = 1$, for multiclass classification with

$$t_{ik} = \begin{cases} 1: & \text{input vector } \mathbf{x}_i \text{ is from } k\text{th class,} \\ 0: & \text{otherwise.} \end{cases} \tag{17}$$

The log-likelihood is given by

$$\ln L = \sum_{i=1}^{n}\sum_{k=1}^{K}\{t_{ik}\ln(p_{ik})\}. \tag{18}$$

For multi-class classification, the one-against-one method (also called pairwise classification) is used to produce a vector of multi-class probabilities for each pair of classes, and then to couple the estimates together [10]. The earliest used implementation for multi-class SVM is probably the one-against-one method of [21]. This method constructs $K(K-1)/2$ classifiers based on the training on data from the kth and lth classes of training set.

The SVM solves the primal formulation [3, 10, 23]

$$\begin{aligned}
\min_{\boldsymbol{\beta}^{kl}, b^{kl}, \xi^{kl}} \quad & \frac{1}{2}\left(\boldsymbol{\beta}^{kl}\right)^T \boldsymbol{\beta}^{kl} + C\sum_t \xi_t^{kl} \\
\text{s.t.} \quad & \left(\boldsymbol{\beta}^{kl}\right)^T \phi(x_t) + b^{kl} \geq 1 - \xi_t^{kl}, \quad \text{if } y_t = k, \\
& \left(\boldsymbol{\beta}^{kl}\right)^T \phi(x_t) + b^{kl} \leq -1 + \xi_t^{kl}, \quad \text{if } y_t = l, \\
& \xi_t^{kl} \geq 0.
\end{aligned} \tag{19}$$

Given K classes of data for any $\mathbf{x}$, the goal is to estimate

$$p_k = \Pr\{t = k \mid \mathbf{x}\}, \quad k = 1, 2, \ldots, K. \tag{20}$$

We first estimate pairwise class probabilities

$$r_{kl} \approx \Pr(t = k \mid t = k \text{ or } l, \mathbf{x}), \tag{21}$$

by using

$$r_{kl} \approx \frac{1}{1 + e^{Af+B}}, \tag{22}$$

where A and B are estimated by minimizing the cross entropy using training data and the corresponding decision values f.

Hastie and Tibshirani [21] proposed minimizing the Kullback-Leibler (KL) distance between r_{kl} and $\mu_{kl} = E[r_{kl}] = p_k/(p_k + p_l)$,

$$\begin{aligned}
l(\mathbf{p}) &= \sum_{k \neq l} n_{kl} r_{kl} \ln\left(\frac{r_{kl}}{\mu_{kl}}\right) \\
&= \sum_{k<l} n_{kl} \left\{ r_{kl} \ln\left(\frac{r_{kl}}{\mu_{kl}}\right) + (1 - r_{kl}) \ln\left(\frac{1 - r_{kl}}{1 - \mu_{kl}}\right) \right\},
\end{aligned} \tag{23}$$

where $r_{lk} = 1 - r_{kl}$ and n_{kl} is the number of training data in the kth and lth classes.

Wu et al. [10] propose the second approach to obtain p_k from all these r_{kl}'s by optimizing

$$\begin{aligned}
\min \quad & \frac{1}{2}\sum_{k=1}^{K} \sum_{k:k \neq l} \left(r_{kl} p_i - r_{kl} p_j\right)^2 \\
\text{s.t.} \quad & \sum_{k=1}^{K} p_k = 0, \quad p_k \geq 0.
\end{aligned} \tag{24}$$

Thus, we can adopt the leaving-one-out CV similar to the method in Section 2.2.

Step 1. From the initial sample $\mathbf{X}$, $\mathbf{X}_i$ are deleted in order to form the training sample $\mathbf{X}_{[i]} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_{i-1}, \mathbf{X}_{i+1}, \ldots, \mathbf{X}_n\}$.

Step 2. Using each training sample, fit a SVM in order to estimate $r_{kl[i]}$ by (22), and predict $(\hat{p}_{[i1]}, \hat{p}_{[i2]}, \ldots, \hat{p}_{[iK]})$ for the deleted ith sample $\mathbf{X}_{[i]}$.

Step 3. Steps 1 and 2 are repeated for $i = 1, 2, \ldots, n$.

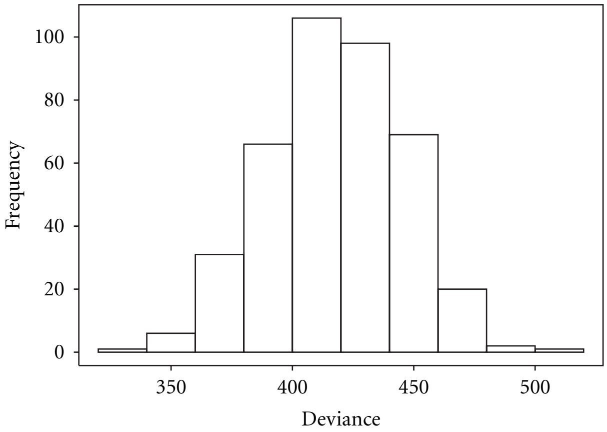

FIGURE 1: Histogram of the bootstrapped Dev* (b) for mackerel-egg survey data.

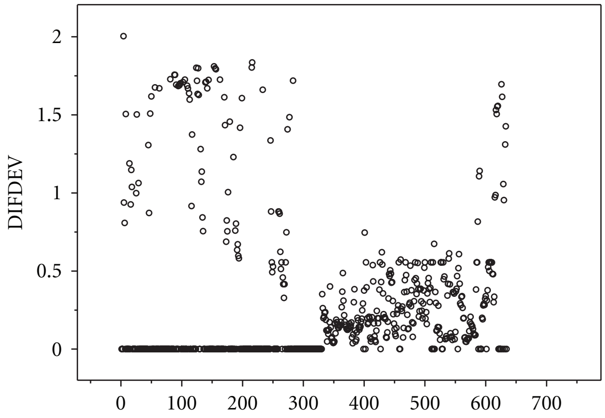

FIGURE 2: DIFDEV for mackerel-egg survey data.

Step 4. The CV score is given by

$$\mathrm{CV} = \sum_{i=1}^{n} \sum_{k=1}^{K} \{t_{[ik]} \ln(\hat{p}_{[ik]})\}. \tag{25}$$

Step 5. Tuning parameters with minimum CV can be determined as optimal by carrying out a grid search over C and γ.

4. Examples

4.1. Mackerel-Egg Survey Data. We consider data consisting of 634 observations from a 1992 mackerel egg survey [24]. There are the following predictors of egg abundance: the location (longitude and latitude) at which samples were taken, depth of the ocean, distance from the 200 m seabed contour, and, finally, water temperature at a depth of 20 m. We first fit a SVM. In the same manner as described in [11], we determine tuning parameters C and γ. The optimum values of the tuning parameters are $(C, \gamma) = (28, 0.09)$. The bootstrap estimator of the percentile for the deviance is Dev*(b) = 444.31. A comparison with the deviance Dev = 443.132 from (15) suggests that the SVM fits the

Table 1: Error rates for mackerel-egg survey data.

Model	Apparent error rate	Leaving-one-out CV error rate
SVM	0.159	0.167
GAM	0.158	0.175
Neural network with three hidden units	0.153	0.181
Fisher's linear discriminant	0.180	0.185
Logistic discriminant	0.192	0.196

Table 2: Error rates for liver disease data.

	Training sample		
Model	Apparent error	Leaving-one-out CV error	Test sample
SVM	0.085	0.209	0.292
Fisher's linear discriminant	0.366	0.360	0.340
Multinomial logistic discriminant	0.229	0.196	0.323

data fairly well. For reference purposes, the histogram of the bootstrapped $\mathrm{Dev}^*(b)$ for $B = 400$ is provided in Figure 1.

We can estimate the apparent errors rate of incorrectly predicting outcome and leaving-one-out CV error rates for several models as shown in Table 1. The smoothing parameters in generalized additive models (GAM) [24] and the number of hidden units in a neural network in Table 1 are determined using the leaving-one-out CV. From Table 1, the leaving-one-out CV error rate for the SVM is the smallest among all models, but the apparent error rate is the smallest for the neural network. The CV scores are 477.04, 509.44, and 541.61 for the SVM, logistic discriminant, and neural network, respectively. This implies that the SVM is the best among these three models from the point of view of CV. Figure 2 shows the index plot of $\Delta\mathrm{Dev}_{[i]}$, which indicates that no. 399 and no. 601 are influential observations at the 0.01% level of significance.

4.2. Liver Disease Data. We apply the proposed method to laboratory data collected from 218 patients with liver disorders [25–27]. Four liver diseases were observed: acute viral hepatitis (57 patients), persistent chronic hepatitis (44 patients), aggressive chronic hepatitis (40 patients), and postnecrotic cirrhosis (77 patients). The covariates consist of four liver enzymes: aspartate aminotransferase (AST), alanine aminotransferase (ALT), glutamate dehydrogenase (GIDH), and ornithine carbamyltransferase (OCT). For each (C, γ) pair, the CV performance is measured by training 70% and testing the other 30% of the data. Then, we train the whole training set by using the pair $(C, \gamma) = (93, 0.20)$, which achieves the minimum CV score (=187.93) and predicts the test set. The apparent and leaving-one-out CV error rates for traing and test samples for several models as shown in Table 2. As shown, the apparent error rate for SVM of training sample and the error rate for SVM of test sample are the smallest among all models, but the leaving-one-out CV error rate for SVM of training sample is larger than that of the multinomial logistic discriminant model.

5. Concluding Remarks

We considered the application of resampling methods to SVMs. Statistical inference based on the likelihood approach for SVMs was discussed, and the leaving-one-out CV was suggested for determining the tuning of parameters and for estimating the bias of the excess error in prediction. Bootstrapping is used to focus on the evaluation of the overall goodness-of-fit with the optimum tuning parameters. Data from a mackerel-egg survey and a liver-disease study are used to evaluate the resampling methods.

There is one broad limitation to our approach: the SVM assumed the independence of the predictor variables. More generally, it may be preferable to visualize interactions between predictor variables. The smoothing spline ANOVA models [28] can provide an excellent means for handling data of mutually exclusive groups and a set of predictor variables. We expect that flexible methods for a discriminant model using machine learning theory [1], such as penalized smoothing splines, will be very useful in these real-world contexts.

References

[1] C. M. Bishop, *Pattern Regression and Machine Learning*, Springer, New York, NY, USA, 2006.

[2] N. Cristianini and J. Shawe-Tylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Method*, Cambridge University Press, Cambridge, UK, 2000.

[3] C.-W. Hsu, C.-C. Chung, and C.-J. Lin, "A practical guide to support vector classification," 2009, http://www.csie.ntu.edu.tw/∼cjlin/papers/guide/guide.pdf.

[4] P. Zhang, "Model selection via multifold cross validation," *Annals of Statistics*, vol. 21, pp. 299–313, 1993.

[5] B. Efron and R .J. Tibshirani, *An Introduction to the Bootstrap*, Chapman & Hall, New York, NY, USA, 1993.

[6] M. Tsujitani and T. Koshimizu, "Neural discriminant analysis," *IEEE Transactions on Neural Networks*, vol. 11, no. 6, pp. 1394–1401, 2000.

[7] M. Tsujitani and M. Aoki, "Neural regression model, resampling and diagnosis," *Systems and Computers in Japan*, vol. 37, no. 6, pp. 13–20, 2006.

[8] M. Tsujitani and M. Sakon, "Analysis of survival data having time-dependent covariates," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 389–394, 2009.

[9] G. Gong, "Cross-validation, the jackknife, and the bootstrap: excess error estimation in forward logistic regression," *Journal of the American Statistical Association*, vol. 81, pp. 108–113, 1986.

[10] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

[11] C. W. Hs and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, pp. 415–425, 2002.

[12] C.-C. Chang and C.-J. Lin, “LIBSVM: a library for support vector machines,” 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[13] J. Platt, “Probabilistic outputs for support vector machines and comparison to regularized likelihood methods,” in *Advances in Large Margin Classifiers*, A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, Eds., MIT Press, Cambridge, Mass, USA, 2000.

[14] A. Karatzoglou, D. Meyer, and K. Hornik, “Support vector machines in R,” *Journal of Statistical Software*, vol. 15, no. 9, pp. 1–28, 2006.

[15] H. T. Lin, C. J. Lin, and R. C. Weng, “A note on Platt’s probabilistic outputs for support vector machines,” *Machine Learning*, vol. 68, no. 3, pp. 267–276, 2007.

[16] H. Akaike, “Information theory and an extension of the maximum likelihood principle,” in *Proceedings of the 2nd International Symposium on Information Theory*, B. N. Petrov and F. Csaki, Eds., pp. 267–281, Akademia Kaido, Budapest, Hungary, 1973.

[17] M. Ishiguro, Y. Sakamoto, and G. Kitagawa, “Bootstrapping log likelihood and EIC, an extension of AIC,” *Annals of the Institute of Statistical Mathematics*, vol. 49, no. 3, pp. 411–434, 1996.

[18] R. Shibata, “Bootstrap estimate of Kullback-Leibler information for model selection,” *Statistica Sinica*, vol. 7, no. 2, pp. 375–394, 1997.

[19] D. Collett, *Modeling Binary Data*, Chapman & Hall, New York, NY, USA, 2nd edition, 2003.

[20] J. M. Landwehr, D. Pregibon, and A. C. Shoemaker, “Graphical methods for assessing logistic regression models,” *Journal of the American Statistical Association*, vol. 79, pp. 61–71, 1984.

[21] T. J. Hastie and R. J. Tibshirani, “Classification by pairwise coupling,” *Annals of Statistics*, vol. 26, no. 2, pp. 451–471, 1998.

[22] E. J. Bredensteiner and K. P. Bennett, “Multicategory classification by support vector machines,” *Computational Optimization and Applications*, vol. 12, pp. 53–79, 1999.

[23] C. W. Hsu and C. J. Lin, “A formal analysis of stopping criteria of decomposition methods for support vector machines,” *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1045–1052, 2002.

[24] S. N. Wood, *Generalized Additive Models an Introduction with R*, Chapman & Hall, New York, NY, USA, 2006.

[25] A. Albert, *Multivariate Interpretation of Clinical Laboratory Data*, Marcel Dekker, New York, NY, USA, 1992..

[26] A. Albert and E. Lesaffre, “Multiple group logistic discrimination,” *Computers and Mathematics with Applications*, vol. 12, no. 2, pp. 209–224, 1986.

[27] E. Lesaffre and A. Albert, “Multiple-group logistic regression diagnosis,” *Computers and Mathematics with Applications*, vol. 38, pp. 425–440, 1989.

[28] Y. Wang, G. Wahba, C. Gu, R. Klein, and B. Klein, “Using smoothing spline anova to examine the relation of risk factors to the incidence and progression of diabetic retinopathy,” *Statistics in Medicine*, vol. 16, no. 12, pp. 1357–1376, 1997.

Advances in Multimedia
The Scientific World Journal
International Journal of Distributed Sensor Networks
Journal of Industrial Engineering
Applied Computational Intelligence and Soft Computing
Advances in Fuzzy Systems
Modelling & Simulation in Engineering
Journal of Computer Networks and Communications
Advances in Artificial Intelligence
Advances in Computer Engineering
International Journal of Computer Games Technology
International Journal of Biomedical Imaging
Advances in Artificial Neural Systems
Advances in Software Engineering
Journal of Robotics
Advances in Human-Computer Interaction
Computational Intelligence and Neuroscience
International Journal of Reconfigurable Computing
Journal of Electrical and Computer Engineering